Topic modelling using hierarchical priors on feature distributions with application to genetic association studies

Yidong Zhang

Linacre College
University of Oxford

A thesis submitted for the degree of
Doctor of Philosophy

Hilary 2022

Abstract

Multiple lines of evidence indicate that different diseases have shared pathological pathways. From an epidemiological perspective, certain diseases frequently co-occur within the same individuals. From a genetic perspective, the phenomenon of pleiotropy is widespread. In this work, I introduce, test and apply statistical methodology that aims to bridge these two perspectives and to better define how genetic risk factors influence the broad spectrum of common human diseases.

The approach taken is to use topic modelling applied to routine healthcare data, specifically hospital records encoded by the International Classification of Disease Version 10 (ICD10) ontology. Due to the sparse nature of the data, we introduce statistical methodology that uses the hierarchical structure of the ontology to create a prior on feature distributions. We combine this with Bayesian non-negative matrix factorization to develop our own methodology, named "treeLFA", to model the multi-morbidity patterns of common diseases as disease "topics". The estimated individuals' weights for topics then provide derived phenotypes for the analysis of association.

We first introduce the methodology and the computational methods used to estimate parameters. Using simulation, we then demonstrate that treeLFA outperforms other commonly used topic models in situation where the training data is small or individuals frequently have multiple topics. We also assess treeLFA's performance under various aspects of mis-specification.

We then apply the methodology to data from UK Biobank (UKB), finding that inferred disease topics align well with current medical understanding and provide additional power for genetic discovery (69 new loci identified for common diseases in the UKB). However, by comparing patterns of genetic association of topics and single disease codes, we find that the majority of genetic effects (about two thirds of topic-associated loci) are readily identified at the single code level. Nevertheless, we also show that, for about two thirds of diseases, genetic risk prediction can be improved by leveraging the GWAS results for topics. We also explore modelling multi-morbidity at different resolutions (i.e., with different numbers of topics) and find that inference results and most associations are stable across different resolutions.

In summary, treeLFA provides a new data-driven approach to model the hidden structure of the high-dimensional and sparse phenome data in biobanks. In addition, its inference results provide additional insights to genetic and epidemiological studies, in terms of both prediction of risks and the understanding of biology.

Topic modelling using hierarchical priors on feature distributions with application to genetic association studies

Yidong Zhang

Linacre College

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

Hilary 2022

Acknowledgements

Personal

My deepest appreciation to my supervisors Gil McVean, Gerton Lunter and Alexander Mentzer. Gil has been an invaluable source of knowledge and idea throughout my doctoral study in Oxford. With his support and judgement I rarely doubted I could make it to finish the exciting project. From him I gradually learnt how to become an qualified and outstanding researcher. I owe a great debt to Gerton Lunter, who agreed to become my co-supervisor on my second year. His encouragement and optimism gave me continual confidence to tackle difficulties, and his scientific insights and curiosities helped me solved countless methodological challenges along the way. I am also extremely grateful to Alexander Mentzer, who gave me opinions from a clinician's point of view, and was creative and productive whenever I sought input of ideas. I also greatly appreciate the helpful discussions with Chris Holmes and Alexander Dilthey.

I was lucky enough to work in two very supportive and friendly groups. My most profound gratitude to Xilin Jiang for getting me on track in the beginning and giving me huge amounts of help in all aspects of my life in Oxford. Thank you to Chaimaa Fadil and Wilder Wohns, it would be forever my precious memory to talk freely and enjoy the formal dinners with you in Oxford. Thank you to Jerome Kelleher, Yan Wong, Robert Hinch, Jason Hendry, Chris Eijsbouts and Marcus Tutert for always being generous with their time to answer my questions. Thank you to Emma Jones for her kindness, patience and quick response to all my enquiries. Thanks also to all members in Gerton's group. Thank you to Chris Cole, LiangTi Dai and Ron Schwessinger for discussing with me your projects and inspiring me with your ideas. Thank you to Edward Sanders and Ravza Gur for making my time in a new group such an pleasant experience.

I would like to express my gratitude to professor Tao dong for creating the opportunities for many Chinese students to come and study in Oxford (me included), and being caring to us and offering us guidance from day one. I also want to thank my previous supervisors Jie He and Yibo Gao in China for encouraging me to take the challenge and follow my dream.

I want to express my massive gratitude to my girl friend Zhiyan Xu. Thank you so much for your unending support and tolerance of my worries along this long

journey. Thank you for helping me become a more mature individual. My deepest gratitude to my parents for supporting me with the best they have for decades and leading by example to pursue the excellence and be fearless to hardships at all times.

Institutional

I am incredibly grateful to the COI-CSC scholarship programme for providing me the funding to further my study in Oxford. Thank you to the computing team in Big Data Institute for helping all students and researchers solve technical problems, and the biomedical research computing (BMRC) team for maintaining the amazing computing infrastructure. Thank you to the UK-Biobank team for building and maintaining such an incredible dataset, and all individuals who participated in this program for the benefits of human's health.

Abstract

Multiple lines of evidence indicate that different diseases have shared pathological pathways. From an epidemiological perspective, certain diseases frequently co-occur within the same individuals. From a genetic perspective, the phenomenon of pleiotropy is widespread. In this work, I introduce, test and apply statistical methodology that aims to bridge these two perspectives and to better define how genetic risk factors influence the broad spectrum of common human diseases.

The approach taken is to use topic modelling applied to routine healthcare data, specifically hospital records encoded by the International Classification of Disease Version 10 (ICD10) ontology. Due to the sparse nature of the data, we introduce statistical methodology that uses the hierarchical structure of the ontology to create a prior on feature distributions. We combine this with Bayesian non-negative matrix factorization to develop our own methodology, named "treeLFA", to model the multi-morbidity patterns of common diseases as disease "topics". The estimated individuals' weights for topics then provide derived phenotypes for the analysis of association.

We first introduce the methodology and the computational methods used to estimate parameters. Using simulation, we then demonstrate that treeLFA outperforms other commonly used topic models in situation where the training data is small or individuals frequently have multiple topics. We also assess treeLFA's performance under various aspects of mis-specification.

We then apply the methodology to data from UK Biobank (UKB), finding that inferred disease topics align well with current medical understanding and provide additional power for genetic discovery (69 new loci identified for common diseases in the UKB). However, by comparing patterns of genetic association of topics and single disease codes, we find that the majority of genetic effects (about two thirds of topic-associated loci) are readily identified at the single code level. Nevertheless, we also show that, for about two thirds of diseases, genetic risk prediction can be improved by leveraging the GWAS results for topics. We also explore modelling multi-morbidity at different resolutions (i.e., with different numbers of topics) and find that inference results and most associations are stable across different resolutions.

In summary, treeLFA provides a new data-driven approach to model the hidden structure of the high-dimensional and sparse phenome data in biobanks. In addition, its inference results provide additional insights to genetic and epidemiological studies, in terms of both prediction of risks and the understanding of biology.

Contents

Appendices

List of Figures

List of Abbreviations

EHR	Electronic health records.
GWAS	Genome-wide association study.
PheWAS	Phenome-wide association studies.
UKB	UK Biobank.
ICD-10	International Classification of Diseases Version 10.
PCA	Principal component analysis.
PCs	Principal components.
CCA	Canonical correlation analysis.
LSA	Latent semantic analysis.
SVD	Singular value decomposition.
LDA	Latent Dirichlet Allocation.
NMF	Non-negative matrix factorization.
NBMF	Non-negative binary matrix factorization.
HES	Hospital Episode Statistics.
KNN	K-nearest neighbour.
tSNE	t-distributed stochastic neighbor embedding.
topic-GWAS	GWAS on topic weights as traits.
Subgroup-GWAS	GWAS performed for a subgroup of a disease.
LD	Linkage disequilibrium.
INT	Inverse normal transformation.
glm	Generalized linear model.
QQ plot	Quantile-quantile plot.
OR	Odd ratios.
rdb	Regulome DB score.
chromHMM	A multivariate hidden Markov model trained on histone modifications to identify chromatin states.

eQTL	Expression quantitative trait loci.
PRS	Polygenic risk score.
ESS	Effective sample sizes.
VB	Variational Bayes.
HLA	Human leukocyte antigen.
MHC	Major histocompatibility complex.
NGS	Next generation sequencing.
WES	Whole exome sequencing.
WGS	Whole genome sequencing.
SNP	Single nucleotide polymorphism.
SBT	Sanger based typing.
HMM	Hidden markov model.
PRG	Population reference graph.
PBS	Peptide binding sites.

1 Introduction

Contents

1.1 Multi-morbidity

1.1.1 Overview of multi-morbidity

In recent decades, with economic development and the advancement of bioscience, medicine and technology, it has become inevitable for us to face the population-aging problem and its consequences. Among them, multi-morbidity is one of the most challenging. According to the report produced by the The Academy of Medical Sciences in the UK [1], the basic definition of multi-morbidity is the co-existence of two or more chronic conditions. Notably, the definition of multi-morbidity and co-morbidity are not entirely the same, though both refer to the co-existence of multiple conditions. For co-morbidity, there is an index condition that receives the major focus, while for multi-morbidty no disease is regarded as the index condition [1]. Multi-morbidity is increasingly common in different age groups across both high-income and low-income countries around the world (Figure 1.1) [2]. According to systemic reviews, estimates of the prevalence of multi-morbidity range from 13 % to 95 % [3, 4]. The large range for the estimates of prevalence is likely to be caused by the differences in the inclusion criterion and definitions of multi-morbidity among studies.

Multi-morbidity has negative impacts on both patients and health-care professionals and requires immediate attentions from clinicians, researchers and policy makers. It was found that patients with multi-morbidity have lower quality of life [5] and worse clinical outcomes [6] compared to those with single conditions; moreover, they also suffer from higher treatment expenses [7]. For health care professionals, the management of patients with multi-morbidity is challenging, since most treatments and guidelines are targeted towards single conditions. A similar situation can be found in the research field, since patients with multi-morbidity are often excluded from standard clinical studies, and most basic scientific projects are also focused on the mechanisms of single diseases. These result in the lack of relevant high quality data, and consequently evidence-based guidelines for multimorbidity, which makes its management less efficient and effective. Due to all these factors, the economic burden caused by multi-morbidity is often larger than the additive costs for multiple single diseases [8].

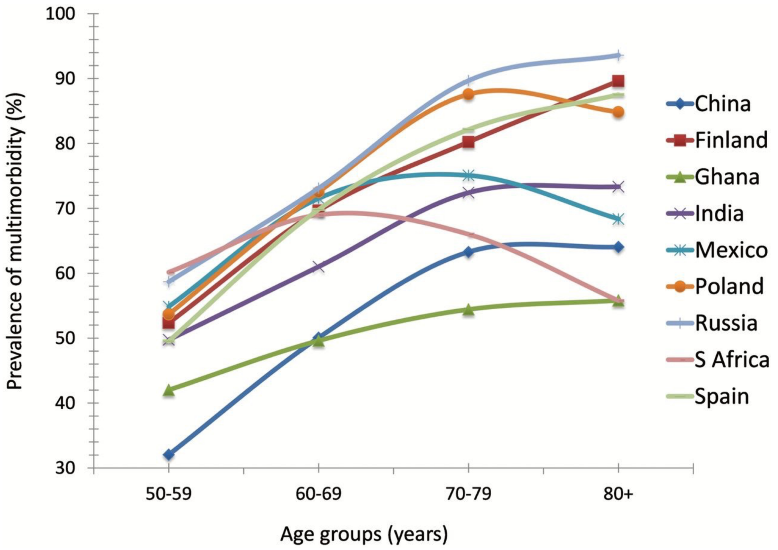


Figure 1.1: Multi-morbidity prevalence across different age groups in different countries. (Source: Garin, *et al.*, 2015)

1.1.2 The study of multi-morbidity

To better understand multi-morbidity, the first step is to identify which diseases are more likely to co-occur, or in other words, to identify the common multimorbidity clusters. A few systematic reviews have summarized multi-morbidity patterns reported in published studies [9, 10]. For instance, depression, cardiometabolic disorder and musculo-skeletal disorders have been identified as the most common components of different multi-morbidity clusters. However, these reviews have only summarized the results from individual observational studies. Nowadays, with the wide use of electronic health records (EHR), more systematic study of multi-morbidity can be carried out. A recent study identified six disease clusters (among which five are organ specific, including musculo-skeletal, endocrine-metabolic, digestive/digestive-respiratory, neurological, and cardiovascular patterns) by analyzing the EHR data of a Spanish cohort using multiple correspondence analysis (MCA) [11]. Other studies tried to build networks of diseases where the relationships among diseases are described via the topological measures of the network (Figure 1.2) [12–15]. Notably, these networks were constructed based on the correlation of occurrence for pairs of diseases, which limits its power in capturing complex structure in the high-dimensional phenotypic space. In the future, more advanced methodologies are still required to further mine the phenotypic data.

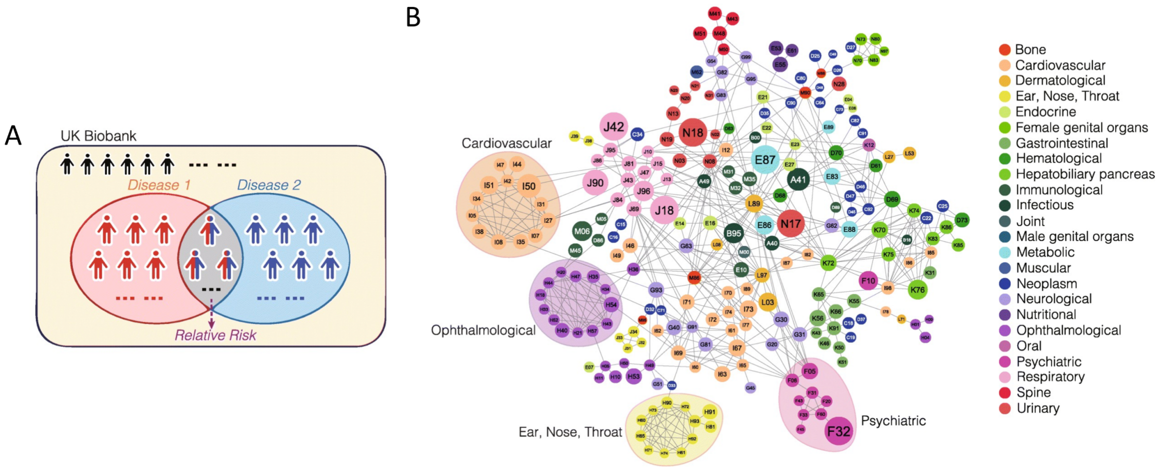


Figure 1.2: Multi-morbidity network based on the co-occurrences of diseases in UKB. Original caption: a, Schematic illustration of how relative risk (RR) is calculated for each pair of diseases. b, The high-confidence multi-morbidity network constructed by only including multi-morbidities with RR > 15. Each node represents a disease and each edge represents a multi-morbid relationship between two diseases. The color code of a node represents the category of the disease. The size of each node is proportional to the number of its multi-morbidities. (Source: Dong, *et al.*, 2021)

With multi-morbidity clusters identified, the next important step is finding their determinants. Both genetic and environmental factors can play key roles in this process, and among these factors there can also be complex interactions. Besides, the causal relationships among diseases in the same multi-morbidity cluster can also be sophisticated. For one thing, diseases may share common causal factors (for instance, psychiatric diseases are known to share many associated genomic loci [16]); for another thing, certain diseases may themselves be causal to other ones (such as the relationship between obesity and type-2 diabetes [17]).

Broadly speaking, there are still only very limited studies focusing on the mechanisms of multi-morbidity instead of individual diseases. To dissect the complexity of multi-morbidity, diseases in the same cluster should be analyzed in an integrated framework, instead of being studied individually and then compared with each other. In the future, studies specifically designed for multi-morbidity are undoubtedly necessary for us to accumulate knowledge and insights about them, which are the prerequisites for achieving more efficient prevention and treatment of multi-morbidity in the long run.

1.1.3 Genetics and multi-morbidity

In past decades, genetic studies have revolutionized the way we understand and (to a lesser extent) manage diseases. It has been widely observed that some seemingly unrelated diseases are genetically correlated [18], which may partly explain their tendency to co-occur. For example, epidemiological studies on depression and cardio-metabolic diseases first found they increase the risks for each other [19]; later, shared genetic factors were identified by analysing the results for their corresponding genetic studies [20, 21].

Some studies have aimed to systematically assess the roles of genetic factors in forming major multi-morbidity clusters. For instance, Bagley *et al.* evaluated the overlap of disease pairs that are likely to co-occur in EHR and disease pairs that are known to be genetically correlated, and found that immune and neuro-psychiatric diseases are the two major classes of diseases for which the multi-morbidity clusters are likely to be driven by genetic factors [22]. More recently, with the wide application of biobanks, integrated analysis of both genotype and phenotype data for the same population has become feasible [23]. A recent study used the hospital inpatient data for 385,335 patients in the UK Biobank (UKB) to investigate the morbidity relationships among 439 common diseases. After constructing the multi-morbidity network for all pairs of diseases that are likely to co-occur, post-GWAS analyses were carried out to reveal their shared genetic components on different levels (loci, network and genetic architecture levels). It was found that 46 % of multi-morbidities have shared genetic components on at least one of the three levels [15], indicating the widespread effect of genetic factors on multi-morbidity clusters.

To date, most studies focusing on systemically investigating multi-morbidity clusters from a genetic perspective are still largely exploratory in nature. The best pipelines and frameworks for carrying out such studies have not been established and much remains to be done to improve the computational algorithms for such work. Moreover, novel findings need to be thoroughly validated, and their biological indications also require further investigation. In spite of the gaps to be filled, it is clear that studying the genetics of multi-morbidity is an efficient strategy to deepen our understanding, since evidence accumulated thus far strongly suggests the importance of genetic factors, and there are sufficient resources (such as the genome and phenome data for the same population in biobanks) now available to researchers.

1.2 Association study with genome and phenome data

1.2.1 Genome-wide association study: GWAS

During the past fifteen years, Genome-wide association study (GWAS) have been the driving force for modern genetic studies. It is used to find genetic variants across the entire human genome that are associated with phenotypes of interest, such as common diseases, expression of genes or behaviours. To date, more than 5,700 GWAS have been conducted on 3,300 traits [24], and tens of thousands of significant variants have been found. In the future, it is expected that more and more significant variants (including rare variants) will continue to be found by GWAS with increasingly larger sample sizes.

A GWAS usually scans through hundreds of thousands of genetic variations across the genome. Typically, there is usually only one phenotype studied by a GWAS each time. This type of study design to a large extent limits the potential of GWAS, in that the recruitment of patients meeting certain inclusion criterion is time-consuming and expensive. In recent years, with the establishment of biobanks and the extensive use of EHR [25, 26], researchers began to have other options for designing and carrying out GWAS.

1.2.2 GWAS with biobank data

In recent years, many population-based biobanks have been established across the world [27–30]. Biobanks, by definition, are collections of biological specimens, such as blood samples that can be used for genotyping or sequencing. Meanwhile, unlike traditional cohorts, in biobanks a large number of variables are measured for all individuals through various approaches including questionnaires, physical examinations, laboratory tests and linkage to people's EHR. Take the UKB as an example; at present, biological measurements (including genetics), lifestyle indicators, biomarkers as well as the imaging data for half a million people have been made fully accessible to researchers around the world [27]. With these resources, it has become more convenient than ever for researchers to extract cases and controls for any specifically designed association studies (Figure 1.3). It is noteworthy that the definition of cases does not have to rely on a single phenotypic variable. For instance, instead of using only one diagnostic code to find people in biobanks with a certain disease, electronic phenotyping algorithms can more accurately define the desired phenotype by combining a few related diagnostic codes [31].

Biobanks are ideal resources for the study of multi-morbidity, because the diagnosis data for tens of thousands of diseases for all individuals are directly available, thus the co-occurrence of any pair of diseases can be directly evaluated. Moreover, phenome data in biobanks are linked to omics data, enabling integrative analyses across data modalities.

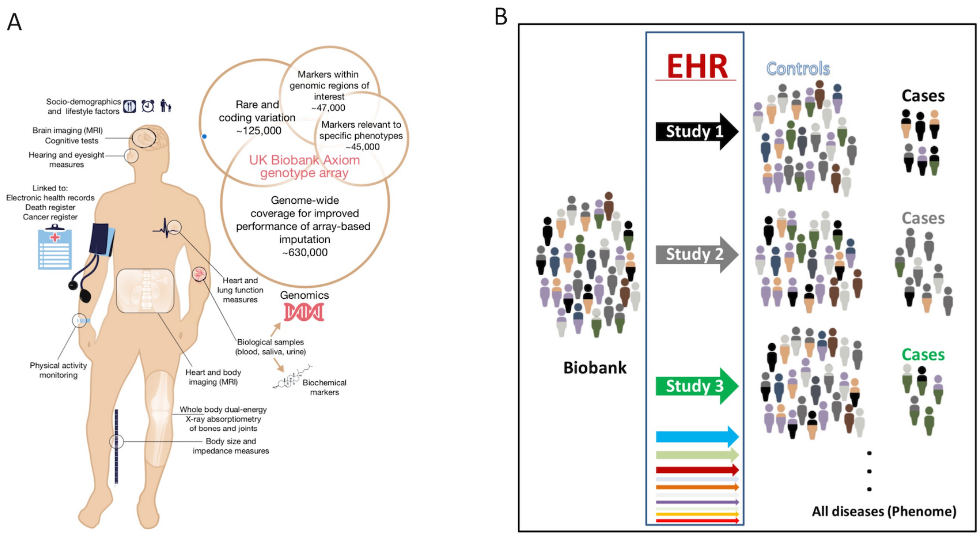

A
Socio-demographics and lifestyle factors
Brain imaging (MRI) Cognitive tests
Hearing and eyesight measures
Linked to: Electronic health records Death register Cancer register
Physical activity monitoring
Heart and lung function measures
Heart and body imaging (MRI)
Whole body dual-energy X-ray absorptiometry of bones and joints
Body size and impedance measures
Biological samples (blood, saliva, urine)
Genomics
Biochemical markers
Rare and coding variation ~125,000
Markers within genomic regions of interest ~47,000
Markers relevant to specific phenotypes ~45,000
UK Biobank Axiom genotype array
Genome-wide coverage for improved performance of array-based imputation ~630,000
B
EHR
Controls
Cases
Study 1
Study 2
Study 3
Biobank
All diseases (Phenome)

Figure 1.3: Resources in UKB and design of GWAS using biobank data. Original caption: a, Summary of the UK Biobank resource and genotyping array content. b, An Example of a Biobank Linked to an Electronic Health Record. (Source: Bycroft *et al.*, 2018; Hebbring, 2018).

1.2.3 Phenome-wide association study: PheWAS

Biobanks paved the way for carrying out more GWAS at much lower cost. However, focusing on one trait each time is not the only and best way to exploit the rich information in biobanks. To find more phenotypic associations for genomic loci, Phenome-wide association studies (PheWAS) were developed, aiming to find phenotypic associations for a single exposure (genetic variants, genetic risk score or a biomarker) from a large number of candidate phenotypes [32]. GWAS are a representative of forward strategy (from phenotype to genotype) for genetic studies. By contrast, PheWAS is a reverse strategy (from genotype to phenotype), therefore PheWAS is often regarded as a complement to GWAS. In 2010, the first proof-of-concept PheWAS was published [33]. In the following years, other PheWAS studies proved that PheWAS could not only replicate GWAS discoveries, but can help to identify new associations as well [34, 35]. Particularly, PheWAS is well-suited to studying cross-phenotype (CP) associations for genetic variants, so-called "pleiotropic" effects.

1.2.4 Widespread pleiotropic effects for common genetic variants

A striking phenomenon discovered by GWAS and PheWAS in the past decade is that many genomic loci are associated with multiple different phenotypes (cross-phenotype association, or CP association), especially different diseases. One famous example of pleiotropy is the FTO gene, which was discovered to be associated with both BMI and type 2 diabetes. Later, PheWAS found it was also associated with sleep apnea [36]. Another well-known example is the human's HLA (human leukocyte antigen) gene complex, the most polymorphic and pleiotropic region in the human genome. Various conditions related to human's health are associated with genetic variations in this specific genomic region, particularly autoimmune diseases and susceptibility to infectious diseases [37, 38].

A recent study quantitatively assessed the extent of pleiotropy across the genome, and found that among the 41,533 significant loci that had been discovered by GWAS, 93.3% are associated with more than one traits [39]. In many cases the discovered CP associations are quite unexpected, since many diseases associated with the same genomic locus are previously thought to be completely unrelated. For instance, it was discovered that variants in the ABO gene (which determines people's blood groups) was associated with the risk of coronary artery disease and tonsillectomy [40]. There are many possible reasons behind the observed CP associations, and genuine pleiotropy (a genomic locus directly affects more than one trait) is only one of them, since there are also spurious CP associations (which can be caused by mis-classification of cases, ascertainment bias or the ambiguity in mapping the true underlying causal variant with the tag SNP identified by the GWAS) and mediated associations (one phenotype is causally related to a second phenotype) [41]. The above mentioned percentage of pleiotropic loci refers to statistical pleiotropy, which include both spurious CP and mediated CP.

The existence of widespread pleiotropy has important research and clinical implications. First, the universal pleiotropic effects provide a motivation for the study of multi-morbidity from a genetic perspective, since it is common for different diseases to share causal variants, and identifying them will point to the underlying biological pathways for these diseases. Furthermore, from a clinical perspective, the genetic components shared by multiple diseases also offer new opportunities for drug re-purposing [41].

1.3 Multi-trait GWAS

1.3.1 Limitations of PheWAS

As discussed in the previous section, the widespread pleiotropic effects for common SNPs and the establishment of biobanks make PheWAS a very appealing tool to researchers. However, PheWAS also has major limitations. First, PheWAS is usually performed for genomic loci for which GWAS have already provided evidence of phenotypic association. In another word, PheWAS is not used to discover new genetic markers - only novel associations. However, it has long been realized that many diseases are polygenic, for which a large number of variants with small effects can explain a non-small proportion of the phenotypic variance [42]. For these diseases, new significant variants are kept being found, enabling us to better depict their genetic architecture. Second, PheWAS evaluates all traits independently and doesn't take into account the phenotypic correlation structure, which can result in a loss of statistical power for discovery. For the phenome data in biobanks, these limitations of PheWAS means it is far from a perfect method. In biobanks, many traits are rare, which makes if difficult to find their genomic associations. On the other hand, in biobanks there are many correlated traits (such as the systolic and diastolic blood pressure), which potentially reflect the same underlying biological pathways from different perspectives. Evaluating these phenotypes independently can result in a significant loss of power.

1.3.2 Applications of multi-trait GWAS methods

In recent years, multi-trait GWAS methods have been under fast development and have received lots of attention. Multi-trait GWAS methods jointly model the association between variants and multiple related phenotypes (diseases), aiming for an increased power for discovery, making use of the correlation structure among phenotypes and the larger sample size achieved by joining individuals with different yet correlated phenotypes into a single group.

To date, multi-trait GWAS have generated important insights about the mechanisms of complex diseases that cannot be obtained using standard single trait GWAS. For instance, based on the summary statistics of single trait GWAS, Julienne *et al.* detected new loci with CP associations that cannot be found by single trait studies. These significant loci were assigned to different clusters according to their phenotypic association profiles, and mapped to distinct biological pathways in various tissues [43]. In another study, by applying Non-negative matrix factorization (NMF) on the summary statistics of 47 significant variants found by single trait GWAS for 47 type-2 diabetes (T2D) related traits, five distinct clusters of T2D associated loci were identified, which all have distinct patterns of tissue specific enhancer or promoter enrichment and are potentially involved in different biological pathways [44].

For the study of multi-morbidity, multi-trait GWAS is a natural and straightforward choice, since diseases in the same multi-morbidity cluster can be analyzed jointly, and their shared genetic components can be directly identified. In the next section, a review of different multi-trait GWAS methods will be given. The major emphasis will be put on the dimension reduction methods, which are suitable for analyzing high-dimensional phenome data in a data-driven and hypothesis free manner, since these methods can automatically extract hidden patterns from the phenome data.

1.4 Multi-trait GWAS methods

As summarized by Hackinger and Zeggini [45], multi-trait GWAS methods can be divided into two classes according to their underlying statistical framework: the univariate methods and multivariate methods. Univariate methods combine the summary statistics of multiple single trait GWAS, while multivariate methods model all the traits and the genetic variant in a unified framework. Multivariate methods can be further divided into two classes: regression based methods (also named as direct methods, since they model the phenotypes without changing its original format) and dimension reduction methods (also named as indirect methods, since the original traits are combined in some ways to form new traits). Unlike univariate methods multivariate methods require access to individual level data. Figure 1.4 provides a schematic summary of the main methods to be discussed in the following sections.

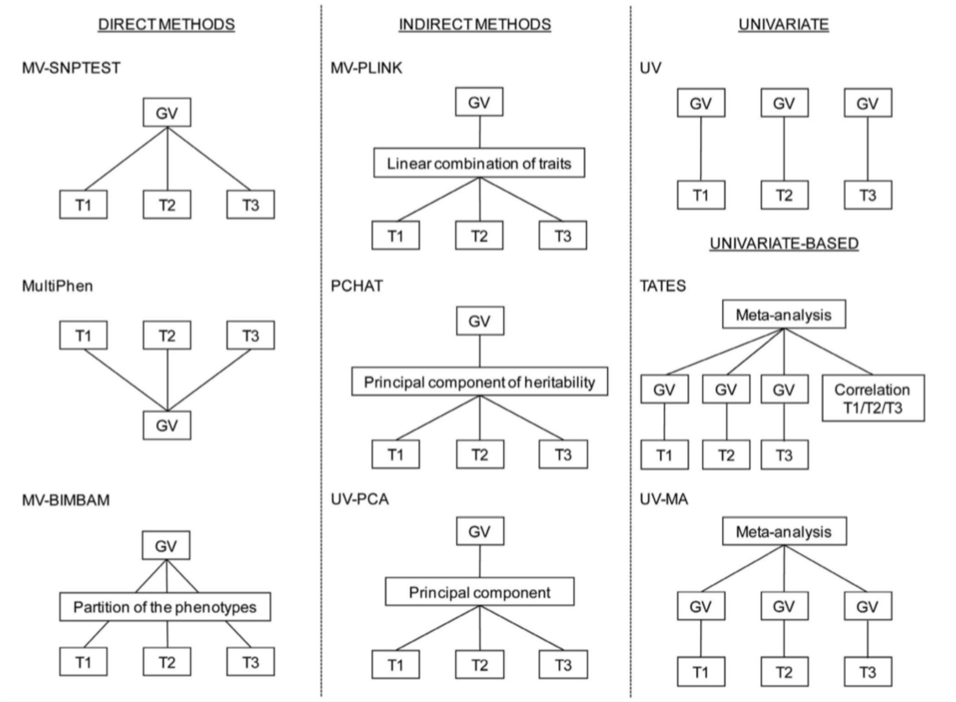


Figure 1.4: Major classes of multi-trait GWAS methods. Original caption: GV indicates genetic variant; MV, multivariate; PCHAT, Principal Component of Heritability Association Test; T1, trait 1; T2, trait 2; T3, trait 3; TATES, Trait-based Association Test that uses Extended Simes procedure; UV-MA, meta-analysis of univariate results; UV-PCA, univariate analysis of first principal component. (Source: Galesloot *et al.*, 2014)

1.4.1 Univariate methods

In general, univariate methods offer more flexibility than multivariate methods. The origin of univariate methods was the meta-analysis methods for GWAS, which combined the P-values or estimated effect sizes given by different studies on the same trait. To analyze traits with heterogeneous genetic associations, cross-phenotype meta-analysis (CPMA) methods were then designed, which compared the expected and observed joint distribution of P-values for all traits at a locus [46]. Later, other more advanced univariate methods were developed. For instance, The "Asset" evaluates all possible subsets of traits to find exactly which traits are associated with a variant [47]; "CPASSOC" combines P-values given by single trait GWAS, and it can model heterogeneous genetic effects on different traits and the overlap of samples for different cohorts [48]; "TATES" also works by combining the P-values for single traits, but takes into account the correlation structure among traits [49]; "MTAG" dissects different sources of the correlation between the estimated effects for correlated traits, and improves the estimates for each trait [50]. There are also Bayesian univariate approaches. For instance, "metaABF" calculates the probabilities of different association models for one variant and multiple correlated traits, with the correlation structure for traits and the relatedness of individuals in different studies encoded using priors on parameters of the model [51]; "CPBayes" uses spike and slab prior for the effect sizes of different traits and relies on the Gibbs sampling to simulate their posterior distributions [52].

1.4.2 Multivariate methods

In recent years, multivariate methods have become increasingly popular because they can make use of the individual level data that is available in many large scale datasets. Theoretically, individual level data contains more information than the summary statistics given by single trait GWAS, therefore multivariate methods should possess larger power for discovery compared to univariate methods.

1. Multivariate mixed models: Linear mixed models (LMM) are an extension to the standard regression model, which incorporates both fixed and random effects. When used as a multi-trait GWAS method, LMM model genetic effects as fixed effects and the inter-trait covariance as random effects [53]. In the recent years, algorithmic improvements have significantly sped up the computation for LMM [54, 55], allowing simultaneous analysis of up to 100 traits [56]. More advanced LMMs also allow non-normally distributed traits (such as binary traits) and even the combination of different types of traits [57, 58].

2. Bayesian approaches: A major advantage of Bayesian approaches is that direct model comparisons can be made. The standard Bayesian multivariate linear model was implemented in the software “SNPTEST” [59]. By comparing Bayes factors for different models of associations, “PleioGRiP” could detect additional phenotypic associations for a variant conditioned on one known association [60]. A unified framework proposed by Stephens *et al.* (implemented in the software “mvBIMBAM”) divides all phenotypes into three categories: unassociated, directly associated and indirectly associated. Models of different partitions of phenotypes into these groups can be compared using their Bayes factors [61].

3. Other approaches: In addition to the major classes of methods discussed above, there are also other unconventional methods. The “Multiphen” [62] performs ordinal regression for the genotype variable against multiple phenotypic variables and tests their associations. Since phenotypes are independent variables in this framework, no assumption needs to be made for their distributions [63]. Multinomial regression (implemented in the software “TRINCULO”) is a generalization of the logistic regression, and can be used to model phenotypic variables with more than two possible values, such as diseases with multiple subtypes [64].

4. Explicit tests for pleiotropy: Most methods discussed above do not explicitly test for cross-trait associations, since a global null hypothesis that assumes no trait is associated with the variant is typically used. As a result, post-hoc analyses are needed to further dissect the association signals. Specific methods were also developed to overcome this problem. For instance, the sequential likelihood ratio test developed by Schaid *et al.* gives the exact number of associated traits. This framework starts with the global null of no associated trait, and tests for one additional associated trait in each step, until the null hypothesis of no more association can no longer be rejected [65].

5. Summary of multivariate methods: There have been a few studies aimed at evaluating and comparing the performance of different multi-trait GWAS methods. Galesloot *et al.* found that all multivariate methods have higher power compared to standard single trait method, even when the correlation between the traits is small, or there is only one trait associated with the variant [66]. Porter *et al.* concluded that most multivariate methods have similar power, which is typically higher than single trait methods. In addition, the relative performance of different multivariate methods depends largely on the specific correlation structure between the traits and the specific combination of genetic effects on all these traits [67].

Notably, most of the methods discussed above rely on experts to select the set of traits to be analyzed together based on their domain knowledge. For the phenome data in biobanks, such expert-led methods are not sufficient, since in addition to analyzing traits that are known to be correlated, we also want extract unknown patterns from the data. To deal with this problem, dimension reduction methods offer a potential solution.

1.4.3 Dimension reduction methods

A common and practical way to deal with very high-dimensional phenotype data is to firstly apply a dimension reduction on the original features in the input data to project them into a space of much lower dimension. With the new set of features, standard multi-trait and single-trait GWAS can then be carried out, with important correlation structure retained and the multiple testing burden relieved.

The most distinguishing feature of dimension reduction methods is the derivation of a new set of traits. They may be understood as something similar to the endo-phenotypes (intermediate phenotypes) used in PheWAS, such as bio-markers [32]. Clearly, these endo-phenotypes are not clinical end-points, but measurements that may reflect latent diseases or pathological processes in an early stage. For the study of multi-morbidity, new traits obtained by dimension reduction algorithms on the original disease occurrence data may represent abstract biological pathways or mechanisms shared by a group of different yet related diseases.

The most commonly used dimension reduction method is principal component analysis (PCA), which finds the linear combination of the original traits that maximize the covariance of these traits. Quite a few studies have used PCA-based methods for combined analyses of multiple related traits (such as different quantitative traits related to the metabolic syndrome) [68–70]. Notably, some studies further applied other multi-trait GWAS methods on the principal components (PCs) of the original traits [68]. Hugues *et al.* discovered with simulations that combining the association signals across all PCs of the original traits may significantly increase the power for detecting both the pleiotropic variants and variants associated with only one trait [71].

One drawback of the basic PCA based multi-trait GWAS is that PCs derived for the phenotypic features are not genetically based. “PHCAT” extends standard PCA and aims to find the single new trait that results in the largest heritability for the variant of interest among all the possible linear combination of the original traits [72]. Simulation showed that PHCAT had larger power than standard PCA based methods, particularly when some traits were not associated with the variant.

CCA (canonical correlation analysis) is another type of dimension reduction method [73–75], which evaluates the linear relationship of two sets of variables (the canonical correlation refers to the correlation between the two sets of variables). For genetic studies, CCA implemented in the software PLINK finds the linear combination of traits that maximizes the covariance between variants and traits. If only one variant is analyzed, the statistical test used for CCA is equivalent to the MANOVA (multivariate analysis of variance) test [63].

In theory, both CCA and PCA work best on continuous variables that are approximately normally distributed. For other types of input data (such as binary input data), it is better to use other latent factor models for dimension reduction, such as topic models developed for discrete input data. The use of topic models on phenotype data is the basis of this thesis, so they will be comprehensively reviewed in the next section.

1.5 Topic models

Topic models were originally developed for text mining and natural language processing. They learn topics from documents based on the counts of all words used in documents in the corpus (Figure 1.5). Words that are frequently used together in the same context in documents will be grouped into the same topic.

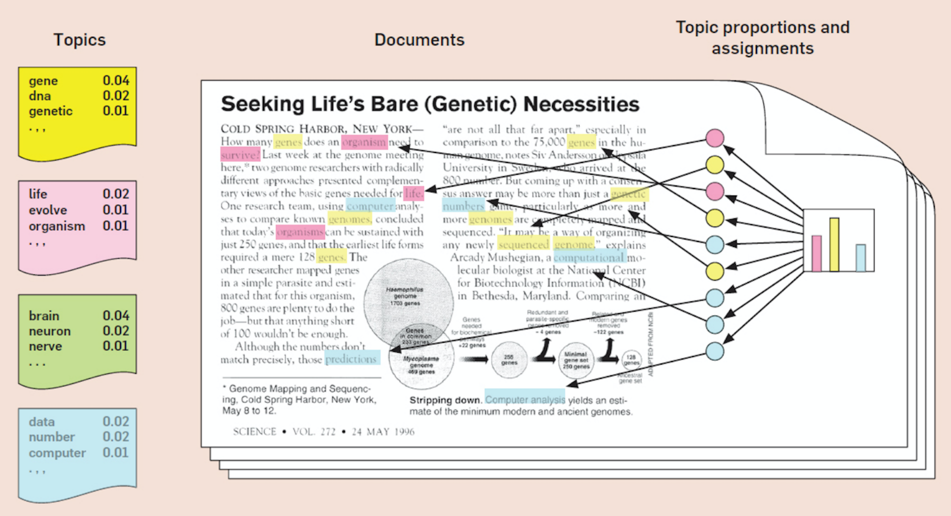


Figure 1.5: Schematic for the learning of topics from documents by topic models. Topics of documents were learnt based on the number of words used in documents. (Source: Blei, 2012)

The basic assumption of topic models is that a document is a "bag of words", which means the model omits the order and the context of words in documents, and pays attention only to the number of times each word in the vocabulary is used in a document. In this way, the input for topic models is simply a word-count matrix for all documents, with each row (or column) represents a document and each column (or row) represents a word in the vocabulary. Each entry in the input matrix records the number of times a word appears in a document. The outputs of topic models are usually two matrices: the topic matrix (which contains the topics used by all the documents) and the topic weight (loading) matrix for documents, which describes the contributions of different topics to different documents.

For the study of multi-morbidity, topic models can also be directly applied to the diagnosis data for patients to project them from the original disease space to the lower dimensional topic space. More specifically, each disease will be regarded as a word, and each person viewed as a document. In another word, people will be treated as "bag of diseases" by topic models. In this way, the occurrences of different diseases on people can be explained by people possessing different weights for different topics 1.6. In the next section, the emphasis will be on technical aspects of topic modelling.

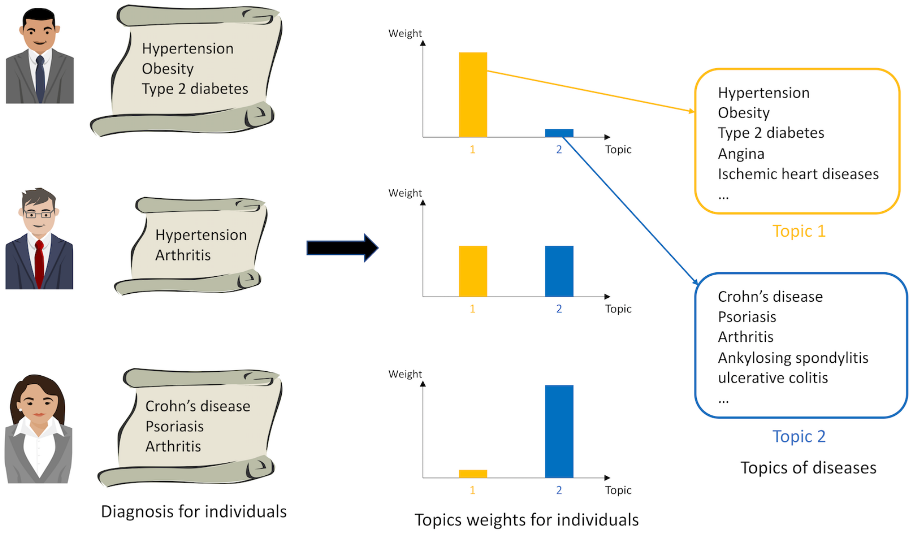


Figure 1.6: Schematic for the study of multi-morbidity with topic models. Based on the diagnosis data for people, topics of diseases and individuals' weight for topics are inferred by topic models.

1.5.1 Overview of major types of topic models

The development of topic models

The origin of topic models can be dated back to 1990, when the latent semantic analysis (LSA) was proposed. LSA is based on the singular value decomposition (SVD) of the transformed document-word (word-count) matrix [76]. One drawback of LSA is that the factorized matrices can contain negative entries. Later, non-negative matrix factorization (NMF) was developed [77], which puts non-negativity constraints on the two factorized matrices. This makes the interpretation of learnt topics and topic weights easier, since only additive contributions of topics to documents are allowed. NMF-based topic models continue to be widely used today.

Aside from adding constraints onto the parameters of the model, another development direction is to build probabilistic models. Inspired by LSA, pLSA, the probabilistic version of LSA was developed in 2001 [78]. pLSA has a proper generative process, in which each word in a document has a corresponding latent topic assignment variable, denoting which topic generates the word. The joint likelihood of a pLSA model can also be written in the form of a matrix tri-factorization [78]. The difference between LSA and pLSA is that LSA uses the L2 norm as the objective function for matrix factorization, while pLSA uses the cross entropy between the empirical distribution and the model [78], which better fits its generative process. Probabilistic models also enabled researchers to use standard statistical tools to deal with tasks like model selection and comparison. A shortcoming of pLSA is that topic weights of documents are individual parameters that have to be learnt separately and independently for each document in the training set.

Latent Dirichlet Allocation

The Latent Dirichlet Allocation (LDA) [79, 80], which was a Bayesian extension to pLSA. The most important advancement for LDA was putting a Dirichlet prior on the topic weights of all documents, which makes LDA a full Bayesian model. For LDA, each topic is a multinomial distribution over words, and each document is a multinomial distribution over topics. To generate the documents (all words in documents), firstly for each document a topic weights vector (a multinomial distribution denoting the weights of topics in the document) was sampled from the Dirichlet prior. Then for each word in the document, a topic assignment is sampled from the topic weight distribution. In the final step, each word is sampled from the corresponding topic according to its topic assignment. The generative process of LDA is summarized below:

```
For each topic $k$:
    sample topic variable $\phi_k \sim Dirichlet(\eta)$;
For each document $d$:
    sample topic loading variabel $\theta_d \sim Dirichlet(\alpha)$;
    For each word placeholder $s$ in document $d$:
        sample topic assignment variable $Z_{ds} \sim Multinomial(\theta_d)$;
        sample word $W_{ds} \sim Multinomial(\phi_{Z_{ds}})$;
```

The graphical model for LDA is shown in Figure 1.7.

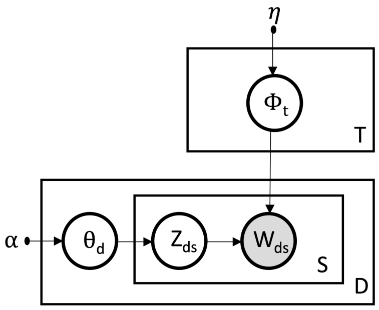


Figure 1.7: The graphical model for LDA.

At present, Latent Dirichlet Allocation is the most widely used topic model in various fields. Since its introduction, a large number of more advanced topic models have been developed based on its framework with diverse functionalities added, such as correlated topic models [81] and supervised topic models [82].

3. Binary matrix factorization methods

Thus far, we have been discussing topic models built for count input data. However, other types of input data can also be analyzed by topic models, such as binary data, in which zeros and ones are used to represent the presence/absence of features for samples (documents).

In fact, LDA can also be used to model binary input data. However, this is not completely appropriate, since in this way all words can at most occur once in a document, and zeros in the input matrix will be ignored by the model. Apart from LDA based topic models, there were also other types of models specifically designed for binary input data based on various matrix factorization techniques. The most common way to model binary input data with matrix factorization can be expressed as:

$$P(W \mid \phi, \theta) = \prod_d \prod_s Bernoulli(W_{ds} \mid \sigma(\theta * \phi)_{ds}) \tag{1.1}$$

In the above equation, W is the input matrix, rows of matrix θ are topic weights for documents, and rows of matrix ϕ are topics. Each binary word variable w_{ds} (word s in document d) is modelled with a Bernoulli distribution, and the matrix of Bernoulli distributions for all words are parameterized with the product of θ and ϕ.

To ensure the validity of this parameterization, all entries in the product matrix of ϕ and θ need to fall between 0 and 1. A common way to achieve this is to apply a link function (such as the logit link function, represented by σ in equation 1) on the product matrix, which makes the model nonlinear. The advantage in using a link function is no extra constraints are needed for θ and ϕ. On the contrary, by adding additional constraints on ϕ and θ, the product matrix can be directly used. This type of linear model is named mean-parameterized binary non-negative matrix factorization (BNMF). If θ and ϕ are treated as variables and given prior distributions, we will have Bayesian BNMF. Alberto, et al. proposed a general framework for such models [83], in which various combinations of prior distributions (including beta prior for individuals entries in the matrices, and Dirichlet prior for rows or columns of the matrices) can be chosen for the components of θ and ϕ.

1.5.2 Applications of topic models

Since its introduction, topic models were quickly adopted in various biological research fields such as population genetics and functional genomics [84]. Topic models were often used as bioinformatic tools for data mining tasks such as clustering, classification and feature extraction. For instance, they were used to model microarray expression data, where each sample was regarded as a document, and each gene a word. Topics in this scenario represented groups of genes which had functional relations [85, 86]. Topic models were also efficient in extracting hidden structure from EHR data. For instance, "MixEHR", a Bayesian multi-view topic model was developed to extract meaningful medical concepts (in the form of topics) from the heterogeneous and noisy EHR data [87]. Based on the inferred topic weights for patients, superior prediction power for undiagnosed phenotypes was achieved compared to other state-of-art machine learning methods.

Topic models for genetic studies

Topic models were also used in genetic studies. In 2002, Pritchard *et al.* developed a Bayesian clustering approach to infer population structure and assign individuals to different population groups using genotype data [80]. Via the use of topic models, simultaneous assignment to multiple populations was allowed, which means alleles of different loci for the same individual can be sampled from the corresponding parametric distributions for different populations.

In 2017 McCoy *et al.* first applied topic models to population based association studies. LDA was used to learn 50 topics for 508 Phecodes from three small biobanks, and GWAS was then carried out using topics traits, instead of single Phecodes [88]. Multiple known disease-associated loci were recovered with smaller P-values compared to the standard case-control GWAS, and new significant loci were also found. After the first proof of concept study, topic modelling of diseases was used in a few following studies aimed to expand our knowledge of the pleiotropic effects of known disease-associated loci. For instance, in one study a polygenic risk score (PRS) for depression was found to be associated with topics of cardiac diseases, suggesting shared genetic components for these diseases [89]. Another study found that the schizophrenia associated gene SLC39A8 was also associated with topics of cerebrovascular diseases [90]. The third study focused on the SNP rs10455872 [91], which is located in the lipo-protein A (LPA) gene and explains 20-30 % of the variation circulating lipo-protein level. A NMF based topic model learnt six topics for 1853 phenotypes from the EHR of 12759 individuals, and a negative correlation between rs10455872 and a topic enriched for lung cancer related codes was discovered, which was not known before.

To summarize, in all the studies discussed in this section, some previous GWAS findings were replicated, and new discoveries were also made. This was achieved by joining diseases into topics to form new traits for the standard GWAS and PheWAS. In spite of these preliminary success, there is still large space for improvement. For instance, these studies all used the most standard topic models (LDA or NMF), and many technical issues related to topic models (such as model selection and hyperparameter learning) were only tackled using some basic heuristic rules. In the following section, some technical issues related to topic models will be briefly discussed, before the research goals of the thesis are given in the final section.

1.6 Technical issues related to topic models

1.6.1 Model selection for topic models

For standard topic models, we need to specify the number of topics to be inferred before training the models. In reality, this number is usually not known a priori. To tackle this problem, people usually train multiple topic models with different number of topics, and then compare their performance using some metrics. To date, there hasn't been a golden standard metric for the evaluation of topic models. A common metric being widely used is the predictive likelihood for a model on the testing dataset. It is calculated based on the topics inferred from the training data, and it evaluates the generalization ability of the inferred topics to unseen new data. On average, a higher likelihood on the testing dataset indicates better inferred topics. The review written by Wallach [92] gives a comprehensive summary of the existing methods to calculate this predictive likelihood, most of which are based on monte-carlo techniques.

In addition to the likelihood on the testing data, there are also other model selection methods based on different ideas. For instance, in practice different secondary tasks such as document classification and information retrieval can be used to evaluate topic models if they are trained for a specific purpose. Furthermore, other metrics such as the coherence and stability of inferred topics [93–96] directly evaluate the inferred topics from different perspectives. Notably, Cao *et al.* proposed to do model selection by calculating the overall correlation among all the inferred topics. The idea was that with both too few or too many topics, the overall/averaged correlation among topics would be large, since ideally the inferred topics should be distinct, which will leads to small overall correlation [97]. Lastly, building non-parametric models is another option. The Dirichlet process [98] can be regarded as the non-parametric version of LDA, for which the number of topics is a also a variable to be learnt from the data.

All the available model selection methods have both advantages and shortcomings. Currently there is not a fit for all method. In general, the predictive likelihood on the testing data is a basic metric that was applied in most studies. But in addition to that, the inference result given by topic models should also be evaluated from other perspectives, depending on the research goals.

1.6.2 Incorporation of domain knowledge into topic models

Medical data in biobanks are often high-dimensional and sparse, which makes the statistical inference very challenging. As a result, pure data-driven approaches may face great difficulties, especially when the input data is noisy and not large enough. On the other hand, during the past decades, people have accumulated large amounts of medical knowledge, which is undoubtedly valuable external information to be exploited. Therefore, incorporating people's prior knowledge into the modelling process should benefit the inference for statistical models.

In recent years, efforts have been put into encoding domain knowledge as priors for hidden variables (topics) of topic models to boost the quality of inference [99, 100]. For the study of multi-morbidity, such priors should reflect the relationship of different diseases, and more specifically for topic models, which diseases are more likely to be the active components in the same topic. Medical ontology built by experts based on their current understanding of the diseases, like the ICD-10 disease coding system [101], can be an ideal choice for constructing informative prior for the structure of topics. Medical ontology usually uses a hierarchical structure to summarize the subordinative relationships of a large number of medical concepts (such as diseases). Take the ICD-10 coding system as an example, tens of hundreds of diseases are firstly divided into dozens of chapters (major categories of human diseases such as infectious diseases, cancers and metabolic diseases). Then layer by layer, increasingly smaller categories of diseases are defined, until individual diseases (or subtypes of individual diseases) are reached (Figure 1.8). In general, diseases that are similar to each other in a broad sense will also be more closely positioned on the hierarchical structure dictated by the medical ontology. With this hierarchical structure, priors for topics can then be constructed using various mathematical tools, with domain knowledge encoded in the form of probabilities.

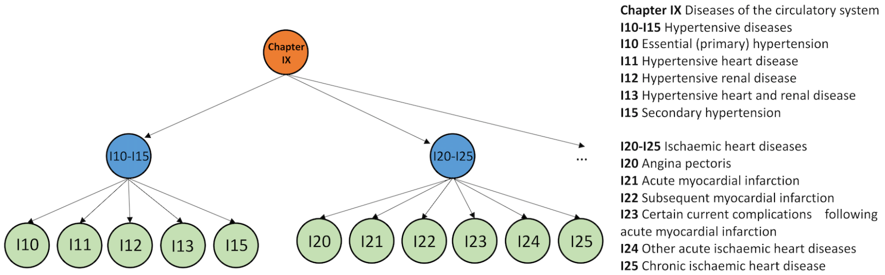

Chapter IX
I10-I15
I20-I25
...
I10
I11
I12
I13
I15
I20
I21
I22
I23
I24
I25
Chapter IX Diseases of the circulatory system
I10-I15 Hypertensive diseases
I10 Essential (primary) hypertension
I11 Hypertensive heart disease
I12 Hypertensive renal disease
I13 Hypertensive heart and renal disease
I15 Secondary hypertension
I20-I25 Ischaemic heart diseases
I20 Angina pectoris
I21 Acute myocardial infarction
I22 Subsequent myocardial infarction
I23 Certain current complications following acute myocardial infarction
I24 Other acute ischaemic heart diseases
I25 Chronic ischaemic heart disease

Figure 1.8: Schematic for the ICD-10 coding system. A very small fraction of the entire hierarchical structure for the ICD-10 coding system is plotted.

Quite a few studies have succeeded in improving the inference for statistical models with priors defined on the basis of medical ontology. For instance, medical ontology was used as a knowledge graph to instruct the learning of embeddings for medical concepts, and this helped with the data insufficiency and interpretation issues for deep-learning algorithms [102]. In another study, researchers used distances between diseases defined on the ICD-10 ontology to regularize topics learned from EHR data by a non-parametric Bayesian topic model named "wddCRF". With the use of decay function, it was more likely for words to be assigned with the same topic if their distance on the ICD-10 ontology was small [103]. For genetic studies, "treeWAS" was designed to increase the statistical power for association study using the hierarchical phenome data in biobanks. It assumed a correlation structure for the effect sizes of a variant on all diseases according to the ICD-10 ontology [104]. Overall, these studies all implied that priors based on medical oncology can significantly improve inference of hidden variables from the high-dimensional and sparse phenotypic data in biobanks.

1.6.3 The learning of hyperparameters for topic models

The main hyperparameters for topic models are the parameters of the prior distributions for topic weights (vector α) and topics (vector β). They control the distribution of people's weights over different topics and the probability of diseases in topics. For both LDA and NBMF, these prior distributions are usually Dirichlet distributions (or Beta distributions, which are Dirichlet distributions with a dimension of two), since they are conjugate priors for multinomial (Binomial) distributions (topics, topic weights for people).

Take α (parameters of the Dirichlet prior for topic weights) as an example, $\alpha = (\alpha_1, \alpha_2, ...\alpha_k)$ can be parameterized as below:

$$t = \sum_k \alpha_k$$
$$m = \alpha/t$$

m is the base measure (mean distribution) of the Dirichlet distribution, and t is the concentration parameter (scale). For standard topic models, normally a symmetric base m is used for the Dirichlet prior α, which means the numbers of assignments of different topics to words in the corpus are assumed to be roughly the same. As for the concentration parameter t, it controls how peaked the Dirichlet distribution is (Figure 1.9). In most cases, the concentration parameter is set to be 1 (non-informative), or a simple grid search can be used to find its optimal value (for instance: try 0.1, 1, 10 for t). There are also heuristic rules to choose values for these hyperparameters [105].

In recent years, there are studies exploring the use of more complex and appropriate priors for topic models. For instance, Wallach et al advocated using asymmetric Dirichlet prior for topic weights and symmetric prior for topics, and he verified with experiments that this parameterization gave the best inference result for LDA [106]. Besides, he also suggested the use of an optimization based method to learn α from the data, since it gave result as good as that achieved by employing a more complex and time consuming full Bayesian approach.

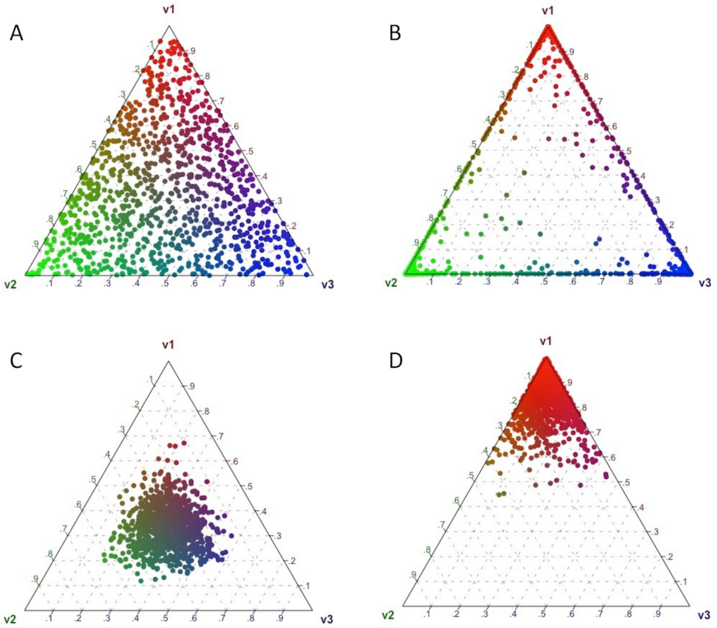


Figure 1.9: Distribution of 1,000 samples from Dirichlet distributions with different parameters. A, Result for Dirichlet(1,1,1). B, Dirichlet(0.1,0.1,0.1). C, Result for the Dirichlet(10,10,10). D, Dirichlet(10,1,1);

1.7 Research goals for the topic modeling of phenotypic data in biobanks

In spite of the success of the previous studies, several issues regarding the use of topic models to analyze phenotypic data in biobanks still remain, and are summarized as follows: Firstly, LDA, the most widely used topic model for biomedical data, takes word-count matrix as input. However, many phenotypic datasets in biobanks are binary. Although there are also topic models specifically designed for binary data, they haven't been widely used in biomedical studies. Secondly, the inference for LDA is entirely data-driven, without the incorporation of the prior knowledge about topics. Thirdly, a few technical problems related to topic models have not been completely solved, including the model selection problem for topic models and the learning of hyperparameters for topic models. Although advanced methods have been developed to deal with these issues [105–107], in biomedical studies they haven't been widely adopted and validated. Lastly, the most efficient way to carry out the downstream analyses (such as the association study) based on the inference results given by topic models is still unknown. In summary, further explorations are needed to answer these questions, before topic modeling algorithms can be widely accepted as one of the standard ways in analyzing the phenotype data in biobanks.

In this thesis, I aim to develop a new topic model to learn individuals' major multi-morbidity clusters in the form of topics of diseases. Meanwhile, the technical issues related to the current mainstream topic models should be properly dealt with. After that, I will explore ways to carry out further analyses based on the inference result given by the topic model to advance our understanding of the mechanisms of multi-morbidity clusters, with an emphasis put on studying the genetic basis of multi-morbidity.

1.8 HLA and human diseases

1.8.1 Genotype data for GWAS

Various types of genotype data can be used for GWAS, including single nucleotide polymorphism (SNP) data, copy number variants and sequence variations. The most widely used data type for GWAS is the SNP data given by microarrays genotyping [24]. In recent years, with the decrease of cost for the next generation sequencing (NGS), we have seen an increasing trend of using NGS data for GWAS, which allows us to find more rare variants and their associations. In human's genome, the HLA gene complex is a special region, in which the common analytic strategies for GWAS do not work very well, because of the high variability and complex structure of this region. In the following sections, genetic variations in this region and their wide association with phenotypes will be briefly introduced. Then the emphasis will be put on using in-silico algorithms and the NGS data to determine people's genetic variations in this region, which is a unavoidable step towards a better understanding of this region and its implications for human's health.

1.8.2 The HLA gene complex

The Human Leukocyte Antigen (HLA), the human's Major Histocompatibility Complex (MHC), is a gene complex in the 6p21.3 region on the short arm of human chromosome 6 [108]. More than 220 genes of diverse functions are located in this region, playing pivotal roles in the functioning of the human's immune system [108], which makes it the most gene dense region in human's genome [109]. The HLA gene complex can be divided into three regions. The class I region contains HLA*A, HLA*B and HLA*C genes, while the class II region contains HLA*DPA1, HLA*DPB1, HLA*DQA1, HLA*DQA2, HLA*DQB1, HLA*DQB2 and HLA*DRB1-5 genes. The major biological role for the two classes of genes is to present processed antigens to immune cells. The class III region doesn't contain HLA genes. However, it is the most genes enriched region in the HLA gene complex, since many genes implicated in the inflammatory response, leukocyte maturation and complement cascade are located in this region [110, 111].

The HLA gene complex is the most polymorphic region in the human genome. However, in spite of the diversity of HLA genes, the recombination rate in this region is low [112], therefore HLA genes are usually inherited as haplotypes [113]. Up to now, the official repository for curated HLA sequences, the IPD-IMGT/HLA database [114], have recorded 30,522 different HLA alleles. In addition to the small genetic variations, there are also large structural variations in the HLA region. In the class II region, people can have different copy numbers for genes that are paralogous to the HLA*DRB1 gene, which include HLA*DRB3, HLA*DRB4 and HLA*DRB5. For example, some people only have the HLA*DRB1 gene, while other people can have both the HLA*DRB1 and the HLA*DRB3 genes [111].

Genetic variations in the HLA gene complex have large influence on human's health. By far, it is the genomic region with the largest number of associated diseases in the GWAS catalog (constitute 6.4 % of all genome-wide significant SNP associations) [109]. During the past decades, genetic variations in the HLA complex were found to be associated with various medical conditions [115], including autoimmune diseases [116], infectious diseases [117], organ transplantation rejection [118], adverse drug effect [119] as well as cancer and patients' responses to immunotherapy [110, 120, 121]. Nevertheless, there is still much work to be done before we can fully understand the mechanisms behind all these associations.

1.8.3 Nomenclature of HLA alleles

The diversity of HLA alleles requires a hierarchical system to precisely classify and name them. The WHO Nomenclature Committee of Factors for the HLA System is in charge of the naming of HLA alleles [122]. Currently, HLA alleles are named using 2-4 fields of numbers. The first two fields are used to name HLA alleles encoding different amino acid sequences. The third field is used for HLA alleles with the same amino acid sequence but different silent mutations in the coding area of the genes, while the last field (the fourth field) is used for variations in the non-coding region of the genes. In addition, HLA alleles with abnormal expressions have a suffix, which is a single letter used for expression variants. Figure 1.10 describes the nomenclature rules for HLA alleles. It is important to understand the nomenclature of HLA alleles because different HLA allele inference algorithms (see the following sections) output HLA alleles at different resolutions (HLA alleles with different number of fields to name them).

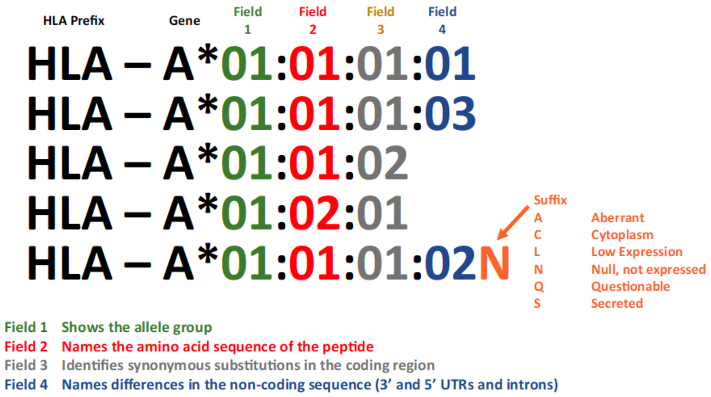

HLA Prefix
Gene
Field 1
Field 2
Field 3
Field 4
HLA – A*01:01:01:01
HLA – A*01:01:01:03
HLA – A*01:01:02
HLA – A*01:02:01
HLA – A*01:01:01:02N
Suffix
A Aberrant
C Cytoplasm
L Low Expression
N Null, not expressed
Q Questionable
S Secreted
Field 1 Shows the allele group
Field 2 Names the amino acid sequence of the peptide
Field 3 Identifies synonymous substitutions in the coding region
Field 4 Names differences in the non-coding sequence (3' and 5' UTRs and introns)

Figure 1.10: Nomenclature of HLA alleles. Up to four fields of digits can be used to name a HLA allele. Different fields are used to represent different types of variations for HLA alleles. (Source: Lee Ann Baxter-Lowe, 2021)

1.9 Inference of HLA alleles

There are many different methods to determine which alleles people have for different HLA genes, including both experimental and computational methods. In history, serology was firstly used for typing HLA alleles, which utilizes antibodies that can recognize epitopes of antigens on the HLA proteins. Later, DNA based typing methods that determine the nucleotide sequences of genes replaced serology methods. Early DNA-based typing methods use polymerase chain reaction (PCR) to amplify specific segments of HLA genes. This method can only detect a few key sequence motifs, which are then used to assign HLA genotypes. Later, sanger based typing (SBT) appeared, which is based on nucleotide sequencing technique. Although usually SBT still couldn't cover the full length of HLA genes (in most cases only exons in the antigen recognition domain are sequenced), it quickly became the golden standard for HLA typing since it has higher resolution than the PCR based methods. However, it is important to note that both SBT and early genotyping based methods have the problem called HLA typing ambiguities, because sometimes multiple HLA alleles in the reference HLA database (such as the IPD-IMGT/HLA database) can be matched to the same typing result [123].

In the past decade, Biobanks were established in many countries, and array based genotyping were done for millions of people. However, genotyping in the HLA complex is challenging because of the high polymorphism, the large structural variations and homology between HLA genes in this region [111]. Besides, interpretation of GWAS hits in this region is also difficult due to the complex Linkage Disequilibrium structure. To deal with these challenges, many genotyping based statistical imputation methods for HLA alleles were developed [124–128], which uses the model built on a reference panel of people with both genotyping and HLA typing data to infer HLA alleles for people with only genotyping data in the imputation panel.

Except for array based genotyping, in the past decade the next generation sequencing (NGS) also gained its popularity quickly. NGS is substantially less expensive than Sanger based sequencing, thus in a short perior of time, large amounts of NGS data have been accumulated. For the HLA region, standard mapping and variant calling methods do not work well, therefore more dedicated algorithms for this highly polymorphic region were developed [129–133]. Among these methodologies, the use of the genome graph for the representation of diverse genomic sequences in the population was widely adopted. In the genome graph, nodes or edges are labelled with nucleotides, and traversals of the graph represent possible genomic sequences. Dilthey, et al firstly adopted the genome graph for the inference of HLA alleles, and developed a series of HLA typing algorithms based on short-read sequencing data [129–131]. Currently, the graph based inference of HLA alleles still have limitations, which is mainly caused by the lack of completely resolved HLA haplotypes that can be used as backbones of the genome graph [111]. In the future, such situation might be improved by long-read sequencing technology and haplotype resolved de novo assembly [111, 134]. Nevertheless, in silico HLA typing methods have already achieved inference accuracy comparable to more traditional sequencing based method such as SBT [135]. Compared to the imputation methods, NGS based methods are less dependent on sample ancestry, which undoubtedly broadens its applications [131]. A summary of different HLA typing methods were shown in Figure 1.11.

In summary, the establishment of biobanks all over the world, which give us access to the genotyping and short-read sequencing data for hundreds of thousands of people, together with the fast developing statistical HLA inference methods, have offered us an amazing opportunity for the characterization of genetic variations in the HLA gene complex. With accurately inferred HLA alleles and the deep phenotyping data in Biobanks, it is obvious that our understanding of the biological roles of HLA alleles will soon be brought to a new level.

1.10 Thesis aims

The overarching goal of this thesis is to develop a new framework to find the major multi-morbidity patterns and their underlying mechanisms using data in biobanks.

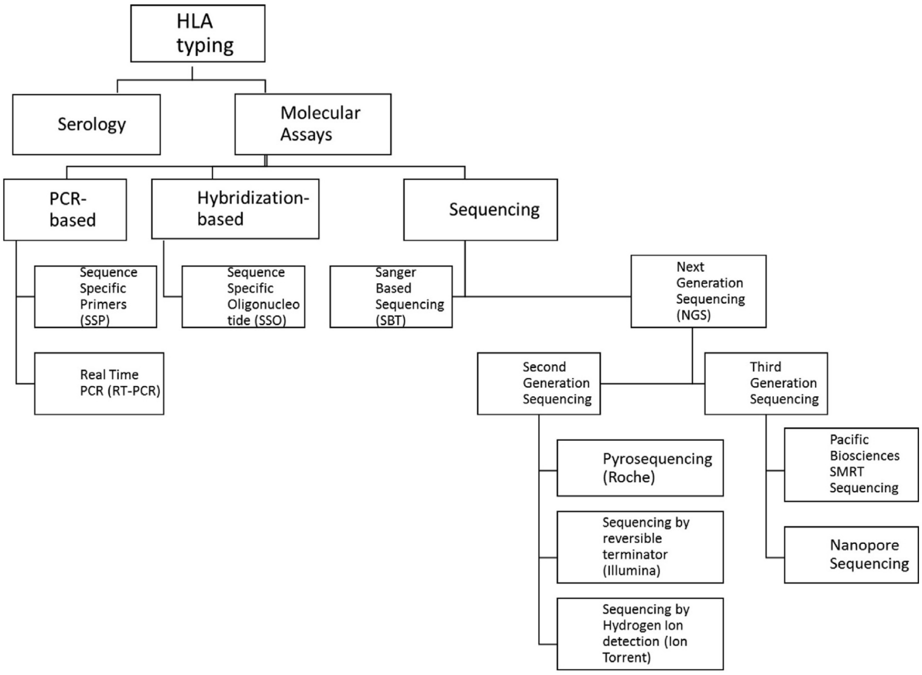

HLA typing
Serology
Molecular Assays
PCR-based
Hybridization-based
Sequencing
Sequence Specific Primers (SSP)
Real Time PCR (RT-PCR)
Sequence Specific Oligonucleotide (SSO)
Sanger Based Sequencing (SBT)
Next Generation Sequencing (NGS)
Second Generation Sequencing
Third Generation Sequencing
Pyrosequencing (Roche)
Sequencing by reversible terminator (Illumina)
Sequencing by Hydrogen Ion detection (Ion Torrent)
Pacific Biosciences SMRT Sequencing
Nanopore Sequencing

Figure 1.11: Summary of different HLA typing methods. (Source: Valia Bravo-Egana, 2021)

More specifically, I will firstly aim to develop a new topic model for the binary diagnosis data in UK Biobank, which also incorporates our prior knowledge about the structure of topics encoded in the hierarchical structure of medical ontology. In a second step, using the inference results (topics of diseases) ascertained using this topic model, various genetic analyses will be carried out to find the genetic components of these multi-morbidity clusters, and their biological implications.

Meanwhile, I aim to validate one of the state-of-art HLA typing algorithms, the HLA*LA, and use it to infer the HLA alleles for people in the UKB, such that in the future we can determine their relations with major multi-morbidity patterns. A summary of chapters follows.

In Chapter 2 we first introduce the topic model we developed (treeLFA) in details and the relevant algorithms for it. Then we comprehensively evaluate its performance with simulated data and investigate the role played by its hierarchical prior for topics. We also compare its performance with the most widely used topic model, LDA. In addition, we evaluate the robustness of treeLFA from a few different aspects.

In Chapter 3 we construct a relatively small input dataset (the top-100 UKB dataset) for treeLFA, which contains people's diagnosis data for 100 common ICD-10 codes. Topics of common diseases are learnt and analyzed, and the performance of several relevant topic models are compared. We also compare the inference result given by treeLFA models with different numbers of topics and study the connections between them. Lastly, we explore the possibility to subtype common diseases based on treeLFA's inference results.

In Chapter 4 we focus on carrying out genetic analyses using the inference result of treeLFA. The emphasis is on running GWAS on topic weights as new traits (topic-GWAS). Results given by topic-GWAS and standard single trait GWAS are compared, and new discoveries of topic-GWAS are validated using multiple methods. In the last section, we explore a different way to perform genetic analyses for topics other than running topic-GWAS.

In Chapter 5 we construct a larger input dataset for treeLFA (the top-436 UKB dataset), and repeat the core analyses done on the top-100 dataset. Results and findings are compared to that obtained on the top-100 dataset, and with new analyses more insights about the topics are gained.

In Chapter 6 we validate the HLA typing algorithm - HLA*LA with the 1000 Genomes dataset, using SBT data as the benchmark. Then we apply HLA*LA onto the WES data for about 50,000 people in the UKB. Typing results are summarized and presented and further compared with SNP-based imputation results.

In Chapter 7 we discuss the key findings, the limitations of the current work and the implications of results obtained on the two UKB datasets, as well as potential future directions.

2 Development and validation of treeLFA

Contents

2.1 Aims of this chapter

The major goal of this chapter is to develop a new topic model capable of learning multi-morbidity patterns of diseases in the form of topics of diseases from the cross-sectional health-care data such as that available from those population-scale biobanks. We built the model such that it makes use of both the data and our prior knowledge about the topics for inference. This prior knowledge for topics is the relationships of all the diseases codes summarized in the hierarchical structure specified by a disease classification system like the ICD-10 coding system. In addition to design and implement the model, we also aim to find reliable solutions to a few common technical issues related to topic models, including choosing the number of topics to be inferred (a model selection problem) and learning the hyperparameters for the model, as well as the identifiability issue of Gibbs sampling algorithms for topic models. After developing the model, we tested its performance with both simulated and real world data, and compared it with two other related topic models so that we could figure out the specific situations in which each of them is most advantageous against the others. Lastly, we tested the robustness of the model from a few different perspectives.

2.2 treeLFA

2.2.1 Model overview

Our topic model is named "treeLFA", which is the abbreviation for "latent factor allocation with a tree structured prior for topics". It retains the basic "document-topic-word" configuration of topic models, but these components are assigned with new meanings to analyze the phenotypic data in biobanks. For treeLFA, each individual in the biobank is viewed as a document, and each disease a word. Topics of diseases capture different constellations of diseases that frequently co-occur on the same individuals. The input for treeLFA is a $D \times S$ binary matrix, where each row corresponds to a person, and each column a disease code. Entries in the input matrix denote which disease codes were diagnosed for individuals. In total, D × S binary disease variables are used to record the diagnostic status of all diseases for all individuals. Like LDA, the output of treeLFA is two matrices: matrix of topics ($T \times S$, each entry is the probability of a disease in a topic), and the matrix of people's topic weights ($D \times T$, each row records an individual's weights for all topics).

The treeLFA is developed on the basis of the Bayesian mean-parameterized binary non-negative matrix factorization (BNMF)[83], which is specifically designed for analyzing sparse binary matrices, and is fundamentally different from the widely used topic model LDA [79, 80] (discussed in detail in Section 1.5.1). treeLFA models the presence and absence of disease codes for people with Bernoulli distributions. For treeLFA, each topic is a sequence of probabilities which parameterize the Bernoulli distributions for all diseases in this topic, instead of a Dirichlet distribution over all the diseases (LDA topics). Therefore, the model is named as latent factor allocation (LFA) to differentiate from the latent Dirichlet allocation (LDA). Furthermore, a prior for topics based on a hierarchical structure for all diseases is incorporated into the model, therefore the word "tree" is added into the model's name before "LFA". The schematic for the inference of treeLFA is shown in Figure 2.1.

To incorporate the prior for topics based on the tree structure of disease codes, for each disease code s in topic t, a binary indicator variable I_{ts} is introduced,

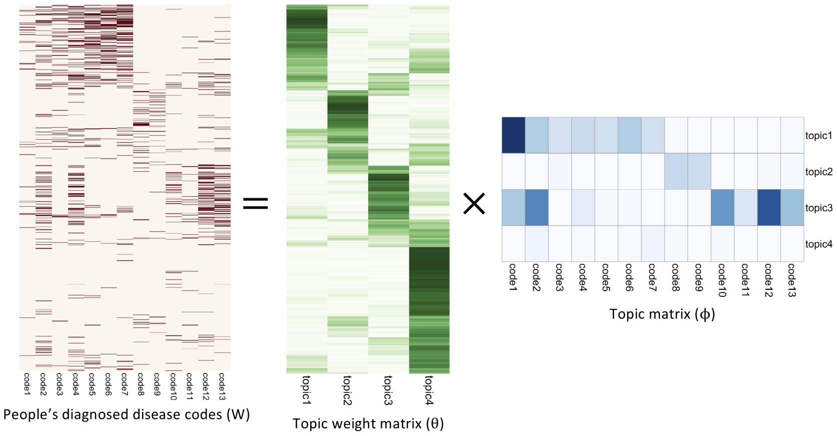
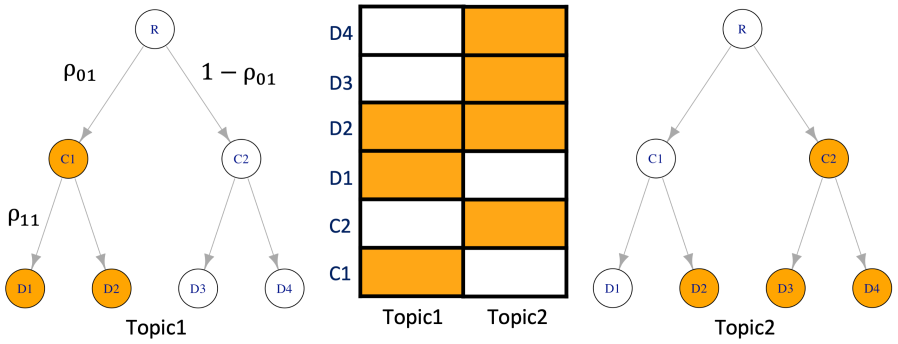


Figure 2.1: Schematic for the inference of treeLFA. The input for treeLFA is a binary matrix, with each row representing a person, and each column a disease code. The output of treeLFA are two matrices, one topic matrix and one topic weight matrix. Active disease codes in topics are visualized on a hierarchical structure of diseases, which is used to construct the prior for topics.

denoting whether disease code s is active in topic t. Active disease codes in a topic have relatively large occurrence probability (large ϕ_{ts}), while inactive disease codes have near-zero ϕ_{ts}. These are attained by putting different Beta priors ($Beta(a_0^0, a_0^1)$ or $Beta(a_1^0, a_1^1)$) on the ϕ_{ts} for inactive and active disease codes. Indicator variables of all disease codes in a topic are generated using a Markov process on the tree structure of disease codes. As is shown in Figure 2.2, each node on the tree corresponds to the indicator variable of a disease code in a topic. Starting from the inactive root node (does not correspond to any disease code), the value of an indicator variable is sampled conditioning on the value of its parent indicator variable on the tree. Going down from the root node to all the terminal nodes, indicator variables are sampled layer after layer. Notably, the sparsity of a topic (which means the total number of active codes in a topic) can be controlled by tuning the transition probabilities of the Markov process, just as what we can do with LDA by tuning the concentration parameter of the Dirichlet prior for topics. Furthermore, we can also manipulate the distribution of active codes in a topic by tuning (see Section 2.5.2 for the relevant discussions).



	Topic1	Topic2
D4	⬜	🟧
D3	⬜	🟧
D2	🟧	🟧
D1	🟧	⬜
C2	⬜	🟧
C1	🟧	⬜

Figure 2.2: Schematic for the Markov process on the tree structure of disease codes. The indicator variables for six disease codes in two topics are shown in the middle of the figure. Each cell in the table corresponds to the indicator variable of a disease code in a topic, with white color representing inactive code, and orange representing active code. The first topic has three active codes, and the second topic has four. Indicator variables for all codes in a topic are generated using a Markov process on the tree structure of codes. ρ_{01} and ρ_{11} are the transition probabilities of this Markov process: ρ_{01} is the probability of getting an active indicator conditioned on an inactive parent indicator, while ρ_{11} is the probability of getting an active indicator conditioned on an active parent indicator.

The model can be more clearly explained by going through its generative process. To generate a topic, we firstly sample the indicator variables $\boldsymbol{I}$ for all disease codes in this topic using a Markov process on the tree structure of all disease codes. Next, conditioned on these indicator variables, we sample probability variable $\boldsymbol{\phi}$ for all disease codes in the topic from the corresponding Beta priors. The topic weight variable $\boldsymbol{\theta}$ for an individual is sampled from the Dirichlet distribution parameterized by $\boldsymbol{\alpha}$. With topics and topic weights, the observable disease variable $\boldsymbol{W}$ (the input binary matrix) can then be generated by sampling from the Bernoulli distributions paramterized by the product of $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$. The full likelihood of the model can be decomposed as below:

$$\begin{aligned}&\boldsymbol{P(W, \theta, \phi, I \mid \rho, \alpha, a, tree)}\\ =&\boldsymbol{P(I \mid tree, \rho) \cdot P(\phi \mid I, a) \cdot P(\theta \mid \alpha) \cdot P(W \mid \phi, \theta)}\end{aligned}$$

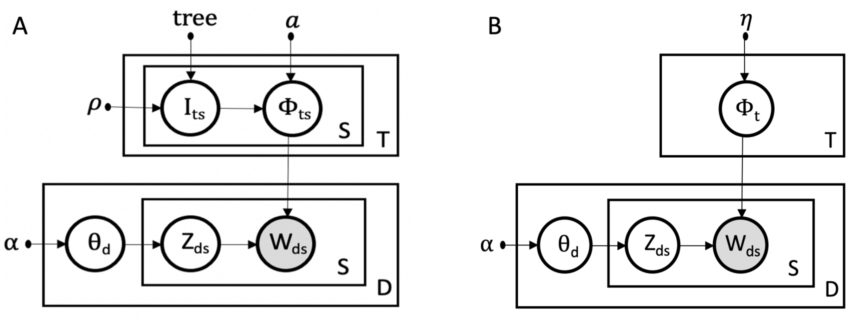


Figure 2.3: A, The probabilistic graphical model for treeLFA. Only the disease variables for people are observable. Each topic has a probability vector $\boldsymbol{\phi}$ for the $\boldsymbol{S}$ disease codes, and each probability variable $\boldsymbol{\phi}$ has a corresponding indicator variable $\boldsymbol{I}$, denoting if this disease code is active in the topic. $\boldsymbol{I}$ for all disease codes in a topic are generated using a Markov process on the tree structure with transition probability vector $\boldsymbol{\rho}$. Different Beta priors $\boldsymbol{a}$ are used for the $\boldsymbol{\phi}$ of active/inactive codes. To generate the disease variable $\boldsymbol{W_{ds}}$ for disease code $\boldsymbol{s}$ for person $\boldsymbol{d}$, its topic assignment variable $\boldsymbol{Z_{ds}}$ is firstly sampled from the categorical distribution parameterized by this person's topic weight vector $\boldsymbol{\theta_d}$, which itself is sampled from a Dirichlet prior distribution parameterized by vector $\boldsymbol{\alpha}$. With the topic assignment and topics, the disease variable $\boldsymbol{W_{ds}}$ can then be sampled from the Bernoulli distribution parameterized by $\boldsymbol{\phi_{Z_{ds},s}}$. B, The graphical model for LDA.

To perform inference for treeLFA using MCMC algorithms, we further augment the model using topic assignment variable $\boldsymbol{Z}$. For disease code $\boldsymbol{s}$ and individual $\boldsymbol{d}$, the topic assignment variable $\boldsymbol{Z_{ds}}$ specifies the topic from which the disease variable $\boldsymbol{W_{ds}}$ is sampled. $\boldsymbol{Z_{ds}}$ is sampled from the categorical distribution parameterized by $\boldsymbol{\theta_d}$. In this way, the model become conjugate, since $\boldsymbol{\theta_d}$ and $\boldsymbol{Z_d}$ follow a compound Dirichlet-Multinomial distribution.

The graphical models for treeLFA and LDA are shown in Figure 2.3. It can be seen that the two models use different topics, but the generative process for topic weight variable $\boldsymbol{\theta}$ and topic assignment variable $\boldsymbol{Z}$ (the lower half of the two graphical models) are the same for the two models. Notations for treeLFA are listed in Table 2.1.

The full treeLFA model is specified as below:

$$\begin{aligned}
&\boldsymbol{P(W, \theta, Z, \phi, I \mid \rho, \alpha, a, tree)}\\
=&\boldsymbol{P(I \mid tree, \rho) \cdot P(\phi \mid I, a) \cdot P(\theta \mid \alpha) \cdot P(Z \mid \theta) \cdot P(W \mid \phi, Z)}
\end{aligned}$$

$$\begin{aligned}
&\boldsymbol{P(I \mid tree, \rho) = \prod_{t=1}^{T} \prod_{s=1}^{S} P(I_{ts} \mid I_{ts}^{pa}, \rho)}\\
&\boldsymbol{P(\phi \mid I, a) = \prod_{t=1}^{T} \prod_{s=1}^{S} P(\phi_{ts} \mid I_{ts}, a)}\\
&\boldsymbol{P(\theta \mid \alpha) = \prod_{d=1}^{D} P(\theta_d \mid \alpha)}\\
&\boldsymbol{P(Z \mid \theta) = \prod_{d=1}^{D} \{ \prod_{s=1}^{S} P(Z_{ds} \mid \theta_d) \}}\\
&\boldsymbol{P(W \mid \phi, Z) = \prod_{d=1}^{D} \prod_{s=1}^{S} P(W_{ds} \mid \phi_{Z_{ds}, s})}\\
&\boldsymbol{P(\theta_d \mid \alpha) \sim Dirichlet(\alpha)}\\
&\boldsymbol{P(Z_{ds} \mid \theta_d) \sim Categorical(\theta_d)}\\
&\boldsymbol{P(I_{ts} \mid I_{ts}^{pa} = 0, \rho) \sim Bernoulli(\rho_{01})}\\
&\boldsymbol{P(I_{ts} \mid I_{ts}^{pa} = 1, \rho) \sim Bernoulli(\rho_{11})}\\
&\boldsymbol{P(\phi_{ts} \mid I_{ts} = 0, a) \sim Beta(a_0^0, a_0^1)}\\
&\boldsymbol{P(\phi_{ts} \mid I_{ts} = 1, a) \sim Beta(a_1^0, a_1^1)}\\
&\boldsymbol{P(W_{ds} \mid \phi_{z_{ds}, s}) \sim Bernoulli(\phi_{z_{ds}, s})}
\end{aligned} \tag{2.1}$$

Notation	Explanation
D	total number of people in the input data
S	total number of disease codes to be analyzed
T	total number of topics to be inferred
d	person d
s	disease code s
t	topic t
$\boldsymbol{Z_{ds}}$	topic assignment variable for disease code s for person d
$\boldsymbol{\phi_{ts}}$	probability variable for disease code s in topic t
$\boldsymbol{I_{ts}}$	indicator variable for disease code s in topic t
$\boldsymbol{I_{ts}^{pa}}$	indicator variable for the parent disease code of disease code s in topic t
$\boldsymbol{I_{ts}^{ch}}$	indicator variables for all children disease codes of disease code s in topic t
$\boldsymbol{\theta_d}$	topic weight vector for person d
$\boldsymbol{W_{ds}}$	binary disease variable for disease code s for person d
$\boldsymbol{tree}$	the fixed hierarchical structure for disease codes specified by a disease classification system
$\boldsymbol{\rho_{01}}$	transition probability of the Markov process on the tree of getting an active code given an inactive parent code
$\boldsymbol{\rho_{11}}$	transition probability of the Markov process on the tree of getting an active code given an active parent code
$\boldsymbol{\alpha}$	parameter vector for the Dirichlet prior for $\boldsymbol{\theta}$
$\boldsymbol{a_0^0, a_0^1}$	parameters of the Beta prior for $\boldsymbol{\phi}$ of inactive codes
$\boldsymbol{a_1^0, a_1^1}$	parameters of the Beta prior for $\boldsymbol{\phi}$ of active codes

Table 2.1: Notations for "treeLFA"

2.2.2 Inference of treeLFA with Gibbs sampling

We use a partial collapsed Gibbs sampler [83, 136] to sample from the posterior distributions of the latent variables of treeLFA. The topic weight variable $\boldsymbol{\theta}$ is integrated out to achieve better mixing for the the Markov Chains, which is now the standard method of performing Gibbs sampling for topic models. It may also be possible to integrate out $\boldsymbol{I}$ or $\boldsymbol{\phi}$, but we haven't explore these options.

Derivations for the updating equations of different hidden variables are as follows:

Topic assignment variable for disease code s for person d: Z_{ds}

$$
\begin{aligned}
P(Z_{ds} = t* \mid Z_d^{\neg s}, \phi, I, W) \propto \int & P(Z_{ds} = t*, Z_d^{\neg s}, \phi, I, W, \theta) d\theta \\
& \propto P(W_{ds} \mid Z_{ds} = t*, \phi) \cdot \int P(\theta_d \mid \alpha) \cdot P(Z_d \mid \theta_d) d\theta_d \\
& = P(W_{ds} \mid Z_{ds} = t*, \phi) \cdot \int [(\frac{\Gamma(\sum_{k=1}^{K} \alpha_k)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \prod_{k=1}^{K} \theta_{dk}^{\alpha_k - 1}) \\
& \prod_{s=1}^{S} \theta_{d,Z_{ds}}] d\theta_d \\
& = P(W_{ds} \mid Z_{ds} = t*, \phi) \cdot \int [\frac{\Gamma(\sum_{k=1}^{K} \alpha_k)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \prod_{k=1}^{K} \theta_{dk}^{\alpha_k - 1} \\
& \prod_{k=1}^{K} \theta_{dk}^{c_{dk}}] d\theta_d \\
& = P(W_{ds} \mid Z_{ds} = t*, \phi) \cdot \frac{\Gamma(\sum_{k=1}^{K} \alpha_k)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \cdot \frac{\prod_{k=1}^{K} \Gamma(\alpha_k + c_{dk})}{\Gamma(\sum_{k=1}^{K} (\alpha_k + c_{dk}))} \\
& \propto P(W_{ds} \mid Z_{ds} = t*, \phi) \cdot \frac{\prod_{k=1}^{K} \Gamma(\alpha_k + c_{dk})}{\Gamma(\sum_{k=1}^{K} (\alpha_k + c_{dk}))} \\
& = P(W_{ds} \mid Z_{ds} = t*, \phi) \cdot \frac{\prod_{k \neq t*} \Gamma(\alpha_k + c_{dk}^{\neg s})}{\Gamma([\sum_{k=1}^{K} (\alpha_k + c_{dk}^{\neg s})] + 1)} \\
& \cdot \Gamma(\alpha_{t*} + c_{dt*}^{\neg s} + 1) \\
& = P(W_{ds} \mid Z_{ds} = t*, \phi) \cdot \frac{\prod_{k} \Gamma(\alpha_k + c_{dk}^{\neg s})}{\Gamma([\sum_{k=1}^{K} (\alpha_k + c_{dk}^{\neg s})] + 1)} \\
& \cdot (\alpha_{t*} + c_{dt*}^{\neg s}) \\
& \propto \phi_{t*s}^{W_{ds}} \cdot (1 - \phi_{t*s})^{1 - W_{ds}} \cdot (\alpha_{t*} + c_{dt*}^{\neg s})
\end{aligned}
$$

$$
P(Z_{ds} = t* \mid Z_d^{\neg s}, \phi, I, W) = \frac{\phi_{t*s}^{W_{ds}} \cdot (1 - \phi_{t*s})^{1 - W_{ds}} \cdot (\alpha_{t*} + c_{dt*}^{\neg s})}{\sum_{k=1}^{K} [\phi_{ks}^{W_{ds}} \cdot (1 - \phi_{ks})^{1 - W_{ds}} \cdot (\alpha_k + c_{dk}^{\neg s})]}
$$

$Z_d^{\neg s}$: topic assignment variables for all disease codes except for disease code s for person d.

c_{dk}: the total number of disease variables assigned with topic k for person d.

$c_{dk}^{\neg s}$: the total number of disease variables except for disease variable s that are assigned with topic k for person d.

To sample from this categorical distribution, for each disease variable W_{ds} we need to calculate the numerator $\phi_{t*s}^{W_{ds}} \cdot (1 - \phi_{t*s})^{1 - W_{ds}} \cdot (\alpha_{t*} + c_{dt*}^{\neg s})$ in the above equation for all topics, and then normalize them. However, we can save time by avoiding repetitive computations. Specifically, depending on whether the disease variable W_{ds} is zero or one, only one of $\phi_{t*s}^{W_{ds}}$ and $(1 - \phi_{t*s})^{1 - W_{ds}}$ needs to be calculated. Take $\phi_{t*s}^{W_{ds}}$ as an example, when $W_{ds} = 1$ we need to calculate $\phi_{t*s} \cdot (\alpha_{t*} + c_{dt*}^{\neg s})$, which equals $\phi_{t*s} \cdot \alpha_{t*} + \phi_{t*s} \cdot c_{dt*}^{\neg s}$. During the sampling of Z_{ds}, $\phi_{t*s} \cdot \alpha_{t*}$ are fixed. We can pre-calculate and cache these values for all t and s. This very simple trick can significantly reduce the time we spend on sampling Z.

Probability variable for disease code s in topic t: ϕ_{ts}

$$\begin{aligned}
P(\phi_{ts} \mid \cdot) \propto\ & P(W \mid Z, \phi_{ts}) \cdot P(\phi_{ts} \mid a, I_{ts}) \\
& \propto [\prod_{d=1}^{D} I(Z_{ds} = t) \cdot P(W_{ds} \mid \phi_{ts})] \cdot P(\phi_{ts} \mid a, I_{ts}) \\
& = [\prod_{d=1}^{D} I(Z_{ds} = t) \cdot (1 - \phi_{ts})^{1-W_{ds}} \cdot \phi_{ts}^{W_{ds}}] \cdot Beta(\phi_{ts} \mid a_{I_{ts}}^{0}, a_{I_{ts}}^{1}) \\
& \propto \phi_{ts}^{\sum_{d=1,Z_{ds}=t}^{D} W_{ds}} \cdot (1 - \phi_{ts})^{\sum_{d=1,Z_{ds}=t}^{D} (1-W_{ds})} \cdot (1 - \phi_{ts})^{a_{I_{ts}}^{1} - 1} \cdot (\phi_{ts})^{a_{I_{ts}}^{0} - 1}
\end{aligned}$$

$$P(\phi_{ts} \mid \cdot) \sim Beta(a_{I_{ts}}^{0} + N_{st}^{1}, a_{I_{ts}}^{1} + N_{st}^{0}). \tag{2.2}$$

N_{st}^{0}: for people who have W_{ds}=0, the total number of them whose disease variables s are assigned with topic t.
N_{st}^{1}: for people who have W_{ds}=1, the total number of them whose disease variables s are assigned with topic t.

$$\begin{aligned}
N_{st}^{0} &= \sum_{d=1}^{D} I(Z_{ds} = t, W_{ds} = 0). \\
N_{st}^{1} &= \sum_{d=1}^{D} I(Z_{ds} = t, W_{ds} = 1).
\end{aligned}$$

Indicator variable for disease code s in topic t: I_{ts}

$$\begin{aligned}
P(I_{ts} = 0 \mid \cdot) \propto\ & P(\phi_{ts} \mid I_{ts} = 0) \cdot P(I_{ts} = 0 \mid I_{ts}^{pa}) \cdot P(I_{ts}^{ch} \mid I_{ts} = 0) \\
& = P(\phi_{ts} \mid I_{ts} = 0) \cdot ((1 - \rho_{11})^{I_{ts}^{pa}} \cdot (1 - \rho_{01})^{1-I_{ts}^{pa}}) \cdot \prod_{I_{ts}^{ch}} \rho_{01}^{I_{ts}^{ch}} \cdot (1 - \rho_{01})^{1-I_{ts}^{ch}} \\
& = Beta(\phi_{ts} \mid a_{0}^{0}, a_{0}^{1}) \cdot ((1 - \rho_{11})^{I_{ts}^{pa}} \cdot (1 - \rho_{01})^{1-I_{ts}^{pa}}) \cdot \prod_{I_{ts}^{ch}} \rho_{01}^{I_{ts}^{ch}} \cdot (1 - \rho_{01})^{1-I_{ts}^{ch}}.
\end{aligned}$$

$$P(I_{ts} = 1 \mid \cdot) = Beta(\phi_{ts} \mid a_{1}^{0}, a_{1}^{1}) \cdot (\rho_{11}^{1-I_{ts}^{pa}} \cdot \rho_{01}^{1-I_{ts}^{pa}}) \cdot \prod_{I_{ts}^{ch}} \rho_{11}^{I_{ts}^{ch}} \cdot (1 - \rho_{11})^{1-I_{ts}^{ch}}. \tag{2.3}$$

$P(I_{ts} = 1 \mid \cdot)$ and $P(I_{ts} = 0 \mid \cdot)$ calculated using the above equations need to be normalized such that they sum to 1.

2.3 Model selection: evaluation of inferred topics on the testing dataset

As was introduced in Chapter 1, there are various methods to select an optimal value for the number of topics for treeLFA. Basically, we adopted the most widely used way to deal with this problem, which is to evaluate the inferred topics on a held-out testing dataset. This requires the calculation of the predictive likelihood on the testing dataset using the topics learnt from the training data by treeLFA.

There are a wide variety of algorithms to calculate the predictive likelihood for topic models, and Wallach's review has a comprehensive summary of them [92].

We didn't comprehensively evaluate most of the available algorithms, since the focus of our study is not to select one single best model in terms of the number of topics. Instead, we are more interested in comparing the inference results given by different models, and studying how the inference result change across them, or whether the inference result is stable across models. Because of this, after implementing and comparing a few basic and widely used algorithms, we chose a basic one to calculate the predictive likelihood for treeLFA models. It should be pointed out that this method might not be the most efficient one in calculating predictive likelihood for topic models, since it is likely that there exists better choices. However, this algorithm is easy to implement and it appears to work on small-scaled simulated datasets (Section 2.6.2), and it also serves as the basis for other more advanced sampling based methods (such as importance sampling based method and bridge sampling based method [137]). In Chapter 5, the limitation of this method and our corresponding solution will be further discussed.

2.3.1 Predictive likelihood on the testing dataset

In this section the method we used to calculate predictive likelihood on the testing dataset for treeLFA models is introduced. We used posterior samples of topics ($\boldsymbol{\phi}$) collected by running the Gibbs sampler on the training data to calculate the model's predictive likelihood on the testing data. With a testing dataset W' and a posterior sample of $\boldsymbol{\phi}$, the predictive likelihood on the testing dataset could be expressed as [138]:

$$\boldsymbol{P(W^{'} \mid W) = \prod_{d=1}^{D} P(W_{d}^{'} \mid \phi, \alpha)}$$

Ideally, calculation of the predictive likelihood requires integrating out all latent variables. Since this is analytically intractable, we used a Monte-Carlo approximation for this integral. For topic weight variable $\boldsymbol{\theta_d}$, we drew P samples from its prior distribution and use them to approximate the integral:

$$\begin{aligned}
&\boldsymbol{\theta_d^p \sim Dirichlet(\alpha)}\\
&\boldsymbol{P(W_d^{'} \mid \phi, \alpha) \approx \frac{1}{P}\sum_{p=1}^{P} P(W_d^{'} \mid \theta_d^p, \phi)}
\end{aligned}$$

Conditioned on the P samples of $\boldsymbol{\theta_d}$, we sum out all topic assignment variables $\boldsymbol{Z_{ds}}$:

$$\begin{aligned}
\boldsymbol{P(W_d^{'} \mid \theta_d^p, \phi)} &= \boldsymbol{\prod_{s=1}^{S}[\sum_{Z_{ds}=1}^{K} P(W_{ds}^{'}, Z_{ds} \mid \theta_d^p, \phi)]}\\
&= \boldsymbol{\prod_{s=1}^{S}[\sum_{Z_{ds}=1}^{K} P(W_{ds}^{'} \mid \phi_{Z_{ds},s}) \cdot P(Z_{ds} \mid \theta_d^p)]}
\end{aligned}$$

In summary, the predictive likelihood could be expressed as:

$$\boldsymbol{P(W' \mid W) \approx \prod_{d=1}^{D}\{\frac{1}{P}\sum_{p=1}^{P}\{\prod_{s=1}^{S}\{\sum_{Z_{ds}=1}^{K}[P(W'_{ds} \mid \phi_{Z_{ds},s}) \cdot P(Z_{ds} \mid \theta_d^p)]\}\}\}} \tag{2.4}$$

2.4 Learn hyperparameters of treeLFA

The choice of hyperparameters can have a large influence on the inference result. There are three hyperparameters for treeLFA, which are vector $\boldsymbol{\alpha}$ that parameterizes the Dirichlet prior for topic weight vector $\boldsymbol{\theta}$, the transition probability vector $\boldsymbol{\rho}$ for the Markov process on the tree structure, and parameter vector $\boldsymbol{a}$ for the Beta priors of the probability variables of disease codes in topics.

2.4.1 Transition probabilities of the Markov process on the tree structure

For transition probabilities $\boldsymbol{\rho}$ of the Markov process, we integrated the inference of them into the overall Gibbs sampling framework by putting Beta priors on them as follows:

$$\begin{aligned} \boldsymbol{P(\rho_{01}) \sim Beta(b_{00}, b_{01})} \\ \boldsymbol{P(\rho_{11}) \sim Beta(b_{10}, b_{11})} \end{aligned} \tag{2.5}$$

And updating equations for these transition probability variables are:

$$\begin{aligned} \boldsymbol{P(\rho_{01} \mid I) \sim Beta(N_1^0 + b_{00}, N_0^0 + b_{01})} \\ \boldsymbol{P(\rho_{11} \mid I) \sim Beta(N_1^1 + b_{10}, N_0^1 + b_{11})} \end{aligned} \tag{2.6}$$

$\boldsymbol{N_1^0}$: the total number of active indicator variables in all topics that have an inactive parent indicator variable on the tree.
$\boldsymbol{N_1^1}$: the total number of active indicator variables in all topics with an active parent indicator variable on the tree.
$\boldsymbol{N_0^0}$: the total number of inactive indicator variables in all topics with an inactive parent indicator variable.
$\boldsymbol{N_0^1}$: the total number of inactive indicator variables in all topics with an active parent indicator variables.

2.4.2 Beta priors for probability variables of disease codes in topics

In topics, the probability variables $\boldsymbol{\phi}$ for active and inactive disease codes were given different Beta priors such that inactive codes always have probability close to 0 while active codes have relatively large probabilities (such as probabilities in the range of 0.1 to 1). Parameters for the two prior distributions were chosen to meet these requirements. In Figure 2.4 we show the histograms of 10,000 samples drawn from Beta(0.3,80) and Beta(2,4), which are the prior distributions we chose for the probability variable $\boldsymbol{\phi}$ of active/inactive codes in topics. It is noteworthy that based on our experience with simulated data, samples of $\boldsymbol{\phi}$ for inactive codes shouldn't be too small compared to $\boldsymbol{\phi}$ for active codes, otherwise the hierarchical prior for treeLFA won't play an important role in the inference.

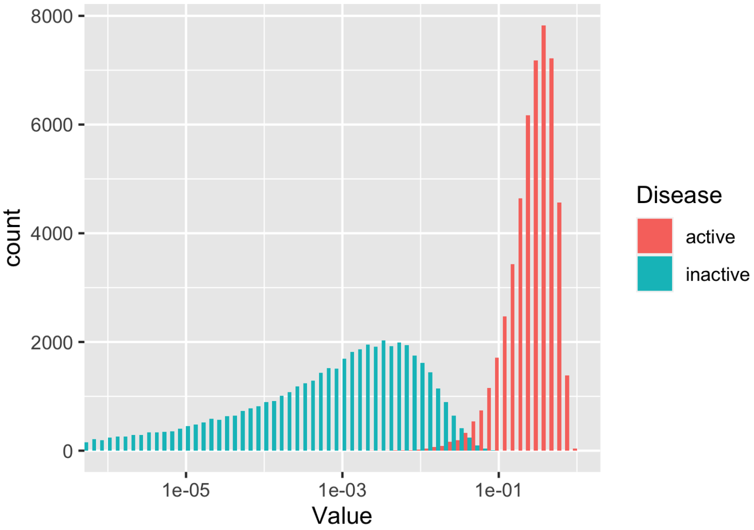


Figure 2.4: Beta priors for probabilities of codes in topics. Different Beta priors are used for the probability variable $\boldsymbol{\phi}$ for active/inactive codes in topics, such that active codes have large probability, while inactive codes have negligible small probability. 10,000 samples were drawn from Beta(0.3,80) and Beta(2,4) respectively, and their histograms were plotted.

2.4.3 Learn asymmetric Dirichlet prior for topic weight variable

Learning hyperparameter $\boldsymbol{\alpha}$ for the Dirichlet prior of the topic weight variable $\boldsymbol{\theta}$ is not straightforward due to the lack of conjugate prior for Dirichlet distribution. For LDA it is common to use a symmetric Dirichlet prior, which implies there are roughly same number of words in all documents assigned with different topics. However, the incidence of different diseases in biobanks varies largely, since there are both rare diseases and very common ones such as essential hypertension (28.2 % people in UKB have it). Under such condition, a symmetric Dirichlet prior may not be appropriat, since different topics can be used at quite different frequencies. To deal with this issue, the use of more complex Dirichlet prior for topic models have been studied by researchers. Wallach et al advocated using asymmetric Dirichlet prior for topic weight variables $\boldsymbol{\theta}$ and symmetric prior for topics $\boldsymbol{\phi}$, and he verified with experiments that this strategy gave the best inference result for LDA [106]. Besides, he also suggested the use of an optimization based method to learn $\boldsymbol{\alpha}$ from the data, since it gave result as good as that can be achieved by employing a more complex and time consuming full Bayesian approach.

We used a stochastic EM algorithm, named “Gibbs-EM” to optimize $\boldsymbol{\alpha}$ [105, 139]. The objective function of the algorithm is the marginal likelihood of the data: $\boldsymbol{P(W \mid \alpha)}$. This marginal likelihood is maximized with respect to $\boldsymbol{\alpha}$ by alternating between two steps. In the E-step, we fix the value of $\boldsymbol{\alpha}$ and maximize a lower bound of the marginal likelihood by approximating the posterior distribution of hidden variables with $\boldsymbol{G}$ samples given by the Gibbs sampler for treeLFA:

$$\begin{aligned} \boldsymbol{logP(W \mid \alpha)} &\geq \sum_{\boldsymbol{Z}} \boldsymbol{P(Z \mid W, \alpha) \cdot logP(W, Z \mid \alpha)} \\ &\approx \frac{\boldsymbol{1}}{\boldsymbol{G}} \sum_{\boldsymbol{g=1}}^{\boldsymbol{G}} \boldsymbol{logP(W, Z^g \mid \alpha)} \\ &= \frac{\boldsymbol{1}}{\boldsymbol{G}} \sum_{\boldsymbol{g=1}}^{\boldsymbol{G}} [\sum_{\boldsymbol{d=1}}^{\boldsymbol{D}} \boldsymbol{logP(W_d, Z_d^g \mid \alpha)}]. \end{aligned} \tag{2.7}$$

In the above equation, hidden variables $\boldsymbol{\phi}$, $\boldsymbol{I}$ and $\boldsymbol{\theta}$ are omitted, and only $\boldsymbol{Z}$ is retained, since it is the only hidden variable related to the optimization of $\boldsymbol{\alpha}$ in the M-step.

In the M-step, we use a fixed point method to optimize $\boldsymbol{\alpha}$ such that the lower bound calculated in the E-step is maximised. Entries of $\boldsymbol{\alpha}$ are optimized one at a time:

$$\boldsymbol{\alpha_t^{new} = \alpha_t \cdot \frac{\sum_g \sum_d \{\Psi(N_{dt}^g + \alpha_t) - \Psi(\alpha_t)\}}{\sum_g \sum_d \{\Psi(S + \sum_t(\alpha_t)) - \Psi(\sum_t(\alpha_t))\}}}. \tag{2.8}$$

$\boldsymbol{N_{dt}^g}$: the total number of disease variables for person $\boldsymbol{d}$ that are assigned with topic t in posterior sample g.
$\boldsymbol{\Psi()}$: the digamma function.

We made a few technical adjustments to the original Gibbs-EM algorithm. Firstly, to speed up the computation, for the first several thousand iterations of the Gibbs-EM algorithm we only collected one posterior sample of all hidden variables in the E-step instead of multiple posterior samples. Then for the following hundreds of iterations, we collected ten posterior samples of hidden variables in the E-step so that the optimization of $\boldsymbol{\alpha}$ in the M-step can be more accurate. Secondly, in the E-step we always ran the Gibbs sampler for 19 burn-in iterations before we collected one posterior sample of hidden variables. Lastly, we set a lower bound of 0.01 and an upper bound of 5 for entries in $\boldsymbol{\alpha}$ in the M-step to avoid numerical underflow or overflow. This can also be understood as putting a truncated uniform prior on $\boldsymbol{\alpha}$ and finding its MAP estimation.

2.5 Validation of treeLFA with simulated data

2.5.1 Overview of the goals of simulation

With the treeLFA model developed and implemented, the next step is to evaluate its performance with simulated data. In addition to verify that topics can be correctly inferred by treeLFA, we also want to investigate if the two distinguishing features of treeLFA (the hierarchical prior for topics and the binary non-negative matrix factorization framework) are of values to the inference. To study the role played by the hierarchical prior for treeLFA, we built another topic model, named "flatLFA", and compared its performance with treeLFA. flatLFA has the same model space as treeLFA, and the only difference is that it uses an non-informative prior for topics. In addition to flatLFA, we also included LDA for comparison, whose model configuration are different from treeLFA and flatLFA. By comparing the three models on various datasets, we figured out the situations in which each of them give the best performance.

In addition to the basic inference algorithm (Gibbs sampling for the hidden variables) for treeLFA, the model selection and hyperparameter learning algorithm (Gibbs-EM) are also validated. Lastly, we also aim to test the robustness of treeLFA from different perspectives, including the robustness of treeLFA to mis-specified hierarchical structure of disease codes and wrongly specified number of topics, as well as the behavior of treeLFA when the occurrences of diseases are completely independent (equivalent to input data generated with single code topics).

2.5.2 Simulation of input datasets

To generate input datasets, we firstly built a tree structure for 20 diseases codes (Figure 2.5). This tree structure has three layers. In the first layer there is only an inactive root node; the second layer has five nodes, and each of them has three children nodes in the third layer.

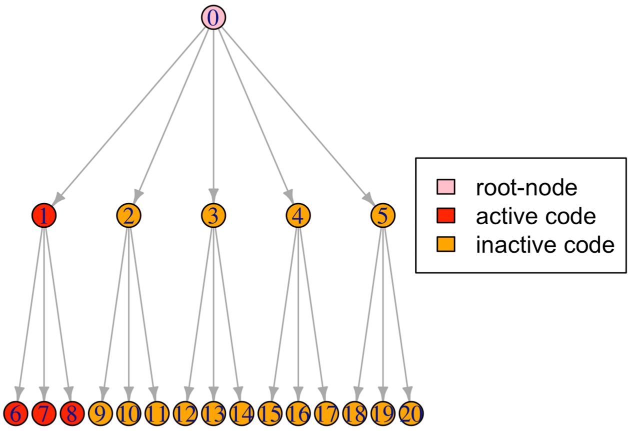


Figure 2.5: The tree structure of 20 disease codes for simulation. The 3-layered tree structure describes the hierarchical relationship of 20 disease codes. The first layer only has a root node. Each node in the second layer has three children nodes in the third layer. The red color highlights active codes in a topic.

With the tree structure, the next step is to generate topics of disease codes. Before introducing the topics we used to simulate the input data for the three topic models (treeLFA, flatLFA and LDA), it is necessary to discuss the relationship between the Markov process on the tree and the structure of topics (the total number and distribution of active codes in topics), since this is closely related to what kind of topics we should use for simulation, and how should we choose reasonable hyperparameters for the model.

A Markov process on the tree structure is used to generate indicator variables for disease codes in topics, thus the two transition probabilities of the Markov process, $\boldsymbol{\rho_{01}}$ and $\boldsymbol{\rho_{11}}$, controls the structure of topics. Choosing small values for both $\boldsymbol{\rho_{01}}$ and $\boldsymbol{\rho_{11}}$ give rise to very sparse topics (because most of the indicators will be inactive), while large values for both generate very dense topics. In addition, $\boldsymbol{\rho_{11}}$ also controls the distribution of active codes in topics. With a large $\boldsymbol{\rho_{11}}$, most children codes of an active parent code will also be active. As a result, active codes will form clusters on different branches of the tree. On the other hand, if $\boldsymbol{\rho_{11}}$ is small, active codes will be sporadically scattered across the whole tree. For our simulation, we preferred to use topics that are likely to be generated by a Markov process with small $\boldsymbol{\rho_{01}}$ and large $\boldsymbol{\rho_{11}}$. This is because for one thing we believe in the real world most topics of disease codes should be sparse (thus we chose small $\boldsymbol{\rho_{01}}$), and for another thing we also think it should be more likely for disease codes close to each other on the tree to be active in the same topic (thus we chose large $\boldsymbol{\rho_{11}}$).

For our simulation, we manually built two sets of topics. The first set of topics are in line with the Markov process with small $\boldsymbol{\rho_{01}}$ and large $\boldsymbol{\rho_{11}}$ on the tree structure, while the second set of topics are against this Markov process on the tree structure. The first set of topics is used to test if knowing the tree structure of codes improves the inference of treeLFA, and the second set of topics is used to test the robustness of treeLFA when the tree structure is wrong. We chose to manually built topics instead of directly simulating them using the generative process of treeLFA and the tree structure. The reason was that based on our experience, it was hard to generate topics with strong structure when there were only limited number of disease codes (20 codes in our case). Without strong structure for topics, it would be difficult to differentiate the performance of different topic models.

Figure 2.6A shows the first set of topics. Among the four topics, the first three contain four active codes which are from the same branches of the tree, and the last topic contains eight active codes coming from two branches of tree (Figure 2.5). In this way, the last topic has a different level of sparsity compared to the first three topics. As was mentioned, these topics are likely to be generated using a Markov process with small $\boldsymbol{\rho_{01}}$ and large $\boldsymbol{\rho_{11}}$ on the tree structure. On the contrary, the second set of topics were against the tree structure (Figure 2.6B). They were constructed by switching some active codes between neighbouring topics in Figure 2.6A. As a result, in these topics all children codes of an active parent code are inactive, while inactive parent codes only have active children codes.

With the manually built topics, input datasets were then simulated for the topic models to be evaluated using the generative process of treeLFA. Training and testing datasets of the same size were simulated. To evaluate the topic models in different situations, multiple groups of datasets were generated using different combinations of two hyperparameters: $\boldsymbol{\alpha}$ for the Dirichlet prior for topic weights $\boldsymbol{\theta}$ and the number of people $\boldsymbol{D}$ in the training and testing datasets.

For the first set of topics, we simulated four groups of datasets using both small

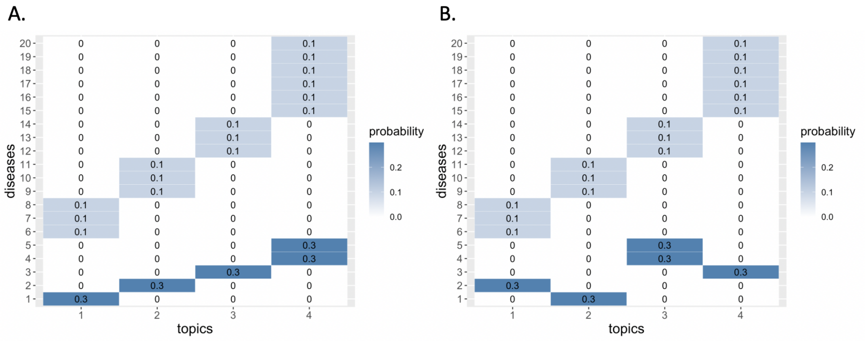


Figure 2.6: Two sets of topics used for simulation. In the heatmap each column is a topic, and each row is a disease code. The 20 codes are numbered/named in the same way as the nodes on the tree in Figure 2.2. The first five codes correspond to the five nodes on the second layer of the tree (internal nodes) in Figure 2.2. Color of cells represent probabilities of codes in topics. A, Topics that are in line with a Markov process with small $\boldsymbol{\rho_{01}}$ and large $\boldsymbol{\rho_{11}}$ on the tree structure. B, Topics that are against a Markov process with small $\boldsymbol{\rho_{01}}$ and large $\boldsymbol{\rho_{11}}$ on the tree structure.

Dataset	**D**	$\boldsymbol{\alpha}$	**Tree structure**
1	2500	1	correct
2	5000	1	correct
3	300	0.1	correct
4	1000	0.1	correct
5	2500	1	incorrect
6	5000	1	incorrect
7	300	0.1	incorrect
8	100	0.1	incorrect

Table 2.2: hyperparameter setting for simulated datasets

and large values for $\boldsymbol{\alpha}$ and $\boldsymbol{D}$. A large value for $\boldsymbol{\alpha}$ means multiple topics have non-small weights for the same person, thus multiple topics will be assigned to non-small numbers of different disease codes for the person. By contrast, a small value for $\boldsymbol{\alpha}$ means most disease codes for the same person will be assigned with the same dominant topic. It is obvious that a large value for $\boldsymbol{\alpha}$ makes the inference difficult. As a result, datasets simulated using large $\boldsymbol{\alpha}$ require large $\boldsymbol{D}$ for the inference of topic models to work. For the second sets of topics, we also simulated four groups of datasets using the same hyperparameters. Table 2.2 summarizes the choice of hyperparameters for different groups of datasets:

2.5.3 Method for validation of treeLFA with simulated data

Comparison of the three related topic models

Three topic models were evaluated and compared using the simulated datasets, which are treeLFA, flatLFA and LDA. flatLFA has the same model space as treeLFA.

The only difference between treeLFA and flatLFA is their tree structure (Figure 2.7). flatLFA uses a non-informative tree structure for topics, in which all codes are placed parallelly under the root node. This tree structure contains no information about the relationship of the disease codes. For treeLFA and flatLFA, we used different Beta priors for the transition probability variable $\boldsymbol{\rho}$ on the Markov process, such that the expected number of active codes in topics generated with the tree structure for the two models are the same. In summary, any difference in the inference results for treeLFA and flatLFA can only be caused by the prior knowledge about topic structure we encoded in treeLFA's hierarchical prior.

In addition to flatLFA, we also included the standard LDA for comparison to evaluate the influence of model configuration on the inference. As was discussed in chapter 1, LDA is designed for count data and uses multinomial distributions for topics instead of Bernoulli distributions.

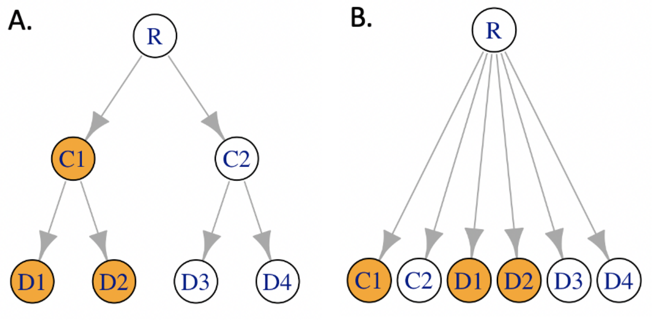


Figure 2.7: Schematics for the tree structure used by treeLFA and flatLFA. treeLFA uses an informative tree structure, on which certain disease codes are more close to each other compared to other codes. flatLFA uses a non-informative tree structure, in which all disease codes are placed in parallel under the root node.

To evaluate the performance of the three topic models, we used two types of metrics. The first metric was the predictive likelihood on the testing dataset for different models; The second metric was the inference accuracy, which means the difference of inferred and true hidden variables (topics). For both $\boldsymbol{\phi}$ and $\boldsymbol{I}$ we used the averaged per disease difference of true and inferred probability in topics to measure the inference accuracy:

$$\begin{aligned} \boldsymbol{\phi_{diff}} &= \frac{\sum_{k=1}^{K}\sum_{s=1}^{S}|\phi_{ks}^{true} - \phi_{ks}^{infer}|}{K \cdot S} \\ \boldsymbol{I_{diff}} &= \frac{\sum_{k=1}^{K}\sum_{s=1}^{S}|I_{ks}^{true} - I_{ks}^{infer}|}{K \cdot S} \end{aligned}$$

Implementation of the three topic models

LDA was implemented using the R package "topicmodels", and treeLFA and flatLFA were implemented by ourselves with the R package "RcppParallel" and "Rcpp". For the hypeparameters of the model, we provided the true value of $\boldsymbol{\alpha}$ to train the three models. Beta priors used for the probabilities of active and inactive disease codes in topics were Beta(2,4) and Beta(0.3,80). These values were chosen empirically to ensure a good performance for both treeLFA and flatLFA. The Beta priors used for the transition probability $\boldsymbol{\rho_{01}}$ and $\boldsymbol{\rho_{11}}$ on the tree structure for treeLFA were Beta(4.8,20) and Beta(20,4.8). For flatLFA only $\boldsymbol{\rho_{01}}$ would be used, and its prior was Beta(7,20). These Beta priors approximately resulted in the same expected number of active codes in topics generated by the corresponding Markov processes for the two models. For LDA we tried a few different values for $\boldsymbol{\eta}$ (0.01, 0.1, 1), which was the Dirichlet prior for topics. We found that with $\boldsymbol{\eta} = \mathbf{0.01}$ LDA had the best performance, so we used this value.

For topics of treeLFA and flatLFA, we initialized all indicator variable $\boldsymbol{I}$ as 0, and then initialized all probability variable $\boldsymbol{\phi}$ via simulating from the Beta prior for inactive disease codes. All topic assignment variable $\boldsymbol{Z}$ were initialized randomly. $\boldsymbol{\theta}$ were not initialized because they were integrated out during the collapsed Gibbs sampling.

On each dataset, ten Gibbs chains were trained for each model, and 50 posterior samples were collected from each Gibbs chain after 15,000 burn-in iterations. Chains for the same model were initialized randomly. We evaluated the accuracy of each posterior sample of topics from each chain, and then combined the statistics for all posterior samples as the final accuracy score for a model. The order of inferred topics in different posterior samples are not guaranteed to be the same. To compare the inferred and true topics, we used the true topics as template and realigned inferred topics to true topics one after another.

2.5.4 Result for validation and comparison of three related topic models

Result on datasets simulated using correct tree structure

Performance of treeFA, flatLFA and LDA on the four groups of datasets simulated using the correct tree structure were compared. In Figure 5.1, traceplots for all ten treeLFA chains on one dataset in each group were shown. The burn-in stage was short, and the mixing was good for all chains. In Figure 2.9 and Figure 2.10 the inference accuracy was evaluated in terms of the probability variable $\boldsymbol{\phi}$ and the indicator variables $\boldsymbol{I}$ in topics respectively. treeLFA achieved accurate inference of topics on all four datasets. On datasets simulated using large $\boldsymbol{\alpha}$ and small $\boldsymbol{D}$ (the most difficult group of datasets), 98.6 % of the $\boldsymbol{I}$ in topics were correctly inferred, compared to 90 % for flatLFA. On the contrary, flatLFA and LDA only gave inference accuracy comparable to treeLFA on certain datasets. LDA had good performance on datasets simulated using small $\boldsymbol{\alpha}$ ($\boldsymbol{\alpha} = \mathbf{0.1}$) (Figure 2.9C-D). On datasets simulated using large $\boldsymbol{\alpha}$, LDA's performance was much worse than the other two models (Figure 2.9A-B). As for flatLFA, it had significantly inferior performance than treeLFA on the two datasets simulated using small $\boldsymbol{D}$ (Figure 2.9A and C). As the size of the training data increased, the performance of treeLFA and flatLFA became very similar (Figure 2.9B and D).

In addition to measuring the inference accuracy on the training data, we also used testing datasets to evaluate the generalization ability of inferred topics. For both treeFLA and flatLFA, we calculated their predictive likelihood on the corresponding testing data. On each dataset, the predictive likelihood calculated using different posterior samples of topics from different chains was averaged, and the per individual ratio of likelihood given by treeLFA and flatLFA was shown in Figure 2.11. On all four groups of datasets, the predictive likelihood given by treeLFA was larger than that for flatLFA. Furthermore, on the two groups of datasets simulated with small $\boldsymbol{D}$ (group 1 and 3 in Figure 2.11), the ratio was larger compared to the other two groups of datasets simulated using large $\boldsymbol{D}$ (group 2 and 4 in Figure 2.11).

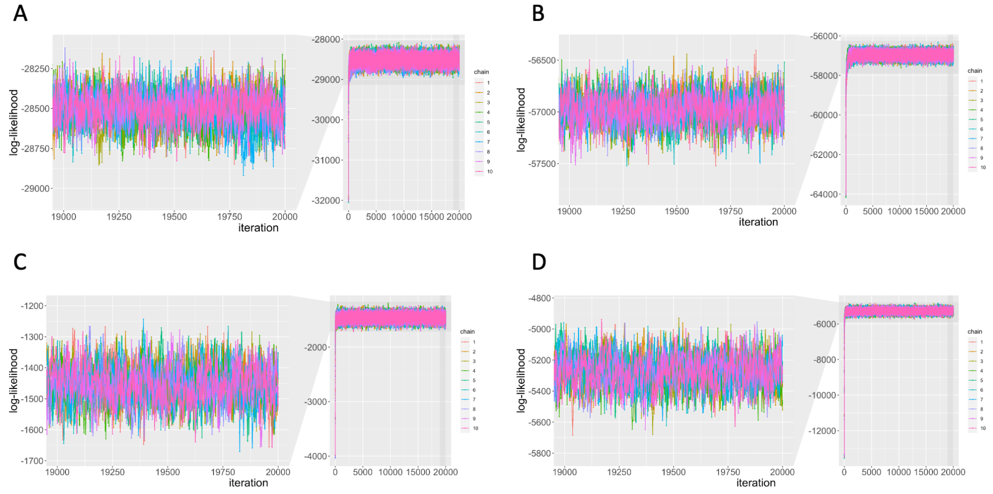


Figure 2.8: Traceplots for all ten treeLFA chains on one dataset from each group were shown. The log-likelihood was plotted against the Gibbs sampling iterations. A, Result for the first dataset simulated using $\boldsymbol{\alpha = 1}$ and $\boldsymbol{D = 2500}$. B, Result for the first dataset simulated using $\boldsymbol{\alpha = 1}$ and $\boldsymbol{D = 5000}$. C, Result for the first dataset simulated using $\boldsymbol{\alpha = 0.1}$ and $\boldsymbol{D = 300}$. D, Result for the first dataset simulated using $\boldsymbol{\alpha = 0.1}$ and $\boldsymbol{D = 1000}$.

Result on datasets simulated using wrong tree structure

To test the performance of treeLFA when the tree structure is incorrect, another four groups of input datasets were simulated using topics that are unlikely to be generated using a Markov process with small $\boldsymbol{\rho_{01}}$ and large $\boldsymbol{\rho_{11}}$ on the tree structure.

Overall, on the second set of datasets, the performance of the three models was similar to each other (Figure 2.12). On all four groups of dataset, the difference in the inference accuracy given by treeLFA and flatLFA was insignificant. The performance of LDA was still significantly different from that for treeLFA and flatLFA, but the absolute difference was much smaller than that on the first set of datasets (Figure 2.9).

By comparing the performance of the three models across the two sets of datasets (Figures 2.9 and 2.12), we see that in general flatLFA and LDA gave

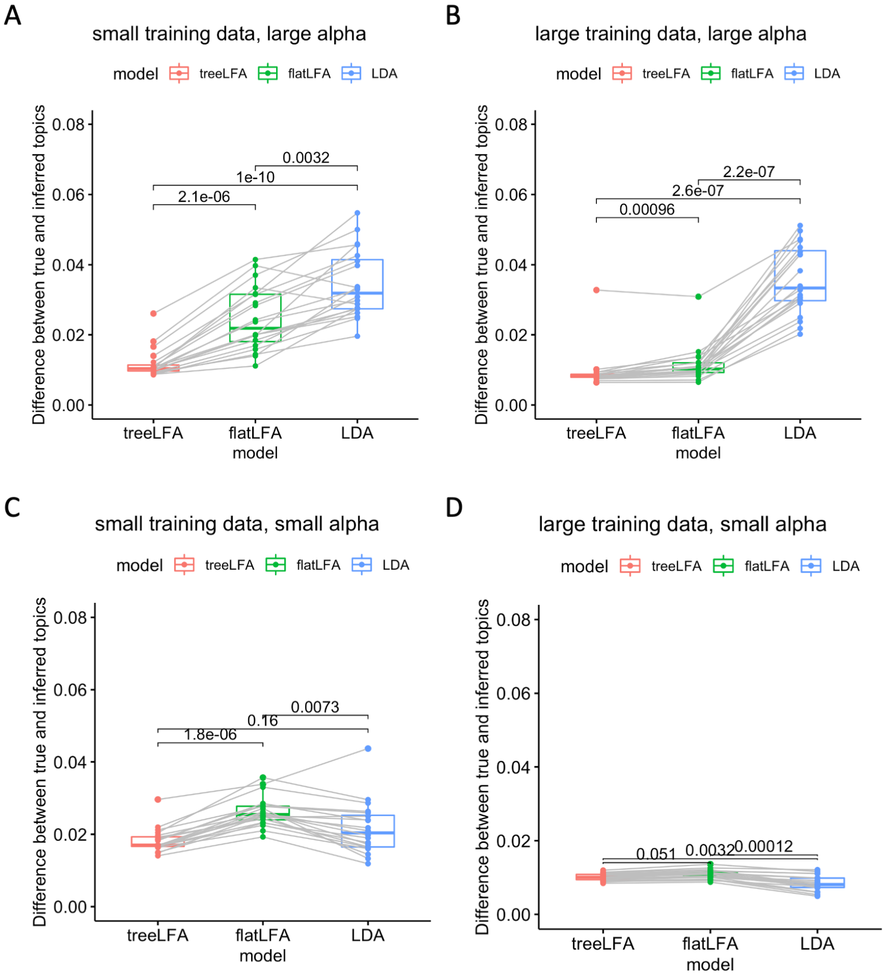


Figure 2.9: Comparison of the inference accuracy for the three topic models on four groups of simulated datasets. Each point in these figures represents the result averaged from ten Gibbs chains for one model on one dataset. The inference accuracy is measured with the averaged per disease difference in probability between true and inferred topics. A, Result on the 20 datasets simulated using $\boldsymbol{\alpha} = \mathbf{1}$ and $\boldsymbol{D} = \mathbf{2500}$. B, Result on datasets simulated using $\boldsymbol{\alpha} = \mathbf{1}$ and $\boldsymbol{D} = \mathbf{5000}$. C, Result on datasets simulated using $\boldsymbol{\alpha} = \mathbf{0.1}$ and $\boldsymbol{D} = \mathbf{300}$. D, Result on datasets simulated using $\boldsymbol{\alpha} = \mathbf{0.1}$ and $\boldsymbol{D} = \mathbf{1000}$.

similar performance on the two sets of datasets (datasets simulated using the same $\boldsymbol{\alpha}$ and $\boldsymbol{D}$, but correct or incorrect tree structure). For instance, for flatLFA the difference between true and inferred topics per disease was $\mathbf{1.96 \pm 0.72}$ in Figure 2.9A and $\mathbf{1.78 \pm 0.62}$ in Figure 2.12A. These were within our expectation since flatLDA and LDA did not use the tree structure of disease codes. Due to the use of a misleading prior for topics, treeLFA failed to correctly infer the true topics as before (the averaged per disease difference of probability increased from $\mathbf{0.94 \pm 0.34}$ to $\mathbf{2.11 \pm 0.73}$). However, with a large enough training data (Figure 2.12B), treeLFA and flatLFA again had the same level of performance. This indicated that with

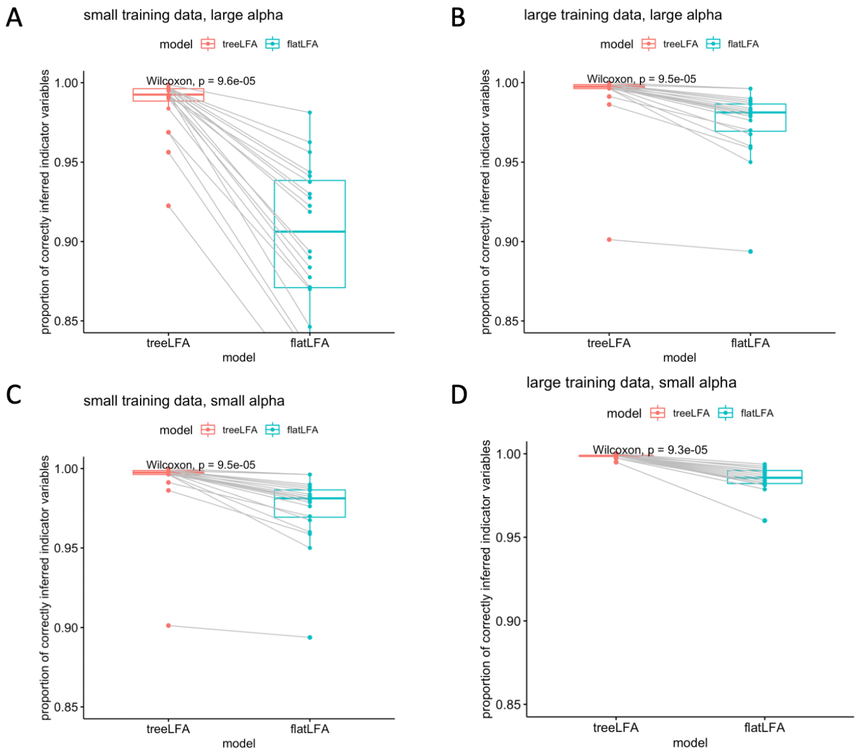


Figure 2.10: Comparison of the proportions of correctly inferred indicator variables for the three topic models on four groups of simulated datasets. A, Result on 20 datasets simulated using $\boldsymbol{\alpha} = \mathbf{1}$ and $\boldsymbol{D} = \mathbf{2500}$. B, Result on datasets simulated using $\boldsymbol{\alpha} = \mathbf{1}$ and $\boldsymbol{D} = \mathbf{5000}$. C, Result on datasets simulated using $\boldsymbol{\alpha} = \mathbf{0.1}$ and $\boldsymbol{D} = \mathbf{300}$. D, Result on datasets simulated using $\boldsymbol{\alpha} = \mathbf{0.1}$ and $\boldsymbol{D} = \mathbf{1000}$.

enough training data, the wrong prior for topics can be overrode.

To assess the magnitude of gain and loss in terms of the inference accuracy with the use of correct and wrong tree structure, the difference between the performance of treeLFA and flatLFA were compared across the two sets of simulated datasets (datasets simulated using correct and incorrect tree structure). As can be seen in Figure 2.13A (datasets simulated using $\boldsymbol{\alpha} = \mathbf{1}$ and $\boldsymbol{D} = \mathbf{2500}$), when the tree structure was correct, treeLFA gave much better result than flatLFA. The difference between true and inferred topics ($\boldsymbol{\phi_{diff}}$) for treeLFA is much smaller than that for flatFLA ($\mathbf{0.94 \pm 0.34}$ for treeLFA, $\mathbf{1.96 \pm 0.72}$ for flatLFA). On the other hand, when the tree structure was incorrect, flatLFA only gave slightly better result than treeLFA. Similar results can be seen in Figure 2.13B (result on datasets simulated using $\boldsymbol{\alpha} = \mathbf{0.1}$ and $\boldsymbol{D} = \mathbf{300}$).

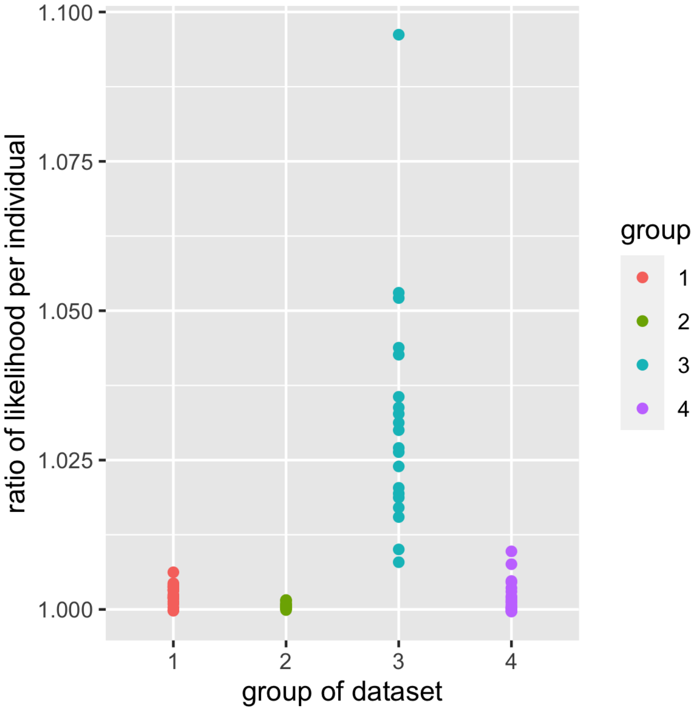


Figure 2.11: Ratio of averaged predictive likelihood per individual in the testing dataset given by treeLFA and flatLFA. The predictive likelihood on different groups of testing datasets was calculated using the topics inferred by treeLFA and flatLFA from the corresponding training datasets. Each point in the plot represents the averaged per individual ratio of likelihood given by the two models on one dataset. There are four groups of datasets in total, and each group contains 20 different datasets. A, Result on the 20 datasets simulated using $\boldsymbol{\alpha} = \mathbf{1}$ and $\boldsymbol{D} = \mathbf{2500}$. B, Result on datasets simulated using $\boldsymbol{\alpha} = \mathbf{1}$ and $\boldsymbol{D} = \mathbf{5000}$. C, Result on datasets simulated using $\boldsymbol{\alpha} = \mathbf{0.1}$ and $\boldsymbol{D} = \mathbf{300}$. D, Result on datasets simulated using $\boldsymbol{\alpha} = \mathbf{0.1}$ and $\boldsymbol{D} = \mathbf{1000}$.

2.6 Validation of the model selection algorithm for treeLFA

2.6.1 Method for validation of the model selection algorithm

For treeLFA, the model selection (the selection of number of topics to be inferred) was done by training multiple models with different numbers of topics and then comparing their predictive likelihood on the testing dataset. Previous simulated datasets were used again to test if the model selection algorithm is able to pick out the best model (if the model with four topics give the largest predictive likelihood on the testing dataset). For the two groups of datasets simulated using large $\boldsymbol{D}$ ($\boldsymbol{D} = \mathbf{1000}$ for $\boldsymbol{\alpha} = \mathbf{0.1}$; $\boldsymbol{D} = \mathbf{5000}$ for $\boldsymbol{\alpha} = \mathbf{1}$) and correct tree structure, we trained nine treeLFA models set with 2-10 topics on five datasets in each group. Ten Gibbs chains were trained for each model on each dataset and 50 posterior samples were collected from each chain. Then the averaged predictive likelihood

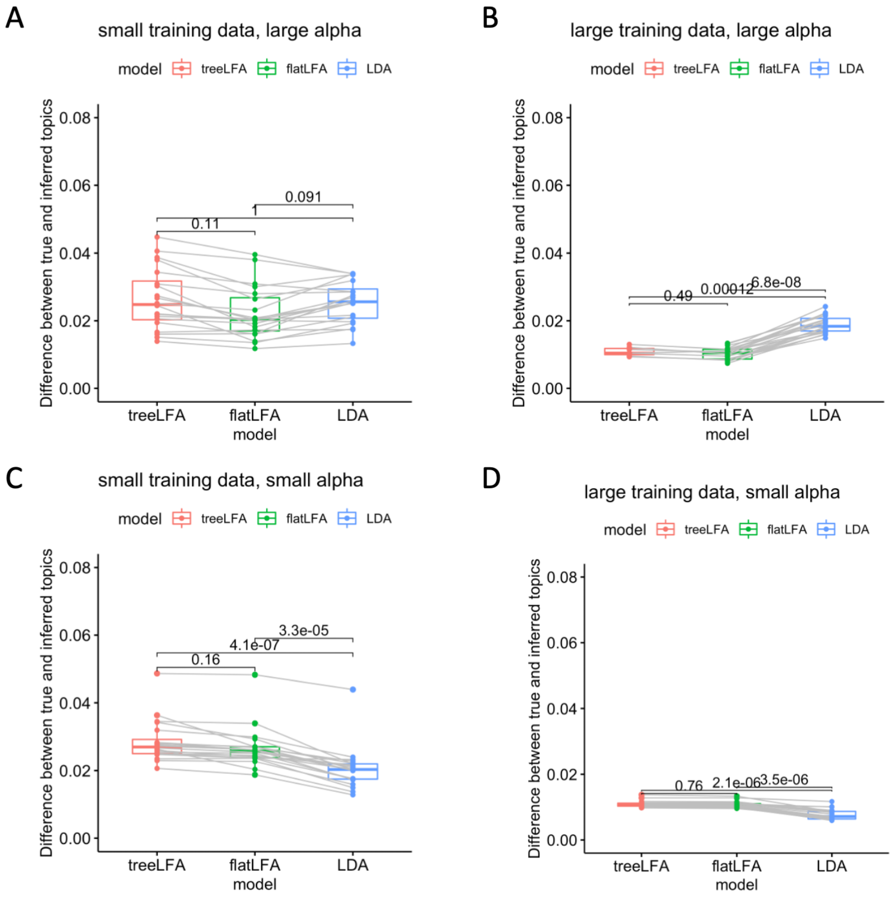


Figure 2.12: Performance of three topic models on datasets simulated using a wrong tree structure. The datasets were simulated using topics whose structure are against the previously used Markov process on the tree. A, Difference between true and inferred topics for three topic models on datasets simulated using $\boldsymbol{\alpha} = \mathbf{1}$ and $\boldsymbol{D} = \mathbf{2500}$. B, Result on datasets simulated using $\boldsymbol{\alpha} = \mathbf{1}$ and $\boldsymbol{D} = \mathbf{5000}$. C, Result on datasets simulated using $\boldsymbol{\alpha} = \mathbf{0.1}$ and $\boldsymbol{D} = \mathbf{300}$. D, Result on datasets simulated using $\boldsymbol{\alpha} = \mathbf{0.1}$ and $\boldsymbol{D} = \mathbf{1000}$.

were calculated using all posterior samples from all the ten chains. Hyperparameters for treeLFA were the same as those used for validation in section 2.5.

2.6.2 Result for validation of the model selection algorithm

Figure 2.14 shows the predictive likelihood for treeLFA models with different number of topics (2-9 topics) on five simulated datasets in each group (results for all chains are shown). It can be seen that in general the model selection algorithm works well. On all five datasets simulated using $\boldsymbol{\alpha} = \mathbf{1}$ (Figure 2.14A) and three datasets simulated with $\boldsymbol{\alpha} = \mathbf{0.1}$ (Figure 2.14B), the treeLFA model with four topics gave the largest predictive likelihood on the testing dataset. On the other two datasets simulated using $\boldsymbol{\alpha} = \mathbf{0.1}$, the treeLFA model with five topics gave the

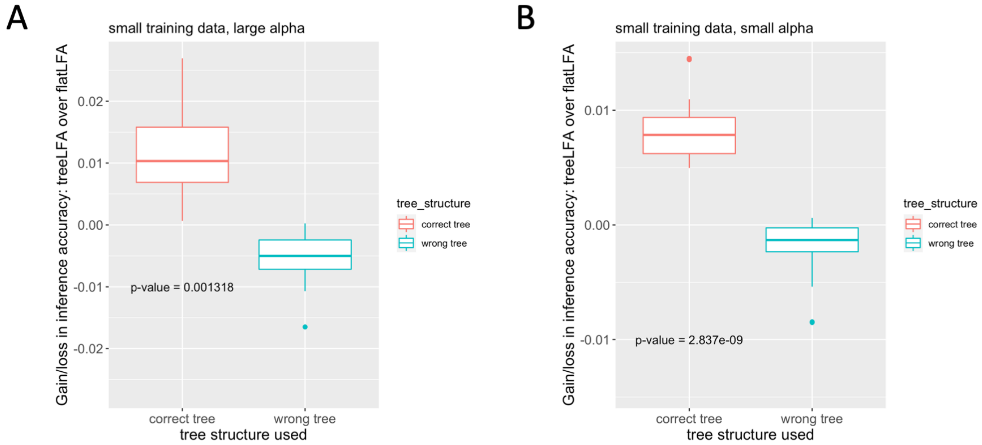


Figure 2.13: Relative gain and loss of inference accuracy with correct and wrong tree structure used for treeLFA. The inference accuracy for treeLFA and flatLFA were compared across two groups of datasets simulated using correct and wrong tree structure and the same hyperparameters. Inference accuracy was measured using the difference between the true and inferred topics ($\boldsymbol{\phi_{diff}}$), and the relative gain and loss of inference accuracy for treeLFA over flatLFA was calculated as:$\boldsymbol{\phi_{diff}^{flatLFA} - \phi_{diff}^{treeLFA}}$. A, Result on datasets simulated using $\boldsymbol{\alpha = 1}$ and $\boldsymbol{D = 2500}$. B, Result on datasets simulated using $\boldsymbol{\alpha = 0.1}$ and $\boldsymbol{D = 300}$.

largest predictive likelihood (Figure 2.14B). The difference of predictive likelihood given by the models with four and five topics was very small. Figure 2.15 shows one posterior sample of the inferred topics given by the treeLFA model with five topics. As can be seen, one of the five learnt topics was an "empty" topic with no active codes, and the other four topics were exactly the four true topics.

2.7 Combining inference results given by different Gibbs chains

For inference done with Gibbs sampling it is routine to train multiple chains and use multiple posterior samples of hidden variables from these chains (for instance, take the average of multiple posterior samples as a point estimate of hidden variables). This is not trivial to do for some models because of the non-identifiability issue. For Bayesian models under symmetric priors, the posterior distributions of hidden variables are invariant to the permutation of their labels. Take topic models as an example, the order of topics in different posterior samples can be different, which means taking their average directly can result in significant error.

An easy way to tackle this problem is to mix multiple posterior samples of different topics together, and cluster them before taking the average within each

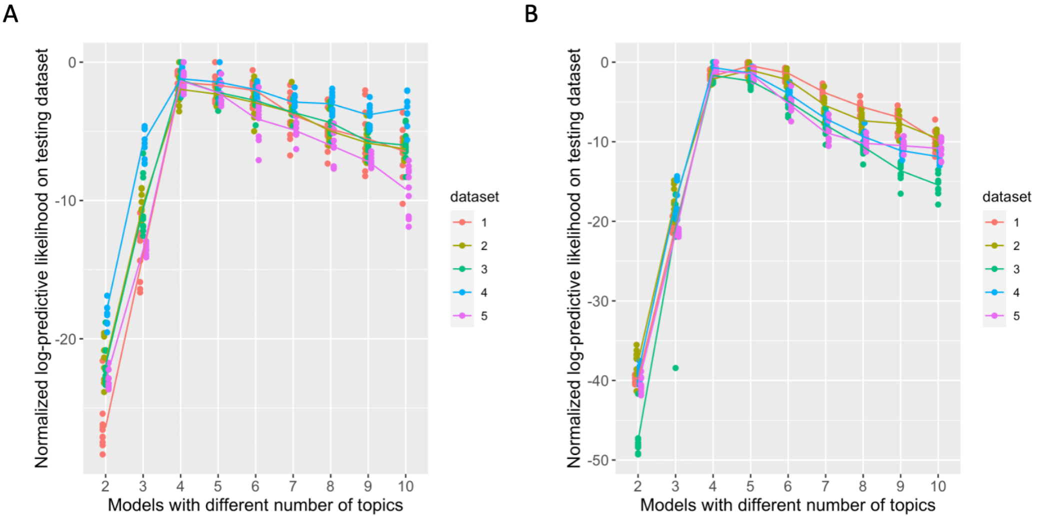


Figure 2.14: Model selection for treeLFA. treeLFA models with 2-10 topics were trained on two groups of datasets simulated using large $\boldsymbol{D}$, and their corresponding predictive likelihood on the testing dataset were calculated. The log-predictive likelihood was normalized such that the largest predictive likelihood for all models on the same dataset is always 0. A, Results on datasets simulated using $\boldsymbol{\alpha} = \mathbf{1}$ and $\boldsymbol{D} = \mathbf{5000}$. B, Results on datasets simulated using $\boldsymbol{\alpha} = \mathbf{0.1}$ and $\boldsymbol{D} = \mathbf{1000}$.

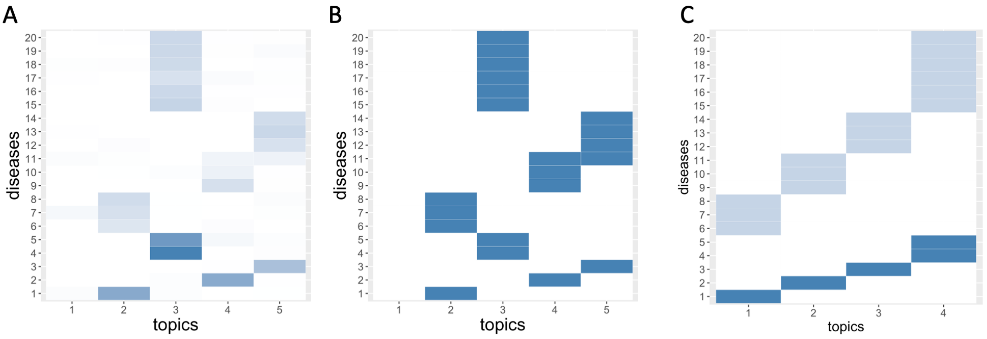


Figure 2.15: Inferred topics given by the treeLFA model with five topics. On some simulated datasets the treeLFA model with five topics had larger predictive likelihood than the treeLFA model with four topics (the correct number of topics). The topics inferred by the treeLFA model with five topics on one of these datasets were checked. A, One posterior sample of the topics inferred by the treeLFA model with five topics. B, One posterior sample of indicator variables for topics inferred by the treeLFA model with five topics. C, The four true topics used to simulated the input data.

cluster [93]. In theory, topics with the same label/identity coming from different chains or posterior samples would be put into the same cluster, regardless of the original order of topics in different posterior samples from different chains.

We used the Louvain clustering algorithm [140] implemented in the R package "igraph" to cluster posterior samples of topics from all Gibbs chains. To visualize the clustering result directly, tSNE (t-distributed stochastic neighbor embedding, a dimension reduction algorithm) was firstly applied to topics to reduce their dimension from 20 to 2. Then posterior samples of topics from all chains trained on a dataset simulated using $\boldsymbol{\alpha} = \mathbf{1}$ and $\boldsymbol{D} = \mathbf{2500}$ were plotted in Figure 2.16. Four distinct clusters of topics could be seen, and each cluster of topics was composed of posterior samples from all the ten Gibbs chains, indicating a consensus reached by the ten Gibbs chains.

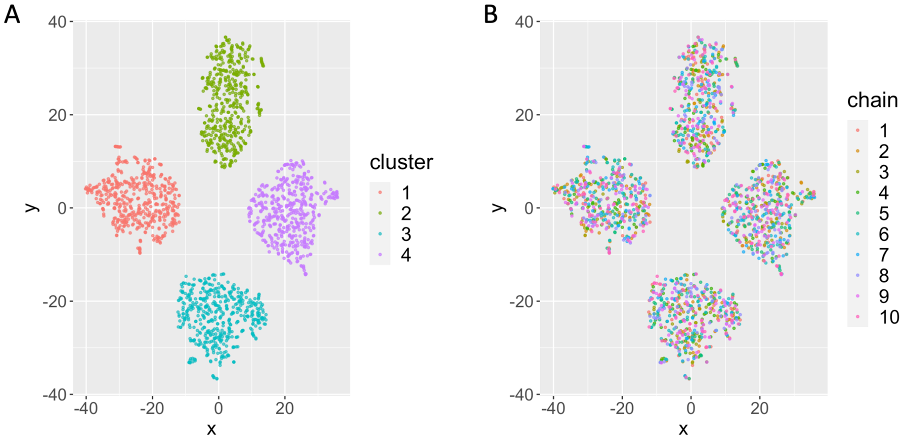


Figure 2.16: Clustering and visualization of posterior samples of topics. Posterior samples of topics from ten Gibbs chains (500 posterior samples of all topics in total) were put together. The Louvain algorithm was used to cluster these topics. tSNE was used for dimension reduction of topics before the visualization of all posterior samples of topics in a 2D plot. Each point in the plot corresponds to one posterior sample of one topic. A, Topics were colored according to their assigned clusters. B, Topics were colored according to the Gibbs chains they come from.

2.7.1 Robustness of inference result to the number of topics

As was discussed in the previous section, by clustering posterior samples of topics we can combine the inference results given by different Gibbs chains. In addition, clustering of topics should also enable us to combine duplicated topics. In reality, the optimal number of topics to be learnt is usually not known. As a result, if the number of topics set for the model is more than the model's actual need, duplicated topics might be learnt from the data. However, all the duplicated topics will be put into the same cluster during the clustering process. And after taking the average for each cluster, duplicated topics will be transformed into a single topic.

On one simulated dataset used in the previous section, we trained ten Gibbs chains for the treeLFA model set with ten topics instead of four (the correct number of topics). Figure 2.17A shows one posterior sample of the ten topics learnt from the data. Although some topics closely resemble the true topics, there are also spurious/incorrect topics. Besides, more than one learnt topics are empty topics with no active codes in them. However, if we applied the clustering procedure onto all posterior samples of topics, we got five distinct averaged topics in total (Figure 2.17B), including one empty topic and the four correctly inferred topics.

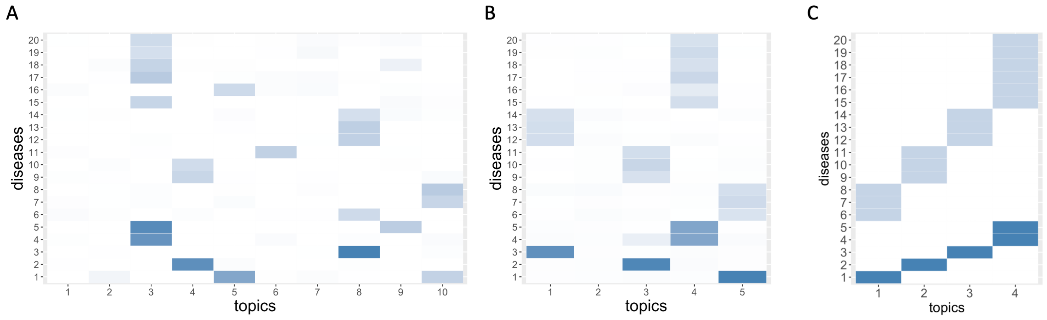


Figure 2.17: Combining the inference results from multiple chains increase the robustness of treeLFA to the number of topics. treeLFA model with ten topics were trained on the input data simulated using four topics. A, One posterior sample of topics given by the treeLFA model with ten topics. B, Five averaged inferred topics obtained by combining the results of ten Gibbs chains through clustering posterior samples of topics. C, Four true topics used to simulated the input data.

2.8 Validation of the hyperparameter learning algorithm for treeLFA

As was discussed in the previous sections, for treeLFA there are three types of hyperparamters, including the Dirichlet prior $\boldsymbol{\alpha}$ for topic weight variable $\boldsymbol{\theta}$, the transition probabilities of the Markov process on the tree structure ($\boldsymbol{\rho_{00}}$ and $\boldsymbol{\rho_{10}}$), and the parameters of the Beta priors for probability variable $\boldsymbol{\phi}$. Among them, the learning of $\boldsymbol{\alpha}$ is the most difficult one, and we adopted and modified the Gibbs-EM algorithm proposed by Minka, et al. [105] to do this. The goal of this section is to test the ability of the Gibbs-EM algorithm to learn different types of $\boldsymbol{\alpha}$.

2.8.1 Method for validation of the Gibbs-EM algorithm

To validate the Gibbs-EM algorithm, three groups of datasets were simulated using three different $\boldsymbol{\alpha}$. Among them, the first two $\boldsymbol{\alpha}$ vectors were symmetric ($\boldsymbol{\alpha} = \boldsymbol{(1,1,1,1)}$ and $\boldsymbol{\alpha} = \boldsymbol{(0.1,0.1,0.1,0.1)}$), and the third $\boldsymbol{\alpha}$ vector was asymmetric ($\boldsymbol{\alpha} = \boldsymbol{(1,0.1,0.1,0.1)}$). The topics used for simulation and the choices of other parameters were same as the previous validations. The three groups of simulated datasets here were larger than the previous ones ($\boldsymbol{D} = \boldsymbol{5000}$ for $\boldsymbol{\alpha} = \boldsymbol{(0.1,0.1,0.1,0.1)}$; $\boldsymbol{D} = \boldsymbol{10000}$ for $\boldsymbol{\alpha} = \boldsymbol{(1,1,1,1)}$; $\boldsymbol{D} = \boldsymbol{10000}$ for $\boldsymbol{\alpha} = \boldsymbol{(1,0.1,0.1,0.1)}$), since doing inference of the hidden variables and learning hyperparameters at the same time was more difficult than doing inference alone with the correct hyperparameters provided, therefore required more training data. For each $\boldsymbol{\alpha}$, ten datasets were simulated, and ten Gibbs chains were trained for treeLFA on each dataset. For all the three groups of datasets, $\boldsymbol{\alpha}$ was initialized as $\mathbf{(1, 1, 1, 1)}$. Both $\boldsymbol{\rho}_{\mathbf{00}}$ and $\boldsymbol{\rho}_{\mathbf{10}}$ were initialized as 0.5, and reasonable Beta priors (Beta(4,20) and Beta(20,4)) were put on them.

Two stages of training with the Gibbs-EM algorithm were implemented. In the first stage (the first 2000 iterations) only one sample of the hidden variables were collected in the E-step for the optimization of $\boldsymbol{\alpha}$ in the M-step; In the second stage (the 200 iterations following the first stage), ten samples of hidden variables were collected in the E-step. One posterior sample of topics and $\boldsymbol{\alpha}$ were taken every ten iterations in the second training stage, thus in total 20 posterior samples of topics and $\boldsymbol{\alpha}$ were taken for the evaluation of the inference accuracy.

Posterior samples of topics from different chains were mixed and clustered as before. Within the same posterior sample of hidden variables, topics and the entries in the $\boldsymbol{\alpha}$ vector had one-to-one correspondence, so we also put entries of different $\boldsymbol{\alpha}$ vectors into the corresponding clusters based on the clustering result for topics. In another word, we also clustered entries in the learnt $\boldsymbol{\alpha}$ vectors given by different chains. Mean was then taken for each cluster of topics and entries of $\boldsymbol{\alpha}$, and inferred topics and $\boldsymbol{\alpha}$ were compared to the true ones to evaluate the performance of the Gibbs-EM algorithm.

2.8.2 Result for validation of the Gibbs-EM algorithm

The inference accuracy of treeLFA models trained with the Gibbs-EM algorithm was evaluated in the same way as before. The per disease difference in probability between true and inferred topics are $\mathbf{0.34 \pm 0.04}$, $\mathbf{0.51 \pm 0.09}$ and $\mathbf{0.59 \pm 0.08}$ for the three groups of datasets, indicating the inference of topics were accurate (with previous validation results used as reference). In Figure 2.18 the averaged $\boldsymbol{\alpha}$ learnt by the Gibbs-EM algorithm for each dataset in each group were plotted. The exact value of each entry in the learnt $\boldsymbol{\alpha}$ vector are shown in the heatmap. The true $\boldsymbol{\alpha}$ for the first group of datasets is $\mathbf{(0.1, 0.1, 0.1, 0.1)}$. Although the entries in the learnt $\boldsymbol{\alpha}$ were always smaller than the true values, we can see that on all datasets the Gibbs-EM algorithm figured out that the true $\boldsymbol{\alpha}$ should be a symmetric vector and entries in it should be small. In Figure 2.18B, entries in the learnt $\boldsymbol{\alpha}$ were also smaller than the true values $\mathbf{(1, 1, 1, 1)}$. Yet as the result for the first group of datasets, learnt values for entries of $\boldsymbol{\alpha}$ still fell in the reasonable range. In Figure 2.18C the true $\boldsymbol{\alpha}$ was a asymmetric vector $\mathbf{(1, 0.1, 0.1, 0.1)}$. The learnt $\boldsymbol{\alpha}$ vectors maintained the ratio of the large and other small entries in the true $\boldsymbol{\alpha}$. Overall, these results demonstrate that Gibbs-EM is reliable in learning both symmetric and asymmetric hyperparamter $\boldsymbol{\alpha}$ for the Dirichlet prior for topic weights.

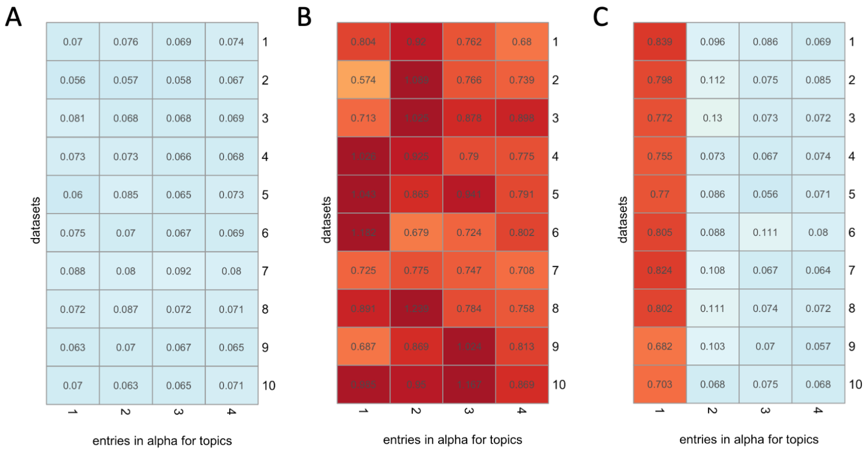


Figure 2.18: $\boldsymbol{\alpha}$ learnt by the Gibbs-EM algorithm for treeLFA on simulated datasets. Three groups of datasets were simulated using different $\boldsymbol{\alpha}$, and each group contains ten different datasets. On each dataset ten Gibbs chains were trained, and the averaged learnt $\boldsymbol{\alpha}$ was calculated using the results given by the ten chains. In each heatmap, each row corresponds to a dataset and each cell corresponds to an entry in the learnt $\boldsymbol{\alpha}$ for this dataset. Entries in all $\boldsymbol{\alpha}$ vectors were re-aligned according to their corresponding inferred topics, using the true topics as template. A, Results on the first group of datasets for which the true $\boldsymbol{\alpha}$ is $\mathbf{(0.1, 0.1, 0.1, 0.1)}$. B, Results on the second group of datasets for which the true $\boldsymbol{\alpha}$ is $\mathbf{(1, 1, 1, 1)}$. C, Result on the third group of datasets for which the true $\boldsymbol{\alpha}$ is $\mathbf{(1, 0.1, 0.1, 0.1)}$.

2.9 Learn topics with single active code

2.9.1 Motivations for studying topics with single active code

Topic models have been used for decades to learn topics of documents based on the co-occurrence of words in them. In this study, our aim is to learn topics of diseases. Words are basic units of language, and they always need to be used together to form sentences. However, a disease is much more complex than a word, and almost every disease has its own unique pathological mechanism. This means it may not always make sense to force single diseases to be grouped into topics. Considering this, we want to test whether treeLFA can put independently occurred diseases into separate topics. A straightforward way to do this is to simulate input data for treeLFA using topics which only contain a single active disease.

2.9.2 Method for testing treeLFA on data simulated using single code topics

We built 20 topics for 20 disease codes, and each topic contains one different active code (Figure 2.19). One input dataset was simulated using these topics, and treeLFA model with 20 topics was trained using the Gibbs-EM algorithm. Two different sets of Beta priors were used for the transition probabilities of the Markov process on the tree structure, since it was likely that the structure of inferred topics would be strongly influenced by the prior when there was no strong pattern in the data. The first set of Beta priors for $\boldsymbol{\rho}$ was the ones we had been using thus far, which were Beta(4,20) and Beta(20,4) for $\boldsymbol{\rho_{00}}$ and $\boldsymbol{\rho_{10}}$. For data simulated using single code topics, this prior was inappropriate because it favored multiple active codes in the same branch of the tree. In another word, this prior did not favor very sparse topics like single code topics. The second set of prior for $\boldsymbol{\rho}$ were Beta(4,20) and Beta(4,20) for $\boldsymbol{\rho_{00}}$ and $\boldsymbol{\rho_{10}}$, which strongly favored very sparse topics. This was because for one thing it was difficult to get an active code given an inactive parent code (because of the large $\boldsymbol{\rho_{00}}$), and for another thing it was also difficult to get an active child code conditioned on an active parent code (because of the large $\boldsymbol{\rho_{10}}$). As before, ten Gibbs chains were trained for each set of Beta priors. After training, posterior samples of topics from all chains were mixed and clustered, and then the average was taken for each cluster.

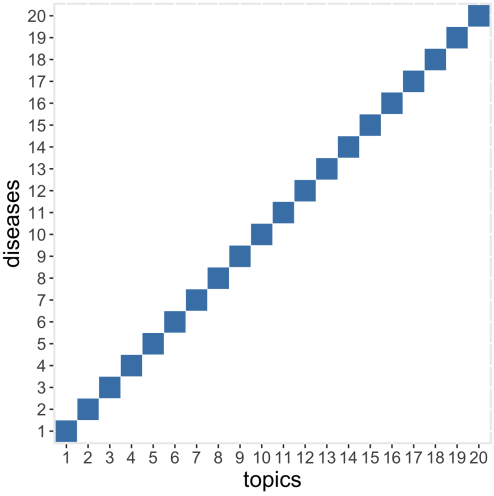


Figure 2.19: Topics used for simulation to test the ability of treeLFA in learning single code topics.

2.9.3 Result for testing treeLFA on data simulated using single code topics

Figure 2.20 shows the averaged inferred topics for treeLFA using the two sets of Beta priors for $\boldsymbol{\rho}$. In Figure 2.20A we see seven distinct clusters of topics. Among them one topic is empty, and one topic only contains a single active code. Most of the remaining topics contain four active codes, and the distribution of active codes in these topics completely follows the tree structure of codes (Figure 2.5). This suggests the prior for topics played an decisive role in the inference. On the contrary, in Figure 2.20B we see 20 clusters of inferred topics given by treeLFA using the second

Hidden variable	**Computational time**
$\boldsymbol{\rho}$	0.001
$\boldsymbol{I}$	0.019
$\boldsymbol{\alpha}$	4.328
$\boldsymbol{\phi}$	4.463
$\boldsymbol{Z}$	10.858

Table 2.3: Time spent on sampling different hidden variables once for the full dataset.

set of Beta priors for $\boldsymbol{\rho}$, which favors very sparse topics. Most of these topics contain only one active code, which means true topics were successfully inferred by treeLFA.

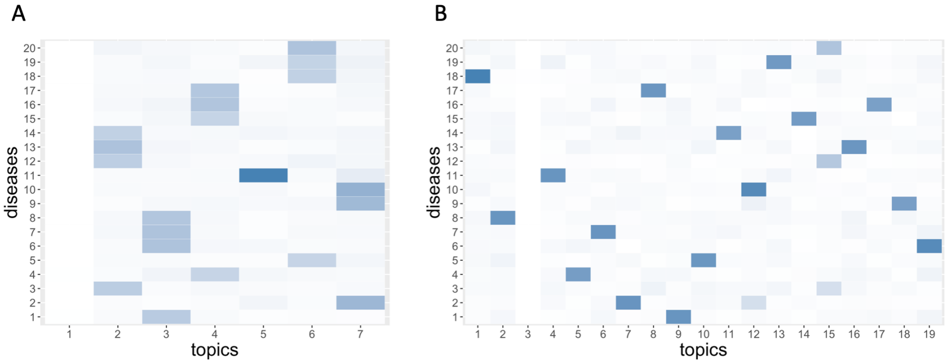


Figure 2.20: Averaged inferred topics given by treeLFA models using different Beta priors for the transition probabilities $\boldsymbol{\rho}$ of the Markov process on the tree. A, Topics inferred by treeLFA using Beta(4,20) and Beta(20,4) for $\boldsymbol{\rho}_{\mathbf{00}}$ and $\boldsymbol{\rho}_{\mathbf{10}}$. B, Topics inferred by treeLFA using Beta(4,20) and Beta(4,20) for $\boldsymbol{\rho}_{\mathbf{00}}$ and $\boldsymbol{\rho}_{\mathbf{10}}$.

2.10 Scalability of treeLFA

The targeted data for treeLFA is the phenotypic dataset in biobanks. On such large-scale datasets, the scalability of the inference algorithm is important. The collapsed Gibbs sampling for treeLFA is composed of the sampling steps for different hidden variables ($\boldsymbol{\phi}$, $\boldsymbol{I}$, $\boldsymbol{Z}$, $\boldsymbol{\rho}$). Since treeLFA is to be applied onto the population based biobank datasets (contain hundreds of thousands of individuals), the rate-limiting step would be the sampling of $\boldsymbol{Z}$ (topic assignment variable), whose number will be several orders of magnitude larger than other hidden variables. Besides, sampling $\boldsymbol{\phi}$ involves checking the topic assignments of all disease variables for all individuals, which is also time-consuming. In Table 2.3 the relative time (the shortest time was set to be 1) spent on updating different hidden variables for one iteration on a large dataset (436 diseases and 400,000 people, 50 topics for the model) is shown.

As can be seen, on a biobank scale dataset most computational time was spent on sampling $\boldsymbol{Z}$ and $\boldsymbol{\phi}$. Although updating $\boldsymbol{\alpha}$ also takes long, for the Gibbs-EM algorithm $\boldsymbol{\alpha}$ will only be updated once after 20 iterations of sampling of $\boldsymbol{Z}$,$\boldsymbol{\rho}$,$\boldsymbol{\phi}$ and $\boldsymbol{I}$, which makes its contribution to the total computational time non-significant.

In theory, the total computational time will roughly be linearly proportional to the number of people in the dataset ($\boldsymbol{D}$), and the number of diseases to be analyzed ($\boldsymbol{S}$), since sampling $\boldsymbol{Z}$ and $\boldsymbol{\phi}$ both require going through each disease variable for each individual. The relationship between the number of topics ($\boldsymbol{K}$) and the total computational time is not obvious. $\boldsymbol{K}$ will undoubtedly be linearly proportional to the time spent on sampling $\boldsymbol{\phi}$. As for the time spent on $\boldsymbol{Z}$, with more topics more time will be needed to define the categorical distribution for $\boldsymbol{Z}$ and sample $\boldsymbol{Z}$. In Figure 2.21 we checked the influence of each parameter ($\boldsymbol{D}$, $\boldsymbol{S}$, $\boldsymbol{K}$) on the total computational time by training treeLFA using the Gibbs-EM algorithm, with only one parameter varied each time. It can be seen that the computational time increased linearly as we increased each of the parameters when analysing a biobank-scale dataset.

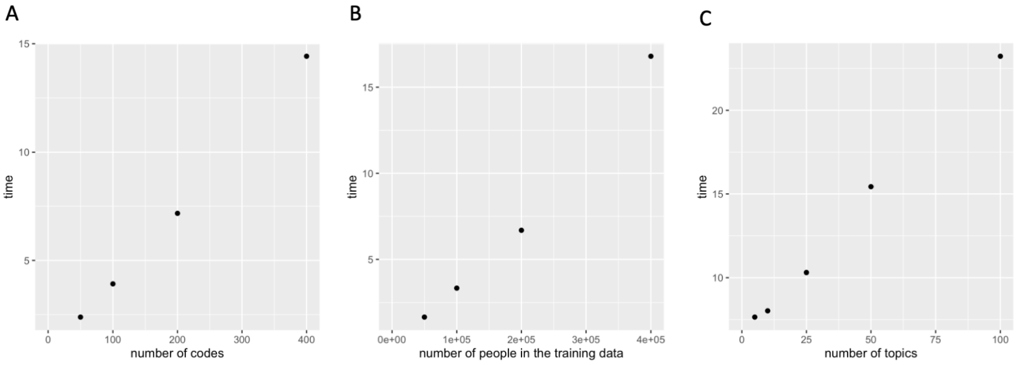


Figure 2.21: Relationship between the computational time and three parameters of treeLFA. The influence of the number of people, disease codes and topics on the total computational time spent on running Gibbs-EM for two iterations without optimizing $\boldsymbol{\alpha}$ were plotted. A, The relationship between the number of disease codes and the computational time. B, Result for the number of people in the training dataset. C, Result for the number of topics to be inferred.

2.11 Discussion

In this chapter we developed a new topic model named treeLFA, which can be used to learn topics of binary variables. The motivation in doing this is to build a topic model that is better than available ones to analyze large scale phenotypic datasets in biobanks, such as the Hospital Episode Statistics (HES) data in UKB. There were two main features we added on this model. Firstly, we wanted it to be specifically adapted to binary input data. Secondly, we wanted to incorporate a prior for the structure of topics based on our current understanding of the relationship of diseases. To realize the first goal, we developed treeLFA on the basis of BNMF instead of the more commonly used LDA. The fundamental difference between treeLFA and LDA is that instead of using multinomial distributions to model the occurrences of diseases (LDA), the status of all diseases for all individuals are modelled with Bernoulli distributions by treeLFA. As a result, a topic inferred by treeLFA is a sequence of Bernoulli distributions for all diseases, while a topic inferred by LDA is a multinomial distribution over all diseases. The basic motivation for using a different model configuration is to model the input data as what it is (a binary matrix). Since LDA is designed for count data, in which the number of times each word in the vocabulary appears in a document is used for inference, it is not entirely appropriate for analyzing binary input data, though it has already been widely used in doing so. In addition, the model configuration for treeLFA also makes it easier to construct an informative prior for topics based on the hierarchical medical ontology. Besides, as shall be seen in Chapter 4, it also makes individuals' inferred topic weights independent with each other, instead of being negatively correlated (the case for LDA), which is beneficial to the downstream analyses.

To incorporate a hierarchical prior for topics based on a disease classification system, we used a Markov process on the fixed hierarchical structure of diseases dictated by the disease classification system to generate indicators for all diseases in a topic. These indicators denote which diseases are active (having large non-negligible probability) in a topic. By choosing specific values for the two transition probabilities of this Markov process, we encourage diseases that are close on the tree structure (sit on the same branch of the tree) to be co-active in the same topic.

After developing treeLFA, we undertook extensive validations for the new model using various simulated datasets. By comparing the performance of three related topic models (treeLFA, flatLFA and LDA) on these simulated datasets, we verified the unique values for both the hierarchical prior for topics as well as the unique model configuration of treeLFA (based on Bayesian non-negative matrix factorization). In Section 2.5.4 we see that when the training data is small, treeLFA has large advantage against flatLFA and standard LDA, because of the additional information contained in the prior for topics. This is reflected by both the inference accuracy on the training data as well as the generalization ability of topics to the testing data. Besides, when $\boldsymbol{\alpha}$ is large, which means multiple topics make non-negligible contributions to the generation of words in documents, both treeLFA and flatLFA have apparent advantages over LDA. Currently the reason behind this observation is not entirely clear. It may be possible that the prior for topics used by treeLFA (hierarchical prior) and flatLFA (flat prior) have more flexibility than the symmetric Dirichlet prior used by LDA. The topics we used for simulation have different levels of sparsity (total number of active codes), and this may partly explain why flatLFA also outperformed LDA on some datasets. On datasets simulated using topics that are against the tree structure, we found that treeLFA didn't give much inferior result compared to flatLFA. In summary, this means that the gain in inference accuracy for treeLFA over flatLFA given the correct tree structure is much larger than the loss of inference accuracy given a wrong tree structure.

To obtain more reliable inference result, we proposed a simple method to combine the results given by multiple Gibbs chains by clustering all posterior samples from all chains. We discovered that this method increases the robustness of treeLFA to the number of topics set for them. After clustering, duplicated topics were put into the same cluster, and spurious topics were merged into the major clusters of topics and then cancelled out by taking the average for each cluster, since major patterns (true topics) in theory should be captured by many posterior samples, while spurious patterns are unstable and of diverse forms. This to some extent lessens the burden of the model selection problem for topic models, which is computationally expensive since it involves training models with different numbers of topics and comparing their performance in some way. In summary, by removing duplicated topics and diluting wrongly inferred topics with clustering and averaging, treeLFA become more robust to the number of topics as long as enough topics are set for the model.

In addition to validating the basic inference algorithm for treeLFA, we also tested the model selection algorithm for treeLFA and the hyperparameter learning algorithm (Gibbs-EM algorithm), since both algorithms are important for the application of treeLFA on complex real-world data. We got reassuring results for both tests. For the model selection algorithm, sometimes the treeLFA model with one more topic than the number of true topics gave the best performance. We observed that when this happened, usually the one extra inferred topic was an empty topic with no active codes. This is in fact a property of treeLFA, that the algorithm tend to learn empty topic with no active codes. The basis for this property is that treeLFA doesn't force topics to be multinomial distributions over all words in the vocabulary as LDA, thus the probabilities of all diseases in a topic don't need to sum to 1. Besides, the empty topic has the highest level of sparsity, thus they also have relatively high likelihood conditioned on the Markov process we used (Markov process with large $\boldsymbol{\rho_{00}}$ and small $\boldsymbol{\rho_{10}}$). Overall, this doesn't affect the major inference result, since important patterns in data will still be captured by non-empty topics.

In the final section of this Chapter, we carried out an additional analysis, which evaluated the performance of treeLFA on datasets simulated using topics with one single active code. We found that on such dataset, the transition probabilities of the Markov process played an large role. With transition probabilities supporting active codes in a topic to be in the same branch of the tree structure, the learnt topics also entirely follow the tree structure. With transition probabilities favoring very sparse topics, true single active code topics were learnt. This behaviour of treeLFA is acceptable to us, since it means that if the occurrences of diseases are independent and no pattern exists in the data, the algorithm will simply re-construct the original tree structure of diseases in the inferred topics, instead of learning meaningless and misleading topics of diseases.

Overall, the validation results in this chapter indicate that treeLFA is a reliable topic model for binary input data with its own unique strength, and is ready to be used on large-scale biobank datasets. In the next four chapters, we will apply treeLFA onto the HES dataset in UKB to infer topics of common diseases (multi-morbidity clusters), and carry out various downstream analyses on the basis of the inference of treeLFA. A major emphasis will be put on studying the genetic associations of multi-morbidity clusters.

3

Application of treeLFA on UK Biobank data

Contents

3.1 Overview of the chapter

After developing the treeLFA model and validating it using simulation, in this chapter we apply the model to the hospital episode statistic (HES) data from the UK Biobank (UKB), which is coded using the ICD-10 ontology. Using treeLFA, topics of common diseases were learnt and found to be aligned with current medical understanding. We also trained flatLFA and LDA models on the UKB data and compared their inference results with those given by treeLFA. In addition to focusing on one treeLFA model, we trained treeLFA models with different numbers of topics, and studied the change of inferred topics across different models. Lastly, we explored the possibility of subtyping common diseases using inferred topic weights and studied genetic heterogeneity among subgroups of diseases.

3.2 The input data for treeLFA

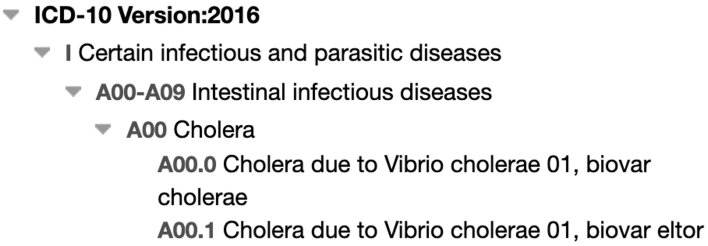
The UKB HES dataset consists of the records of tens of thousands of diseases (coded using ICD-10 codes) for half a million people. At present, it is not realistic for treeLFA to analyze all these ICD-10 codes simultaneously because the computational time required to so would not be affordable. Moreover, we also wished to explore alternative ways of performing inference and downstream analyses on a smaller set of data before scaling everything up. Therefore, we first constructed a restricted UKB dataset (named the top-100 dataset) from the HES data in UKB. This dataset was composed of the top 100 most frequent ICD-10 codes from the first 13 chapters of the ICD-10 coding system for all people in UKB. This selection of chapters provided a balance between breadth of phenotype and depth within any one chapter (it was desired that the number of codes from each chapter was not too small) so that the potential benefits of treeLFA can be explored. Notably, many of the excluded chapters contained codes that were not well-defined diseases (such as Chapter 19: Injury, poisoning and certain other consequences of external causes)

In the top-100 dataset, zeros and ones were used to represent the absence and presence of diagnosed ICD-10 codes for individuals. If an individual was diagnosed with the same disease code several times, one would still be used. The full top-100 dataset (containing 502,537 individuals belonging to all different ancestry groups in UKB) was randomly split into a training dataset and a testing dataset, containing the records of ICD-10 codes for 80% and 20% of all people respectively.

3.2.1 The hierarchical structure of ICD-10 codes

Diagnoses in UKB are coded using the ICD-10 billing system, which uses a hierarchical (or tree-like) structure with five layers to classify and organize all diseases [141]. As introduced in Chapter 2, for treeLFA a prior for topics is constructed on top of this tree structure. The first layer of this tree structure is simply the root node; the second layer is composed of chapters of diseases represented by capital English letters; the third layer contains blocks of disease categories; the fourth layer contains single disease categories; and lastly, the bottom layer contains sub-categories of diseases, which can be, for instance, the same disease occurring at different sites of the human body, or different subtypes of a disease. An example of one branch of this hierarchical structure from the root node to the leaf (terminal) node is shown in Figure 3.1.

In UKB, most of the diagnosed diseases were encoded using subtypes of diseases (codes on the bottom layer of the tree). For the top-100 dataset we constructed, ICD-10 codes on the fourth layer of the tree (disease categories) were used to encode people's diseases and to build the input dataset for treeLFA. We replaced terminal ICD-10 codes with the parental code on the ICD-10 tree. In this way, different subtypes of a disease were re-coded into the same one. The tree structure of the ICD-10 codes in the top-100 dataset can be seen in Figure 3.2.



▼ **ICD-10 Version:2016**
- ▼ **I** Certain infectious and parasitic diseases
  - ▼ **A00-A09** Intestinal infectious diseases
    - ▼ **A00** Cholera
      - **A00.0** Cholera due to Vibrio cholerae 01, biovar cholerae
      - **A00.1** Cholera due to Vibrio cholerae 01, biovar eltor

Figure 3.1: The hierarchical structure of the ICD-10 coding system. An example is used to demonstrate the five levels of the hierarchical structure of the ICD-10 system. From the top to the bottom are the root node, the chapter one of ICD-10 codes, the block of categories of diseases A00-A09, the disease category A00, and sub-categories A00.0 and A00.1.

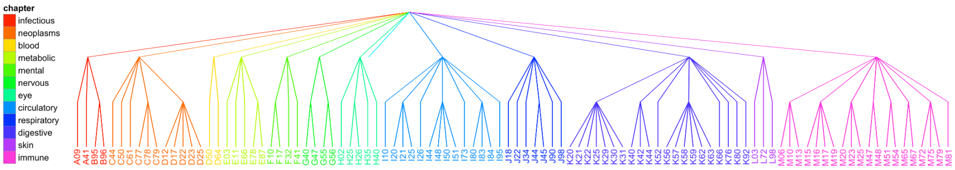

chapter
infectious
neoplasms
blood
metabolic
mental
nervous
eye
circulatory
respiratory
digestive
skin
immune
A09 A41 B95 B96 C44 C50 C61 C77 C78 C79 D12 D17 D22 D23 D25 D50 D64 E03 E11 E66 E78 E87 F10 F17 F32 F41 G40 G47 G55 G56 H02 H25 H26 H35 H40 I10 I20 I21 I25 I26 I44 I48 I50 I51 I73 I80 I83 I84 I95 J18 J22 J34 J44 J45 J90 J98 K20 K21 K22 K25 K29 K30 K31 K40 K42 K44 K52 K56 K57 K58 K59 K62 K63 K66 K76 K80 K92 L03 L72 L98 M06 M10 M13 M15 M16 M17 M19 M20 M23 M25 M47 M48 M51 M54 M65 M67 M72 M75 M79 M81

Figure 3.2: The hierarchical structure for codes in the top-100 UKB dataset. The tree structure has four layers which correspond to the first four layers of the ICD-10 tree. Edges and nodes on the tree structure are colored according to the ICD-10 chapters.

3.2.2 Multi-morbidity in the top-100 dataset

For treeLFA to perform well individuals in the UKB need to demonstrate sufficient multi-morbidity, so that treeLFA can learn from the patterns of co-occurrence of diseases. The distribution of the total numbers of diagnosed ICD-10 codes for all individuals in the the top-100 dataset is shown in Figure 3.3. Among the 400,000 individuals in the training dataset (of all different ancestry groups), 214,993 were diagnosed with multiple diseases, although there are also 122,704 individuals who have no recorded diagnoses and 62,303 individuals who were only diagnosed with a single disease.

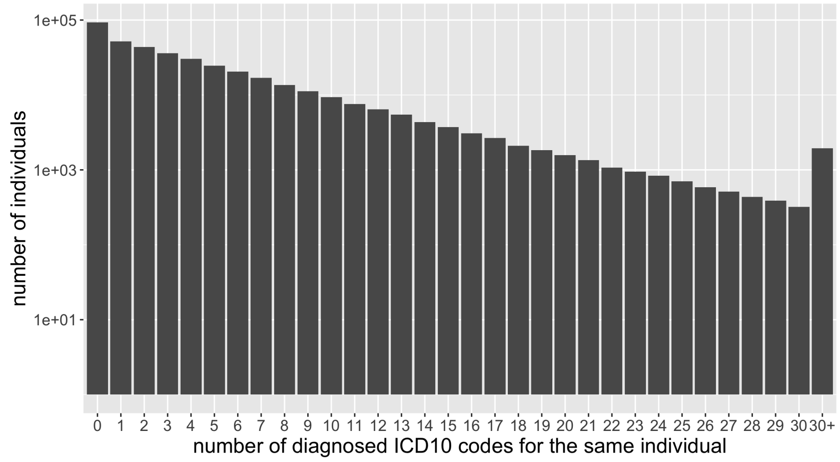


Figure 3.3: Multi-morbidity in the top-100 UKB dataset. Histogram of the total number of different diagnosed ICD-10 codes for people in the training dataset of the top-100 dataset.

3.3 Inference with the top-100 UKB dataset

To further evaluate treeLFA's performance with the phenotypic (HES) data in UKB, and to study the topics of disease codes at different resolutions, we trained multiple treeLFA models with different numbers of topics (2-20, 50 and 100 topics) on the training data of the top-100 dataset. We also trained flatLFA and LDA models and compared their inference results to that for treeLFA, comparable to the analyses presented in Chapter 2.

3.3.1 Implementation of treeLFA

The training strategy

For each treeLFA model, we firstly used the Gibbs-EM algorithm to learn the hyperparameter $\boldsymbol{\alpha}$ in two stages. In the first stage we ran 1000-2000 iterations of the Gibbs-EM algorithm. In the E-step of each iteration, we ran 19 burn-in iterations for the Gibbs sampler before collecting one posterior sample of the hidden variables ($\boldsymbol{Z}$), which was used to optimize $\boldsymbol{\alpha}$ in the M-step. In the second stage we continued to run another 200 iterations of the Gibbs-EM algorithm, during which we collected ten posterior samples of the $\boldsymbol{Z}$ in each E-step. As with the first training stage, before collecting each posterior sample we still ran 19 burn-in iterations of the Gibbs sampler. As a result, in total $\mathbf{200((19+1)\cdot 10 = 200)}$ iterations of Gibbs sampling were run to collect ten posterior samples of $\boldsymbol{Z}$ in each E-step in this stage. The reason to have two stages of training was to balance the computational speed with the inference accuracy. During the first stage, the training was fast so that we could quickly get close to the optimal value of $\boldsymbol{\alpha}$. In the second stage, the optimization of $\boldsymbol{\alpha}$ was more stable and accurate since it is based on ten posterior samples of $\boldsymbol{Z}$.

Once the training with the Gibbs-EM algorithm was completed, the collapsed Gibbs sampler was then used to simulate posterior distributions of all hidden variables ($\boldsymbol{Z}$, $\boldsymbol{I}$, $\boldsymbol{\phi}$,$\boldsymbol{\rho}$), with $\boldsymbol{\alpha}$ fixed at the values provided by the last iteration of the Gibbs-EM algorithm. 5000 iterations of Gibbs sampling were run and posterior samples of hidden variables were collected every 100 iterations. For each model, ten Gibbs chains were constructed, and 50 posterior samples were collected from each chain.

Initialization of hidden variables

The initialization for hidden variables and hyperparameters influences the speed and accuracy of the inference. The initialization of $\boldsymbol{\alpha}$ is most important, since learning $\boldsymbol{\alpha}$ takes a long time, and a smart initialization for $\boldsymbol{\alpha}$ can significantly shorten the training with the Gibbs-EM algorithm. Based on our experience with the treeLFA model and the UKB dataset, among all the inferred topics there will always be an empty topic, in which all disease codes are inactive. A large fraction of individual's disease variables will be assigned to this empty topic, since the majority of people in UKB only have a few diagnosed disease codes (most disease variables for most individuals are zero, hence the empty topic is the most likely topic to be assigned to them). Taking this into account, we initialized $\boldsymbol{\alpha}$ with the vector $\boldsymbol{(1, 0.1, \ldots, 0.1)}$. The first entry in the $\boldsymbol{\alpha}$ vector corresponds to the empty topic, thus has a larger weight than the other entries. We also tried to initialize $\boldsymbol{\alpha}$ in different ways, such as using $\boldsymbol{(1, \ldots 1)}$, and we found that the final inference results were the same, and the learnt $\boldsymbol{\alpha}$ was usually most close to $\boldsymbol{(1, 0.1, \ldots, 0.1)}$.

For topic assignment variable $\boldsymbol{Z}$, we assigned the empty topic to all disease variables for all individuals without any disease codes. For individuals with at least one diagnosed disease code, all topics were initially assigned to all disease variables randomly.

For topics, all indicator variables ($\boldsymbol{I}$) for disease codes in topics were initialized as 0, with the probability variable $\boldsymbol{\phi}$ randomly sampled from Beta(1,5,000,000). Beta priors used for $\boldsymbol{\phi}$ were Beta(0.3,80) for inactive codes and Beta(2,4) for active codes. Priors used for the transition probabilities $\boldsymbol{\rho_{01}}$ and $\boldsymbol{\rho_{11}}$ of the Markov process were Beta(3,20) and Beta(3,3) respectively, chosen to impose sparsity on topics.

Training of treeLFA/flatLFA

Eight CPU cores on the BMRC (Biomedical research computing) computing cluster were used for the training of each chain for treeLFA/flatLFA. The computational time depends on the number of inferred topics and the number of iterations. For the model with 11 topics, about one day was spent on the training with the Gibbs-EM algorithm (learning of $\boldsymbol{\alpha}$, 1,100 Gibbs-EM iterations, equivalent to 40,000 Gibbs sampling iterations), and about four hours was spent on the training with the Gibbs sampling (collection of posterior samples of hidden variables, 5,000 iterations).

3.3.2 Inference result for the top-100 dataset

Figure 5.1 shows the traceplots for the ten Gibbs chains during the Gibbs-EM training stage. 1,100 Gibbs-EM iterations were run, corresponding to 40,000 Gibbs sampling iterations. At the end of the training, all the ten chains converged. Besides, in the second stage of the Gibbs-EM training (when multiple samples were taken in the E-step for the optimization of $\boldsymbol{\alpha}$), the fluctuation of the log-likelihood was smaller than that in the first stage, indicating the optimization of $\boldsymbol{\alpha}$ is more accurate when multiple samples of $\boldsymbol{Z}$ are used.

Figure 3.5 shows a single posterior sample of all inferred topics for the treeLFA model with 11 topics, together with the tree structure of the 100 ICD-10 codes. The treeLFA model with 11 topics was chosen arbitrarily to enable inspection of the structure of inferred topics; treeLFA models with differing numbers of topics will be discussed in the following sections. Probabilities of the ICD-10 codes in topics and their corresponding indicator variables are shown in the two heatmaps in Figure 3.5.

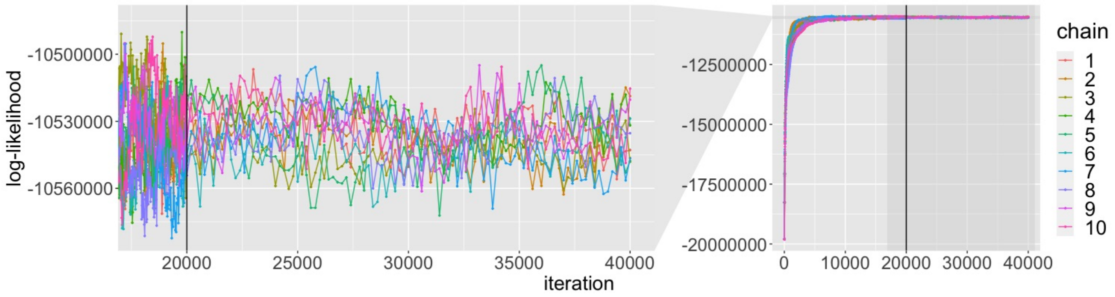


Figure 3.4: Traceplots for ten treeLFA chains during the Gibbs-EM training stage. Log-likelihood was plotted against the iterations of Gibbs sampling. Iterations 18,000 to 40,000 were zoomed in for better visualization. In the first stage of the Gibbs-EM training (iterations to the left of the vertical line in the figure), ***alpha*** was optimized every 20 Gibbs iterations using a single posterior sample of $\boldsymbol{Z}$ taken in the E-step, and the log-likelihood was calculated after each optimization of $\boldsymbol{\alpha}$. In the second stage of the Gibbs-EM training (iterations to the right of the black vertical line), $\boldsymbol{\alpha}$ was optimized every 200 Gibbs iterations using ten posterior samples of $\boldsymbol{Z}$ taken in the E-step, and the log-likelihood was calculated after each optimization of $\boldsymbol{\alpha}$.

Among the 11 inferred topics there is an empty topic (the first topic in Figure 3.5), in which all disease codes are inactive. The corresponding weight for this topic in the learnt $\boldsymbol{\alpha}$ vector is much larger than that for other topics, which means on average about 65 % of people's disease variables were assigned to the empty topic. Topics other than the empty topic all have active codes in them. Some topics are very dense (have a large number of active codes), such as Topic 7 and Topic 10. Other topics are more sparse and contain less active codes.

To provide insights into the nature of the inferred topics we extracted their top active codes (codes with the largest probabilities). Table 3.1 shows the top active codes in the 11 topics (codes with probability larger than 0.2). The active codes in topics are in accordance with our prior knowledge about the relationships of diseases. For instance, I10 (hypertension), E11 (diabetes) and I20 (angina)

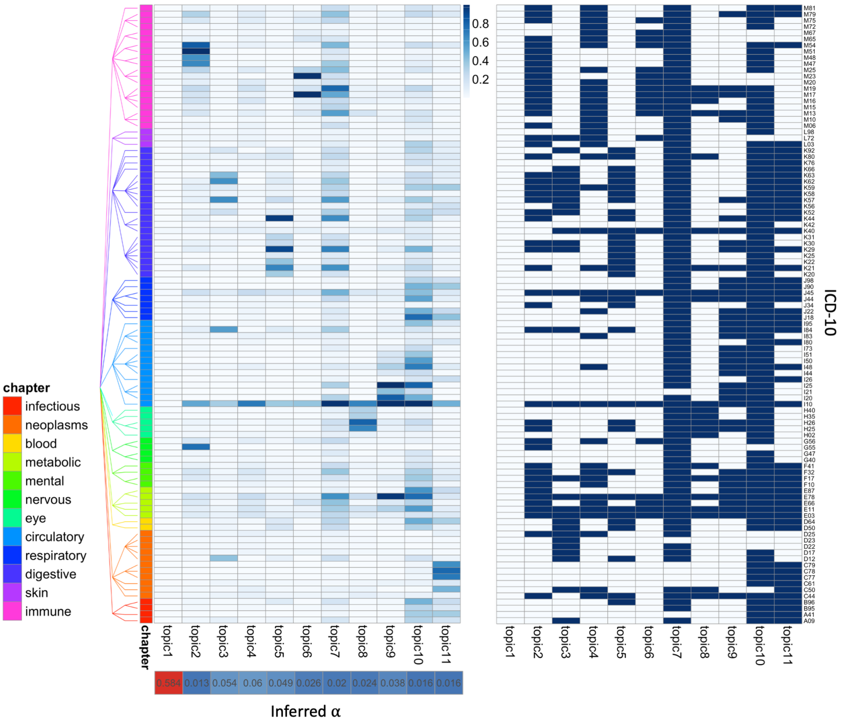


Figure 3.5: Topics inferred by treeLFA for the top-100 dataset. The left heatmap shows one posterior sample of the probability variable $\boldsymbol{\phi}$ for all disease codes in the 11 inferred topics. Each row is an ICD-10 code and each column is a topic. The ICD-10 codes are ordered alphabetically and numerically. As a result, the bottom row is code A09 (the first code), and the top row is M81 (the last code). Codes that are close to each other on the tree structure will also be close in the heatmap. The tree structure of the 100 ICD-10 codes is shown to the left of the inferred topics. The single row of heatmap below the inferred topics shows the $\boldsymbol{\alpha}$ vector learnt by the Gibbs-EM algorithm. The right heatmap shows one posterior sample of the indicator variables of disease codes in the 11 inferred topics. Indicator variables can only take the value of 0 or 1, denoting active and inactive disease codes in a topic.

are components of metabolic syndrome [142], which is known to be associated with an increased risk for cardiovascular diseases (CVD) [143]. This association is reflected in Topic 9, since among the seven top active codes, four are codes for heart diseases (I20 angina, I21 myocardial infarction, I25 chronic ischemic heart disease, I48 Atrial fibrillation and flutter). By contrast, active codes in Topic 6 have a completely different profile, since three among the four of them come from Chapter 13 of the ICD-10 coding system (Diseases of the musculo-skeletal system and connective tissue). For most sparse disease topics, their active codes come from one to two dominant ICD-10 chapters. It is also worth noticing that most ICD-10 codes are only active in 1-2 topics, which indicates that most disease codes have stable comorbidity profiles. However, I10 (hypertension) is an exception, since it is active in almost all topics. This might be due to the fact that hypertension is an extremely common diseases (with a prevalence of 22.5% in UKB) and it co-occurs with a wide variety of different disease codes.

topic	**Active codes**	Topic name
1		Empty Topic
2	G55, I10, J45, M19, M25, M47, M48, M51, M54, M79	Topic of spine diseases
3	D12, I10, I84, K52, K57, K62, K63	Topic of lower GI diseases
4	E78, I10, J45	Topic of hypertension
5	I10, K20, K21, K22, K29, K30, K31, K44, K57	Topic of upper GI diseases
6	I10, M17, M23, M25	Topic of joint diseases
7	D12, D50, D64, E03, E11, E66, E78, F32, F41, G56, H26, I10, I20, I25, I84, J44, J45, K21, K29, K30, K31, K44, K52, K57, K58, K59, K62, K63, K80, K92, M06, M13, M15, M16, M17, M19, M23, M25, M47, M54, M75, M79, M81	Dense topic 1
8	E78, H25, H26, H35, H40, I10	Topic of eye diseases
9	E11, E78, I10, I20, I21, I25, I48	Topic of heart diseases
10	A09, A41, B95, B96, D50, D64, E11, E66, E78, E87, F10, F17, F32, H26, I10, I20, I21, I25, I44, I48, I50, I51, I73, I95, J18, J22, J44, J45, J90, J98, K21, K29, K44, K52, K57, K59, K62, K63, K92, L03, M19, M54, M79	Dense topic 2
11	A41, B96, C50, C77, C78, C79, D64, E87, I10, I26, J18, J22, J90, K52, K59	Topic of cancer

Table 3.1: Codes with probabilities larger than 0.2 in the 11 inferred topics

In addition to the topics of ICD-10 codes, another type of hidden variable inferred by treeLFA are individual's weights for topics, ($\boldsymbol{\theta}$). Figure 3.6 shows a single posterior sample of the topic weights for 2000 randomly selected healthy people (Figure 3.6A) and 2000 people with at least one diagnosed ICD-10 code (Figure 3.6B). For people without recorded diagnoses (informally referred to as healthy people), the empty topic (the first column in the heatmap) always has a weight that is close to one (which means other topics have weights close to 0), while for people who were diagnosed with diseases, their weights for the empty topic are usually much smaller than one, since some weight is assigned to other disease topics. We found a strong negative correlation (Pearson correlation -0.853) between the total number of diagnosed codes and people's weights for the healthy topic, which means the more codes a person has, the less weight the empty topic will possess.

3.3.3 Comparison of inference results given by the three related topic models

With simulated data, we performed comprehensive comparisons between the three related topic models (treeLFA, flatLFA and LDA) on datasets simulated using the generative process of treeLFA, and found that treeLFA significantly

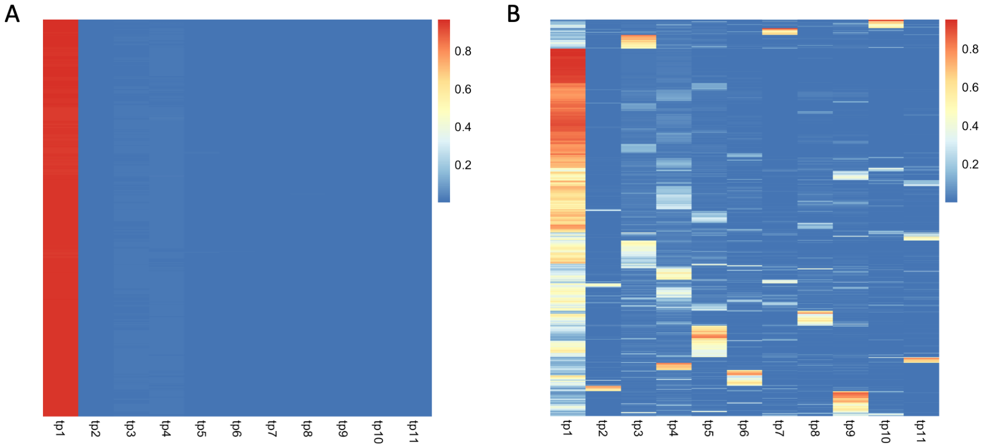


Figure 3.6: Inferred topic weights for people in the top-100 UKB dataset. A, The heatmap shows one posterior sample of the inferred topic weights for 2,000 healthy people, who have no diagnostic records for the 100 ICD-10 codes. Each column in the heatmap is a topic, and each row is a person. The 11 topics here are in the same order as those in Figure 3.5, so the first topic is the empty topic. Each row (all topic weights for a person) sums to 1, since the topic weight vector for a person is a probability vector sampled from a multinomial distribution over topics. B, One posterior sample of the inferred topic weights for 2000 people with at least one diagnosed ICD-10 code.

outperformed the other models in most cases. On UKB data, although we do not know the truth, comparison between topic inferences made using different approaches provides insights into the stability of inferences and the potential advantages of the treeLFA approach.

Topics inferred by the three approaches

In addition to the treeLFA model with 11 topics, we also trained the flatLFA model with 11 topics and the LDA model with ten topics using the same input data. The reason the LDA model with one less topic was trained was that LDA was unable to infer the empty topic, since topics inferred by LDA are multinomial distributions over disease codes, in which probabilities of all codes need to sum to 1.

Figure 3.7 compares the topics inferred by treeLFA to the topics inferred by flatLFA and LDA. Topics inferred by the three models were realigned such that similar topics are adjacent. To compare topics inferred by treeLFA and LDA directly, we normalized treeLFA inferred topics such that the probabilities of all codes sum to 1 in all topics. It can be seen in Figure 3.7A that the topics inferred by treeLFA and flatLFA are very similar, which indicates the size of the input data is large enough such that prior knowledge about the structure of topics plays only a minor role in inference. As for topics inferred by LDA, in Figure 3.7B we see that if we leave out the empty topic, most topics inferred by LDA are also quite similar to the treeLFA inferred topics, but differences also exist (for instance, the first topic inferred by the two models are quite different). However, we also notice that for the dense topics inferred by treeLFA (Topics 7 and 10), after normalization they also have very similar counterparts (topics) in the LDA-based inference.

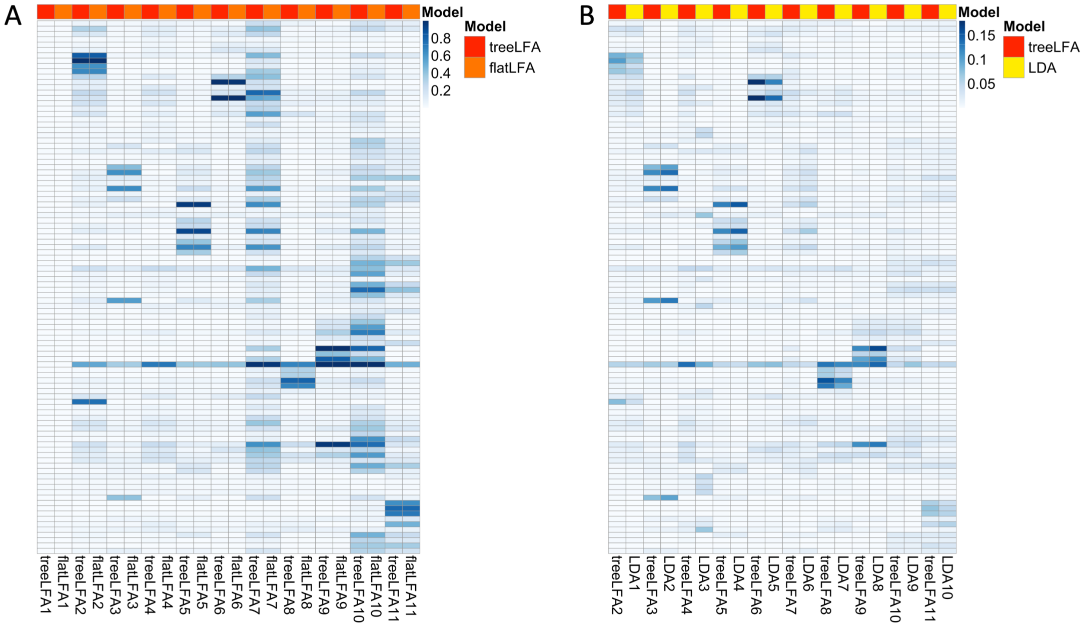


Figure 3.7: Comparison of topics inferred by treeLFA, flatLFA and LDA. Topics inferred by the three models were re-ordered such that similar topics inferred by different models are adjacent. A, Comparison of topics inferred by treeLFA and flatLFA. B, Comparison of topics inferred by treeLFA and LDA. treeLFA inferred topics were normalized to be categorical probability vectors. The empty topic inferred by treeLFA is not shown.

Predictive likelihood for treeLFA and flatLFA

Predictive likelihood on the testing dataset was calculated using the averaged inferred topics of each treeLFA and flatLFA chain. Figure 3.8 shows that there is no difference between the distribution of predictive likelihood for treeLFA and flatLFA chains. Besides, the estimation of predictive likelihood also has a high level of error, as is indicated by the range of likelihood for different chains on the log-scale. These results suggest that by calculating the predictive likelihood, the performance of treeLFA and flatLFA chains can not be differentiated.

Overall, results in this section suggest that the topics inferred by the three models are similar, which suggests that the same multi-morbidity patterns were captured by different models. This adds to the reliability of the inference result, though points to the prior playing a weak role in the analysis of this dataset.

3.3.4 Post-processing of the inference result

Up to now, we have only looked at single posterior samples of topics and topic weights. In order to combine the results given by multiple Gibbs chains,

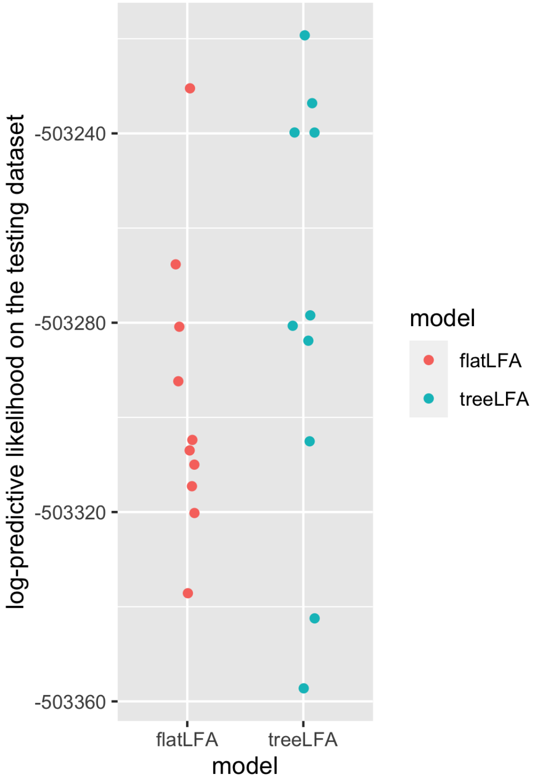


Figure 3.8: Predictive log-likelihood on the testing dataset for treeLFA and flatLDA chains. Ten Gibbs chains were trained for treeLFA and flatLFA, and their predictive likelihood on the testing dataset was calculated using topics averaged from the 50 posterior samples of topics from each chain.

further processing of the inference results are required, as was performed for the simulated data in Chapter 2. In this section, we will perform analyses using only the inference result for treeLFA.

Clustering of posterior samples of inferred topics

For each treeLFA model we trained ten Gibbs chains. To combine the inference results given by all Gibbs chains for a model, we combined posterior samples of topics from all Gibbs chains and clustered them using the Louvain algorithm [140]. After clustering, the mean values for samples within a cluster can be taken as the input for downstream analyses.

In addition to the topics ($\boldsymbol{\phi}$), we also assigned posterior samples of other types of hidden variables (such as the topic weight variable $\boldsymbol{\theta}$) to corresponding clusters according to the topic clustering results. The topic weight variable $\boldsymbol{\theta}$ was integrated out during the collapsed Gibbs sampling, so their posterior samples were estimated using posterior samples of topic assignment variables $\boldsymbol{Z}$ and the $\boldsymbol{\alpha}$ learnt from the data, according to the method proposed by Griffiths and Steyvers in 2004 [136]:

$$\boldsymbol{\theta_{dt}} = \frac{\boldsymbol{N_{dt}} + \boldsymbol{\alpha_t}}{\boldsymbol{N_d} + \sum_t(\boldsymbol{\alpha_t})}, \tag{3.1}$$

where $\boldsymbol{N_{dt}}$ is the total number of disease variables assigned with topic $\boldsymbol{t}$ for person $\boldsymbol{d}$ and $\boldsymbol{N_d}$ is the total number disease variables.

Visualization of inferred topics

To evaluate the consistency of inference results given by different chains for the same model, we can visualize posterior samples of topics as in Chapter 2. Ideally, posterior samples of topics should form distinct clusters, and each cluster should contain posterior samples given by different Gibbs chains.

By applying tSNE on inferred topics we can plot posterior samples of all topics in the same figure. In Figure 3.9, 11 distinct clusters of topics can be seen, and each cluster contains topics from all ten Gibbs chains, indicating a consensus was reached by all chains.

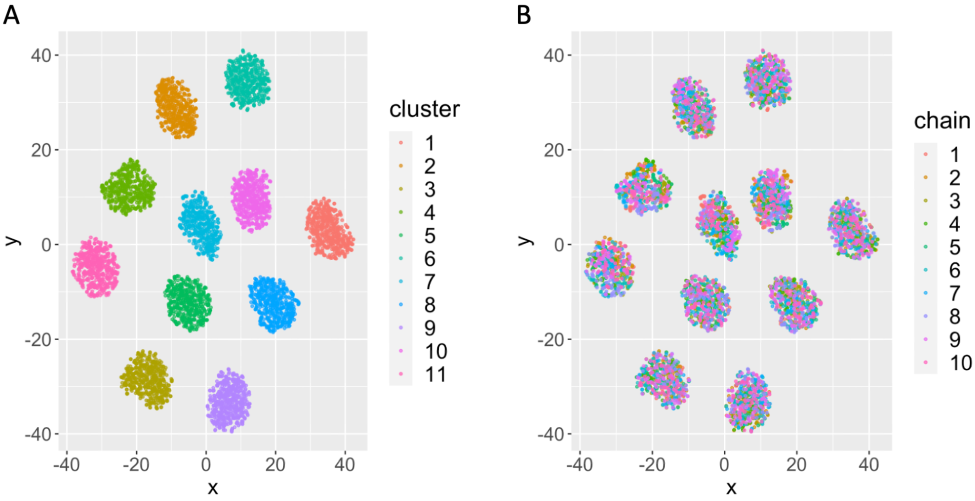


Figure 3.9: Visualization of posterior samples of topics given by different Gibbs chains for the same model. Ten Gibbs chains were trained for the treeLFA model with 11 topics, and 50 posterior samples were collected from each chain. Posterior samples were mixed and clustered using the Louvain algorithm and visualized in 2D plots after applying the tSNE algorithm on topics. A, Posterior samples of topics were colored according to their cluster assignments. B, Posterior samples of topics were colored according to the Gibbs chains they come from.

Post-processing of inference results for models with a large number of topics

In the previous section, the post-processing of treeLFA's inference result was discussed. For the treeLFA model set with 11 topics, exactly 11 distinct topics were inferred (among them there is a single empty topic), and different chains gave consistent inference results. As mentioned at the start of this chapter, for the top-100 dataset we trained many treeLFA models with different numbers of topics. For models set with a very large numbers of topics (for instance, the models set with 50 or 100 topics), the post-processing of inference result can be more difficult and challenging.

For these models, even after the clustering of posterior samples of topics with the Louvain algorithm, there would still be multiple near empty topics (instead of one empty topic). For instance, for the treeLFA model set with 100 topics, 43 clusters of topics remained after the clustering of topics (Figure 3.10A). Among them, about ten topics looked like empty topics. These topics were in fact slightly different with each other. In Figure 3.10B, the last ten topics in Figure 3.10A were plotted with a different color range. Despite differences between these topics were marginal, the Louvain algorithm was sensitive to them when the whole topic vectors were almost empty. As a result, these near-empty topics were still assigned to distinct clusters.

To combine these near-empty topics into a single one, we further applied a Hierarchical clustering on the 43 averaged topics in Figure 3.10A. The dissimilarity between each pair of topics was measured, and similar topics were kept being combined until all the remaining topics were different enough with each other. The "distinctiveness" for all topics was used as the stopping criterion for this hierarchical clustering of topics. It was the smallest pairwise distance for all topics [93], which was defined as: $\boldsymbol{min_{i\neq j}d(\phi_i, \phi_j)}$, with d in the equation denotes a measure of distance between a pair of topics (topics $\boldsymbol{i}$ and $\boldsymbol{j}$). In our analysis, the Manhattan distance was used to measure the distance between topics, and 0.3 was chosen as the threshold for the distinctiveness of topics, which means that two topics can be regarded as different only if their Manhattan distance is larger than 0.3. The choices of using Manhattan distance and the threshold for distinctiveness were arbitrary, but they only decide the number of near empty topics remained, so they don't have important influence on the downstream analyses.

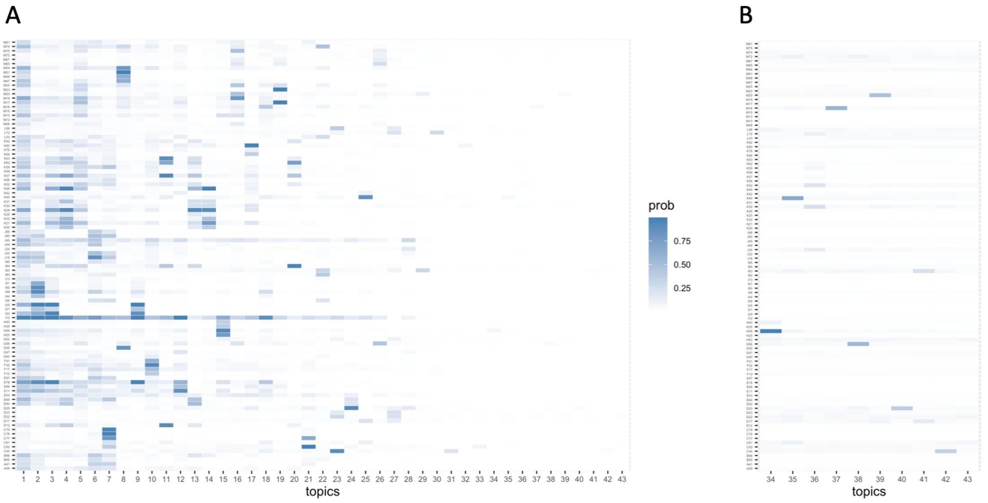


Figure 3.10: Averaged topics obtained by clustering of inferred topics for the model with 100 topics. A, Averaged topics for the 43 clusters of topics given by the Louvain algorithm. Topics were in a descending order based on their density (sum of the probabilities of all disease codes in a topic). B, The last ten topics in Figure 5.3 were plotted alone using a different color range to highlight their minor differences.

3.3.5 Inference results given by treeLFA models with different numbers of topics

On the top 100 UKB dataset, treeLFA models with different numbers of topics were trained so that we can study the change of inferred topics across models. Figure 5.3 shows the averaged inferred topics given by selected treeLFA models with different numbers of topics. For all these models an empty topic was inferred. For the model with two topics (Figure 5.3A), the only non-empty topic is a very dense topic, which undoubtedly was used to explain the occurrences of all diseases. As the number of topics increases, sparse topics begin to appear (Figure 5.3B), indicating more specific multi-morbidity patterns were learnt by the model. For the model with 100 topics, after the application of the 2-step post-processing procedure (Louvain clustering and hierarchical clustering) discussed in the previous section, 32 clusters of topics remained, indicating a limited number of meaningful multi-morbidity patterns exist in the data. By comparing the inference results across models, we can see that many topics were repeatedly inferred by different models. Specifically, we checked and matched every topic inferred by models with 20, 50 and 100 topics, and found that 19 topics inferred by the model with 20 topics were also inferred by the model with 50 topics (the remaining one topic is a near empty topic, which may not be of significance), and all topics inferred by the model with 20 topics were also inferred by the model with 100 topics. This result suggests that important topics will not be missed if excess topics are provided to the model.

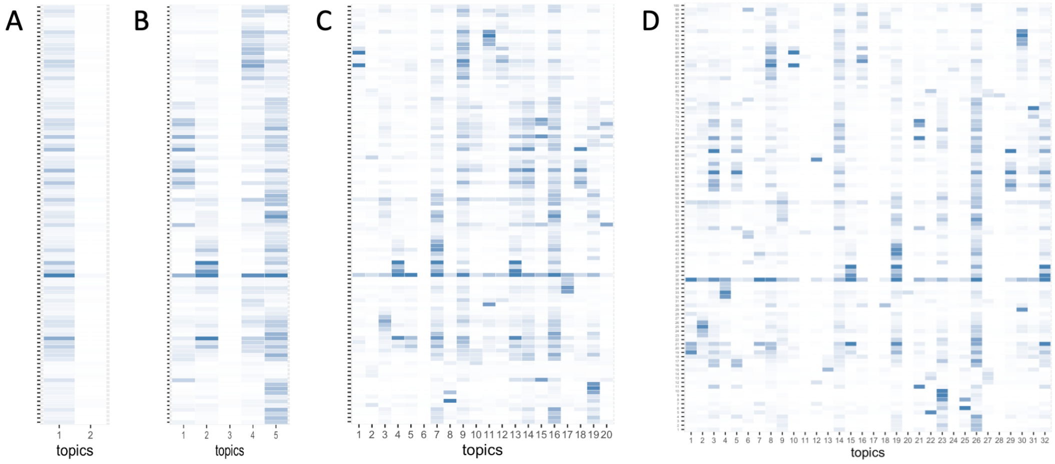


Figure 3.11: Averaged inferred topics given by treeLFA models set with different numbers of topics. All posterior samples of topics from all chains were mixed and clustered, and the averaged topics were calculated for these clusters as the final point estimate of topics for these models. A, Averaged topics given by the model set with two topics. B, Result given by the model with 5 topics. C, Result given by the model with 20 topics. D, Result given by the model with 100 topics.

To provide a high-level summary of the inference results for all models, the number of clusters of topics given by different models and the predictive likelihood for these models on the testing dataset were plotted in Figure 3.12. For models with less than or equal to 20 topics, the number of clusters of topics almost always equals the original number of topics set for the models, which means with more topics added, more distinct multi-morbidity patterns were learnt from the data. For models with 50 and 100 topics, many fewer clusters of topics remained after clustering of posterior samples of topics, indicating redundancy among inferred topics. As the number of topics increases, the predictive likelihood also keeps increasing, but the slope of the curve rapidly decreases (Figure 3.12B). However, it should be noted that there is unavoidable error in estimating the predictive likelihood with the current algorithm, especially for models with large numbers of topics. Consequently, it is not reliable to do model selection purely based on the predictive likelihood.

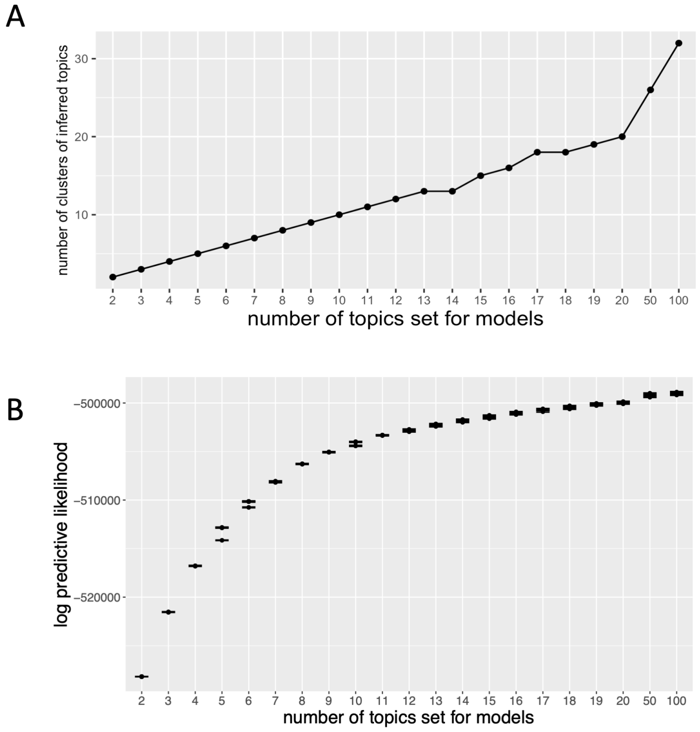


Figure 3.12: Summary of inference results for models with different numbers of topics. A, The numbers of clusters of topics for different models. B, The predictive log-likelihood on the testing dataset for different treeLFA models.

We also evaluated the stability of the inference results given by different chains for different treeLFA models via visualization of the posterior samples of topics. For the model with 100 topics, only ten posterior samples of topics from each Gibbs chain were used so that the size of the input data was not excessive for the tSNE algorithm. For other models, all 50 posterior samples from each Gibbs chain were used.

In Figure 3.13 the posterior samples of topics for selected treeLFA models were shown. For models with few topics (Figure 3.13A-B), the clusters of topics are well-defined. For the model with 20 topics there are also 20 clusters in total, but for some clusters of topics different chains gave slightly different inference results, which is reflected by distinct small sub-clusters formed by topics assigned to the same cluster (Figure 3.13C). For the model with 100 topics (Figure 3.13E-F), there is a huge cluster in the middle of the figure, which is the cluster of empty and near-empty topics. Meanwhile, most of the remaining clusters of topics still have comparable sizes, indicating stable non-empty topics were inferred.

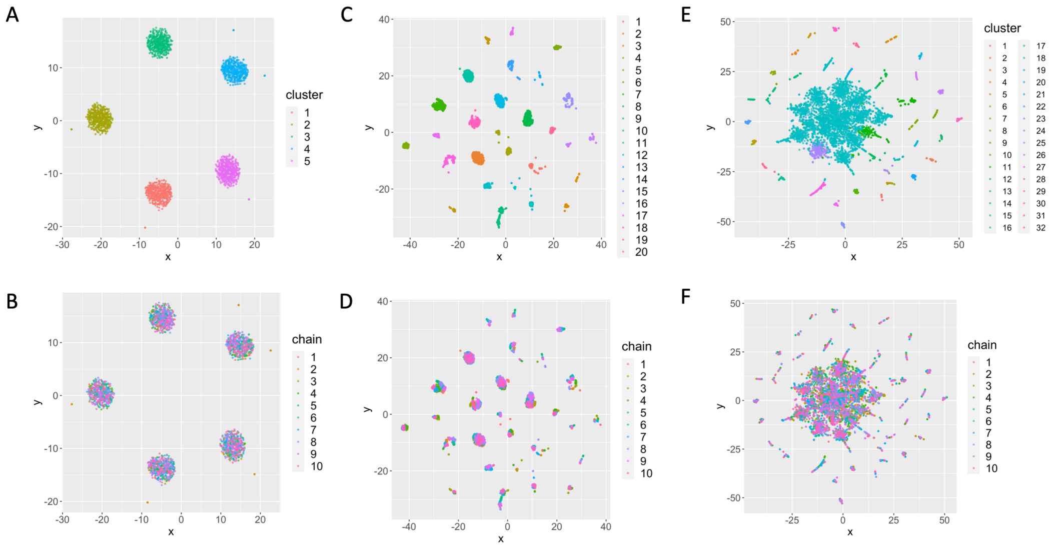


Figure 3.13: Visualization of posterior samples of topics for different treeLFA models. A, Clusters of topics for the model with five topics. Topics are colored according to the clusters they are assigned to. B, Result for the model with five topics. Topics are colored according to the Gibbs chains they come from. C-D, Results for the model with 20 topics. E-F, Results for the model with 100 topics.

3.3.6 Relatedness of topics in different models

As discussed in Section 3.3.5, many topics were consistently inferred by treeLFA models, even with quite different numbers of topics. This raises the question of how the inferred topics evolve as the number of topics for the treeLFA model varies. In this section, we are going to use an concise and straightforward way to consider how topics inferred by different treeLFA models are related to each other.

To provide insight into how topics evolve across treeLFA models with different numbers of topics, we used a tree structure to summarize the relationships of inferred topics from different models. In Figure 3.14 we connected topics from different models to form a tree structure. Each node on the tree is an inferred topic from a treeLFA model. There are 21 layers of nodes on the tree, which correspond to the 21 treeLFA models (models with 2-20,50 and 100 topics). Nodes in each layer correspond to all topics inferred by a specific model. Each node (topic) on the tree is connected to its most similar one in the above layer. In other words, for each topic in the model set with $\boldsymbol{N}$ topics, we found its most similar topic (its parent topic) in the model set with $\boldsymbol{N-1}$ topics. The similarity between two topics was defined using the Pearson correlation between the two topic vectors $\boldsymbol{\phi}$.

To make the tree of topics more understandable, in Figure 3.14A a single branch of the tree is highlighted with red colour, and all topics in this branch are plotted in order in Figure 3.14B. This means that each red node in Figure 3.14A corresponds to a column (topic) in the heatmap in Figure 3.14B. Although these topics come from different treeLFA models they resemble each other, which enables us to track the change of topics across different models. The first few topics in Figure 3.14B are dense (have a large number of active codes), while the following ones are more sparse. This is because the first few topics come from the models set with very few topics, while the following topics come from models set with a large number of topics. It is also noteworthy that from the 8th topic in Figure 3.14B, the subsequent topics almost stay the same. This means that for all models set with more than nine topics, this specific topic was repeatedly inferred.

In addition to using the Pearson correlation to measure the similarity between topics, other similarity metrics were also considered; the tree of topics built using them are shown in Figure 3.15. In general, we obtained similar tree structures using different metrics for the similarity between topics, although some minor differences exist.

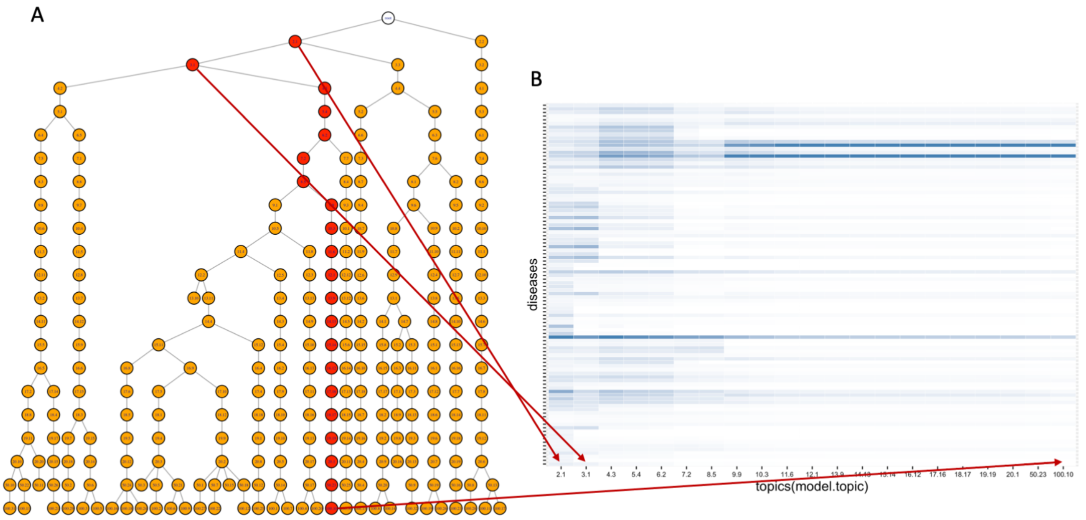


Figure 3.14: Tree structure of topics from different models. Pearson correlation was used to define the similarity between topics. A, The tree structure of topics from all models (models with 2-20, 50 and 100 topics). Each node on the tree is a topic and nodes in each layer of the tree correspond to topics inferred by a model. Each topic is connected with its parent topic (the most similar topic) from the model set with one fewer topics. B, All topics in one branch of the tree (corresponding to all red nodes in Figure 3.14A) are displayed. Each topic is named using the model it comes from and its index in the model. For instance, topic 20.1 means the first inferred topic from the model set with 20 topics.

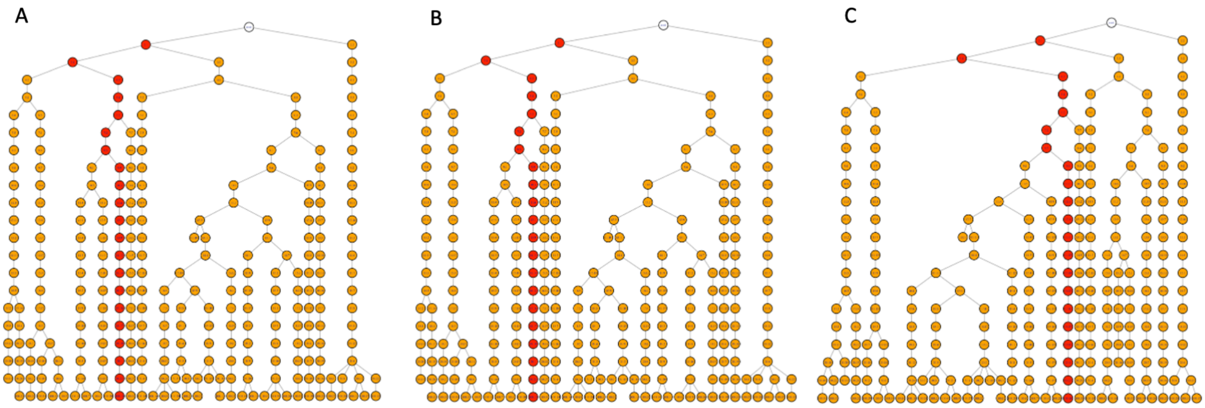


Figure 3.15: Tree structure of topics built with other similarity metrics for topics. A, The tree structure of topics built using the cosine distance to measure dissimilarity between topics. B, The tree structure built with the extended Jaccard distance. C, The tree structure built with the Manhattan distance.

3.4 Subtyping common diseases

The inferred topics and topic weights for individuals enable many downstream analyses to deepen our understanding of the mechanisms of diseases. In the next chapter, the focus will be on using inferred topics for genetic study. In this section, we will make direct use of individual's inferred topic weights to study subgroups of common diseases.

3.4.1 Defining subgroups of diseases using individual's topic weights

Heterogeneity in symptoms, patterns of comorbidity and progression among people diagnosed with the same disease are frequently seen. There are many ways to define subtypes of diseases, such as using patient clinical manifestations [144], clinical lab results [145] and omics data[146], or a combination of all these data types. Since the output of treeLFA defines different patterns of disease comorbidity (in the form of topics), in this section we aim to examine whether these can be used to subtype diseases.

The assumption here is that different comorbidity (reflected in individuals' topic weights) for people diagnosed with a certain disease may hint at different subgroups of this disease. In our study, for all people diagnosed with a certain ICD-10 code in UKB, we defined subgroups for them by clustering their inferred topic weights ($\boldsymbol{\theta}$). As a result, people with similar weights for topics (thus similar profiles of comorbidity) will be put into the same cluster.

Figure 3.16 shows ten subgroups of people diagnosed with I10 (essential hypertension) in UKB identified by performing hierarchical clustering on individual topic weights. Ten was chosen for the number of clusters to find because their are ten disease topics in total. There are 76,727 British ancestry patients diagnosed with I10 in the training dataset. This number is too large for the matrix based hierarchical clustering, so we applied hierarchical clustering on 20,000 patients to define the ten subgroups, and then used the KNN (K-nearest neighbour) algorithm to assign the remaining patients to these subgroups. For each subgroup we randomly selected 1000 people and plotted their comorbidity profiles (their diagnostic status for all the 100 ICD-10 codes) in Figure 3.16B.

Among the ten subgroups of I10, Subgroup 2 is the largest, while other subgroups are quite small in size. People in different subgroups have distinct and characteristic comorbidity patterns. First of all, it is obvious that people in Subgroup 2 have the fewest comorbidities among all subgroups. This is reasonable, since people in this subgroup have large weights for Topics 1 and 4. Topic 1 is the empty topic. In Topic 4, most codes except for I10 have relatively small probabilities (Figure 3.5A), which means people with large weights for Topic 4 will be unlikely to be diagnosed with most ICD-10 codes. On the contrary, many people in Subgroup 3 were diagnosed with E78 (hyperlipidemia), I20 (angina) and I25 (Chronic ischaemic heart disease). This is because these codes are all top active codes in Topic 9 (Figure 3.5A) together with I10, and most people in this subgroup have large weight for Topic 9. Similarly, people in Subgroup 6 have large weight for Topic 6. This is also reflected in their comorbidities, since people in this subgroup have more disease codes related to the knee (M17 and M23, which are top active codes in Topic 6). Lastly, people in the last two subgroups have more broad spectra of comorbidity, which can be explained by their large weights for Topics 7 and 10. These two topics are dense topics with a large number of active codes (Figure 3.5A).

Similar subtyping results for E78 (Disorders of lipoprotein metabolism and other lipidaemias) are shown in Figure 3.17. For E78, there are two major large subgroups. People in Subgroup 1 have a large weight for Topic 9, and many are diagnosed with I20, I21 and I25. Meanwhile, people in Subgroup 4 have the fewest comorbidities, and large weight for Topic 4.

3.4.2 Genetic heterogeneity among subgroups of diseases

As a final exploratory analysis, we studied potential genetic heterogeneity among subgroups of a disease.

Firstly, for I10 we ran two case-control GWAS only using individuals having British ancestry in certain subgroups of I10 as cases (subgroup-GWAS). Covariates (sex, age and the first ten principle components) were controlled for the GWAS analyses, and only common SNPs (SNPs with minor allele frequency larger than 0.01) in UKB were used. The first GWAS considered all patients in Subgroup 2 (35,232 individuals, about half of all I10 patients) as cases, and the second GWAS used all patients who were not in Subgroup 2 (35,957 individuals, other I10 subgroups altogether) as cases. The same control group (all British people in UKB who were not diagnosed with I10, 266,279 individuals) were used for the two subgroup-GWAS analyses. Figure 3.18A shows the Manhattan plots for the two subgroup-GWAS. Both subgroup-GWAS gave a few unique GWAS hits. The GWAS

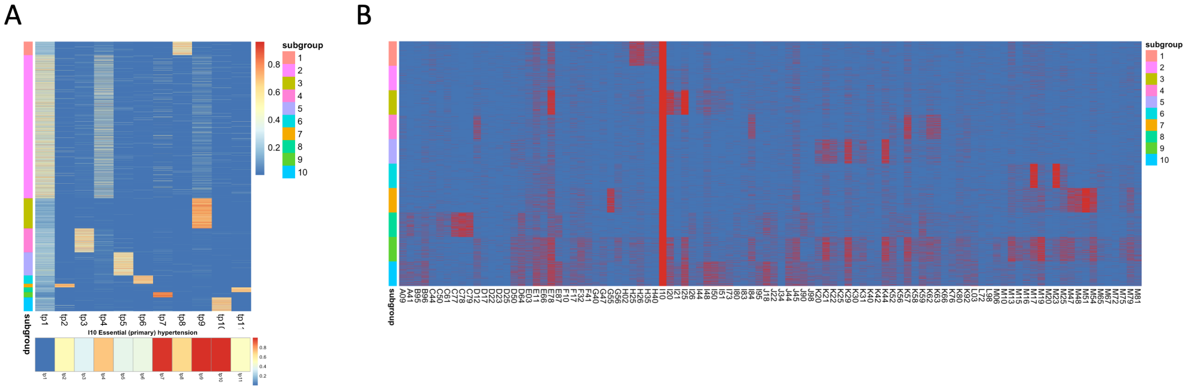


Figure 3.16: Subgroups of patients diagnosed with I10 (essential hypertension). A, I10 patients were divided into ten subgroups according to their inferred topic weights. Weights for the 11 topics were plotted for all people who were diagnosed with I10 in the training dataset (each row in the heatmap is a patient, and each column is a topic), together with their subgroup membership (colored bar to the left of the heatmap), as well as the probability of I10 in the 11 inferred topics (the lower heatmap). B, Comorbidity for 1000 random patients from each subgroup of I10. Each row in the heatmap is a patient and each column is an ICD-10 code. Red cells denote which ICD-10 codes were diagnosed for these people.

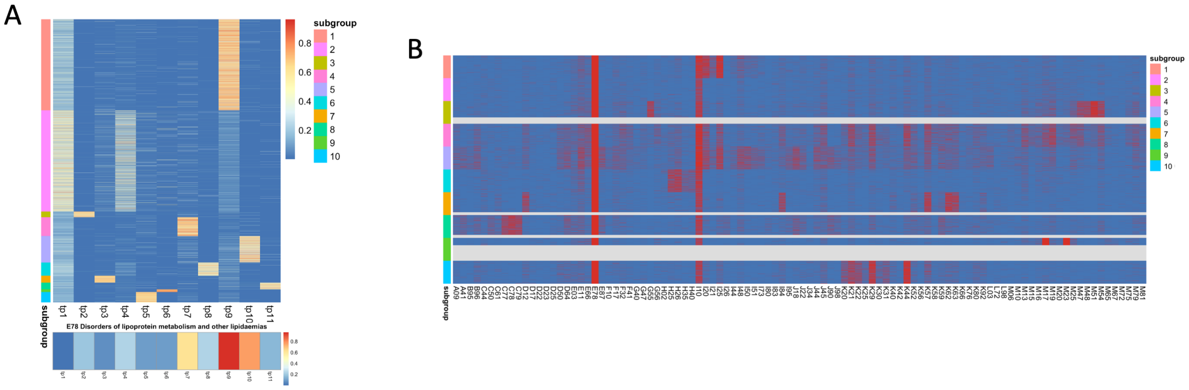


Figure 3.17: Subgroups of patients diagnosed with E78 (Disorders of lipoprotein metabolism and other lipidaemias). A, E78 patients were divided into ten subgroups according to their inferred topic weights. B, Comorbidity for 1000 random patients from each subgroup of E78.

results obtained on all British I10 patients (full-GWAS) were compared with these two subgroup-GWAS results. The full GWAS found 121 significant loci in total ($\boldsymbol{P < 5 \times 10^{-8}}$ used as the threshold for significant P-value, and $\boldsymbol{r^2 > 0.1}$ used in clumping of SNPs), while the two subgroup-GWAS found 46 and 30 significant loci respectively. Notably, three loci found by one of the subgroup-GWAS analyses were not found by the full GWAS. In Figure 3.18B the effect sizes given by the two subgroup-GWAS for the same SNPs (all significant SNPs found by either of the two subgroup GWAS) are plotted. It can be seen that, although for most SNPs the two subgroup-GWAS give similar effect sizes, there are also a few SNPs that have large effects for one subgroup of patients and very small effects for the other subgroup. This result suggests that some loci may only be associated with a fraction of I10 patients, instead of all I10 patients.

Similar results for E78 are shown in Figure 3.19, where we can see different significant loci found by GWAS for the two subgroups, and loci with significantly different effect sizes for the two subgroups.

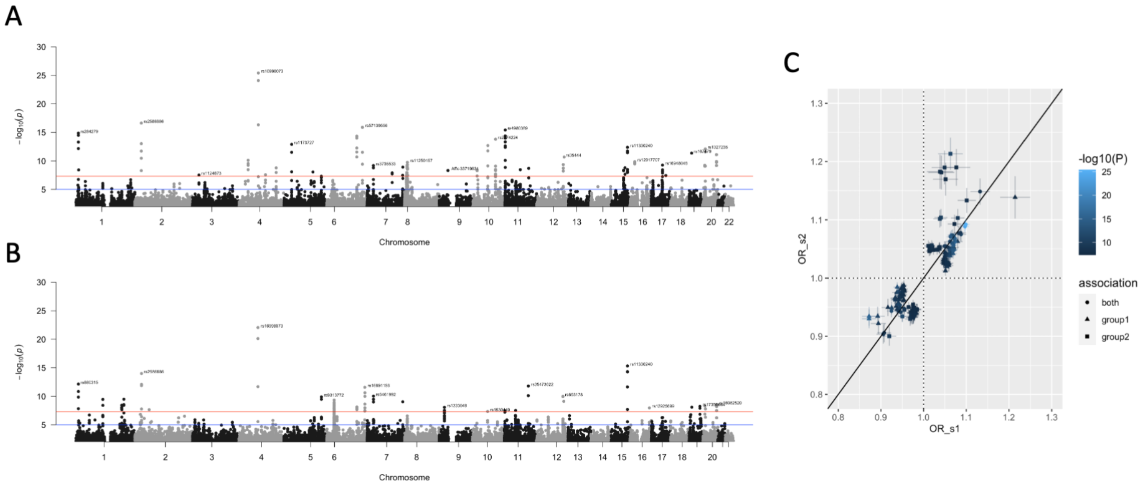


Figure 3.18: Subgroup-GWAS for I10. GWAS analyses were run for people in subgroup 2 of I10 and people in all other subgroups of I10 separately. People who were not diagnosed with I10 were used as the control group for the two subgroup GWAS. A-B, Manhattan plots for the two subgroup GWAS for I10. B, Effect sizes (Odds ratio: OR) given by the two subgroup GWAS for all SNPs found as significant by either of the subgroup-GWAS analyses.

3.5 Discussion

In this chapter, treeLFA was firstly applied to a restricted phenotypic dataset constructed using individuals' diagnosed ICD-10 codes in UKB (the top 100 UKB dataset). Topics of the 100 ICD-10 codes and topic weights inferred by treeLFA were analysed, and the inference results given by the three topic models were compared. In addition, we explored the insights provided by the post-processing methods for treeLFA's inference results developed in Chapter 2 . We also studied how the inferred topics change across models with different numbers of topics. We proposed an innovative way to describe the relationships of topics inferred by different models, and found that most topics are stable across models if the models are set with enough topics. Lastly, we explored the possibility to subtype common diseases using

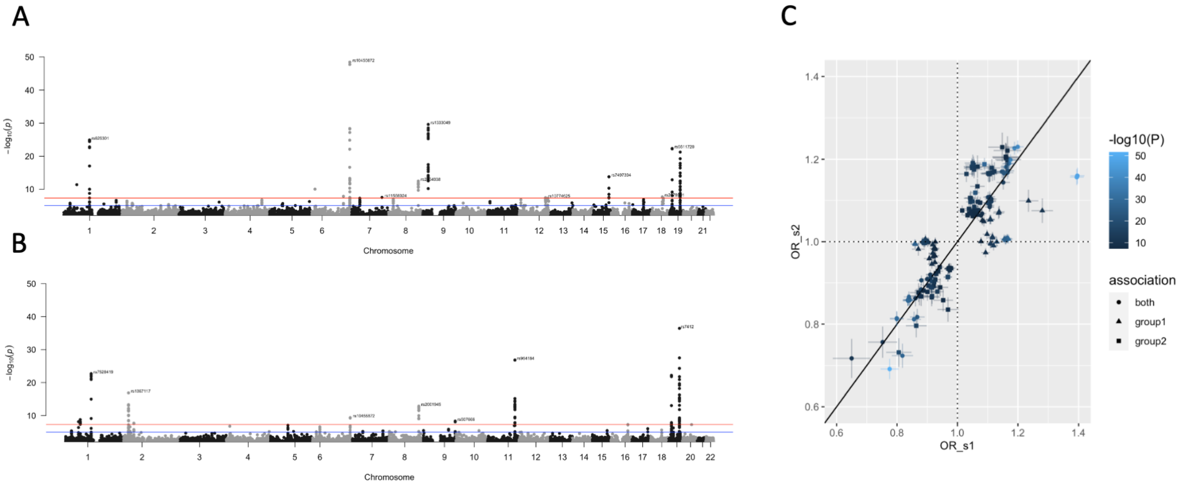


Figure 3.19: Subgroup-GWAS result for E78. GWAS analyses were run for people in subgroups 1 and 2 of E78 separately. Poeple who were not diagnosed with E78 were used as the control group for the two subgroup GWAS. A-B, Manhattan plots for the two subgroup GWAS. B, Effect sizes (Odds ratio: OR) given by the two subgroup GWAS for all SNPs found as significant by either of the subgroup-GWAS analyses.

individuals' inferred topic weights, and as an example we found potential genetic heterogeneity among different subgroups of two ICD-10 codes (I10 and E78).

3.5.1 The input data

In this chapter, the input dataset (top100 dataset) for treeLFA was constructed on the basis of the HES data in UKB encoded using the ICD-10 coding system. The HES data has a few limitations. Firstly, the coding of diseases can be inaccurate for some patients, and the misdiagnoses were not specifically accounted for (for instance, using the number of times a diagnosis was made). Secondly, ICD-10 also may not be most ideal choice for the coding system. One of the reasons is that in many cases the resolution of ICD-10 codes is too high, which decreases the power of association studies and the ability for topic models to infer topics. In this study, we chose to use the ICD-10 codes on the fourth layer of the hierarchical structure, such that subtypes of diseases were combined. However, we didn't verify that this procedure was appropriate for all the involved terminal ICD-10 codes. It should be noted that Phecodes may be a better choice for research purpose, and should be considered in future studies.

3.5.2 Inferred topics with UKB data

In this chapter, topics of the 100 most frequent ICD-10 codes in UKB were inferred. Among the inferred topics there is always an empty topic, whose weight is negatively correlated with people's total number of diagnosed codes. The remaining topics can be further divided into two types: dense topics and sparse topics, which have large and small numbers of active codes respectively. Sparse topics undoubtedly reflect that occurrence of a small number of highly correlated diseases. As for the dense topics, it is possible that they are mainly used by the model to describe the occurrences of a large number of unrelated diseases, or they may also reflect that some risk factors are shared by a large number of diseases, thus their occurrences are more strongly correlated compared to the other ones. In the following chapters, dense topics will be further studied.

By checking the top active codes in topics, it can be seen that most topics focus on one or two ICD-10 chapters, which supports the rationale of commonly used disease classification systems (such as the ICD-10 coding system). However, for most topics there are also codes from ICD-10 chapters other than the dominant one, which demonstrates that data-driven approaches like treeLFA can be a good supplement to the current disease classification system built upon expert medical knowledge.

The topics inferred by treeLFA, flatLFA and LDA were similar in general, implying common multi-morbidity patterns were captured by the three different topic models. Besides, it also indicates that for common diseases in the biobank-scale datasets, the size of data should be large enough such that the prior knowledge about topics no longer have very important roles in inference, which means the inference is mainly driven by the data. However, this may not hold for larger phenotypic datasets containing more relatively rare diseases, as will be seen in chapter 5.

3.5.3 Model selection strategy for treeLFA

In this chapter, treeLFA models with different numbers of topics were trained. In Section 3.3.5, the predictive likelihood for different treeLFA models were calculated and compared. For models with relatively small numbers of topics, as the number of topics set for models increased, the predictive likelihood also increased sharply, suggesting a significant improvement in the ability of the models to explain the data. For models set with relatively large number of topics, the increase in the predictive likelihood with more topics provided was small. Considering that the precision of predictive likelihood estimates given by the naive Monte-Carlo algorithm implemented in this thesis will get worse when there is a large number of topics, it may not be reliable to choose the best model according to the predictive likelihood alone. In other words, a more advanced algorithm for the calculation of predictive likelihood or other model selection strategies (such as prediction tasks on the testing data) might be needed to select the single best model.

It is also noteworthy that the treeLFA model with 100 topics had larger predictive likelihood than all the other models with less topics. 100 topics is apparently too many, therefore in theory this model should have worse performance compared to models with set enough and less topics. However, as was mentioned in Section 3.3.4, when an excess number of topics is set for treeLFA, multiple empty (or near empty/very sparse) topics will be inferred. Having multiple empty/near empty topics in theory should not largely affect the ability of the model to fit the data. Besides, this also does not affect the inference of other meaningful disease topics by the model. Taking all this into account, although the lack of a peak in Figure 3.12

may be an artefact, it still indicate that providing an excessive number of topics to the model would not severely affect the ability of the model to fit the data.

In Section 3.3.6, inferred topics given by all models were organized into a tree structure to describe their relationships. The results show that for models set with a small number of topics, the inferred topics change largely from model to model. However, once the number of topics set for a model is large enough, most inferred topics become stable across models. As was discussed in Chapter 2, this implies a practical and simple model selection strategy for treeLFA, which is to train one model with very large number of topics, and then cluster posterior samples of all topics to remove duplicated ones. With this strategy, we may only need to train one model instead of trying many different values for the number of topics, which is one of the most important hyperparameters for topic models. Up to now, this strategy has been verified with both simulated and real-world data. Besides, this result also adds to the reliability of our inference results, since we obtained similar results by fitting slightly different models.

In the next chapter, this strategy will be further validated with genetic analyses based on the inferred topics and topic weights, before it is finally put into full use in Chapter 5.

3.5.4 Subtyping common diseases

In the last section of the chapter, we tried to define subgroups for all people who were diagnosed with a certain ICD-10 code by clustering their inferred topic weights. For most diseases, we found that the number of stable clusters roughly equals the number of topics inferred. An interesting phenomenon is that for many disease codes there are one to two major subgroups which contain a very large percent of patients. For instance, for I10 (essential hypertension) about half of the patients are in Subgroup 2, and for E78 about one third of patients are in subgroups one and two respectively. These results might indicate that there are stable subgroups for many common diseases which may have distinct pathological components.

After finding the subgroups of common diseases, case-control GWAS was run on these subgroups separately (subgroup-GWAS). Results show that different subgroups may have different associated loci. GWAS was also run on all patients who were diagnosed with a certain code (full-GWAS), and although in most cases the significant loci found by subgroup-GWAS were also found by the full-GWAS, there were also exceptions. Besides, for those significant loci found by both subgroup-GWAS and full-GWAS, it is possible that some of them may mainly work on a specific subgroup, as was reflected by the different effect sizes given by different subgroup-GWAS for some loci. In general, the full-GWAS found more significant loci compared to the subgroup-GWAS, suggesting that in most cases increasing the sample size and joining different subgroups of patients can boost the statistical power for detecting associations. However, when it comes to understanding the functions of these loci, we should bear in mind that they may have heterogeneous effects.

Overall, these results indicate that there might be multiple pathological trajectories leading to the same endpoint (the acquisition of a disease). Furthermore, these results may also suggest the applicability of taking comorbidity into account when identifying subtypes of diseases.

4

GWAS for topics of diseases

Contents

4.1 Overview of the chapter

Using the inferred disease topics, along with individuals' weights for these topics, estimated from the previous chapter, various downstream analyses can be carried out to provide more insight about the mechanisms of diseases. In this chapter, we will focus on genetic association studies that make use of the inference results of treeLFA.

Traditionally, GWAS is performed for one trait (disease) at a time. As is discussed in Chapter 1, in recent years various multi-trait GWAS methods have been developed, aimed at finding genetic variants that have pleiotropic effects on multiple correlated traits with larger statistical power. Topics inferred by treeLFA summarize groups of diseases that may share common pathological factors, therefore individuals' weights for these topics can be used as newly defined traits for GWAS. To put it in another way, in theory an individual's weight for a topic of diseases can be considered as a quantitative measure of their abnormality for a pathological pathway that is related to multiple diseases.

In this chapter, we carried out GWAS on individuals' inferred topic weights as continuous traits (topic-GWAS). Meanwhile, we also ran the standard single code GWAS, and compared the results given by these two GWAS methods. Additionally, we compared the topic-GWAS results based on the topic weights inferred by the three related topic models, and we also studied the change of association signals across treeLFA models set with different numbers of topics. These comparisons are related to those reported in Chapter 3, but here they were carried out from a perspective of genetic association. To obtain more evidence supporting the validity of the topic-GWAS results, we validated the topics associated loci using a few different methods. Lastly, we also explored a few ways to further utilize the topic-GWAS results, including improving the prediction of risks for single diseases and jointly using the inference result given by treeLFA and currently available multi-trait GWAS methods.

4.2 GWAS on inferred topic weights as continuous traits

4.2.1 Methods for performing GWAS on topic weights

We used standard GWAS (linear regression) to find common genetic variants that are associated with weights for different topics (topic-GWAS). It is important to note that for this type of GWAS, the traits are not the presence/absence of a single ICD-10 code, but estimated weights for different topics, which are continuous real numbers that fall in the range of 0 to 1.

For the topic-GWAS, we only used common SNPs (SNPs with a minor allele frequency larger than 0.01 in UKB) and individuals who self-reported as having

British ancestry (343,006 people in total). We used standard linear regression (implemented with the PLINK-2.0) for the GWAS, with sex, age and the first ten principal components of genomic variation (PCs) controlled for. Since topic weights are real numbers between 0 and 1, technically the basic assumptions of standard linear regression do not hold, because topic weights are not allowed to take all real values. Moreover, the distributions of topic weights are usually very skewed, as can be seen in Figure 4.1. This means the assumption for linear regression that residuals should approximately follow a normal distribution also may not hold. The reason for this kind of distribution of topic weights is that normally for a disease topic, most people have very small weight (a weight that is close to 0), and only a small number of people have non-small weights because they were diagnosed with active codes in this topic.

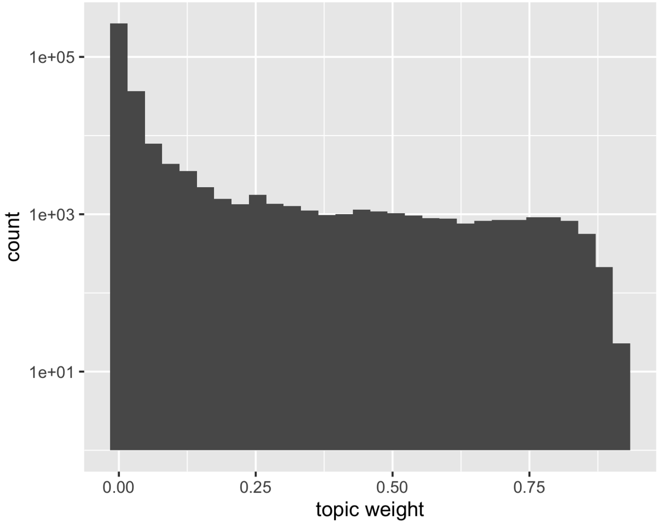


Figure 4.1: Distribution of inferred weights for Topic 9 from the treeLFA model with 11 topics on the top-100 dataset. The y-axis is on the log10 scale.

We considered various methods to deal with this issue. First of all, we applied a logit transformation on topic weights before running the GWAS. The logit transformation can map numbers between 0 and 1 to real numbers. Another option for the transformation of the response variable is using the rank based inverse normal transformation (INT) on topic weights, which is achieved by replacing sample quantiles with quantiles from the standard normal distribution. Lastly, we can also build a generalized linear model (glm), using a logit link function and a Gaussian error. It is noteworthy that this is different from using a logit transformation directly on the response variable. The former method (glm) assumes that the residuals of topic weights follow a normal distribution, and the mean of this distribution can be calculated by applying a logistic function on a linear combination of independent variables. On the contrary, the latter method assumes that the residuals of the transformed response variable follow a normal distribution, the mean of which is a linear combination of independent variables. The two equations below more clearly state the difference between these two methods.

$$\begin{aligned} \boldsymbol{g(Y) \sim Normal(\,(B_0 + B_1X_1 + ... + B_nX_n),\, \sigma^2\,)} \\ \boldsymbol{Y \sim Normal(\,g(B_0 + B_1X_1 + ... + B_nX_n),\, \sigma^2\,),} \end{aligned} \tag{4.1}$$

Here, $\boldsymbol{g()}$ is the logit function, $\boldsymbol{Y}$ is the response variable (topic weight), $\boldsymbol{E()}$ denotes the mean of a random variable, $\boldsymbol{B_i}$ represents the coefficients of the regression and $\boldsymbol{X_i}$ represents the independent variables (genotype and covariates).

The first equation above expresses applying a logit transformation on the response variable before fitting a standard linear model, while the second equation expresses the generalized linear model with a Gaussian error for the response variable and a logit link function.

In Figure 4.2, the distributions of standard residuals for topic weights resulted from running topic-GWAS using the four methods discusses above with the same input data (topic weights, genotype data and covariates) are plotted. The four GWAS methods include linear regressions without transformation and with logit/INT transformation of the response variable, as well as the glm method. Without any transformation (Figure 4.2A), the distribution of the standardized residuals is right skewed, which should be caused by the small number of people with very large weights for the topic. Moreover, the left-hand part of the distribution is also not bell-shaped. As for the generalized linear model (Figure 4.2B), the distribution is markedly different from a Gaussian distribution. Figure 4.2C shows the residuals after applying the logit transformation on the topic weights. The left-hand part of the distribution looks more like a Gaussian distribution, while there is still a heavy right tail. Lastly, Figure 4.2D shows the result for the inverse rank transformation on topic weights. The distribution of residuals is symmetric and very much like the standard Gaussian distribution. Overall, these results indicate that the glm is not an ideal method for running topic-GWAS, while the other three methods should be acceptable. Further comparisons of these three methods will be given in the following sections.

4.2.2 Single-code GWAS

To compare the topic-GWAS results with the results given by standard GWAS using single disease codes as binary traits (single code GWAS), we also ran standard logistic regression (PLINK-2.0) for all the 100 ICD-10 codes on the same cohort. In addition, we also ran single code GWAS using the phecodes as traits. Phecodes are defined by systematically grouping ICD-10 codes into more applicable medical terms based on the judgements of clinicians and researchers, which reduces the granularity of ICD-10 codes [147], as is illustrated in Figure 4.3. It is also worth mentioning that phecodes also have a hierarchical structure like ICD-10 codes, in which several terminal phecodes are placed under a common parent code which represents a more general class of diseases. To map the 100 ICD-10 codes to phecodes, we firstly extracted all the terminal ICD-10 codes which correspond to the 100 level-4 ICD-10 codes we used for treeLFA (details can be seen in the Section 3.2.1), and then mapped them to terminal phecodes. In total, there were 296 terminal phecodes mapped from the 100 ICD-10 codes used.

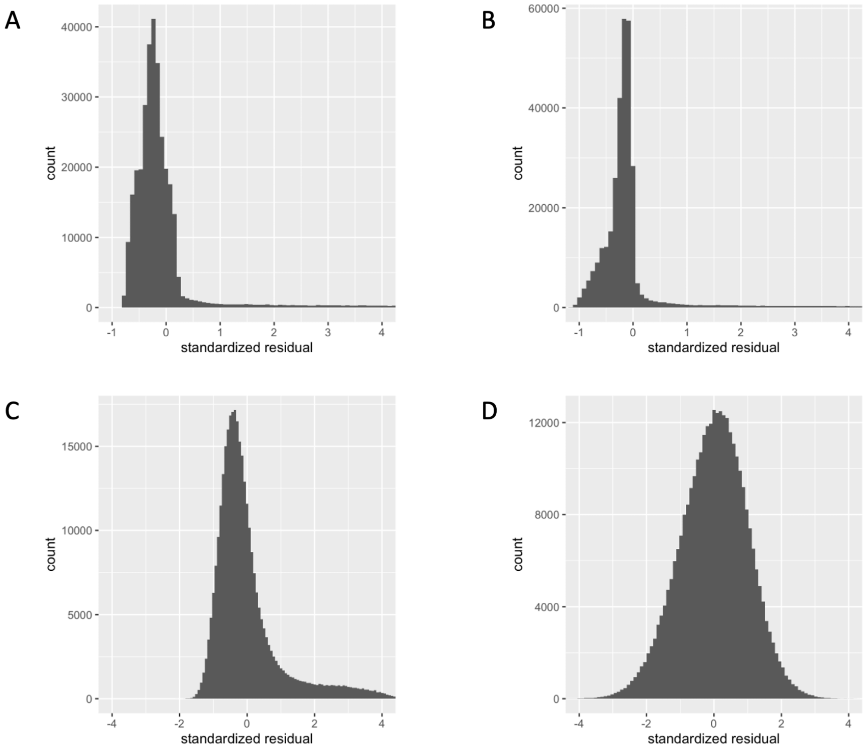
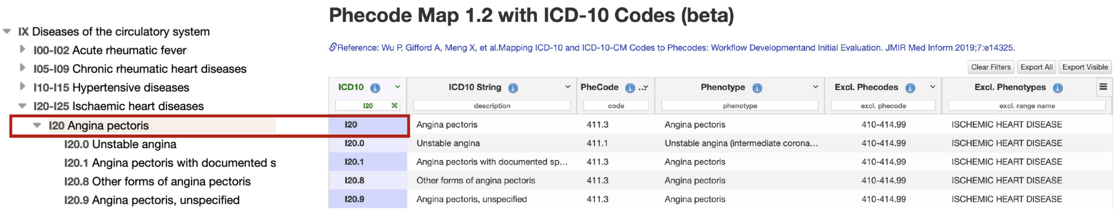


Figure 4.2: Distribution of the standardized residuals for different topic-GWAS methods on the same input data. A, No transformation of topic weight was applied before fitting the regression model. B, Generalized linear model was fitted with Gaussian errors for topic weights and a logit link function. C, Logit transformation on topic weights. D, Rank transformation on topic weights.



Phecode Map 1.2 with ICD-10 Codes (beta)

Reference: Wu P, Gifford A, Meng X, et al.Mapping ICD-10 and ICD-10-CM Codes to Phecodes: Workflow Developmentand Initial Evaluation. JMIR Med Inform 2019;7:e14325.

- IX Diseases of the circulatory system
  - I00-I02 Acute rheumatic fever
  - I05-I09 Chronic rheumatic heart diseases
  - I10-I15 Hypertensive diseases
  - I20-I25 Ischaemic heart diseases
    - I20 Angina pectoris
      - I20.0 Unstable angina
      - I20.1 Angina pectoris with documented s
      - I20.8 Other forms of angina pectoris
      - I20.9 Angina pectoris, unspecified

ICD10	ICD10 String	PheCode	Phenotype	Excl. Phecodes	Excl. Phenotypes
I20	description	code	phenotype	excl. phecode	excl. range name
I20	Angina pectoris	411.3	Angina pectoris	410-414.99	ISCHEMIC HEART DISEASE
I20.0	Unstable angina	411.1	Unstable angina (intermediate corona...	410-414.99	ISCHEMIC HEART DISEASE
I20.1	Angina pectoris with documented sp...	411.3	Angina pectoris	410-414.99	ISCHEMIC HEART DISEASE
I20.8	Other forms of angina pectoris	411.3	Angina pectoris	410-414.99	ISCHEMIC HEART DISEASE
I20.9	Angina pectoris, unspecified	411.3	Angina pectoris	410-414.99	ISCHEMIC HEART DISEASE

Figure 4.3: Mapping between ICD-10 codes and phecodes. Both coding systems have a hierarchical structure. In most cases, terminal ICD-10 codes have higher granularity compared to terminal phecodes. For example, there are four subtypes of angina (highlighted in the red box) according to the ICD-10 coding system. On the contrary, only two phecodes are mapped by the four ICD-10 codes related to angina.

4.2.3 Processing GWAS results

To define genomic loci from SNPs that are significantly associated with topics, we clumped all significant SNPs found by the topic-GWAS using the clumping function implemented in PLINK-1.9, using $\boldsymbol{r^2 > 0.1}$ as the threshold for linkage disequilibrium (LD), and $\boldsymbol{P < 5 \times 10^{-8}}$ as the threshold of P-value for lead SNPs in these loci.

To check the overlap of significant loci found by topic-GWAS and single code GWAS, we clumped the lead SNPs of significant loci found by one GWAS method (single code GWAS or topic-GWAS) to the lead SNPs of significant loci found by the other method. The number of loci (lead SNPs) that can be clumped with loci found by the other method is the number of significant loci found by both methods. As before, PLINK-1.9 was used, with $\boldsymbol{r^2 > 0.1}$ the threshold for LD, and $\boldsymbol{P < 5 \times 10^{-8}}$ the threshold for P-values of lead SNPs.

4.2.4 Results for GWAS on topic weights

In this section, the topic-GWAS result for the treeLFA model with 11 topics on the top-100 dataset will be discussed in detail.

An example of the topic-GWAS result

Figure 4.4 shows the topic-GWAS results for the 11 topics inferred by treeLFA. Figure 4.4A-B show the Manhattan plot and QQ plot (quantile-quantile plot) for the topic-GWAS result of one topic (Topic 9). In Figure 4.4C, weights for this topic were permuted across individuals to check the topic-GWAS result under the null hypothesis of no association. In Figure 4.4D, the topic-GWAS results (number of topic associated loci) for all topics are shown. For each topic, the associated loci were divided into two groups, one group contained loci found by both the single code GWAS on all the 100 ICD-10 codes and the topic-GWAS for this topic, and the other group contained loci only found by topic-GWAS but not the single code GWAS for any of the 100 ICD-10 codes. Loci in these two groups were colored differently in Figure 4.4D. For all the 11 topics there are significant loci found by both the topic-GWAS and single code GWAS (in total 82 loci for the 11 topics), and for most topics there are also significant loci only found by topic-GWAS (in total 46 loci for the 11 topics), indicating the increment of statistical power for discovery joining diseases into topics.

Comparison of the results given by different topic-GWAS methods

As discussed in the previous section, we tried four different methods to perform topic-GWAS. Figure 4.5 compares the total number of topics associated loci (for all the 11 topics in total) found by the three topic-GWAS methods (no transformation of topic weights, logit and rank transformation on topic weights). Among the three methods, the topic-GWAS using logit transformation on topic weights found the largest number of significant loci (128 loci), followed by topic-GWAS using no transformation (103 loci). topic-GWAS using rank based INT on topic weights found the least number of significant loci (72 loci). The overlap of loci found by the topic-GWAS using logit transformation and the other two methods are plotted in Figure 4.5. Overall, the overlap of significant loci found by different methods

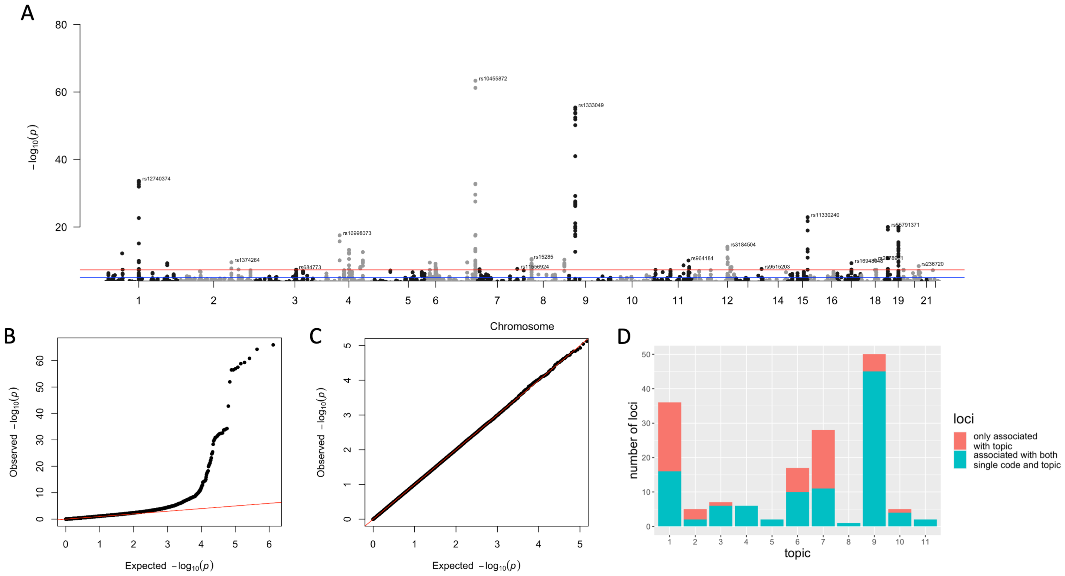


Figure 4.4: topic-GWAS results. A, The Manhattan plot for the topic-GWAS result for Topic 9 from the treeLFA model with 11 topics. B, QQ-plot for the topic-GWAS result of Topic 9. C, QQ-plot for the topic-GWAS result of Topic 9 with the topic weights permuted. D, The total numbers of significant loci found by topic-GWAS for the 11 inferred topics, and among them the numbers of topics associated loci that are found as significant by both the topic-GWAS and single code GWAS (blue bars), as well as the numbers of topics associated loci that are only found as significant by topic-GWAS (red bars).

is large (around 80 % of the loci found by other methods will also be identified by topic-GWAS using logit transformation).

To further validate the use of logit transformation on topic weights, QQ plots for topic-GWAS results obtained using the three transformation methods are compared in Figure 4.6. There is no significant difference in the patterns of the three QQ plots, indicating the heavy right tails in the residual plots (Figure 4.2) caused by applying logit transformation or no transformation on topic weights do not significantly inflate the P-values. Overall the P-values given by topic-GWAS using rank transformation are larger than the other two methods, suggesting a loss of power as a result of applying the rank transformation.

Inflation of P-values are observed for all three methods. This can either be resulted from true polygenicity of the trait, or stratification in the population. To differentiate these two possibilities, we carried out the LD score regression (LDSC) [148] using the summary statistics of topic-GWAS for the 11 topics, and compared the genomic control inflation factor $\boldsymbol{\lambda_{GC}}$ and the intercept of LDSC. A large $\boldsymbol{\lambda_{GC}}$ and small intercept (given by LDSC) for the same trait suggest true polygenicity instead of stratification. In Table 4.1, for all topics the intercepts of LDSC are much smaller than the $\boldsymbol{\lambda_{GC}}$. Among the 11 topics, topic-1 (the empty topic) and topic-7 (one of the two dense topics) have relatively large inflation in P-values caused by stratification. This is reasonable since compared to other sparse disease topics these two topics are related to the risks of much more diseases. As a result, they could also have more un-controlled confounding.

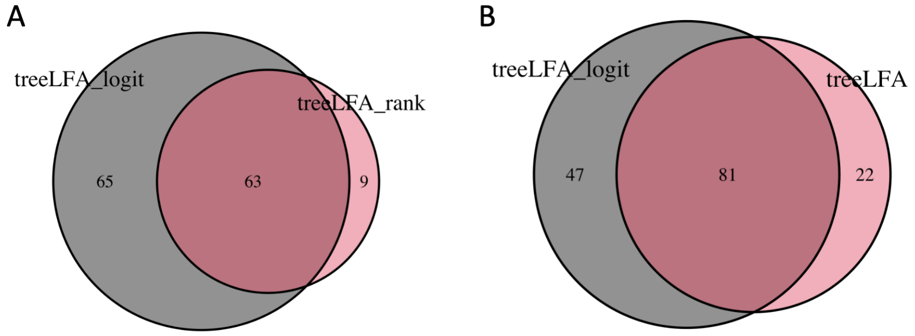


Figure 4.5: The overlap of significant loci found by three topic-GWAS methods. A, The total number of significant loci ($\boldsymbol{P < 5 \times 10^{-8}}$) found by topic-GWAS using logit transformation on topic weights and rank based inverse normal transformation on topic weights, and the overlap of these two sets of loci. B, The total number of significant loci found by topic-GWAS using logit transformation on topic weights and no transformation, and the overlap.

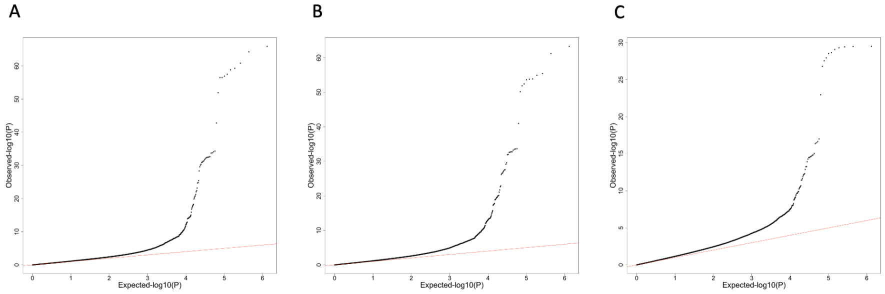


Figure 4.6: QQ plots for results of topic-GWAS with different transformations on traits. A, QQ plot for the topic-GWAS result for topic-9 without transformation on topic weights. B, QQ plot for the topic-GWAS with logit transformation. C, QQ plot for the topic-GWAS with rank-based inverse normal transformation.

In the following sections, we will use the linear regression with a logit transformation on topic weights as the standard topic-GWAS method, since the distribution of residuals and QQ plot are appropriate. Overall, this method enabled us to find more significant loci than the other two methods (topic-GWAS with no transformation or rank transformation on topic weights).

Topic	1	2	3	4	5	6	7	8	9	10	11
λ_{GC}	1.45	1.19	1.12	1.16	1.16	1.17	1.4	1.08	1.26	1.25	1.06
Intercept (LDSC)	1.1 (0.012)	1.02 (0.011)	1.02 (0.011)	1.02 (0.011)	1.03 (0.01)	1.01 (0.011)	1.1 (0.014)	1.02 (0.009)	1.05 (0.012)	1.04 (0.012)	1.02 (0.001)

Table 4.1: Genomic control inflation factors ($\boldsymbol{\lambda_{GC}}$) and LD score regression intercepts for the 11 topics. Values in the brackets following the intercepts are the standard deviations of estimates.

Comparison of the results given by topic-GWAS and single-code GWAS

For the 100 ICD-10 codes and the 296 phecodes mapped from the 100 ICD-10 codes, we ran the single code GWAS. Meanwhile, with the 11 topics inferred for the 100 codes, we also ran topic-GWAS. Figure 4.7 compares the total number of significant loci found by the topic-GWAS and single code GWAS using ICD-10 codes and phecodes as traits. It can be seen that about two thirds of the topic-associated loci are also associated with at least one ICD-10 code and phecode. However, a large number of loci (90 % of ICD-10 codes associated loci and 89 % of Phecodes associated loci) are associated with single codes but not topics, which suggests that most loci are specific to individual diseases, and only a minority of loci are shared by multiple diseases, therefore being associated with topics.

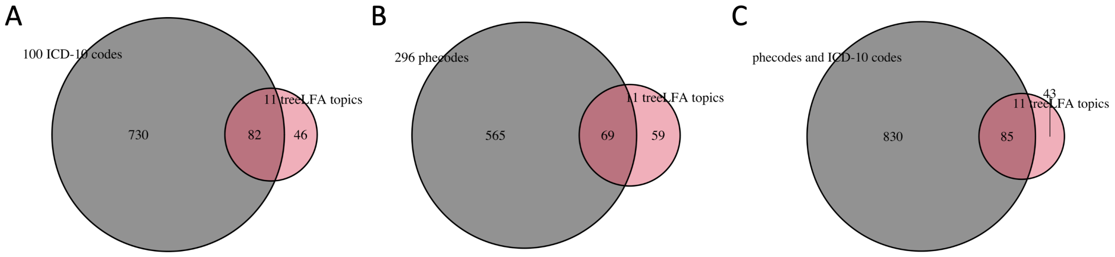


Figure 4.7: The overlap of significant loci found by topic-GWAS and single code GWAS. A, The total number of significant loci ($\boldsymbol{P < 5 \times 10^{-8}}$) found by topic-GWAS and single code GWAS using ICD-10 codes as traits, and the overlap of these two sets of loci. B, The total number of significant loci found by topic-GWAS and single code GWAS using phecodes, and their overlap. C, The overlap of significant loci found by topic-GWAS and GWAS using either ICD-10 codes or phecodes.

Defining new traits related to multiple disease codes using existing clinical ontology

treeLFA learns topics of diseases based on the co-occurrence of diseases, which in essence defines new traits that are related to multiple diseases in a data-driven way. Another common option to group multiple diseases into new traits is to use an expert-led method. For instance, based on the hierarchical structure encoded in the available disease classification systems, we can easily join several diseases (subtypes of diseases) into a single trait for GWAS.

To be more specific, we used all the internal codes (blocks of categories of diseases and chapters of diseases) of the two disease classification systems (ICD-10 and phecodes system) as binary traits for GWAS. For instance, if codes A and B are both under a common parent code C on the hierarchical structure of disease codes, then C will be used as the new trait, and all patients who were diagnosed with either A or B will all be positive for C. For the 100 ICD-10 codes there are 68 internal codes in the above two layers, and for the 296 phecodes there are 136 internal codes.

GWAS was run on these internal codes as binary traits, and the overlap of internal codes associated loci and topic-associated loci were plotted in Figure 4.8. With internal codes used as traits, fewer significant loci were found compared to GWAS using terminal codes as traits. However, there are still many topic-associated loci that cannot be identified by the single code GWAS on internal codes as traits, which indicates the added value of data-driven approaches like treeLFA.

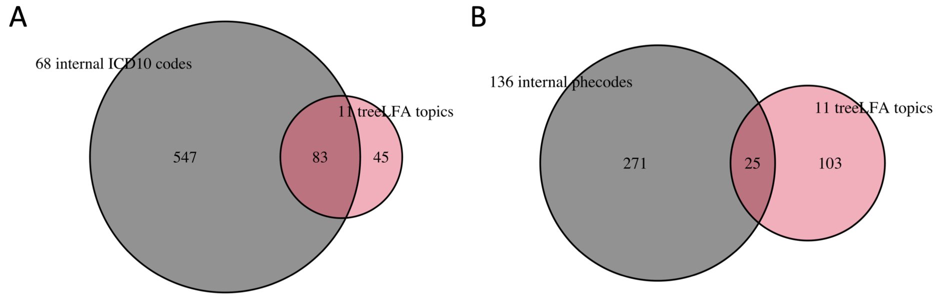


Figure 4.8: The overlap of significant loci found by topic-GWAS and single code GWAS for categories of codes (internal nodes on the tree structure of codes). A, Numbers of significant loci found by topic-GWAS and single code GWAS with internal ICD-10 codes as traits. B, Numbers of significant loci found by topic-GWAS and single code GWAS with phecodes as traits.

Comparison of topic-GWAS results based on topic weights inferred by the three related topic models

In Chapter 3, we trained three types of related topic models on the top-100 dataset, and found that they inferred similar topics. In particular, treeLFA and flatLFA's inferred topics were almost identical. On the other hand, the topics inferred by treeLFA and LDA are slightly different. In this section, we compared the topic-GWAS results for the three models.

In addition to topics inferred by treeLFA, topic-GWAS was also run on topic weights for the 11 topics inferred by flatLFA and the 10 topics inferred by LDA. For LDA, only people with at least one diagnosed disease code can be used as input for inference. For topic-GWAS, there are two options to deal with individuals without any diagnoses. We can either exclude them from the topic-GWAS or we can include them and give them small random weights for all the ten disease topics. We considered both methods, and found that excluding these healthy people resulted in larger power for topic-GWAS. With healthy people excluded, 68 significant loci were found for the ten disease topics. With them included, only 41 significant loci were found. Based on this result, we decided not to include healthy people into the topic-GWAS for LDA.

Figure 4.9 shows the topic-GWAS results for the three topic models. Topics are realigned such that the similar topics inferred by the three models are adjacent. We find that treeLFA and flatLFA topics have very similar numbers of significant loci, which is reasonable considering topics inferred by the two models are almost identical. Meanwhile, treeLFA topics identify more significantly associated loci than LDA topics. In addition, two thirds of the loci associated with LDA topics are also associated with treeLFA topics, suggesting running topic-GWAS with treeLFA's inference results provides us with increased power.

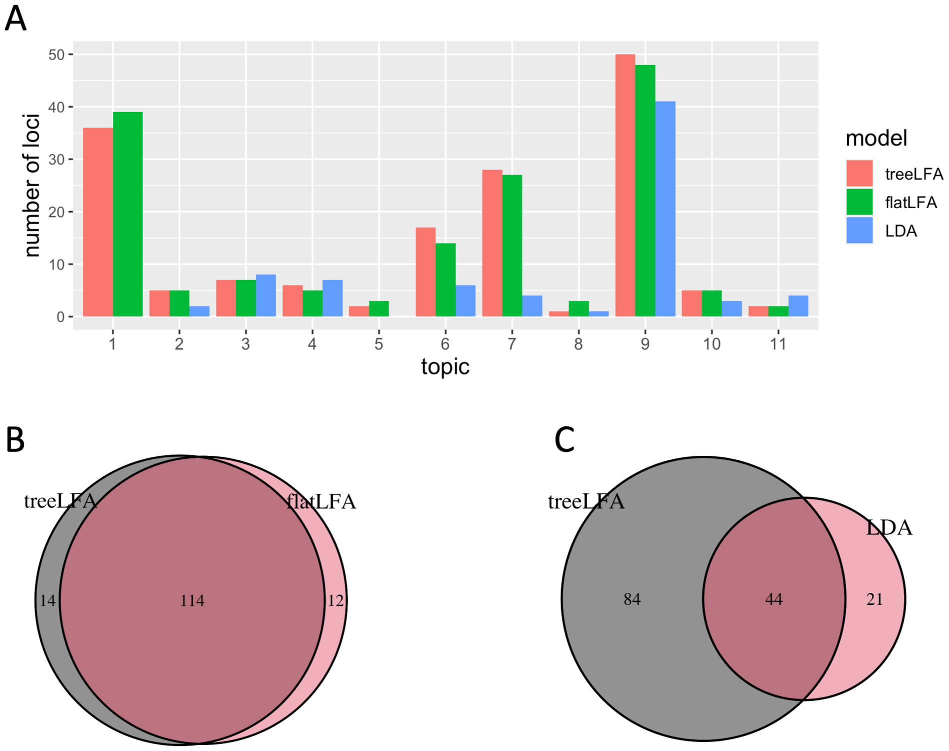


Figure 4.9: Comparison of topic-GWAS results for the three related topic models. A, The numbers of significant loci ($\boldsymbol{P < 5 \times 10^{-8}}$) for topics inferred by treeLFA, flatLFA and LDA. The first topic is the empty topic, which was only inferred by treeLFA and flatLFA but not LDA. B, The total numbers of significant loci for all the treeLFA and flatLFA inferred topics, and their overlap. C, The total numbers of significant loci for all the treeLFA and LDA inferred topics, and their overlap.

4.2.5 Topic-associated loci

After performing topic-GWAS we checked detailed association signals in the topic-associated loci. The software "locuszoom" [149] was used to generate regional Manhattan plots for topic-associated loci. To compare the P-values given by the topic-GWAS and single code GWAS for the same set of SNPs, regional Manhattan plots for single code GWAS results for the top active codes in the corresponding topic were also generated.

Figure 4.10 shows the regional plots for a significant locus for Topic 9 inferred by treeLFA as well as the regional plots for this locus for five top active codes in this topic (E78, I10, I20, I21, I25). The lead SNP in this locus is rs1374264, which is associated with the topic but not any of the top five active codes in the topic. In fact, this SNP is not associated with any of the 100 ICD-10 codes. However, for codes I10 and I25, the P-values for their lead SNP are smaller than $\mathbf{10^{-5}}$, which is considered suggestive. This demonstrates that running topic-GWAS provides larger statistical power for certain loci. Figure 4.11 shows another example (the lead SNP is rs1183910). However, in this case the lead SNP is not only associated with the topic but also the code E78 (Disorders of lipo-protein metabolism and other lipidaemias). It is noteworthy that the P-value given by the topic-GWAS for this SNP is larger than the P-value given by the single code GWAS for E78, indicating the single code GWAS for this variant may have larger power. Figure 4.12 plots the P-values for all the 50 Topic 9 associated loci (lead SNPs) given by the topic-GWAS and the single code GWAS for the five top active codes in Topic 9. For 49 among the 50 topic-associated loci, there are no more than two single code GWAS give smaller P-values for the lead SNP than the topic-GWAS.

4.2.6 topic-GWAS results for models with different numbers of topics

In Section 3.3.6 we studied the evolution of inferred topics across treeLFA models with different numbers of topics. We found that when the number of topics is sufficiently high, the inferred topics are stable across different models. Based on the previous results, in this section we studied how the strength of genetic associations varies across models. To do this, we carried out topic-GWAS for all topics in all treeLFA models (models set with 2-20, 50 and 100 topics).

First, we checked the total number of loci associated with all topics in different models, as well as the number of loci that are only associated with topics but not single ICD-10 codes (Figure 4.13). With increasing numbers of topics, the total numbers of topic-associated loci also increases. In contrast, for different models the numbers of loci that are only associated with topics but not any ICD-10 code do have such an increasing trend. However, this doesn't necessarily mean that all models found the same set of loci that are uniquely associated with topics (for instance, different models can have different topic-associated loci).

To visualize the associations between one SNP and topics in different models, the tree structure of topics from different models (see Section 3.3.6) was used again. For all topics on the tree, regressions of their topic weights against the genotype

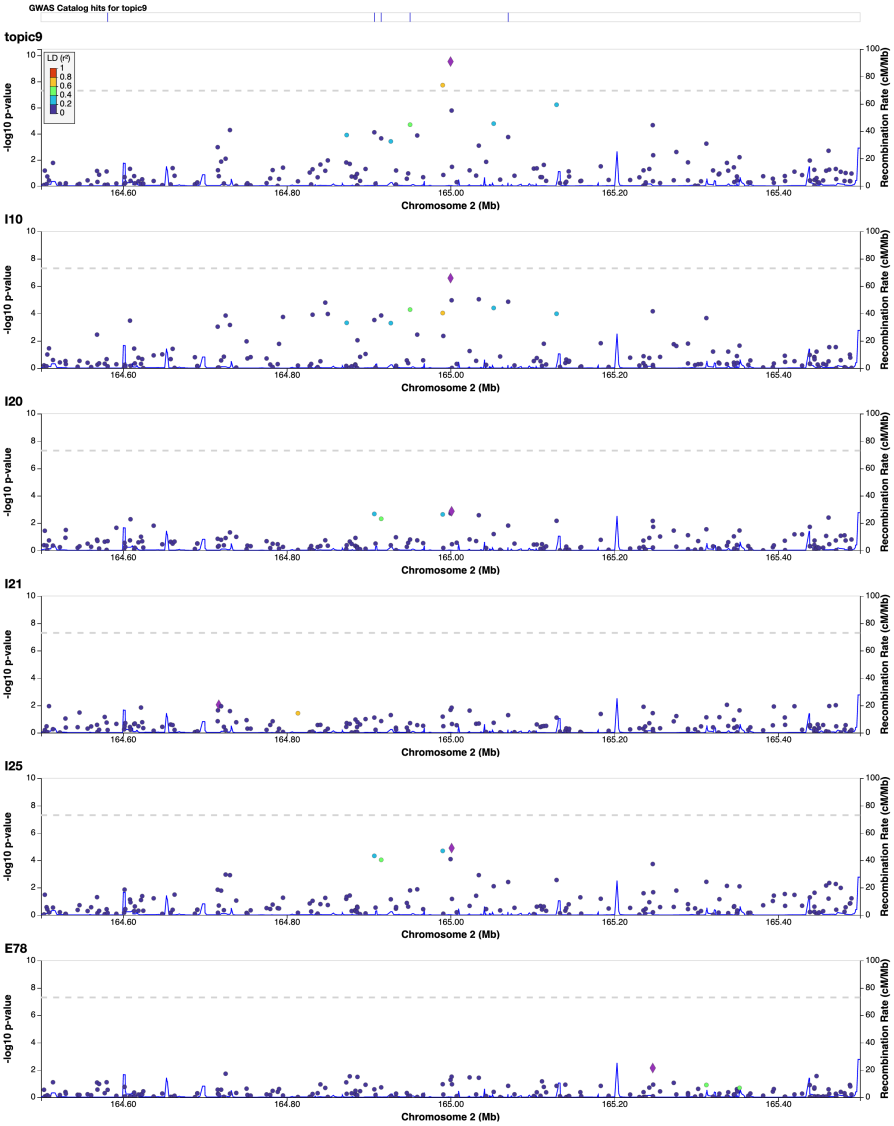


Figure 4.10: Regional plots for a significant locus for Topic 9 and five top active codes in the topic. The lead SNP of this locus is rs1374264 (2:164999883). SNPs within this locus having records in the GWAS catalog are marked on the top of the figure.

for the SNP of interest were performed, controlling for standard covariates (age, sex and the first ten PCs). In Figure 4.14, the results for two SNPs are shown. Nodes (topics) on the tree were colored using the -log10 (P-value) for the genotype variable in the corresponding regressions.

This procedure was repeated for many SNPs, and we discovered that most SNPs were mainly associated with a single branch of nodes in the tree. This suggests the genetic associations are also stable across the tree structure of topics.

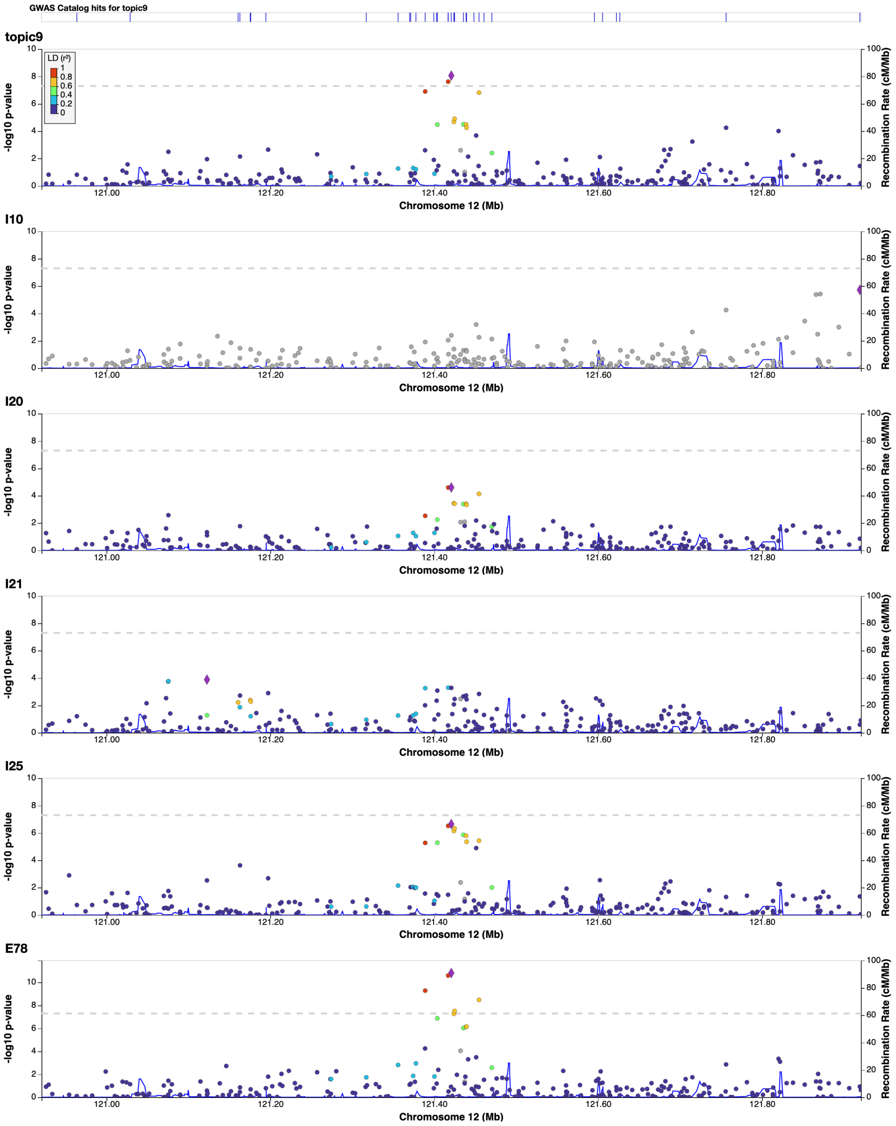


Figure 4.11: Regional plots for a significant locus for Topic 9 and five top active codes in the topic. The lead SNP of this locus is rs1183910 (12:121420807).

4.3 Validation of topic-associated loci

4.3.1 Replication of GWAS on the testing data

With the full UKB dataset split into a training (80 % individuals) and a testing (20 % individuals) dataset, for the purpose of validation we replicated GWAS on the testing data using both the 100 ICD-10 codes and the 11 topics as traits. Topic weights for individuals in the testing dataset were unknown and needed to be inferred. We ran the Gibbs sampler for treeLFA on them to get posterior samples of topic assignment variables ($\boldsymbol{Z}$) for them and approximated their topic weights ($\boldsymbol{\theta}$), with topics ($\boldsymbol{\phi}$) and $\boldsymbol{\alpha}$ fixed at values inferred from the training dataset. The genetic variants, covariates and methods used for GWAS were same as the training

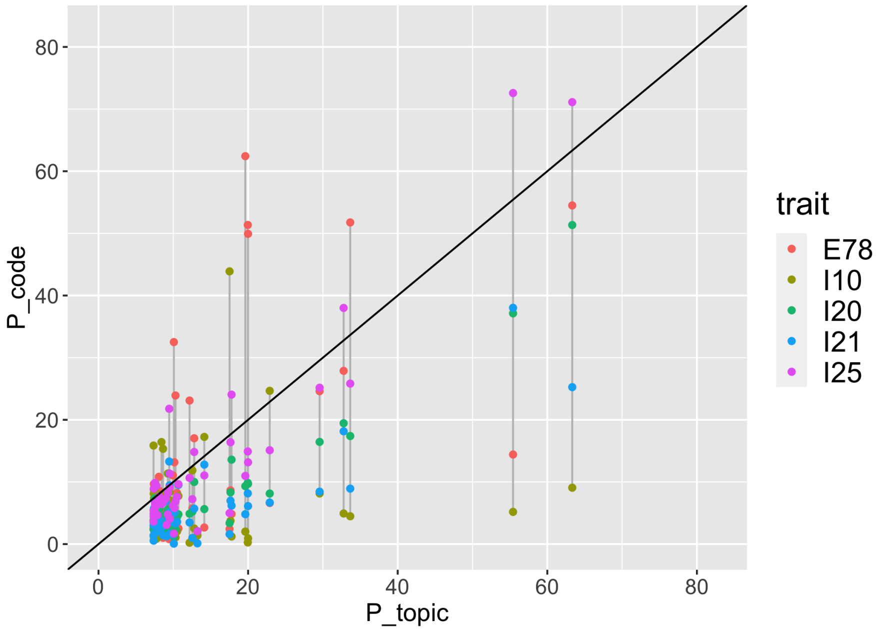


Figure 4.12: Comparison of the P-values for topic-associated loci obtained by single code GWAS and topic-GWAS. For the 50 lead SNPs associated with Topic 9, their -log10 (P-values) given by the topic-GWAS and the single code GWAS for the top five active codes in Topic 9 are plotted on x and y-axes respectively. Each line corresponds to a locus, and there are five nodes on each line, showing the P-values given by the five single-code GWAS for the same SNP (12:121420807).

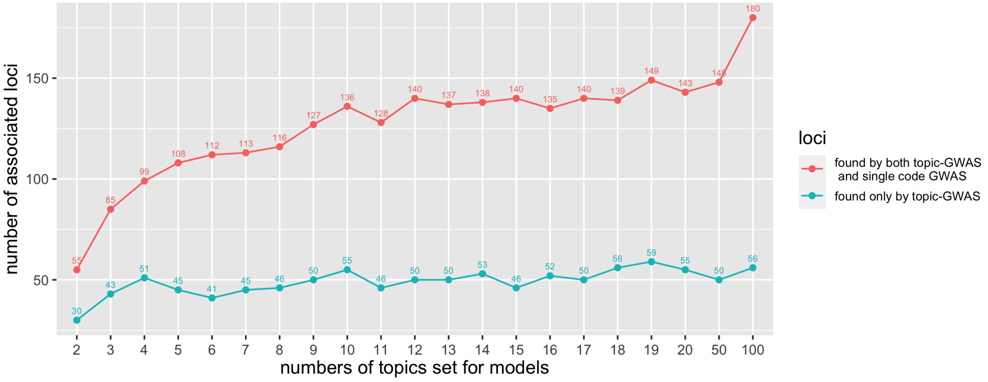


Figure 4.13: The total number of loci associated with any topic in treeLFA models. Models with 2-20,50,100 topics were trained and posterior samples of inferred topics were clustered. topic-GWAS was performed for all topics in these models. The red line shows the total number of loci associated with any topic in models, the blue line shows the number of loci found by topic-GWAS but not single code GWAS for any topic in these models.

data. In total, there were 88,157 British individuals in the testing data, which meant the testing dataset was about one fourth of the size of the training data.

P-values given by GWAS on the testing data for significant lead SNPs found on the training data were checked. There were 159 significant lead SNPs for the 11

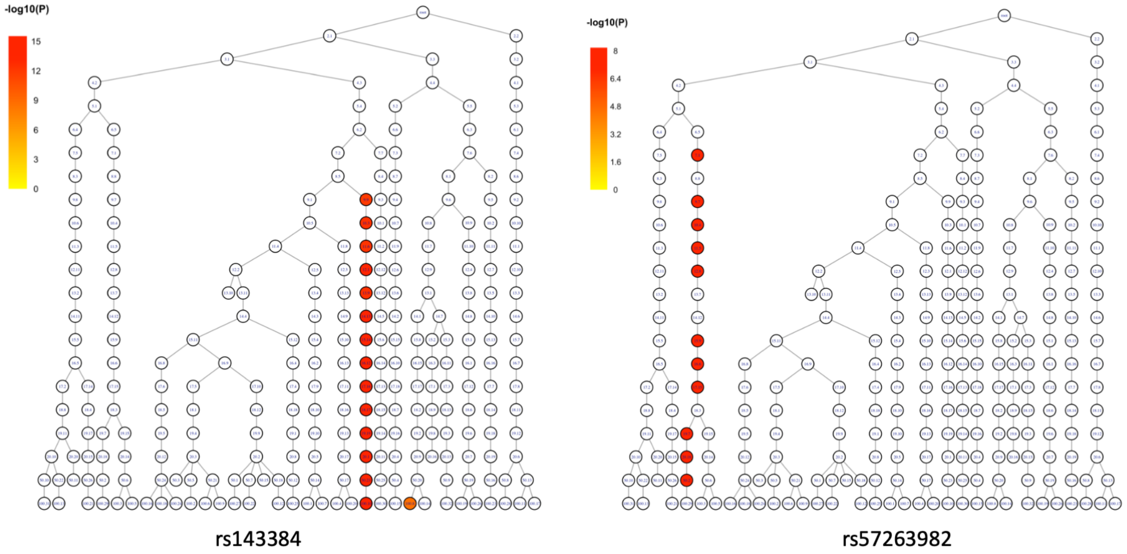


Figure 4.14: Associations between one SNP and topics in all models. Each node on the tree is a topic inferred by a specific model, and nodes in each layer of the tree correspond to all topics inferred by a model. Nodes are colored according to the P-value for the genotype variable in the corresponding regression. Only topics that are significantly ($\boldsymbol{P < 5 \times 10^{-8}}$) associated with the SNP of interest are highlighted with color. A, Result for SNP rs143384. B, Result for SNP rs57263982.

topics, and 1,168 significant lead SNPs for the 100 ICD-10 codes (some loci were associated with multiple topics/codes, and they were counted multiple times here, so these numbers were different from those in Section 4.2.4). The density plots for P-values given by single code/topic-GWAS had similar profiles (Figure 4.15), though differences also existed. Overall, P-values given by topic-GWAS were slightly larger than the P-values given by single code GWAS for significant lead SNPs found on the training data, but in general these results suggested that topic-GWAS and single code GWAS had similar generalization abilities.

4.3.2 Validation using the GWAS catalog

Due to the small size of the testing data, many discoveries made by single code/topic-GWAS were not replicated in the previous section. In addition to repeating GWAS on the testing data, there are other ways to validate topic-associated loci. Although none of these analyses would be decisive, they can still provide supporting evidence for the reliability of topic-GWAS results. In this section, we will use two other methods to validate the topic-associated loci.

Firstly, we used the GWAS catalog [150] as an external empirical criterion to validate the topic-GWAS results. The GWAS catalog is a publicly available, manually curated collection of GWAS results. At 1st September 2016 it had already

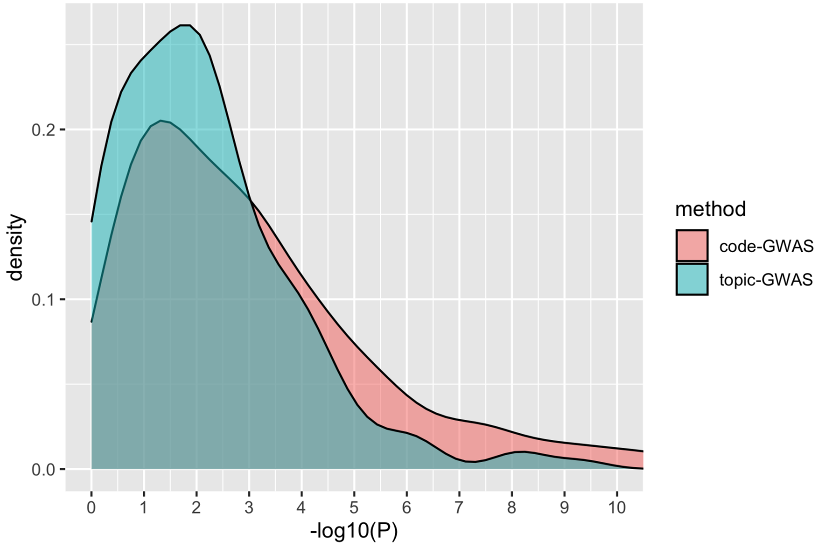


Figure 4.15: Density plots for P-values given by single code/topic-GWAS on the testing data. Both single-code/topic-GWAS were replicated on the testing data, using the 100 ICD-10 codes and weights for the 11 topics as traits. P-values of SNPs found by GWAS on the training data as significant lead SNPs were checked, and their density plots were shown.

contained 24,218 unique SNP-trait associations reported in 2,518 publications [151]. For our topic-associated lead SNPs, if SNPs in LD with them (SNPs in the same locus) had already been recorded in the GWAS catalog, then it will be more likely that the topic-GWAS results are reliable, since SNPs recorded in the GWAS catalog are more likely to be functional and have phenotypic associations compared to SNPs that are not in the GWAS catalog. Notably, SNPs in the GWAS catalog that were found by studies using UKB data were also included, and the threshold of P-value for SNPs in the GWAS catalog is $\mathbf{10^{-5}}$.

We downloaded the full GWAS catalog from its official website [150], and clumped all SNPs in the GWAS catalog to topics associated lead SNPs using PLINK ($\boldsymbol{r^2 > 0.5}$). Figure 4.16 shows the total numbers of loci (lead SNPs) associated with topics, the numbers of loci associated with topics but not any single code, and the proportions of them with a record in the GWAS catalog (which means at least one SNP in LD with it had a record in the GWAS catalog). As can be seen, for most topics at least 60% of their associated loci had already been discovered by other genetic studies. This is also the case for topic-associated loci that are not associated with any ICD-10 code.

4.3.3 Validation using functional genomics resources

With significant loci found by GWAS, usually the next thing to do is carrying out functional analyses to shed light on its biological functions. Our study does not intend to dive deep into the functional genomics analyses at the current stage. However, many standard functional analyses can be used to validate the topic

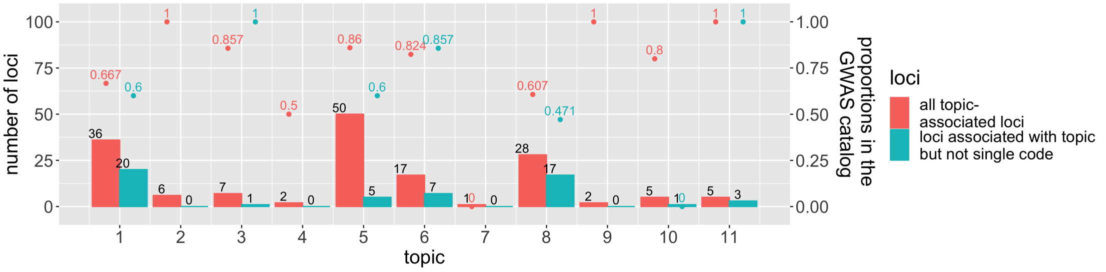


Figure 4.16: Proportions of topic-associated loci with a record in the GWAS catalog. The red and green bars show the total numbers of loci associated with topics, and the numbers of loci associated topics but not any single code. The dots denote the proportions of loci with a record in the GWAS catalog. For topics with no associated loci, the corresponding proportions are not shown.

associated loci.

Firstly, we can annotate the lead SNPs found by the topic-GWAS based on data in various genomic databases. If most topic-associated SNPs have high functional activities, then the reliability of topic-GWAS results will be further increased. At present there are a variety of scoring systems to annotate SNPs. For instance, the Regulome DB (rdb) score [152] is used to evaluate the regulatory potentials of SNPs. The latest version of the rdb score (a probability that a variant is regulatory) is given by a computational model which combines features from the original RegulomeDB database and the DeepSEA to predict the effects of variants on expression in promoters and enhancers [153]. Besides, we also use the chromHMM states [154] predicted by an algorithm based on epigenomic markers to classify the functional role of a genomic region in which a specific SNP sits. In different tissues the chromHMM states for the same locus can be different, and there are 127 tissues for which we have predicted chromHMM states. As rdb score, a smaller value for the chromHMM state means a genomic region has a more active functional role.

Currently, analyses such as annotations of SNPs can be easily done using software like “FUMA” [155], in which many sorts of functional analyses are implemented and integrated. Besides annotating SNPs, FUMA can also map SNPs to genes using different methods, including positional mapping, eQTL (expression quantitative trait loci) mapping and chromatin interaction (CI) mapping. With the mapped genes, a wide variety of downstream analyses such as gene set enrichment analysis and tissue enrichment analysis can be carried out. For the purpose of validation, we calculated the proportions of topic-associated lead SNPs that are eQTL or having CI in various tissues, and compared these proportions with the proportions of a large number of random SNPs.

Annotation of topic-associated loci

To assess if topic-GWAS found true genetic associations, we randomly selected 10,000 SNPs from all the common SNPs ($\boldsymbol{MAF > 0.01}$, 656285 SNPs in total) in

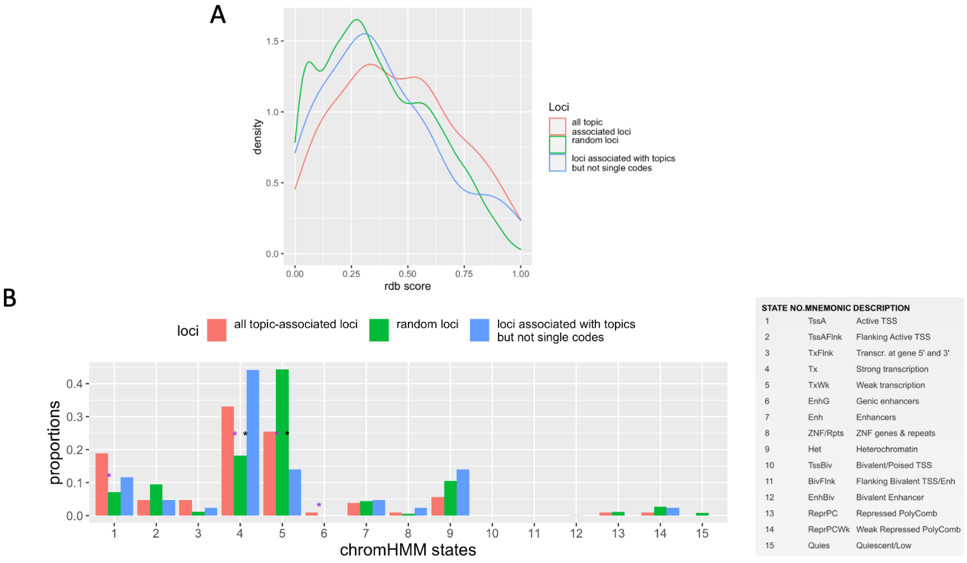
UKB as reference, and carried out the same functional analyses on three groups of SNPs: all lead SNPs associated with topic but not any single code, all topic-associated lead SNPs, and the random SNPs. Results for the three groups of SNPs are compared to see if the two groups of topic-associated lead SNPs have similar profiles that are different from that of the random SNPs. Since most topics only have a small numbers of associated loci, we combined the lead SNPs for different topics as the input for functional analyses. The software FUMA was used to carry out the functional analyses. The latest version of the rdb scores for SNPs were obtained from the official website (https://regulomedb.org/regulome-search/).

In Figure 4.17A, the distribution of rdb scores among the three groups of lead SNPs are compared. Overall, the two groups of topic-associated lead SNPs have larger rdb probability (of being functional variants) than random SNPs. However, the differences are not large. Besides, the group of all topic-associated SNPs have larger rdb scores than the group of SNPs only associated with topic. In Figure 4.17B the proportions of lead SNPs in different chromHMM states are shown for the three groups. The two proportion z-test was used to compare the proportions of random SNPs with the corresponding proportions for the other two groups. For chromHMM states 1 and 4 (correspond to transcription starting sites, or genomic regions that are strongly transcribed), topic-associated lead SNPs have much larger proportions than random SNPs. While for states 2 and 5, random SNPs have larger proportions. For the remaining states, the three groups have negligible differences. Overall, the rdb scores and chromHMM states for the three groups of SNPs provides some evidence that the two groups of topic-associated SNPs have different functional enrichment profiles compared to random SNPs. However, most differences between the random SNPs and SNPs associated with only topics are insignificant. These are likely to be caused by the small number of lead SNPs that are only associated with topics.

In Figure 4.18, the proportions of SNPs that are eQTL and have CI in various tissues are shown for the three groups. The eQTL data used in the analysis comes from the "GTEv8" dataset, and the CI data comes from the "GSE87112" dataset. It is very apparent that the profiles for the two groups of topic-associated lead SNPs are very similar, which are quite different from that for random SNPs. For most tissue types, the group of all topic-associated lead SNPs have significantly larger proportions compared to random SNPs, while the differences between random SNPs and lead SNPs only associated topics are mostly insignificant, which again suggests the lack of power for detecting statistical significance for this small group of SNPs.

4.4 Genetic risk prediction based on topic-GWAS result

In addition to the two methods discussed in the previous section, a more convincing way to validate the topic-GWAS results is to use them for prediction on the testing dataset. Because individual variants' effects on traits of interest are usually small, a common way to estimate individuals' genetic liability to diseases is to aggregate the effects of a large number of variants to build polygenic risk



STATE NO.	MNEMONIC	DESCRIPTION
1	TssA	Active TSS
2	TssAFlnk	Flanking Active TSS
3	TxFlnk	Transcr. at gene 5' and 3'
4	Tx	Strong transcription
5	TxWk	Weak transcription
6	EnhG	Genic enhancers
7	Enh	Enhancers
8	ZNF/Rpts	ZNF genes & repeats
9	Het	Heterochromatin
10	TssBiv	Bivalent/Poised TSS
11	BivFlnk	Flanking Bivalent TSS/Enh
12	EnhBiv	Bivalent Enhancer
13	ReprPC	Repressed PolyComb
14	ReprPCWk	Weak Repressed PolyComb
15	Quies	Quiescent/Low

Figure 4.17: Validation of topic-associated loci with functional annotations. A, The distribution of rdb scores for three groups of SNPs (all topic-associated lead SNPs, 10,000 random SNPs and lead SNPs associated with topic but not any single code) were compared. B, The proportions of the three groups of SNPs having different chromHMM states were compared. The chromHMM states for all genomics regions were predicted in 127 tissues. A genomic locus can be in different chromHMM states in different tissues. In our study we used the minimum chromHMM state across all the 127 tissues for each locus. The two proportion z-test was used to compare the proportions for random SNPs with the proportions for each of the other two groups of lead SNPs, and significant results were highlighted with $*$. The purple $*$ corresponded to the comparison between random SNPs and all topic-associated lead SNPs, while the black $*$ corresponded to the comparison between random SNPs and lead SNPs only associated with topic. Bonferroni correction was used to adjust for multiple hypotheses testing.

scores (PRS) for diseases [156].

4.4.1 PRS for topics

The full top-100 UKB dataset was split into a training dataset (80 % people) and a testing dataset (20 % people). With topics viewed as newly defined continuous traits, PRS for topic weights can be constructed based on topic-GWAS results on the training data, and evaluated on the testing data.

The software "PRSice-2" was used to build the optimal PRS for topics [157]. PRSice-2 uses a "C+T (clumping and thresholding)" method for constructing PRS, which means for each trait (topic) the software tries multiple thresholds for the P-values of SNPs, and then decides the best threshold. Meanwhile, clumping will be done for SNPs such that the selected SNPs are largely independent with each other. For each topic, the optimized PRS for all individuals in the testing dataset are output by PRSice, together with various summary statistics for the evaluation of the PRS.

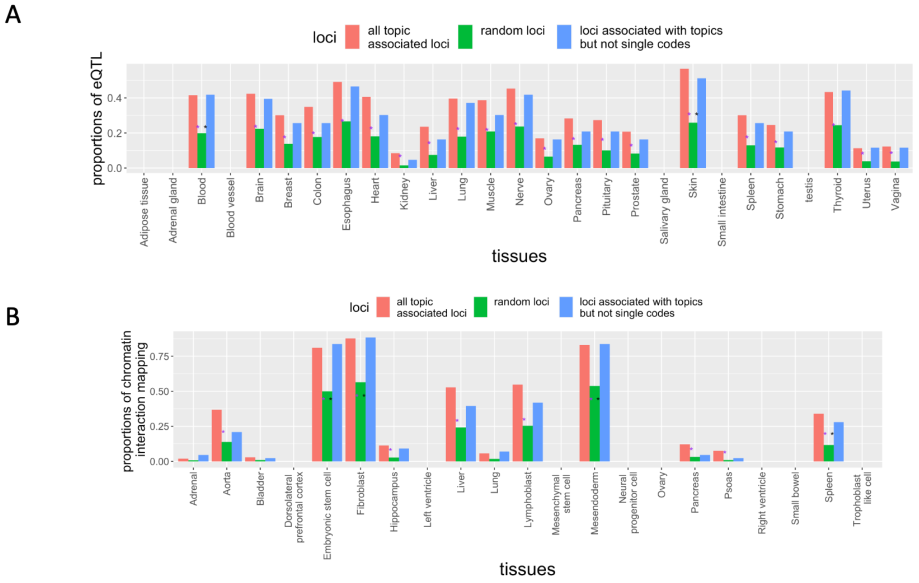


Figure 4.18: Validation of topics associated loci using eQTL and chromatin interaction data. The proportions of all topic-associated lead SNPs and lead SNPs only associated with topic but not any ICD-10 code that are eQTL in different tissues (GTEv8 dataset) were compared to the proportion of random SNPs. The two proportion z-test was used to find significant differences between proportions. Significant results were highlighted with $*$. The purple $*$ corresponded to the comparison between random SNPs and all topic-associated lead SNPs, while the black $*$ corresponded to the comparison between random SNPs and lead SNPs only associated with topic. Bonferroni correction was used deal with multiple hypotheses testing. B, The proportions of the three groups of SNPs that have chromatin interaction in different tissues (GSE87112 dataset) were compared.

Table 4.2 displays the performance of PRS for the 11 topics on the testing dataset, together with the heritabilities of the topics calculated using the LD score regression [148]. All PRS for topics had significant associations with the corresponding topic weights on the testing dataset. Both the PRS.R2 and the P-value indicate that PRS for Topics 1, 7 and 9 had the best performance on the testing dataset. In Figure 4.4 we can see these three topics have the largest numbers of significantly associated loci, which may indicate they have stronger genetic basis.

4.4.2 PRS for single codes based on topic-GWAS results

According to the generative process of treeLFA, to generate a positive disease code for a person, a topic will be firstly sampled for the corresponding disease variable and assigned to it. This means the generation of disease codes are mediated by multiple topics, and a person's probability of getting a disease code is related to the his or her weights for certain topics (in which the disease code is active).

topic	PRS.R2	Heritability	P	NUM_SNP	Threshold
1	0.021	0.09	0	224380	1
2	0.0056	0.04	$\mathbf{1.36 \cdot 10^{-110}}$	224544	1
3	0.0013	0.024	$\mathbf{8.04 \cdot 10^{-27}}$	68706	0.186
4	0.004	0.032	$\mathbf{8.89 \cdot 10^{-81}}$	224745	1
5	0.005	0.032	$\mathbf{3.13 \cdot 10^{-98}}$	224558	1
6	0.004	0.036	$\mathbf{2.59 \cdot 10^{-80}}$	121653	0.383
7	0.019	0.075	0	224386	1
8	0.0017	0.017	$\mathbf{7.74 \cdot 10^{-36}}$	93983	0.28
9	0.0099	0.068	$\mathbf{5.65 \cdot 10^{-212}}$	89123	0.244
10	0.0075	0.052	$\mathbf{2.05 \cdot 10^{-154}}$	128564	0.404
11	0.0006	0.01	$\mathbf{1.76 \cdot 10^{-13}}$	93153	0.28

Table 4.2: The performance of PRS for the 11 topics on the testing dataset. All results are output by the software PRSice-2, except for the heritability of topics, which are given by LDSC. PRS.R2: phenotypic variance explained by the PRS. P: P-value of the model fit. NUM_SNP: the number of SNPs used to construct the PRS. Threshold: threshold used by PRSice-2 for the P-values of SNPs.

Considering this relationship between topics and single disease codes, we seek to predict the risks of single disease codes using the topic-GWAS results.

In this section, we used PRS for topics to construct PRS for single ICD-10 codes. To be more specific, we constructed the PRS for an ICD-10 code as the sum of PRS for the 11 topics weighted by the probabilities of the ICD-10 code in the 11 topics. Meanwhile, as the benchmark we also constructed PRS for the 100 ICD-10 codes with single code GWAS results in the standard way. In summary, two types of PRS for ICD-10 codes were constructed, based on single code GWAS results and topic-GWAS results respectively. The performance of the two types of PRS were evaluated using their AUC on the testing dataset, and compared with each other.

In Figure 4.19, the performance of the two types of PRS are compared and the same results are presented in different forms. Figure 4.19A directly plots the AUC of the two types of PRS for the 100 ICD-10 codes, and Figure 4.19B plots the difference of AUC for the two types of PRS. For 66 ICD-10 codes, PRS based on topic-GWAS results (topic-PRS) have larger AUC than the PRS based on single code GWAS results (code-PRS), though the differences of AUC are usually not large (for most of these ICD-10 codes the increase in AUC range from 0.01 to 0.04).

Notably, ICD-10 codes from different chapters in the ICD-10 coding system have different patterns regarding the relative performance of the two types of PRS. For instance, for all the four ICD-10 codes from Chapter 5 (mental and behavioural disorders), the topic-PRS have larger AUC than code-PRS, and the differences of AUC are all quite large ($\boldsymbol{AUC_{diff}}$ = 0.017, 0.011, 0.023 and 0.038 for codes F10, F17, F32 and F47), suggesting shared genetic components for these diseases. Among the 20 ICD-10 codes from Chapter 13 (Diseases of the musculo-skeletal system and connective tissue), 14 codes have larger AUC for topic-PRS, and only six have larger AUC for code-PRS. A similar result was seen for ICD-10 codes from Chapter 11 (diseases of the digestive system), since 18 ICD-10 codes in this chapter have larger AUC for topic-PRS, and only three ICD-10 codes have larger

AUC for code-PRS. On the contrary, for Chapter 2 (Neoplasms) seven out of 11 codes have larger AUC for code-PRS.

In addition, most codes for which code-PRS performed better had large numbers of associated loci, and by contrast, most codes for which topic-PRS performed better had zero or very few associated loci (Figure 4.19C).

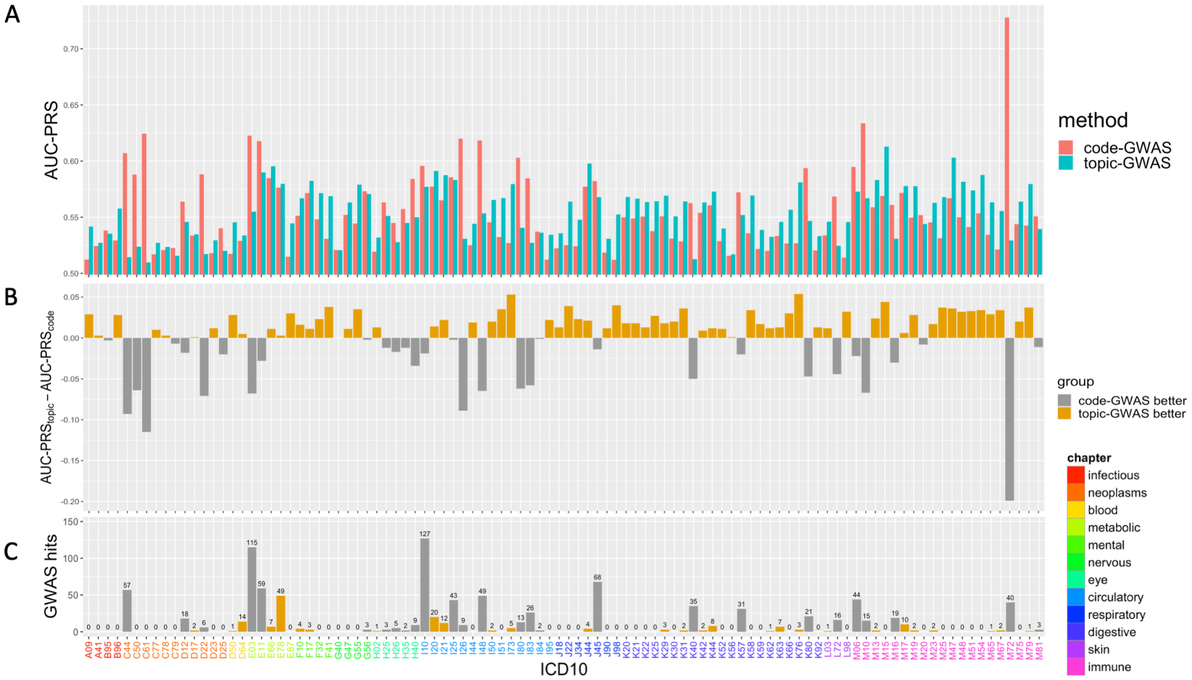


Figure 4.19: Comparison of PRS for ICD-10 codes constructed with topic-GWAS and single code GWAS results. A, AUC of PRS for the 100 ICD-10 codes constructed using single code and topic-GWAS results. B, Difference in the AUC of the two types of PRSs. Bars are colored according to which type of PRS have better performance. C, The numbers of significant loci found by single code GWAS for the 100 ICD-10 codes. Bars are colored the same way as in Figure 4.19B.

4.4.3 Comparison of treeLFA and LDA based on the performance of topic-PRS

Till now, the comparison of the three topic models (treeLFA, flatLFA and LDA) have been done from multiple aspects (inference result, predictive likelihood and topic-GWAS result). We have found that on the top-100 dataset, treeLFA and flatLFA had very similar inference and topic-GWAS results. For LDA, its inferred topics are slightly different from that for treeLFA, and in Section 4.2.4 we saw that LDA-inferred topics have fewer associated loci than treeLFA topics, which possibly indicates that treeLFA provides more power for genetic discovery.

To further verify the above observation that topic-GWAS based on treeLFA's topics provides more power, we built PRS for single codes based on the topic-GWAS results for treeLFA and LDA and compared the performance of these two PRS.

In Figure 4.20 the AUC of PRS constructed using treeLFA and LDA's topic-GWAS results are compared. For 99 ICD-10 codes, the AUC of PRS based on treeLFA's topics are larger. This result indicates that topic-GWAS using treeLFA's topics have a better generalization ability.

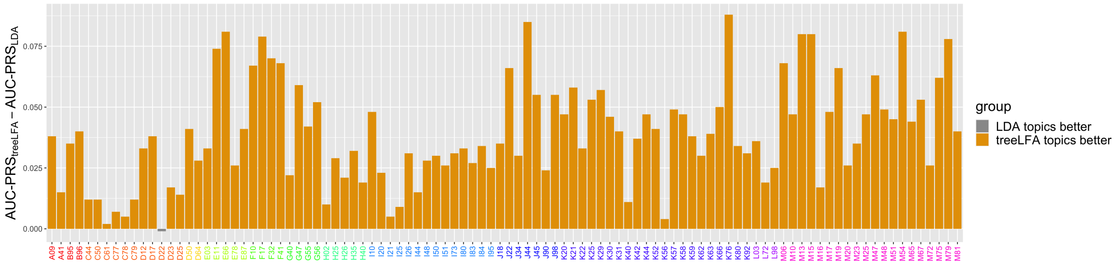


Figure 4.20: Comparison of AUC of PRS for ICD-10 codes based on treeLFA and LDA inferred topics. Bars are colored according to which type of PRS have better performance.

4.4.4 Comparison of different topic-GWAS methods based on the performance of topic-PRS

In Section 4.2.4, three topic-GWAS methods using different transformations on topics weights (no transformation, logit transformation and rank transformation) were compared. The results showed that with logit transformation on topic weights, the largest numbers of topics associated loci were found. In this section, we further verified this findings via comparing the topic-PRS for single codes based on the three topic-GWAS methods.

As is shown in Figure 4.21, for 87 ICD-10 codes the topic-PRS based on topic-GWAS using logit transformation had larger AUC than the topic-PRS based on topic-GWAS with rank transformations. Meanwhile, for 95 ICD-10 codes the topic-PRS based on topic-GWAS using logit transformation had larger AUC than that using no transformations (Figure 4.21B).

Overall, these PRS results further support the previous findings, that topic-GWAS using treeLFA topics and logit transformation on topic weights is the best methods among those under comparison.

4.5 Other GWAS methods for topics

4.5.1 Univariate multi-trait GWAS methods for active codes in inferred topics

The topic-GWAS used in previous sections is a natural choice for running GWAS on inferred topic weights. However, it is not always feasible to generalize this method to other cohorts. Because to carry out the topic-GWAS, the full diagnostic records for all people in the cohort are required to infer their weights for the topics.

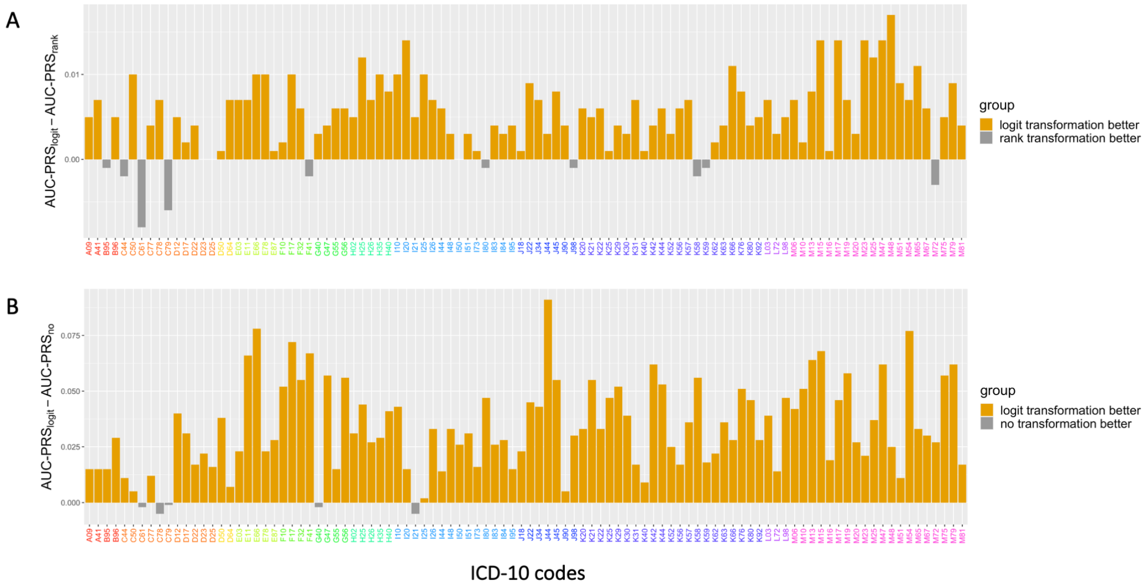


Figure 4.21: Comparison of the AUC of topic-PRS for single codes given by different topic-GWAS methods. Bars are colored according to which type of PRS have better performance. A, Comparison of topic-PRS based on topic-GWAS using logit and rank transformation on topic weights. B, Comparison of topic-PRS based on topic-GWAS using logit and no transformation on topic weights.

For many large cohorts, only limited phenotypic information is available, making it difficult to apply multivariate GWAS methods that require individual level data.

As was discussed in Chapter 1, a more flexible way to carry out multi-trait GWAS is to use univariate methods that only require the summary statistics of single trait GWAS as input. Researchers usually select a few traits of interest, and use univariate methods to jointly analyse the single trait GWAS results for these traits. Now with topics of diseases learnt from biobanks, it is reasonable to choose the top active disease codes in topics as the traits of interest for univariate multi-trait GWAS methods, since top active codes are identified by treeLFA as being more likely to co-occur on the same individuals.

We tried this idea using the flexible meta-analysis framework developed by zhu, et al. (2015) named "CPASSOC" (Cross Phenotype Association) [48], which uses the summary statistics given by single trait GWAS for multiple correlated traits as input. CPASSOC allows integrated analysis of different types of traits (binary, continuous). The key assumption of CPASSOC is that the t-statistics of the traits follow a multivariate normal distribution with zero mean under the null. As for the output, CPASSOC defines a new summary statistics for all traits, which measures the evidence that at least one trait is associated with the variant being analyzed. The effects of the variant on traits can be either homogeneous or heterogeneous, and two different statistics ($\boldsymbol{S_{homo}}$ and $\boldsymbol{S_{het}}$) are defined for the two scenarios to achieve the largest power. The details of the method is provided in Appendix A.

4.5.2 Example results with the top-100 UKB data

To experiment with CPASSOC, which is used as a representative of univariate multi-trait GWAS methods, we selected two topics inferred by treeLFA and used the top active codes in these topics as the traits for CPASSOC. These two topics are Topics 9 and 6 inferred by the treeLFA model with 11 topics for the top100 UKB dataset. In Topic 9, the top five active codes (E78,I10,I20,I21,I25) were chosen for CPASSOC. In Topic 6, three top active codes were chosen (M17, M23 and M25). The summary statistics given by the single code GWAS for these codes were used as the input for CPASSOC, and the result given by CPASSOC is a single P-value for all the traits for each variant. The total numbers of significant loci given by the topic-GWAS and CPASSOC were compared.

For Topic 9, 68 and 150 significant loci were found by CPASSOC ($\boldsymbol{S_{homo}}$ and $\boldsymbol{S_{het}}$), which are more than the number of significant loci found by the topic-GWAS (50 loci). For Topic 6, seven ($\boldsymbol{S_{homo}}$) and 17 loci ($\boldsymbol{S_{het}}$) were identified by CPASSOC. The overlaps of significant loci found by the topic-GWAS and CPASSOC ($\boldsymbol{S_{homo}}$ and $\boldsymbol{S_{het}}$) are plotted in Figure 4.22.

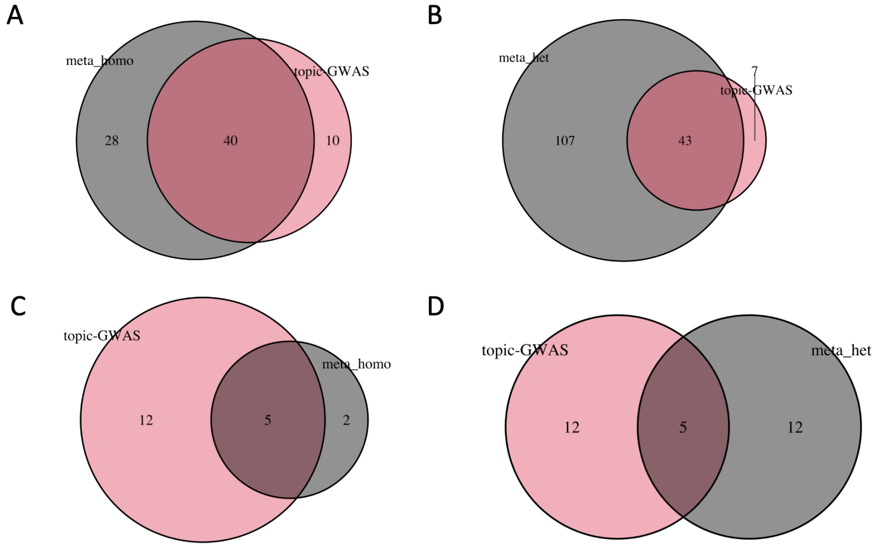


Figure 4.22: Comparison of the GWAS results given by topic-GWAS and a univariate multi-trait GWAS method. topic-GWAS was performed for topic 9 and 6, and CPASSOC (both $\boldsymbol{S_{homo}}$ and $\boldsymbol{S_{het}}$) was performed for the top active codes in the two topics respectively. A, Numbers of significant loci found by topic-GWAS and CPASSOC ($\boldsymbol{S_{homo}}$) and their overlap for topic 9. B, Numbers of significant loci found by topic-GWAS and CPASSOC ($\boldsymbol{S_{het}}$) and their overlap for topic 9. C, Results given by topic-GWAS and CPASSOC ($\boldsymbol{S_{homo}}$) for topic 6. D, Results given by topic-GWAS and CPASSOC ($\boldsymbol{S_{het}}$) for topic 6.

4.6 Discussion

In this chapter, we focused on carrying out genetic association study using the inference result given by treeLFA. First of all, we ran topic-GWAS on people's inferred weights, and compared the result with that given by the standard single code GWAS. Many single codes associated loci were found to have pleiotropic effects (associated with topics), and new loci were also identified by topic-GWAS. As before, the topic-GWAS results for different topic models and treeLFA models with different numbers of topics were compared respectively, and the results were in agreement with our findings in the previous chapters, that treeLFA outperformed LDA, and association signals are stable across different treeLFA models. Loci associated with topics were validated in different ways, and topic-GWAS results were used to construct PRS for both topics and codes, which for one thing validated the topic-GWAS results using the testing data, and for another thing provided insights for better risk prediction for single codes. Lastly, we explored methods other than the topic-GWAS to make use of the inference result given by treeLFA, and got promising preliminary results.

4.6.1 topic-GWAS results

topic-GWAS and single code GWAS

The standard way of running GWAS is to focus on one trait (disease code) at a time. In our study, GWAS was run on inferred topic weights as continuous traits. In general, single code GWAS found much more significant loci than topic-GWAS. The overlap of significant loci found by the two GWAS methods was quite large relative to the number of topics associated loci, indicating some known single codes associated loci have pleiotropic effects. It would be interesting to compare the effect sizes of the same loci on single codes and topics, but this is not straightforward to do, since topics and single codes are two different types of traits, and different GWAS methods were used for them.

On the other hand, topic-GWAS also found non-small numbers of significant loci that were not identified by single code GWAS. By checking the regional Manhattan plots, we found that for most loci that were only associated with topics, their P-values given by single code GWAS for some active codes in these topics were only slightly below the genome wide significance threshold. These findings are within our expectation, since topics associated loci in theory should influence the risks of at least some active codes in the corresponding topics. However, single code GWAS might not be powerful enough to detect these associations. By joining single codes into topics, we in fact have more cases for the GWAS, thus we can now find these associations with a larger power provided by the topic-GWAS.

As for the large number of loci that are only found as significant by the single code GWAS, the reason might be that most loci are still specifically associated with limited number of disease codes. Since these loci do not have effects on other active codes in the same topic, topic-GWAS for the whole topic would be unable to detect them. Besides, it is worth mentioning here that treeLFA inferred the topics based entirely on the phenotype data, instead of aiming to maximize the power to find phenotypic associations for a certain locus, as some of the methods discussed in Chapter 1 did (for example, the "PHCAT"). This is a reasonable choice, since the thesis put more emphasis on understanding the high-dimensional phenome data, instead of developing new methods for the in-depth study of certain important loci.

topic-GWAS for treeLFA and LDA

As was discussed in Section 4.2.4, more significant loci were found for treeLFA topics than LDA topics. Besides, most loci associated with LDA topics were also associated with treeLDA topics, indicating a larger statistical power for the topic-GWAS using treeLFA inferred topics. These two topic models are different in their configurations, therefore the differences in inference and topic-GWAS results are not surprising. However, the exact reasons for the differences are still not completely clear.

There are a few possible explanations to these differences, and we made some attempts in exploring them. Firstly, topic weights inferred by treeLFA may have larger variance, and as a result, more associations can be detected for them. We calculated the variance of topic weights for the ten disease topics inferred by both treeLDA and LDA, and found that the topic weights inferred by LDA have larger variance. This is reasonable since treeLFA usually assigns large weight to the empty topic for most people such that the remaining disease topics have to share less total weight, while LDA doesn't infer the empty topic. This result indicates the larger power for treeLFA is not resulted from introducing larger variance to the traits.

Secondly, the difference in topic-GWAS results may also be caused by the correlation structure among topic weights. We examined the correlation between topic weights for the two models (Figure 4.23). For treeLFA, the empty topic has weak negative correlations with all the other diseases topics, while the disease topics have no correlation with each other (Figure 4.23A). For LDA, all the inferred topics have weak negative correlations (Figure 4.23B). This is in fact a property of the Dirichlet distribution, that weights for different topics will be negatively correlated [158]. The approximate independence between treeLFA's topic weights is desirable in terms of defining new traits for GWAS, while the negative correlation between the empty topic and other disease topics is also reasonable from a biological point of view. Overall, different correlation structure among the inferred topics for the two models may be one of the reasons for their different topic-GWAS results.

4.6.2 PRS for topics

With the topic-GWAS results, we constructed PRS for topics, and evaluated them on the testing dataset. The performance of PRS for topics on the testing dataset was good, and was also consistent with the total numbers of associated loci we found for topics, suggesting different topics may have different levels of genetic relevance.

More interestingly, we constructed PRS for single codes using the topic-GWAS results. This is to some extent equivalent to decomposing the risk for a single disease code into different pathways in the form of topics of diseases. Surprisingly, we

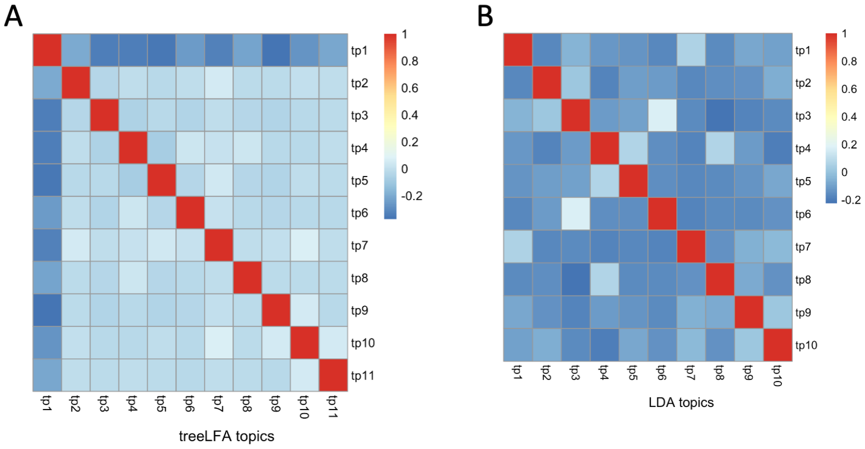


Figure 4.23: Correlation of topic weights for topics inferred by treeLFA and LDA. A, Correlation of topic weights for the 11 topics inferred by treeLFA. B, Correlation of topic weights for the ten topics inferred by LDA.

found that for a large number of ICD-10 codes, the performance of the PRS based on topic-GWAS result (topic-PRS) are better than the PRS based on single code GWAS result (code-PRS). Since treeLFA is a linear matrix factorization method, in theory the linear decomposition of disease risk shouldn't result in large improvement in prediction. Diving into these results, we obtained a few important insights.

Firstly, for most codes for which topic-PRS have better performance, their single code GWAS didn't find many significant loci (many of these single codes have no associated loci). This is very likely caused by the lack of enough statistical power for the single code GWAS. Topic-GWAS, which attains larger power for discovery by joining single disease codes into topics, achieves more accurate estimates for the effect size of associated variants, and therefore improves the risk prediction for these single codes. On the contrary, for those codes for which code-PRS gave better performance, their single code GWAS usually found a large number significant loci (Figure 4.19). This indicates that these single codes have very strong genetic components, and it is also reasonable to assume that many of their associated loci with large effect sizes are specifically associated with these codes, therefore joining these codes with other codes into topics can interfere with the risk prediction for these codes.

Another possible reason for topic-PRS giving better performance is that certain loci may have opposite or different effects on different subtypes of a disease. By decomposing the occurrence of a disease into separate pathways (topics), heterogeneous effects of some loci can be detected and used for prediction. However, considering the large number of loci used for constructing the best PRS for topics 4.2, this shouldn't be the main reason for the superior performance given by topic-PRS over code-PRS.

Secondly, we also noticed that the relative performance of the two types of PRS is related to the ICD-10 chapters where the single codes come from. For mental diseases, GI (gastro-intestinal) diseases and immune diseases, we found that most codes benefited from the topic-GWAS in terms of risk prediction, which suggests shared genetic components among these diseases. Overall, our analyses systematically demonstrate for which groups of diseases should we devote more efforts in understanding their common genetic risk factors.

Apart from the biological insights, PRS for single codes based on PRS for topics also provides us with a criterion to directly compare different topic models and topic-GWAS methods. This is because risk prediction for single codes is an objective metric to measure the performance of different models and methods in the same way. In agreement with previous results, we verified that for topic-GWAS, using treeLFA inferred topics gave better result than using LDA topics, and applying a logit transformation on topic weights is the best topic-GWAS method among those under comparison. Lastly, evaluation of PRS for topics and PRS for single codes based on topic-GWAS results demonstrated that the topic-GWAS results obtained on the training dataset can be generalized to the testing dataset, which in essence are also validations of the topic-GWAS results.

4.6.3 Other genetic analyses making use of inferred topics

With topics and topic weights inferred by treeLFA, the most straightforward way to carry out genetic association study is to directly perform topic-GWAS on topic weights as continuous traits. However, this analysis requires individual level data for all people, and the availability of the full multi-morbidity information to infer people's topic weights, which to some extent restricts its application to the deep phenotyped biobank data.

In the last section, we proposed another way to perform GWAS for topics, which is to apply univariate multi-trait GWAS methods on top active codes in a topic. In addition to widen the range of treeLFA's application, another reason for doing this is that topic-GWAS has its own limitations. For instance, we know almost for certain that most topics associated loci have effects on most but not all the active codes in a topic. For loci with heterogeneous effects on active codes in a topic, topic-GWAS would have inferior power. Some multi-traits GWAS methods can better deal with this situation (such as the $\boldsymbol{S_{het}}$ statistics for CPASSOC). Results in Section 4.6 show that other multi-trait GWAS methods may have additional power for discovery compared to topic-GWAS in certain scenarios. Undoubtedly, this result still needs to be further validated. Nevertheless, in general it suggests that topic-GWAS is not the only choice for studying the genetic associations of topics.

Regardless of the GWAS method to be used in future studies, analyses in this chapter provides people with a new perspective on studying the genetic components of multiple related diseases, and shows the value of integrating high dimensional phenotype data in biobanks to boost the power of genetic association studies.

5

Analyses with a larger UK Biobank dataset

Contents

5.1 Overview of the chapter

In the previous chapters, we applied treeLFA to the top-100 UKB dataset constructed from the HES data in UK Biobank (UKB), and carried out downstream analyses using the inference results. The top-100 dataset, however, was deliberately limited in size, since we wished to train many different models for comparison. Beyond the top 100 most frequent ICD-10 codes, in UKB there are still many diseases with a prevalence suitable for inclusion into the topic modelling framework. In this chapter, we constructed a larger input dataset for treeLFA, which contains the records for 436 ICD-10 codes for all people in the UKB. Key analyses that had been performed on the top-100 dataset were repeated on this new dataset, and additional functional analyses were carried out for the topic-associated loci.

5.2 The input dataset

The top-436 dataset was constructed in the same way as the top-100 dataset used in the last two chapters, and contains the records of the top 436 most frequent ICD-10 codes from the first 14 chapters of the ICD-10 coding system for all people in UKB. These codes are all the ones in the UKB with a prevalence of at least 0.001 at the date of selection (note that continued data collection means that prevalence will tend to increase over time), corresponding to approximately 500 cases in UKB. The prevalence threshold of 0.001 was chosen both for computational reasons (treeLFA chains were run for about 10-12 days using 8 slots/"CPU cores" in parallel on the cluster of BMRC (Biomedical Research Computing) platform) and because there must be sufficient occurrence of disease from which to discover patterns of multi-morbidity. As with the top-100 dataset, we also partitioned the full top-436 dataset into training (80 %) and testing (20 %) datasets. The top-436 dataset and the top-100 dataset used different partitions for the training and testing datasets.

5.3 Inference results for the top-436 UKB dataset

5.3.1 Training strategy for the top-436 dataset

The top-436 dataset is more than three times larger than the top-100 dataset, increasing the computational requirements for training. For the top-100 dataset, treeLFA models with different numbers of topics were trained and compared. We found that when we set an excess number of topics for the model, inferred topics are stable across models with different numbers of topics (after clustering and collapsing, particularly for the empty topics). Therefore, for the top-436 dataset, instead of training models with different numbers of topics, we only trained the treeLFA and flatLFA model with 100 topics.

5.3.2 Inference result for the top-436 dataset

Training of treeLFA on the top-436 dataset

Three Gibbs chains were trained for treeLFA and flatLFA, and were initialized independently. Beta priors used for $\boldsymbol{\phi}$ were Beta(0.1,3000) for inactive codes and Beta(1.2,3) for active codes in topics, to account for rare diseases in the larger dataset. Initialization of the remaining hidden variables, and settings of hyperparameters were same as that on the top-100 dataset. The training strategy with the Gibbs-EM algorithm and Gibbs sampler was also the same as the top-100 dataset.

12 CPU cores on the BMRC (Biomedical research computing) computing cluster were used for the training of each chain. About three weeks were spent on the training with the Gibbs-EM algorithm (1,500 Gibbs-EM iterations with one posterior sample of $\boldsymbol{Z}$ collected in the E-step for the optimization of $\boldsymbol{\alpha}$, 300 Gibbs-EM iterations with ten posterior samples of $\boldsymbol{Z}$ collected in the E-step. The 1,800 Gibbs-EM iterations

require 90,000 Gibbs sampling iterations in total), and about one day was spent on the training with the Gibbs sampling (5,000 iterations).

Figure 5.1 shows the traceplots for the three Gibbs chains for treeLFA. Ranges of log-likelihood for the three chains were different, indicating that they were at different local optimum of the log-likelihood function $\boldsymbol{lnP(W|\alpha)}$.

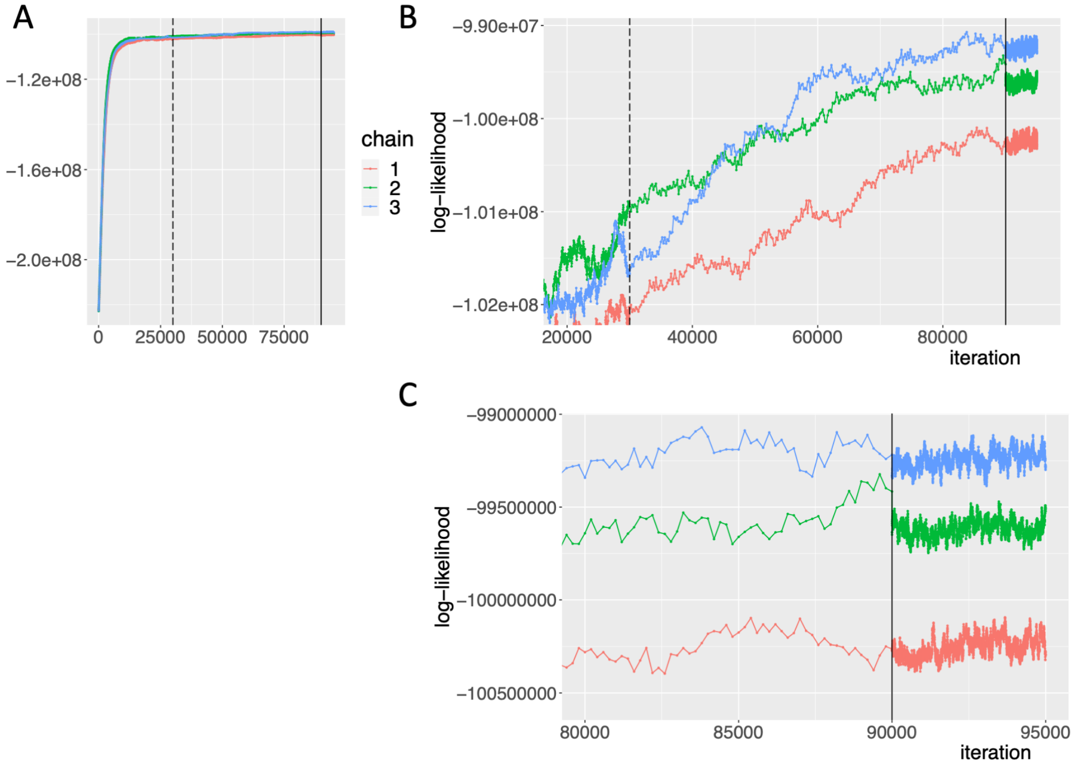


Figure 5.1: Traceplots for treeLFA chains on the top-436 dataset. Log-likelihood was plotted against the iterations of Gibbs sampling. A, The full traceplots for the three treeLFA chains during the Gibbs-EM and Gibbs sampling training stage. Iterations to the left of the dashed line corresponded to the first Gibbs-EM training stage, during which one posterior sample of $\boldsymbol{Z}$ was taken in the E-step every 20 iterations for the optimization of $\boldsymbol{\alpha}$ in the M-step. Log-likelihood was calculated after each optimization of $\boldsymbol{\alpha}$ (every 20 iterations). Iterations between the dashed line and the solid line corresponded to the second Gibbs-EM training stage, during which ten posterior sample of $\boldsymbol{Z}$ was taken in the E-step. Log-likelihood was calculated every 200 iterations. Iterations to the right of the solid line corresponded to the Gibbs sampling stage, during which the $\boldsymbol{\alpha}$ was fixed, and the posterior distribution of other hidden variables were simulated. Log-likelihood was calculated every iterations. B, Iterations 20,000 to 95,000 were zoomed in for better visualization. C, B, Iterations 80,000 to 95,000 were zoomed.

The variability of inference result

The traceplots in the previous section showed that different chains gave different optimal value for $\boldsymbol{\alpha}$. Considering this, we clustered and collapsed the posterior samples of topics for each chain separately, and retained 40 topics for each chain (most of the remaining topics are empty or near empty topics).

The averaged inferred topics given by the three Gibbs chains for treeLFA were checked, and they were slightly different (not plotted). The 120 averaged topics (40 topics given by each one of the three chains) given by the three chains were mixed and clustered into 54 groups using the standard hierarchical clustering algorithm. The cosine distance was used to measure the dissimilarity of two topics ($\boldsymbol{x}$ and $\boldsymbol{y}$), which was defined as below:

$$\begin{aligned}
&\boldsymbol{cosine\,distance = 1 - cosine\,similarity},\\
&\boldsymbol{cosine\,similarity} = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2} \cdot \sqrt{\sum_{i=1}^{n} y_i^2}},\\
&\boldsymbol{x = (x_1, x_2, ..., x_n)},\\
&\boldsymbol{y = (y_1, y_2, ..., y_n)}.
\end{aligned}$$

where X and Y are two vectors of length $\boldsymbol{n}$. The criterion for choosing the number of groups for topics was that the largest cosine distance between topics in the same group should be less than 0.1. Among them, 29 groups contained topics inferred by all the three chains, 8 groups contained topics inferred by two chains, and 17 groups contained topic inferred by only one chain. The three chains for flatLFA also gave slightly different inference results. On the other hand, inferred topics were stable across different posterior samples from the same chain. With Louvain clustering applied to all posterior samples of topics, for each chain exactly 40 clusters were obtained, and each cluster was composed of topics from all posterior samples. These results indicated that different Gibbs chains converged to different local optimum of the posterior distribution of the hidden variables.

Predictive likelihood on the testing dataset

To compare the generalization ability of topics inferred by different chains to a new dataset, the predictive likelihood on the test data were estimated for the three Gibbs chains of treeLFA. Since 40 topics were retained, which was a large number for the calculation of the predictive likelihood (a Monte-carlo approximation was used to approximate an integral to calculate the predictive likelihood), the number of samples required to provide enough precision was unknown. We performed a sensitivity analysis for the number of Monte-carlo samples of topic weights drawn from their prior distribution (Figure 5.2A). With more samples for topic weights, the predictive likelihood of the testing dataset kept increasing. This was because more samples of topic weights meant a larger probability to obtain samples that fit the test data better, and thus give a larger likelihood. However, the order of the magnitude of predictive likelihood for the three chains remained the same, suggesting their generalization abilities were differentiated.

In Figure 5.2B, the predictive likelihood for the three chains for treeLFA and flatLFA were compared. For each chain, the predictive likelihood was calculated for ten times using the same averaged topics to evaluate the uncertainty in estimating them with the implemented algorithm for predictive likelihood. Overall, the three treeLFA chains had larger predictive likelihood than the flatLFA chains, and had a smaller range, suggesting the informative prior used by treeLFA were helpful in extracting meaningful patterns from the data. In addition, compared to the variation caused by the differences in the inference results for different chains, the uncertainty caused by the algorithm for predictive likelihood was much smaller. For the following analyses, only the inference result given by the chain with the largest predictive likelihood for each model was used as input. Meanwhile, for each topic inferred by the chain with the largest predictive likelihood, we also annotated the number of chains that inferred it from the data, so that the reliability of each topic can be assessed.

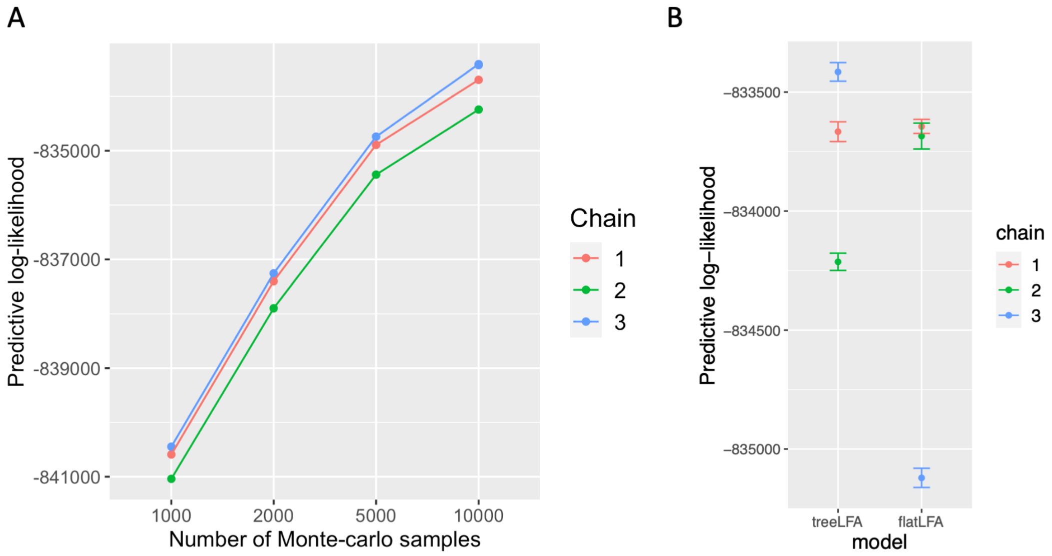


Figure 5.2: Predictive likelihood on the testing data. A, Sensitivity analysis for the number of Monte-carlo samples used to calculate the predictive likelihood. The predictive likelihood for the testing data were calculated using topics inferred by the three treeLFA chains, with different numbers of Monte-carlo samples of topic weights drew from its prior distribution (Dirichlet($\boldsymbol{\alpha}$)). B, Comparison of the predictive likelihood for treeLFA and flatLFA chains. For each treeLFA and flatLFA chain, the predictive likelihood was calculated for ten times using the same topics averaged from 50 posterior samples. The error bars showed the uncertainty in calculating the predictive likelihood using the algorithm.

Comparison of topics inferred by treeLFA and flatLFA

In Figure 5.3, topics inferred by treeLFA and flatLFA are shown. The two models inferred 48 distinct topics in total, among which 32 topics were inferred by both models, and 8 topics were uniquely inferred by one model. For treeLFA, 29 topics were inferred by all three chains, and 5 topics were inferred by two chains. For flatLFA, 32 topics were inferred by all three chains, and 3 topics were inferred by two chains. These suggest that for both models, most topics were inferred by multiple chains from the data. As the inference result for the top-100 dataset, among the inferred topics there are both dense topics with a large number of active codes and sparse topics. In both types of topics, top active codes (codes with the largest probabilities) formed small clusters in the heatmap, suggesting they are near to each other within the ICD-10 ontology (since codes are arranged according to their chapters and coding numbers in the heatmap). In sparse topics, top active codes come from a specific, or a few, ICD-10 chapters. While in dense topics, active codes come from multiple chapters.

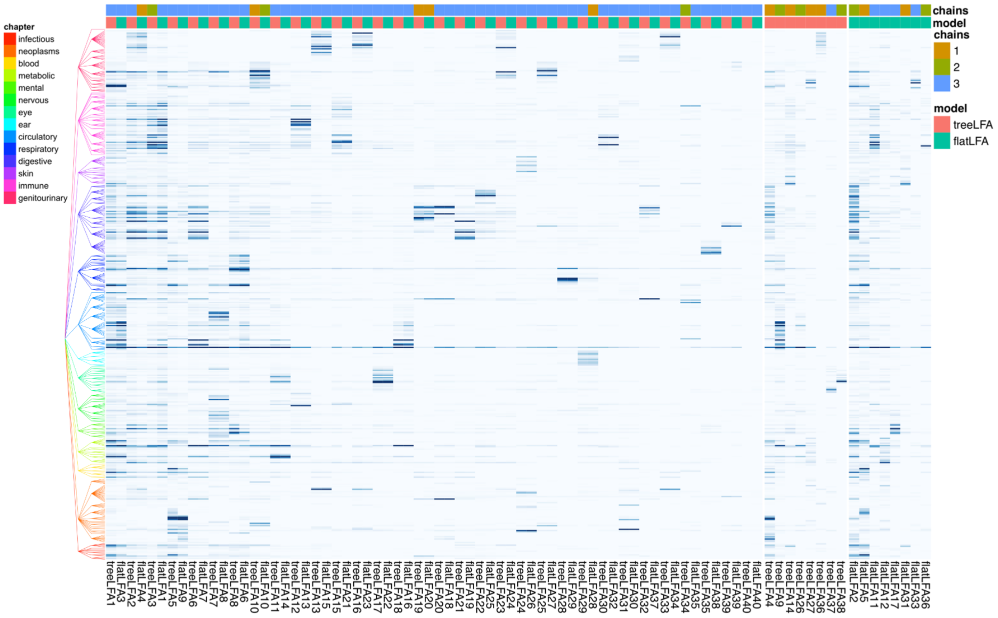


Figure 5.3: Topics inferred by treeLFA or flatLFA on the top-436 UKB dataset. In the heatmap each row is an ICD-10 code. ICD-10 codes are ordered according to the chapters they come from and their coding numbers. Codes from Chapter 1 are placed at the bottom. The tree structure for the 436 codes are plotted to the left of the inferred topics. 32 topics were inferred by both models, while 8 topics were uniquely inferred by only one model. From the left side of the heatmap, the 32 common topics are plotted first, then the topics inferred by only one model. Topics inferred by each model are numbered and ordered according to their density (the sum of probabilities of all constituent codes). Topic inferred by treeLFA having the largest density is placed at the leftmost position. The top colour bar shows the number of chains that inferred each topic.

Interpretation of inferred topics

To provides a better understanding of the inferred topics, we first counted the number of codes in topics with a relatively large probability. Considering different codes have different prevalence in UKB, the probabilities of each code in all topics were firstly normalized so that the largest probability in all topics equalled 1. After that, the number of codes with a probability of at least 0.3 in different topics were shown in Figure 5.4A.

We then examined which chapters made major contributions to each topic. In Figure 5.4B, the numbers of active codes (codes with a normalized probability defined in the previous paragraph of at least 0.3) in topics coming from different ICD-10 chapters are shown. For most sparse topics, their top active codes come from a limited numbers of chapters (typically 1-3). This result is in consistent with the result for the top-100 dataset (Section 3.3.2). Topics were then named according to the major contributing chapters. In Figure 5.4B, ICD-10 chapters were firstly named using single words summarizing the categories of diseases in them (labels on the y-axis). After that, topics were were named using the ICD-10 chapters enriched among their active codes (Fisher exact test, P-values corrected for the number of chapters). Four chapters were enriched for Topic 1 and 4. Most remaining topics focused on 1-2 chapters (only 4 topics were named using three chapters). The top active codes in all topics are shown in Appendix B.

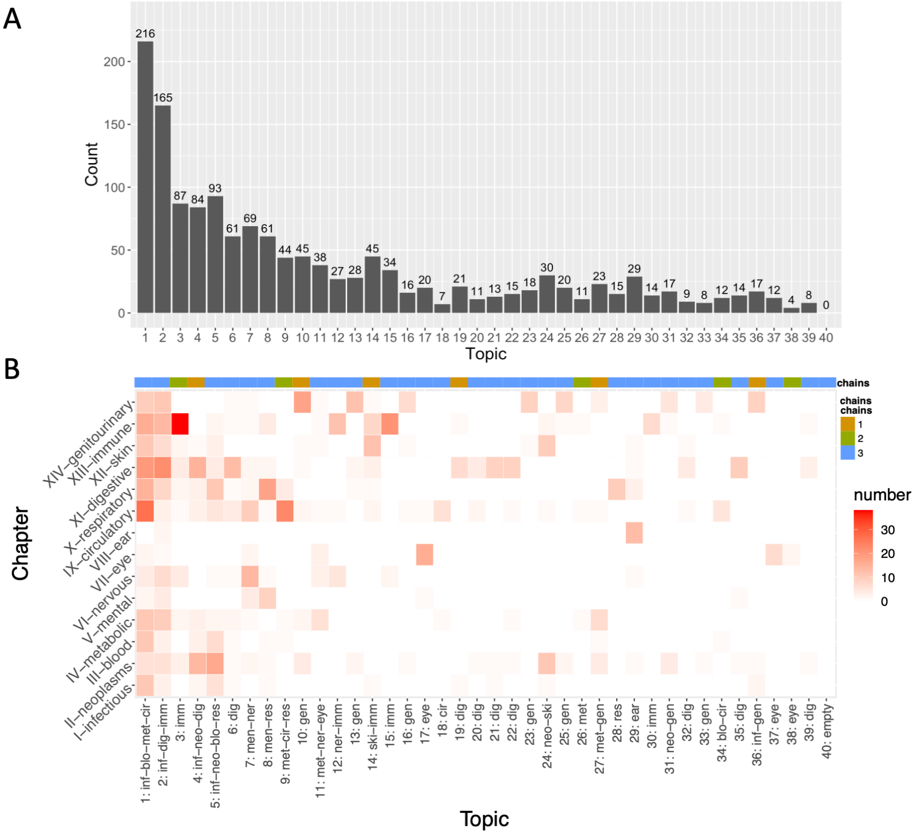


Figure 5.4: Top active codes in topics. A, The numbers of codes with normalized probability of at least 0.3 in topics inferred by treeLFA. B, Numbers of active codes in topics coming from different ICD-10 chapters. Each column in the heatmap is a topic, and each row is an ICD-10 chapter. Topics were ordered according to their density, and were named using the ICD-10 chapters (the first three letters of the names of the chapters on the y-axis) enriched among the active codes in them. For each topic, the number of chains that found them was also shown with the colour bar on top of the heatmap.

On the top-100 dataset, about 30 distinct topics were inferred. To make comparison of the inference results on the two UKB datasets, topics inferred by treeLFA from the two datasets were plotted together in Figure 5.5. The order of the 436 ICD-10 codes were rearranged, such that the top-100 most frequent codes (appeared in both datasets) were put together and shown first (from the bottom of the figure), followed by the remaining codes only appeared in the top-436 dataset. Notably, for topics inferred from the top-100 UKB dataset, only the first 100 codes can have non-zero probabilities.

If we only look at the first 100 most frequent ICD-10 codes in topics (about the bottom one fourth of the heatmap), we see that 22 topics inferred by treeLFA from the top-100 dataset have corresponding topics inferred from the top-436 dataset (one topic inferred from the top-100 dataset has two corresponding topics from the top-436 dataset). There are ten topics uniquely inferred from the top-100 dataset, and about half of them are very sparse topics. There are 17 topics uniquely inferred from the top-436 dataset, and about half of them contain a substantial numbers of active codes from Chapter 14 (Diseases of the genito-urinary system), which was not used to construct the top-100 dataset. Most remaining topics uniquely inferred from the top-436 dataset also contain active codes that are not included in the top-100 dataset (from the first 13 ICD-10 chapters). Overall, this result indicates that the inference results on the two UKB datasets have a high level of consistency, and the same multi-morbidity patterns were captured from the two different datasets.

5.3.3 topic-GWAS result for the top-436 UKB dataset

As with the top100 dataset, topic-GWAS was carried out in the same way for all topics inferred from the top-436 dataset using inferred topic weights as traits. Since the top-436 dataset is much larger than the top100 dataset, to increase the inference accuracy for topic weights, we used Gibbs sampling to re-estimate individuals' topic weights, with topics ($\boldsymbol{\phi}$) and $\boldsymbol{\alpha}$ fixed at values averaged from all posterior samples from the chain with the largest predictive likelihood for each model. With topics and $\boldsymbol{\alpha}$ fixed, the Gibbs sampling does not have the identifiability issue, so the results given by different chains can be combined directly. Ten Gibbs chains were trained to re-estimate topic weights, and averaged topic weights from these chains were used as the input for the topic-GWAS.

Comparison of the topic-GWAS results for treeLFA and flatLFA

The numbers of loci associated with topics inferred by treeLFA and flatLFA were plotted in Figure 5.6A. For almost all topics, the numbers of loci associated with treeLFA and flatLFA inferred topics are close to each other (except for Topic 12 inferred by flatLFA). Overall, we found 278 significant loci associated with the 40 topics inferred by treeLFA, compared to 260 loci for flatLFA inferred topics. These numbers are larger than those obtained on the top-100 dataset (180 loci for 32 topics inferred by treeLFA). The overlap of these two sets of significant loci is 207 (Figure 5.6B). To verify if the significant loci identified by only model were marginally below the significant threshold for another model, we also plotted the

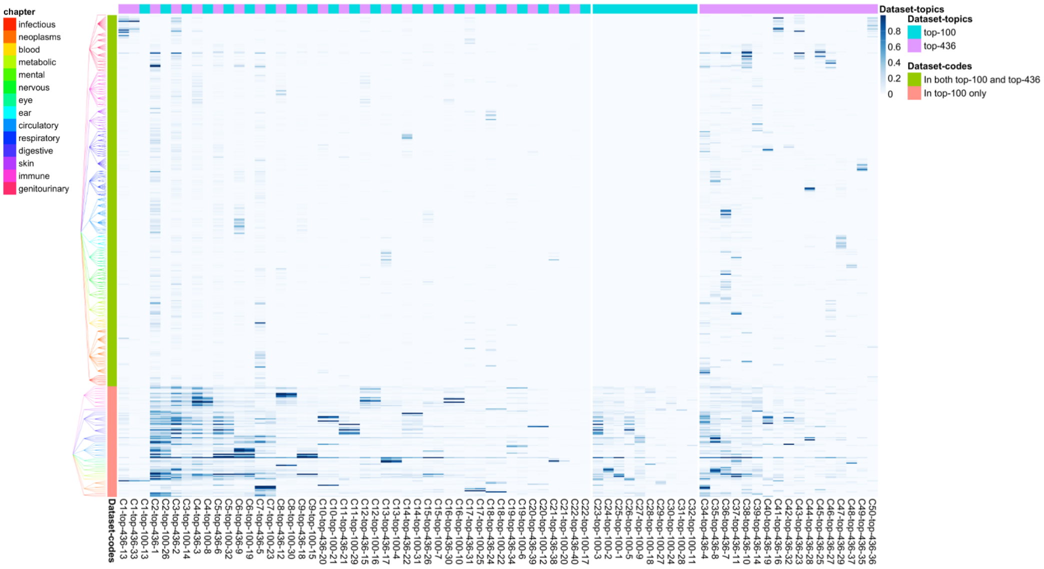


Figure 5.5: Comparison of topics inferred by treeLFA from the two UKB datasets. 32 topics inferred from the top-100 dataset and 40 topics inferred from the top-436 dataset were shown together. In the heatmap, each column is a topic, and each row is an ICD-10 codes. The order of the ICD-10 codes is re-arranged such that codes in the top-100 dataset are shown first in their original order from the bottom of the heatmap, followed by codes only included in the top-436 dataset in their original order. Topics inferred from different datasets but have similar pattern (probabilities) for the first 100 codes are sited adjacent to each other and shown first from the left.

overlap of sub-significant loci ($\boldsymbol{P < 10^{-5}}$) found by one model and the significant loci ($\boldsymbol{P < 5 \times 10^{-8}}$) found by another model (Figure 5.6C-D). As can be seen, more than 80 % of significant loci found by only one model were found as sub-significant by another model, indicating overall the two models' topic-GWAS results have a high level of agreement.

Comparison of the topic-GWAS result and single code GWAS result

The topic-GWAS result for the 40 treeLFA inferred topics and the single code GWAS result for the 436 ICD-10 codes in the top-436 dataset were compared in Figure 5.7. There are 1,093 loci associated with 436 codes, and 278 loci loci associated with the 40 topics. Although most topic-associated loci are also associated with ICD-10 codes (consistent with the topic-GWAS result for the top-100 dataset), there are also 80 significant loci that are only found by topic-GWAS. In Figure 5.7B, the total numbers of loci associated with topics are shown. Most of these loci are associated with both topics and single codes. The numbers of loci only associated with topics but not any single code are highlighted in red. As with the top-100 dataset, the empty topic (Topic 40) has a large number (21) of new significant loci (not found by the single code GWAS). Topics 3 and 30 have large proportions

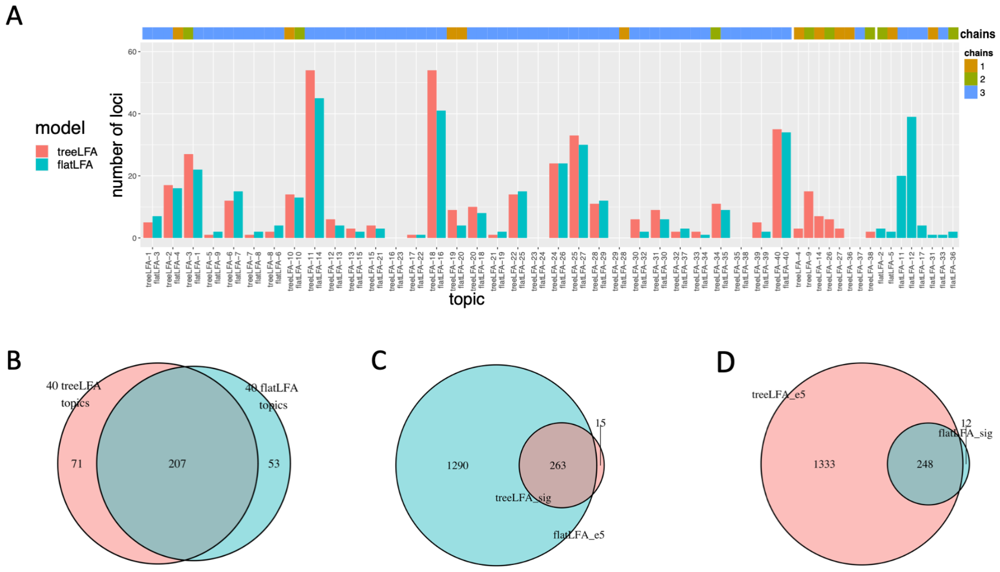


Figure 5.6: Topic-GWAS results for treeLFA and flatLFA. A, Numbers of loci associated topics inferred by treeLFA and flatLFA are shown. The topics are in the same order as those in Figure 5.3. For each topic, the number of chains that found them was also shown with the colour bar on the top. B, The total numbers of loci associated with treeLFA and flatLFA topics and their overlap. C, The overlap of sub-significant ($\boldsymbol{P < 10^{-5}}$) loci for treeLFA topics and significant ($\boldsymbol{P < 5 \times 10^{-8}}$) loci for flatLFA topics. D, The overlap of significant loci for treeLFA topics and sub-significant loci for flatLFA topics.

of loci (81.5 % and 83.3 %) only found by topic-GWAS, and most of their active codes are from Chapter 13 (Diseases of the musculo-skeletal system and connective tissue). Topic 11 and 18 have 8 and 7 loci only found by topic-GWAS. Topic 11 contains many active codes from Chapter 4 (Endocrine, nutritional and metabolic diseases) and Chapter 7 (Diseases of the eye and adnexa), while Topic 12 is mainly about hyperlipidemia and heart diseases (Chapter 9: Diseases of the circulatory system). Topic 25 has 6 new loci, and most of its active codes are from Chapter 14 (Diseases of the genitourinary system). Topic 2 is a dense topic, and also has a substantial numbers of new loci (11), but it is harder to interpret than sparse topics. The remaining topics only have very few new significant loci.

Figure 5.7C-D focus on the comparison of effect sizes for different groups of significant loci. Three groups of loci were evaluated, including loci only associated with single code, loci associated with both topic and single code, and loci only associated with topic. Figure 5.7C compares the distributions of log-OR (odds ratio) given by single code GWAS for lead SNPs associated with both topic and code and lead SNPs only associated with single code. If a variant is associated with multiple codes, then the largest effect size (absolute value) will be used. Compared with SNPs only associated with single code, a larger proportion of SNPs that are associated with both topic and single code have large positive effects. For example, 29.2 % SNPs have an log-OR of at least 0.2 among SNPs associated with both single code and topic and having positive effects, while the proportion for SNPs only associated with single code is 19.4 %. Figure 5.7D compares the distributions of effect sizes given by topic-GWAS for lead SNPs associated with both topic and code and lead SNPs only associated with topic. A larger proportion of lead SNPs only associated with topic have negative effects compared to lead SNPs associated with both topic and single code (52.5 % and 41.4 %). For lead SNPs with positive effects, in general the magnitude of effect sizes for SNPs only associated with topic are slightly smaller than SNPs associated with both topic and code. For instance, 38.3 % SNPs have an BETA of at least 0.03 among SNPs associated with both topic and single code and having positive effects, while the proportion for SNPs only associated with topic is 21.1 %.

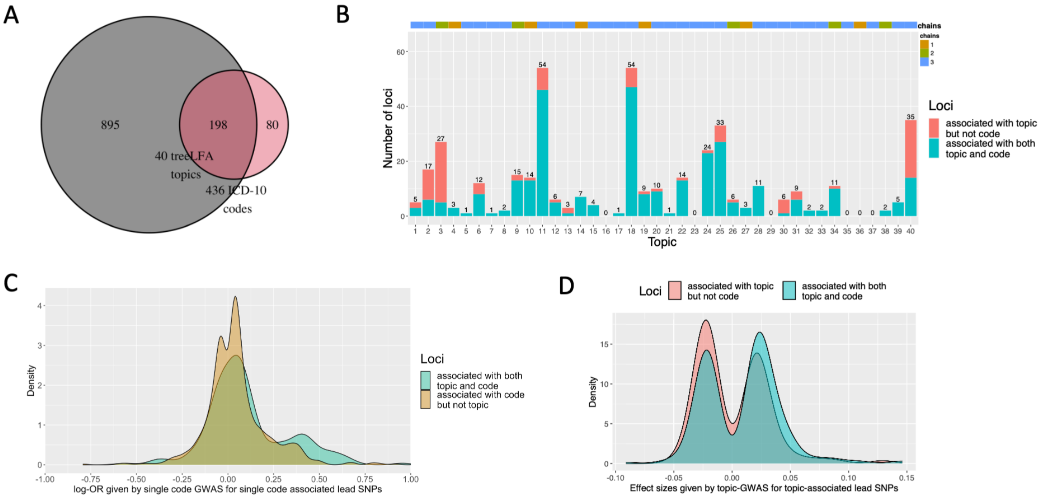


Figure 5.7: Comparison of topic-GWAS result and single code GWAS result. A, The total numbers of loci associated with the 40 topics (treeLFA), the 436 ICD-10 codes and their overlap. B, The numbers of significant loci only found by topic-GWAS and the numbers of significant loci found by both topic-GWAS and single code GWAS for all treeLFA inferred topics. For each topic, the number of chains that found them was shown with the colour bar on the top. C, Density plot for the effect sizes given by single code GWAS for lead SNPs associated with both topic and code and lead SNPs only associated with code. D, Density plot for the effect sizes given by topic-GWAS for SNPs associated with both topic and code and SNPs only associated with topic.

5.3.4 Validation of topic-GWAS results

As with the top-100 dataset, loci associated with the topics for the top-436 dataset were validated using the same approaches as previously.

Overlap with the GWAS catalog

First, the GWAS catalog was used to check for records of the topic-associated loci. All SNPs in the GWAS catalog were clumped to lead SNPs associated with treeLFA inferred topics ($\boldsymbol{r^2 > 0.5}$). Among the 278 topic-associated loci, 248 have a record in the GWAS catalog (89.2 %). Among the 80 significant loci uniquely found by topic-GWAS, 63 have a record in the GWAS catalog (78.8 %). These results are consistent with the validation result for the top-100 datase (for the treeLFA model with 100 topics, 75 % topic-associated loci had a record in the GWAS catalog, and 62.5 % loci uniquely associated with topics had a record in the GWAS catalog).

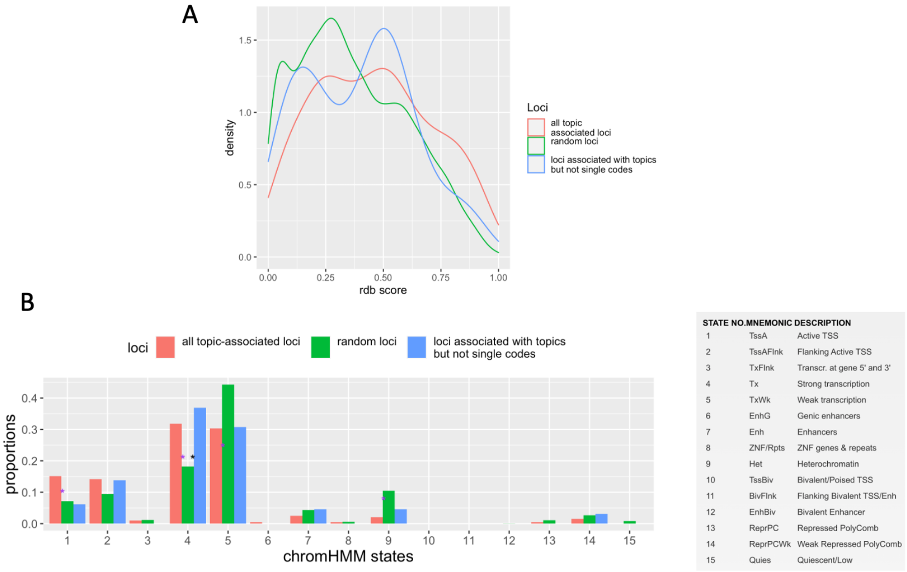
Functional genomic enrichment of topic-associated loci

Validation with the functional genomics resources was also performed. As with the top-100 dataset, the distributions of rdb scores and chromHMM states among three groups of SNPs (all topic-associated lead SNPs; lead SNPs associated with topic but not any single code; random SNPs) were compared. Lead SNPs associated with different topics were combined as the input for the functional analyses.

Figure 5.8A shows that overall the two groups of topic-associated lead SNPs have larger rdb probability than the random SNPs. For example, 40.3 % random SNPs have a rdb probability of at least 0.4, while the proportions for SNPs only associated with topic and all topic-associated lead SNPs are 52.0 % and 55.3 %. For chromHMM states, Figure 5.8B shows that overall, the two groups of topic-associated lead SNPs have similar profiles, which are different from that of the random SNPs. The differences are most apparent for states 2 and 4 (regions flanking active TSS (transcription start sites), and regions with strong transcription, where the two groups of topic-associated lead SNPs have larger proportions), as well as states 5 and 9 (regions with weak transcription, and heterochromatin, where the random SNPs have larger proportions). In Figure 5.9 the proportions of lead SNPs that are eQTLs and have chromatin interactions (CI) for the three groups of SNPs in various tissues are compared. In all tissues in which the proportions of eQTL or CI are not zero, the two groups of topic-associated lead SNPs have larger proportions than the random SNPs. In most tissues, the differences between the two groups of topic-associated SNPs and the random SNPs are significant. Overall, the functional validation results on the larger UKB dataset are more convincing than the results on the top-100 dataset. This is likely to be caused by the increase in the number of topic-associated lead SNPs, which provides more power for detecting differences between groups.

We also performed different enrichment analyses for genes mapped from topic-associated lead SNPs. FUMA's positional mapping function was used to map topic-associated lead SNPs to their nearby genes. With the mapped genes, gene set enrichment analyses can be performed using various reference gene sets. Specifically, FUMA allows us to use genes associated with different traits in the GWAS catalogue as the reference gene sets.

Figure 5.10 shows the gene set enrichment results for two topics of different types as example, which are Topic 40 (the empty topic) and Topic 24 (a sparse



STATE NO.	MNEMONIC	DESCRIPTION
1	TssA	Active TSS
2	TssAFlnk	Flanking Active TSS
3	TxFlnk	Transcr. at gene 5' and 3'
4	Tx	Strong transcription
5	TxWk	Weak transcription
6	EnhG	Genic enhancers
7	Enh	Enhancers
8	ZNF/Rpts	ZNF genes & repeats
9	Het	Heterochromatin
10	TssBiv	Bivalent/Poised TSS
11	BivFlnk	Flanking Bivalent TSS/Enh
12	EnhBiv	Bivalent Enhancer
13	ReprPC	Repressed PolyComb
14	ReprPCWk	Weak Repressed PolyComb
15	Quies	Quiescent/Low

Figure 5.8: Functional validation for topic-associated loci for the top-436 UKB dataset. A, Validation of topic-associated loci using the Regulome DB (rdb) scores. The distribution of the probability of being functional variants (rdb probability) for the two groups of topic-associated lead SNPs (group A: all topic-associated lead SNPs; group B: lead SNPs associated with topic but not any single code) are compared to that of 10,000 randomly selected SNPs. B, Validation of topic-associated lead SNPs using the chromHMM annotations. The proportions of the two groups of topic-associated lead SNPs that are in different chromHMM states are compared to the proportion for the random SNPs. Significant differences ($\boldsymbol{P < 0.05/number\ of\ chromHMM\ states}$) are highlighted with $*$ (purple and black $*$ correspond to the comparisons between group A/B and random SNPs, respectively).

disease topic, with a few diseases of the skin tissue as top active codes). Enriched gene sets (correspond to traits in the GWAS catalogue) for the two topics are very different. For Topic-24, most enriched traits are skin diseases. As for the empty topic, most enriched traits are life-style factors.

5.4 Discussion

In this chapter, treeLFA was applied to a second UKB dataset, the top-436 dataset, which is three times larger than the top-100 dataset. Topics of ICD-10 codes were inferred, and their relations to topics inferred from the top-100 dataset were investigated. topic-GWAS was performed, and further validation and functional analyses were also done to provide biological insights.

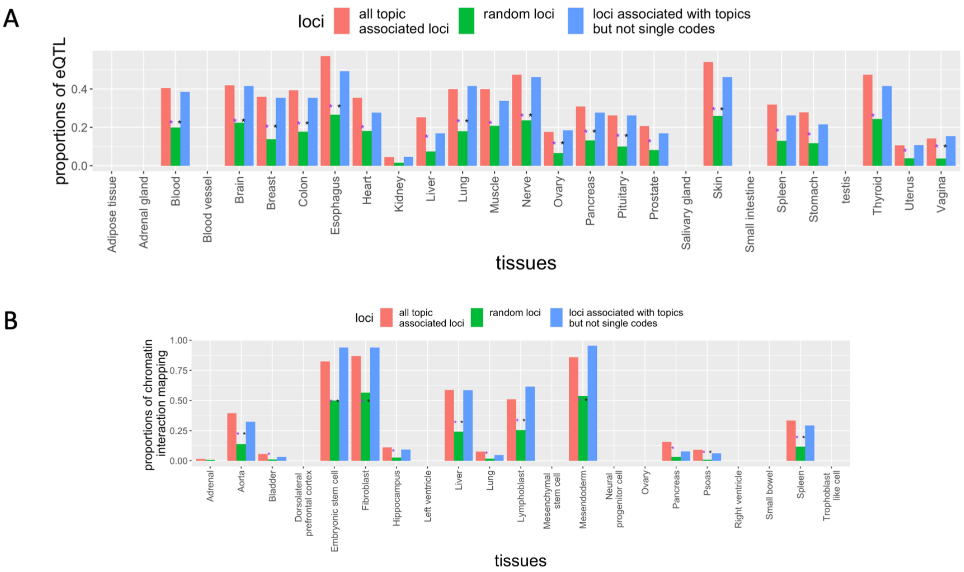


Figure 5.9: Validation of topic-associated loci using eQTL and chromatin interaction (CI) data. A, The proportions of the two groups of topic-associated lead SNPs (group A: all topic-associated lead SNPs; group B: lead SNPs associated with topic but not any of the ICD-10 code) that are eQTL in different tissues (GTEv8 dataset) were compared to the proportion of 10,000 random SNPs. Significant differences ($\boldsymbol{P < 0.05/number\ of\ tissues}$) are highlighted with $*$ (purple/black $*$ correspond to the comparisons between group A/B and random SNPs, respectively). B, The proportions of SNPs that have CI in different tissues for the three groups of SNPs.

5.4.1 Inferred topics for the top-436 dataset

In this chapter, we trained multiple chains for treeLFA and flatLFA models with 100 topics. Unlike the results for the top-100 dataset, different chains for the same model gave slightly different results, which was reasonable considering the size and sparsity of the input data. After clustering of posterior samples of topics, about 40 distinct topics remained. This number of distinct topics is consistent with the results in previous studies aiming to find multi-morbidity patterns of common diseases using topic models [88], in which 50 topics for 508 Phecodes were learnt. The topics inferred by treeLFA and flatLA were compared using the predictive likelihood on the testing dataset, and overall the three chains for treeLFA had larger predictive likelihood than flatLFA chains. This suggested that on a larger and more sparse dataset the prior for topics used by treeLFA provides advantage in terms of inference, which also rationalized the use of medical ontology as prior for topics. However, it is important to note that treeLFA/flatLFA chains didn't all converge to the stationary distribution, indicating that some chains were stuck in the local optimum and failed to explore the full posterior distribution. Besides, the calculation of predictive likelihood is also not very accurate, as is suggested by the variance of likelihood caused by the algorithm in Figure 5.2. Therefore, results in this section only suggest that treeLFA inferred topics are likely to have better

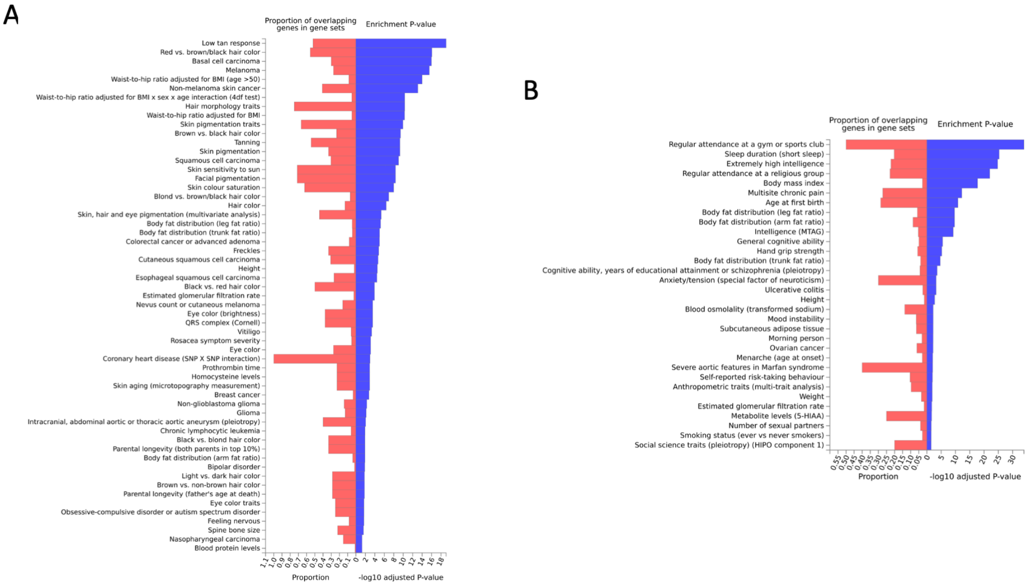


Figure 5.10: Gene set enrichment analysis for loci associated with two topics of different types. Genes associated with different traits in the GWAS catalogue were used as the reference gene sets. A, Result for the empty topic. B, Result for Topic 24, which is a sparse topic centered on skin diseases.

generalization ability than flatLFA topics, but this conclusion cannot be reliably drawn, and requires additional analyses to further verify.

As with the top-100 dataset, among the inferred topics there was one empty topic, several very dense topics and a lot of sparse topics. They dense topics contain active codes from multiple ICD-10 chapters. They might be the baseline topics used by the model to handle background noise in the population, or they might also capture very complex multi-morbidity patterns. It is possible that diseases from many chapters have shared risk factors, therefore they are put in the same topic. By contrast, the top active codes in most sparse topics come from 1-2 ICD-10 chapters. Diseases from different major chapters for the same topic might have important connections that are worth of further explorations.

We also compared the topics inferred from the two UKB datasets (top-100 and top-436 dataset). Focusing on the probabilities of the 100 codes that appeared in both datasets, we can see that near identical patterns for these 100 codes were presented in topics inferred from the two datasets. Additionally, there were also topics dominated by codes that are only present in the top-436 dataset. Specifically, about six distinct topics were centered on codes from Chapter 14 (Diseases of the genitourinary system). Overall, these results indicate that limited number of stable multi-morbidity patterns exist for the common diseases in UKB, since they were independently captured by models trained on different datasets.

As was introduced in Chapter 1, four previous studies also focused on finding topics of diseases using topic models and EHR data. These studies all used relatively small cohorts to infer topics of diseases (sample sizes ranged from about 12,000 to

20,000). Three of these studies were carried out by the same group using the same analytic framework. However, the complete raw data for the inferred topics have not been found by far, so comparison of topics have not been made. Nevertheless, the researchers empirically chose 50 topics to describe all the multi-morbidity patterns in the cohorts, which was in agreement with our results. The other previous study only inferred six disease topics using NMF. The top active codes in these topics are shown in Figure 5.11. Three topics among the six ones (Topic-1, 2 and 4) have corresponding topics inferred by treeLFA on the UKB data (Topic-10, 15 and 9). For the remaining three topics, although many of their top active codes are also included in the top-436 UKB dataset, they were not inferred by treeLFA. Instead, top active codes in these topics were mainly included in the few dense topics, suggesting no characteristic co-morbidity patterns were identified for them. This difference might be caused by the difference in the input data. The previous study used a cohort of 12,759 individuals collected from a single medical center, which was small in size and might have selection bias. Besides, the previous study also included codes for symptoms in the input data, which might cause change to the structure of topics. Overall, our study is based on one of the largest biobanks to date, and should be more reflective of the major multi-morbidity clusters in the UK population from an epidemiological perspective.

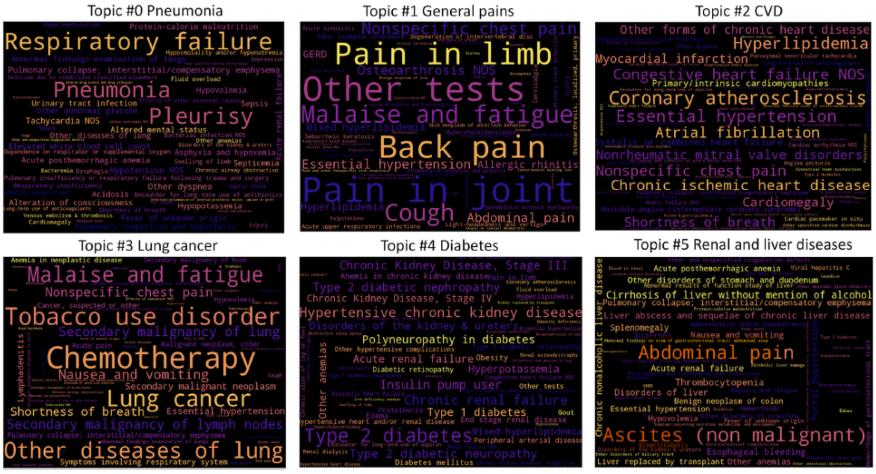


Figure 5.11: Topics of diseases inferred in a previous study. Original caption: Word clouds for six topics. The size of the words (phecode) in each cloud indicates the weights of the phenotypes on the topic. Phenotypes with larger-sized words have greater influence on the topic compared to phenotypes with smaller-sized words. For each word cloud, we listed the top 60 words (Source: Zhao *et al.*, 2019)

5.4.2 GWAS on topic weights

topic-GWAS result

As before, topic-GWAS was carried out on inferred topic weights as continuous traits. For the 40 topics inferred by treeLFA, 278 topic-associated loci were found.

This number is larger than the 180 loci found for the 32 topics inferred by treeLFA on the top-100 UKB dataset. Among the 40 topics inferred by treeLFA on the top-436 dataset, there are seven topics (Topic 10, 13, 16, 23, 25, 27, 33) that are centered on diseases from Chapter 14 of the ICD-10 system. Codes in this chapter was not included in the top-100 dataset. With these seven topics excluded, the remaining topics still have 262 associated loci. These suggest that the genetic associations for topics inferred on the larger UKB dataset were not entirely driven by disease codes with large prevalence in UKB (such as those included in the top-100 dataset), and it is important to include diseases with smaller prevalence into the analyses.

Among the 278 loci associated with the 40 topics, 80 are only found by topic-GWAS, which is also consistent with the result obtained from the top-100 UKB dataset (for models with different numbers of topics, about 50 loci are only found by topic-GWAS). The overlap of single code associated loci and topic-associated loci is large (198) relative to the total 280 topic-associated loci, suggesting most topic-associated loci have large effects, and can be readily found by the standard GWAS. This also shows that a substantial fraction (18.1 %) of single code associated loci have pleiotropic effects on multiple related diseases, and that topic-GWAS provides additional power for discovery (new loci that couldn't be found by single code GWAS). As for the effect sizes of significant loci, it is found that compared to loci only associated with single code, loci associated with both topic and single code are more likely to have larger effect sizes for certain codes. Meanwhile, for those loci only associated with topic but not single code and having positive effects, their effect sizes are slightly smaller than those associated with both topic and single code. This again verifies the additional power provided by topic-GWAS, since it enables the discovery of variants with smaller effect sizes.

Among the inferred topics, two topics centered on metabolic diseases (Topic 11 and 18) have the largest numbers of associated loci (54 and 54 loci), followed by the empty topic (35 associated loci, 21 only found by topic-GWAS). A few other topics centered on the urinary system (Topic 10 and 25), the immune system (Topic 3), the skin tissue (Topic 24), and the digestive system (Topic 6 and 22) also have a substantial numbers of topic-associated loci. It is noteworthy that for a two topics centered on immune diseases (Topic 3 and 30), a large proportions of topic-associated loci (81.5 % and 83.3 %) were not found by single code GWAS. This result is consistent with the findings of a few recent studies that focused on applying multi-trait analysis on immune diseases [159–161], and indicates that analyzing multiple immune disease together is of great value. Topic 2, a dense topic, also has quite a few associated loci that can only be identified by topic-GWAS. Similar results were also seen for the top-100 dataset. This may reflect one of the advantages of treeLFA over standard LDA. On the top-100 dataset, we found that most sparse disease topics inferred by treeFLA and LDA were very similar, while the dense topics were slightly different (Section 3.3.3). For topic-GWAS, sparse disease topics inferred by treeLFA and LDA have similar numbers of associated loci, while treeLFA found more significant loci for dense topics than LDA (Section 4.2.4). Overall, these results demonstrate that the configuration of treeLFA allows for better quantification of people's weights for dense topics. On the contrary, the sparse disease topics are easier to infer, and both models (treeLFA and LDA) can capture them very well.

Validation of topic-GWAS results

As before, we used the GWAS catalogue and functional genomics resources to validate the topic-associated loci. The validation results are within out expectations and are similar to that for the top-100 dataset, which again verifies that the loci found by topic-GWAS are reliable.

With genes mapped from the lead SNPs associated with a specific topic, tissue and gene-set enrichment analyses were performed using the mapped genes as input. For most topics the tissue enrichment analyses didn't give significant results (not shown). This might reflect the complexity of the biological pathways and the multitude of types of tissue and cell involved in the pathogenic processes for diseases in the same topic. Meanwhile, it is also important to point out that the functional analyses done for the topic-associated SNPs are far from comprehensive. The main goal of the functional analyses in this study is to validate the findings of topic-GWAS, instead of diving deeply into the biological mechanisms behind these associations. In the future, more carefully designed functional analyses should be carried out for the topic-associated SNPs. Besides, considering the complexity of biological mechanisms involved, performing functional analyses on all the topic-associated loci may not be the best choice. Instead, it might be better to first replicate the topic-GWAS results on an independent cohort, and then focus on the strong and stable associations.

For the gene set enrichment analyses, an interesting result is that for dense and sparse topics, different enrichment patterns were observed. For instance, as was presented in Section 5.3.4, the enriched gene sets for the empty topic are mainly associated with life-style factors in the GWAS catalogue, such as regular attendance to the gym and having religious belief, while for sparse disease topics, most enriched gene sets are directly related to the topics' top active codes in the GWAS catalogue. The results for sparse disease topics are intuitive and understandable, since they reflect that the topic-associated loci have pleiotropic effects on multiple active codes in the topic, which is also the theoretical basis for topic-GWAS. The result for the empty topic is more interesting, and further studies are needed to dissect the associations. In addition to the empty topic, we also obtained similar result for a few dense topics (not shown), and it is reasonable to assume that the results for these topics share common underlying reasons.

For the loci observed to be associated with both the empty topic and various human's behaviours and life-style related traits, it is possible that they are indirectly associated with people's weights for the empty topic, as a result of the life-style factors' influence on the risks of multiple diseases. In another word, weight for the empty topic is a trait correlated to many life-style related traits. Based on previous GWAS results [162–164], a potential mechanism for behaviour-associated loci is they influence the gene expression in human's brain tissues. Unfortunately, according to tissue enrichment results given by FUMA, brains tissues were not enriched for genes mapped to empty topic associated loci. In the future, the empty topic (as well as the dense topics) can be analyzed together with life-style traits in the same framework to further dissect their relationships with the associated variants. Notably, a large fraction (about 33.3 %) of the empty topic associated loci do not have record in the GWAS catalog. If their associations with the empty topic are true, then it will be likely that in previous GWAS traits were not properly defined for the discovery of them. In addition, we also should not overlook the possibility that some empty topic associated loci may be directly associated with both the empty topic and other life-style traits. It this is the case, then these loci may be involved in unknown biological pathways which influence people's general health condition. Regardless of which assumption is true, these results imply the possibility that constellations of large number of diseases share common genetic and non-genetic risk factors.

6
HLA typing and validation

Contents

6.1 Introduction

This chapter summarises the work I did on a side project during the first year of my study in Oxford. This chapter is included in the thesis because it fits into the big picture of my major project. The focus of my thesis is the study of multi-morbidity clusters from the genetic perspective, and this chapter is about the characterization of genetic variations in a special region in human's genome, the HLA gene complex. At the time I started working on this side project, the input data was only available for 49,960 people in the UKB. In the near future, the data for all people in the UKB will become available. By that time, I will be able to finish the analyses that will be discussed in this chapter using the full UKB data, which will lay a foundation for other future projects.

The HLA gene complex is one of the most important and complex regions in human's genome. Genetic variations in this region have been discovered to be associated with a wide variety of diseases and health conditions. Furthermore, it is very difficult to determine people's genotype for genes in this region due to the polymorphism, the homology between different HLA genes, the large structural variants and the long range haplotype structure in this region [111].

In recent years, the cost of the next generation sequencing (NGS) and genotyping has decreased significantly. As a result, biobanks have been set up around the world with genotype data for hundreds of thousands of people collected and linked to the nationwide electronic health record system, which enables more flexible design and implementation of genotype-phenotype association studies. At the same time, statistical HLA inference algorithms have been under constant development and improvement. The newest generation of these algorithms have already achieved accuracy as good as traditional sequencing based methods such as Sanger based typing (SBT) [135].

In spite of the huge progress in methodological development, these inference algorithms for HLA loci have rarely been comprehensively validated on biobank datasets. In this chapter, we will firstly use the SBT data in the 1000 Genomes dataset to validate one of the cutting-edge in-silico HLA inference algorithms, "HLA*LA", before applying it on the UKB. Although there is no traditional SBT data available for individuals in the UKB, we will use their genotype calls and the HLA imputation to benchmark the algorithm.

6.2 Method

6.2.1 Input data

For our study, the whole exome sequencing (WES) data for 49,960 people in the UKB was used for HLA typing by HLA*LA [165]. Furthermore, we used the imputed HLA alleles given by the HLA imputation algorithm "HLA*IMP:02" (detailed introduction was in the next section) for all people in the UKB [166].

For the validation of HLA*LA, the high coverage whole genome sequencing (WGS) data, the WES data [167] and the SBT data for a fraction of people in the 1000 Genomes dataset was used [168].

6.2.2 Imputation of HLA alleles

Various imputation algorithms for HLA alleles based on genotyping data have been developed in the past decade. For all people in the UKB, their HLA alleles have been imputed with the HLA*IMP:02 algorithm [166]. With a five-fold cross validation among the reference panel samples with European ancestry, the estimated imputation accuracy for the four-digits maximum posterior probability genotype is above 93.9 % for all 11 HLA loci [166].

The HLA*IMP:02 is based on a multi-population reference panel, which enables the imputation in diverse populations. The are two important innovations of HLA*IMP:02. Firstly, it allows for single HLA types to appear on heterogeneous backgrounds of SNP haplotypes (a phenomenon called haplotypic heterogeneity). Secondly, it takes into account the possibility of genotyping errors, which is common in the HLA region [125].

As mentioned above, imputation under HLA*IMP:02 is based on a graphical model of the haplotype structure of the HLA region constructed using the reference panel of samples, since it is well suited to model the LD relationships spanning both long and short physical distances, which is a distinguishing feature for the LD structure in the HLA region. Figure 6.1 shows a schematic for this haplotype graph. The graph is leveled such that it can be used to represent different genomic positions. Each edge in the graph carries either a nucleotide or a HLA allele. A full path through the graph specifies both the HLA allele at a HLA locus and the SNP structure in its neighbouring area.

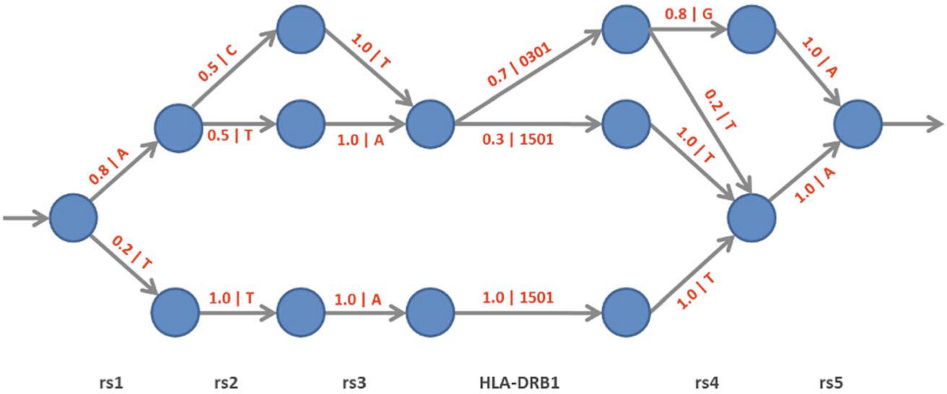


Figure 6.1: Schematic for the haplotype model used by HLA*IMP:02. The haplotype model is a leveled directed graph, with its edges representing different genomic loci. Each edge in the graph carries a symbol, which can be either a SNP or an allele for a HLA locus. The symbol on a edge will be emitted if the edge is traversed. As a result, a path through the model will generate a specific haplotype in the HLA region. Each node in the graph has a probability distribution over all the edges attached to it, which specify the probability of traversing different edges conditioned on being at that node. (Source: Alexander Dilthey, 2013)

The haplotype graph induces a hidden markov model (HMM). States of the HMM correspond to the edges of the haplotype graph, and transition probabilities between states are defined by the edge probability distribution at nodes of the graph. With the haplotype graph and its underlying HMM model constructed from the reference panel of samples, imputation of unknown HLA alleles can be done based on the observed genotype for SNPs in its surrounding region. Mathematically, the inference can be expressed as: $\boldsymbol{P(h' \mid h, M)}$, in which $\boldsymbol{M}$ is the constructed haplotype HMM, h is the observed haplotype for this region with some unknown HLA loci, while $\boldsymbol{h'}$ is a haplotype with all loci known.

6.2.3 Typing of HLA alleles

In the past decade, many NGS based HLA typing algorithms were also developed [129–133, 169]. Chen, et al [135] evaluated and compared these algorithms, and found that with both WES and WGS data these algorithms provided typing accuracy comparable to SBT for most common HLA alleles. For our study, we chose HLA*LA for the HLA typing with the UKB data.

As with HLA*IMP:02, HLA*LA also relies on the use of a graph model, the population reference graph (PRG, Figure 6.2) to represent known HLA sequences in the reference database (such as IMGT) as paths through the graph. The known HLA sequences used for construction of the graph include HLA haplotypes (spanning the whole HLA region), gene haplotypes and exon sequences. With the PRG constructed, the second step is to align the NGS reads to the reference genome, and then project the alignments in HLA loci onto the PRG. In the last stage, the inference of HLA alleles for a locus is performed by finding the pair of alleles which maximises the likelihood of all reads mapped to the PRG, which can be expressed in the equation below:

$$\boldsymbol{L(R \mid (a1, a2)) = \prod_{r \in R} [\frac{1}{2} \times score(r \mid a1) + \frac{1}{2} \times score(r \mid a2)].}$$

In the equation, $\boldsymbol{R}$ represents all the read pairs mapped onto the graph for a specific HLA locus; $\boldsymbol{r}$ represents one pair of aligned reads; $\boldsymbol{a1}$ and $\boldsymbol{a2}$ are the pair of underlying HLA alleles for this locus. The score function evaluates the match between the alleles and the reads. Likelihoods for all possible pairs of alleles are calculated and normalized to a probability distribution [130, 131].

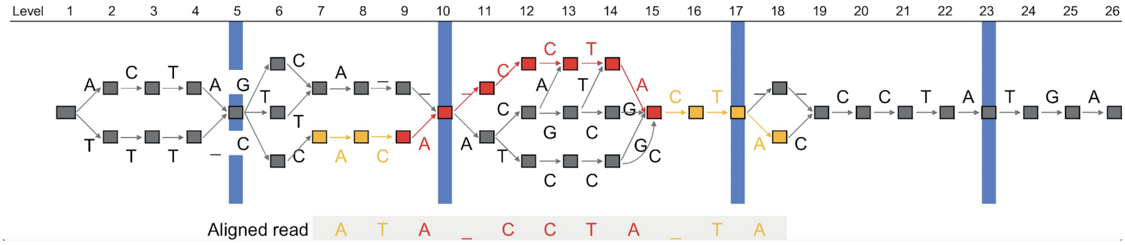

Level 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26
Aligned read A T A _ C C T A _ T A

Figure 6.2: Schematic for the Population reference graph used by HLA*LA. PRG is a leveled graph, with each level corresponding to a genomic position. Paths through the graph are different sequences of HLA genes in the reference panel of samples. NGS reads are mapped to the graph so that the inference of unknown alleles can be performed in a likelihood framework. (Source: Alexander T. Dilthey, 2019).

It is worth mentioning that inference of HLA alleles can be performed at different resolutions (see Chapter 1 for more detailed introduction). HLA*LA outputs G-alleles (inferred alleles at the G-resolution), which specify the sequences of exons that encode the peptide binding sites (PBS) of HLA proteins (exons 2 and 3 for HLA class I genes and exon 2 for HLA class II genes) [130]. In another word, HLA alleles differ with each other outside of the PBS encoding region are put into the same G-group. The concept of G-alleles is further explained in Table 6.1.

G-resolution allele	*Alleles in G-group*
A*01:01:01G	01:01:01:01/01:01:01:02N/01:01:01:03 01:01:01:04/01:01:01:05/01:01:01:06/01:01:01:07/01:01:01:08/01:01:01:09/01:01:01:10/01:01:01:11/01:01:01:12/01:01:01:13/01:01:01:14/01:01:01:15/01:01:01:16/01:01:01:17/01:01:01:18/01:01:01:19/01:01:01:20/01:01:01:21/01:01:01:22/01:01:01:23/01:01:01:24/01:01:100/01:01:38L/01:01:51/01:01:83/01:01:84/01:01:91/01:01:93/01:01:94/01:01:95/01:04:01:01N/01:04:01:02N/01:22N/01:32/01:37:01:01/01:37:01:02/01:45/01:56N/01:81/01:87N/01:103/01:107/01:109/01:132/01:141/01:142/01:155/01:177/01:212/01:217/01:234/01:237/01:246/01:248Q/01:249/01:251/01:252/01:253/01:261/01:274/01:276/01:277/01:280/01:281Q/01:288/01:291/01:295/01:296/01:297/01:300/01:305
A*01:03:01G	01:03:01:01/01:03:01:02/01:287N
A*01:09:01G	01:09:01:01/01:09:01:02

Table 6.1: Explanation of G-alleles. A G-allele is a group of different HLA alleles which encode the same peptide binding site (PBS). Three examples of G-alleles (G-groups) are shown. Each G-group is consisted of many different HLA alleles. HLA*LA outputs the inferred HLA allele at G-resolution.

[h!]

6.2.4 Validation of HLA*LA with 1000 Genomes dataset

We used the SBT data for people in the 1000 Genomes dataset as the benchmark for the validation of HLA*LA. The SBT data was available for five HLA loci (A, B, C, DRB1 and DQB1), for which the typing accuracy of HLA*LA were evaluated. There were 946 individuals in the 1000 Genomes data whose WGS and SBT data were both available, and 985 individuals whose WES and SBT data were both available. As HLA*LA, SBT also output G-alleles, which were a list of possible alleles for a locus. To undertake the validation, the method in the original HLA*LA paper was adopted [130]. More specifically, to conclude HLA*LA and SBT gave the same result, at least one possible allele output by SBT needed to belong to the G-group inferred by HLA*LA. The formal definition of the number of alleles correctly typed by HLA*LA for a locus can be expressed as below:

$$\begin{aligned} \boldsymbol{correct(I_1, I_2, V_1, V_2)} &= \boldsymbol{max(correct2(I_1, I_2, V_1, V_2), correct2(I_1, I_2, V_2, V_1))} \\ \boldsymbol{correct2(I_x, I_y, V_x, V_y)} &= \boldsymbol{same\_G\_group(I_x, V_x) + same\_G\_group(I_x, V_x)} \end{aligned}$$

where the function $\boldsymbol{correct(I_x, I_y, V_x, V_y)}$ counts the number of alleles correctly inferred by HLA*LA for a HLA locus (the count can only be 0, 1 or 2); the function

$\boldsymbol{correct2(I_x, I_y, V_x, V_y)}$ counts the number of correctly inferred alleles using a specific one-to-one matching between the pair of inferred alleles and the pair of reference alleles (given by SBT); the function $\boldsymbol{same\_G\_group(I_x, V_x)}$ tests if the two groups of alleles belong to the same G-group. $\boldsymbol{I_1}$ and $\boldsymbol{I_2}$ are the pair of G-alleles inferred by HLA*LA for an individual at a specific HLA locus, and $\boldsymbol{V_1}$ and $\boldsymbol{V_2}$ are the pair of G-alleles given by SBT.

6.2.5 HLA typing with the WES data in UKB

HLA*LA was used to type the HLA loci for 49,960 people in the UKB for whom the WES data was available. According to the relevant publications, these people are overall representative of all individuals in the UKB, and the coverage of the exome sequencing was high (exceeded **20×** at 94.6 % of sites on average) [165].

For individuals in the UKB, their SBT results were not available. To roughly check if the typing result given by HLA*LA was reliable, its agreement with the imputed HLA alleles given by HLA*IMP:02 was checked. Since HLA*IMP:02 output imputed alleles at the 4-digits resolution, while HLA*LA output typed alleles at the G-resolution, a specific protocol was designed to check their agreement. In brief, if the 4-digits allele given by HLA*IMP:02 had overlap with alleles in the G-group inferred by HLA*LA (changed to 4-digits resolution), we concluded that a consensus was reached by the two algorithms. Moreover, since HLA*LA and HLA*IMP:02 used different reference panels, we neglected the disagreements caused by calling HLA alleles that only existed in one reference panel.

6.3 Results

6.3.1 Validation of HLA*LA

Typing accuracy evaluated with the 1000 Genomes dataset

The overall inference accuracy with WGS data for all individuals in the 1000 Genomes dataset for the five HLA loci had a mean of 98.2 %, ranging from 97.1 % for HLA*DRB1 to 99.2 % for HLA*B (Figure 6.3A). The accuracy was reduced at all loci when the inference was based on WES data, ranging from 89.8 % for HLA*A to 97.7 % for HLA*B (Figure 6.3B).

The inference accuracy was also measure for different populations in the 1000 Genomes dataset separately. With WGS data (Figure 6.4A), for most populations (except for the Yoruba population, which had an averaged accuracy of 94.8 %) the averaged inference accuracy was above 97 % for the five HLA loci. With WES data, the inference accuracy was much reduced, and the differences between populations increased (the averaged accuracy for the five loci ranged from 92.7 % to 97.1 % for different populations). The British population had the highest averaged accuracy (97.1 %), while the Chinese and Japanese population has the lowest averaged accuracy (92.7 %). Among the five loci, the inference accuracy of locus A was usually lower than the other four loci. For most populations the

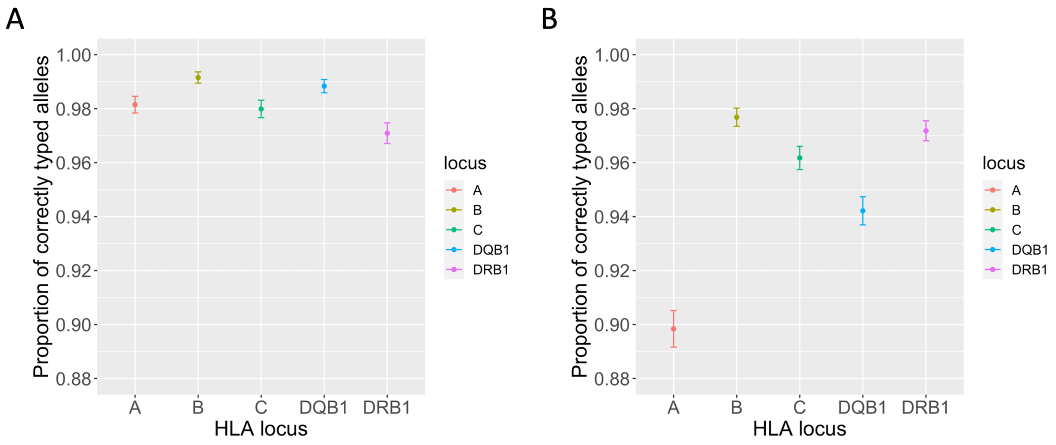


Figure 6.3: Typing accuracy of HLA*LA for five HLA loci was evaluated with the 1000 Genomes dataset. The SBT result was used as the benchmark to calculate the proportion of correctly typed HLA alleles for each locus. The standard error of a proportion $\boldsymbol{p}$ was calculated as $\sqrt{\boldsymbol{p \cdot (1-p)/N}}$, where $\boldsymbol{N}$ is the sample size. A, Typing accuracy of HLA*LA with WGS data across all populations. B, Typing accuracy with WES data.

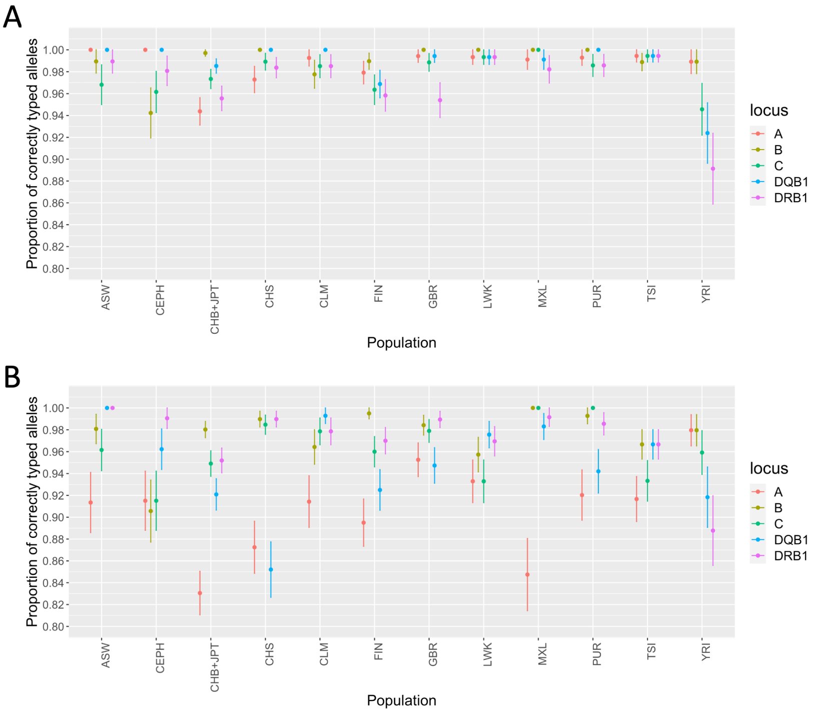
accuracy of locus A was less than 92 %, and for the JPT+CHB population the accuracy was only 83.1 %. In addition, the typing accuracy for the locus DQB1 was also not high in most populations. Overall, this result was in agreement with the result for the whole population (Figure 6.3B).

By comparing the typing accuracy with WES and WGS data for the same locus in different populations, it can be seen that in most cases, the accuracy with WES data was lower than the accuracy with WGS data. However, there were also exceptions. For instance, the accuracy for locus HLA*DRB1 with WES data (98.9 %) was higher than the accuracy with WGS data (95.4 %) for British people.

Relationship between allele frequency and typing accuracy

Through the inspection of the alleles wrongly typed by HLA*LA with the WES data, it was discovered that many of them were rare alleles. It was reasonable to assume that rare alleles were more difficult to infer, since for them the training data for the typing algorithm was more scarce than more common alleles.

To further investigate the relationship between allele frequency and typing accuracy, the frequency of HLA alleles in the 1000 Genomes dataset (calculated using the SBT result) and their typing accuracy were plotted in Figure 6.5. For the typing done with WGS data across all populations (Figure 6.5A), there were 380 different alleles typed for the five loci for 946 individuals. Among the 380 alleles, 293 alleles (77.1 % of all alleles) were all typed correctly (typing accuracy 100 %), and 333 alleles (87.6 % of all alleles) had typing accuracy larger than 95 %. For the typing done with WGS data for the British population (Figure 6.5B), 98 different alleles were typed for the five loci for 87 individuals, and 90 alleles



	code	population
1	CHS	Southern Han Chinese
2	FIN	Finnish
3	PUR	Puerto Rican
4	ASW	African Ancestry SW
5	YRI	Yoruba
6	LWK	Luhya
7	JPT	Japanese
8	CEU	CEPH
9	CHB	Han Chinese
10	CLM	Colombian
11	TSI	Toscani
12	MXL	Mexican Ancestry
13	GBR	British

Figure 6.4: Typing accuracy for the five HLA loci was calculated for different populations in the 1000 Genomes dataset separately. A, Typing accuracy with WGS data. B, Typing accuracy with WES data.

(77.1 % of all alleles) were all typed correctly (typing accuracy 100 %), and 92 alleles (93.9 % of all alleles) had typing accuracy larger than 95 %. For the typing done with WES data across all populations (Figure 6.5C), there were 389 different alleles typed for the five loci for 985 individuals. Among them, the instances of 249 alleles (64 % of all alleles) were all typed correctly (typing accuracy 100 %), and 301 alleles (77.4 % of all alleles) had typing accuracy larger than 95 %. For the typing done with WES data for the British population (Figure 6.5D), 99 different alleles were typed for the five loci for 95 individuals, and 85 alleles (85.9 % of all alleles) were all typed correctly (typing accuracy 100 %), and 88 alleles (88.9 % of all alleles) had typing accuracy larger than 95 %.

With both WGS and WES data, in general the typing accuracy was proportional to the allele frequency, which meant it was more likely for common alleles to have better typing accuracy than rare alleles. Most of the alleles with relatively low typing accuracy (for instance, typing accuracy below 80 %) had less than 1 % frequency. With WGS data, 90 % of the alleles with frequency larger than 0.025 had typing accuracy larger than 0.95 (Figure 6.5A). However, with WES data, only 57.5 % of the alleles with frequency larger than 0.025 had typing accuracy larger than 0.95 (Figure 6.5C). For British people, typing results for most alleles were accurate, with either WES or WGS data.

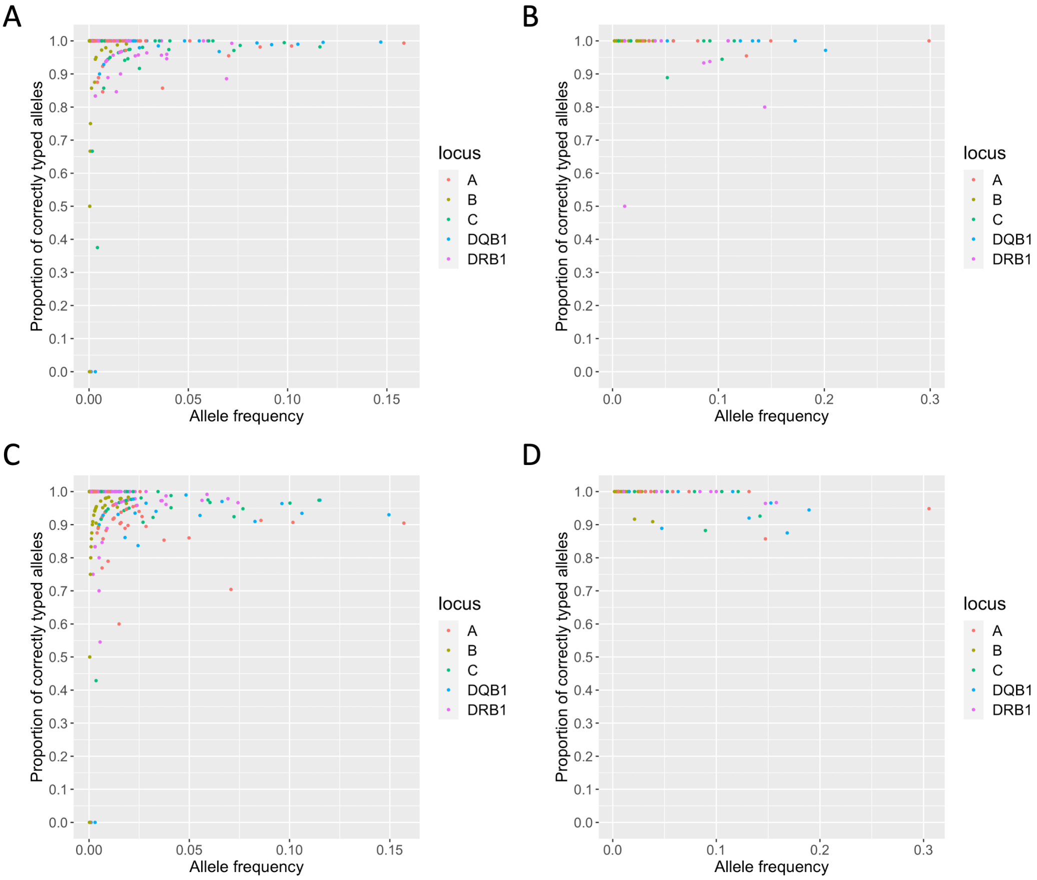


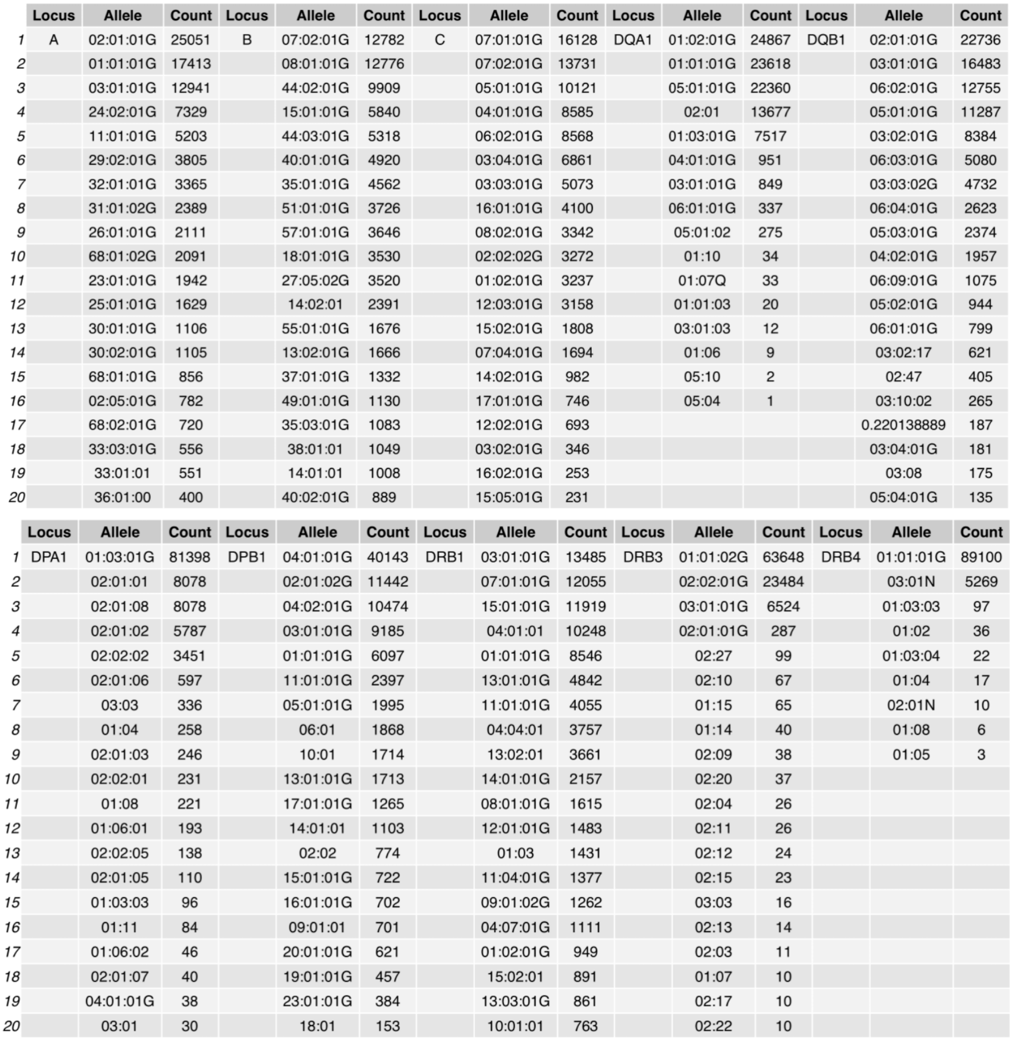
Figure 6.5: The relationship between typing accuracy and allele frequency. For all alleles of the five HLA loci typed for individuals in the 1000 Genomes dataset, allele frequency was calculated using the SBT result, and plotted together with the corresponding allele specific typing accuracy. A, The typing accuracy with WGS data and the allele frequency across all populations. B, The typing accuracy with WGS data and the allele frequency for British people. C, The typing accuracy with WES data and the allele frequency across all populations. D, The typing accuracy with WES data and the allele frequency for British people.

6.3.2 HLA typing with UKB data and validation of typing result

After the validation of HLA*LA with the SBT data for the 1000 Genomes dataset, the algorithm was applied to the WES data of 49,960 people in the UKB. In this section the summary statistics of the typing result was presented, then the agreement between the typing result given by HLA*LA and the imputation result given by HLA*IMP:02 was checked.

Top frequent alleles called by HLA*LA for people in UKB

Figure 6.6 lists the top 20 most frequent alleles called by HLA*LA for ten HLA loci. As can be seen, HLA*LA called not only the G-group alleles, but also four or six digits alleles as well, since some alleles cannot be clustered into any G-group. HLA*LA also called some very rare alleles, like DQA1*05:04, for which there was only one copy among the 49,960 individuals.



	Locus	Allele	Count	Locus	Allele	Count	Locus	Allele	Count	Locus	Allele	Count	Locus	Allele	Count
1	A	02:01:01G	25051	B	07:02:01G	12782	C	07:01:01G	16128	DQA1	01:02:01G	24867	DQB1	02:01:01G	22736
2		01:01:01G	17413		08:01:01G	12776		07:02:01G	13731		01:01:01G	23618		03:01:01G	16483
3		03:01:01G	12941		44:02:01G	9909		05:01:01G	10121		05:01:01G	22360		06:02:01G	12755
4		24:02:01G	7329		15:01:01G	5840		04:01:01G	8585		02:01	13677		05:01:01G	11287
5		11:01:01G	5203		44:03:01G	5318		06:02:01G	8568		01:03:01G	7517		03:02:01G	8384
6		29:02:01G	3805		40:01:01G	4920		03:04:01G	6861		04:01:01G	951		06:03:01G	5080
7		32:01:01G	3365		35:01:01G	4562		03:03:01G	5073		03:01:01G	849		03:03:02G	4732
8		31:01:02G	2389		51:01:01G	3726		16:01:01G	4100		06:01:01G	337		06:04:01G	2623
9		26:01:01G	2111		57:01:01G	3646		08:02:01G	3342		05:01:02	275		05:03:01G	2374
10		68:01:02G	2091		18:01:01G	3530		02:02:02G	3272		01:10	34		04:02:01G	1957
11		23:01:01G	1942		27:05:02G	3520		01:02:01G	3237		01:07Q	33		06:09:01G	1075
12		25:01:01G	1629		14:02:01	2391		12:03:01G	3158		01:01:03	20		05:02:01G	944
13		30:01:01G	1106		55:01:01G	1676		15:02:01G	1808		03:01:03	12		06:01:01G	799
14		30:02:01G	1105		13:02:01G	1666		07:04:01G	1694		01:06	9		03:02:17	621
15		68:01:01G	856		37:01:01G	1332		14:02:01G	982		05:10	2		02:47	405
16		02:05:01G	782		49:01:01G	1130		17:01:01G	746		05:04	1		03:10:02	265
17		68:02:01G	720		35:03:01G	1083		12:02:01G	693					0.220138889	187
18		33:03:01G	556		38:01:01	1049		03:02:01G	346					03:04:01G	181
19		33:01:01	551		14:01:01	1008		16:02:01G	253					03:08	175
20		36:01:00	400		40:02:01G	889		15:05:01G	231					05:04:01G	135

	Locus	Allele	Count	Locus	Allele	Count	Locus	Allele	Count	Locus	Allele	Count	Locus	Allele	Count
1	DPA1	01:03:01G	81398	DPB1	04:01:01G	40143	DRB1	03:01:01G	13485	DRB3	01:01:02G	63648	DRB4	01:01:01G	89100
2		02:01:01	8078		02:01:02G	11442		07:01:01G	12055		02:02:01G	23484		03:01N	5269
3		02:01:08	8078		04:02:01G	10474		15:01:01G	11919		03:01:01G	6524		01:03:03	97
4		02:01:02	5787		03:01:01G	9185		04:01:01	10248		02:01:01G	287		01:02	36
5		02:02:02	3451		01:01:01G	6097		01:01:01G	8546		02:27	99		01:03:04	22
6		02:01:06	597		11:01:01G	2397		13:01:01G	4842		02:10	67		01:04	17
7		03:03	336		05:01:01G	1995		11:01:01G	4055		01:15	65		02:01N	10
8		01:04	258		06:01	1868		04:04:01	3757		01:14	40		01:08	6
9		02:01:03	246		10:01	1714		13:02:01	3661		02:09	38		01:05	3
10		02:02:01	231		13:01:01G	1713		14:01:01G	2157		02:20	37			
11		01:08	221		17:01:01G	1265		08:01:01G	1615		02:04	26			
12		01:06:01	193		14:01:01	1103		12:01:01G	1483		02:11	26			
13		02:02:05	138		02:02	774		01:03	1431		02:12	24			
14		02:01:05	110		15:01:01G	722		11:04:01G	1377		02:15	23			
15		01:03:03	96		16:01:01G	702		09:01:02G	1262		03:03	16			
16		01:11	84		09:01:01	701		04:07:01G	1111		02:13	14			
17		01:06:02	46		20:01:01G	621		01:02:01G	949		02:03	11			
18		02:01:07	40		19:01:01G	457		15:02:01	891		01:07	10			
19		04:01:01G	38		23:01:01G	384		13:03:01G	861		02:17	10			
20		03:01	30		18:01	153		10:01:01	763		02:22	10			

Figure 6.6: The top frequent alleles called by HLA*LA for ten HLA loci using the WES data for 49,960 individuals in the UKB.

Agreement between typing and imputation results

In contrast to the results above presented for the 1000 Genomes dataset, SBT results are not available for UKB. Therefore we used the availability of HLA*IMP:02 calls in place of SBT calls to assess the accuracy of HLA*LA on 49,960 individuals in UKB.

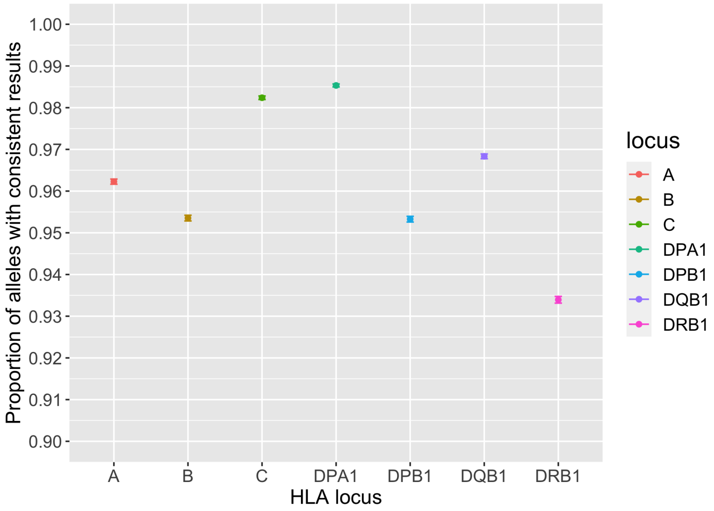


Figure 6.7: Agreement between inference results given by HLA*LA and HLA*IMP:02. The proportion of individuals in the UKB for whom the two HLA inference algorithms gave consistent results was used to represent the level of agreement between the two methods.

The agreement between the typing and imputation results for seven HLA loci was shown in Figure 6.7. Overall, the agreement was high for all the seven loci (ranging from 93.5 % for locus DRB1 to 98.5 % for locus DPA1). Except for checking the overall agreement for each HLA locus, we also checked the agreement across all the seven loci for different populations in the UKB. The agreement for about half of the populations was above 95 %, especially the major populations in UK (including various British populations and the Irish population) and other white populations. Meanwhile, the agreement for people who self-reported as having mixed background is low Figure 6.8.

With the 1000 Genomes dataset, it was found that the inference accuracy for some alleles was significantly lower than that for other alleles, and this was also related to the allele frequency (Figure 6.5). With the inference results for individuals in the UKB, we tried to only keep the alleles which had been proved to have high typing accuracy (larger than 95 %) with the 1000 Genomes dataset, and checked the agreement with the imputation result again for these alleles. In Figure 6.9, the level of agreement before and after excluding the alleles low typing accuracy was compared. It can be seen that for both all individuals and British individuals only, with rare alleles (not in the 1000 Genomes dataset) and alleles which had low typing accuracy excluded, the proportion of people with consistent results increased for 1-2 % for most loci.

Results for HLA*DQA1, HLA*DRB3 and HLA*DRB4 were separately shown in Figure 6.10A, since the agreement reached by the two methods was very low (around 35 % for the two DRB1 loci). We investigated the possible reasons behind this, and found that for HLA*DRB3 and HLA*DRB4, HLA*IMP:02 called many 99:01 alleles, which means the absence of one copy of allele for this locus (people can have different copy numbers for these loci, see Section 1.1.1 for more details).

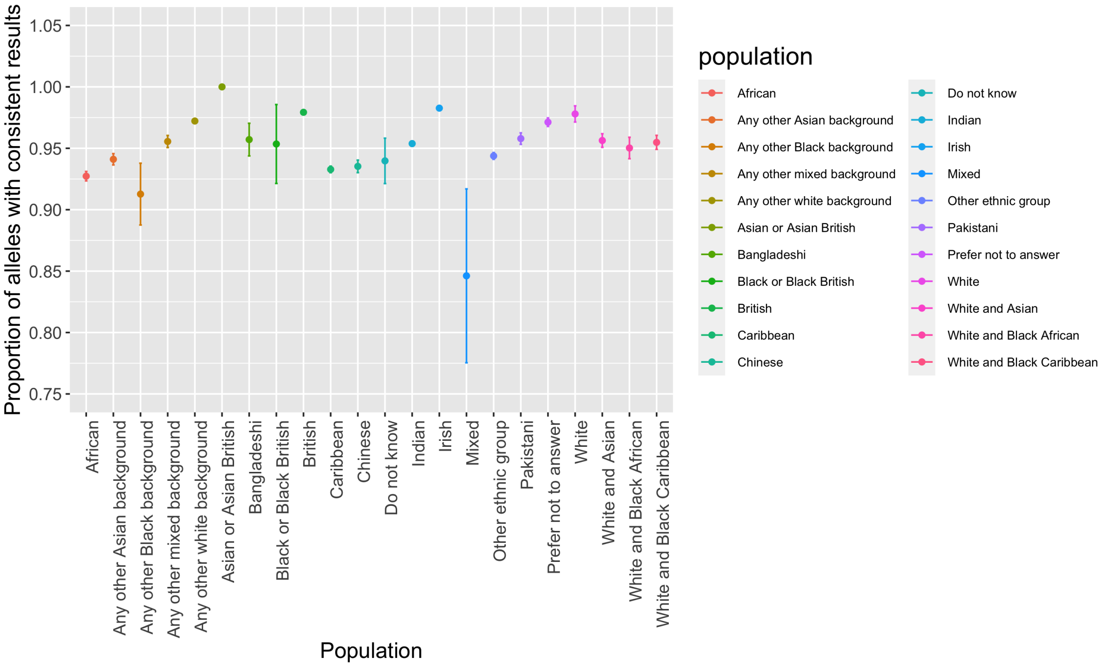


Figure 6.8: Agreement between the inference results given by HLA*LA and HLA*IMP:02 for different populations in the UKB. People in the UKB were put into different populations according to their self-reported ancestry. The agreement between the two algorithms was measured across all the seven loci.

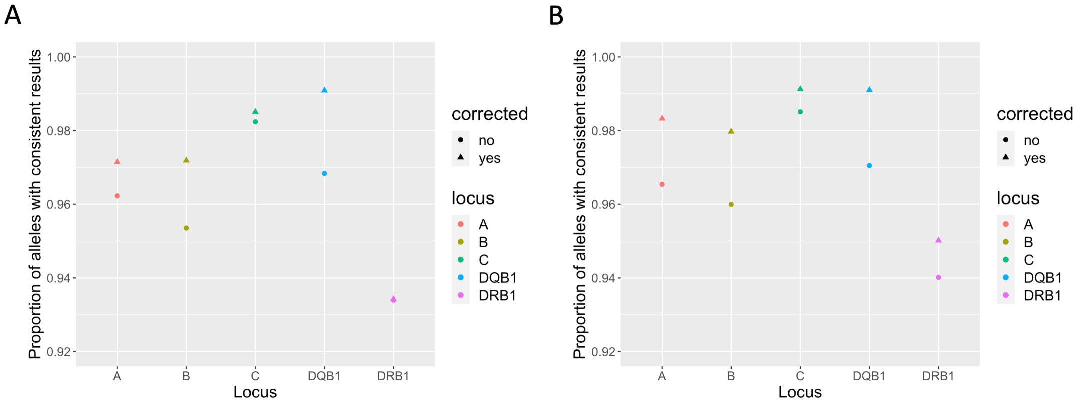


Figure 6.9: Agreement between inference results given by HLA*LA and HLA*IMP:02 before and after excluding alleles with low typing accuracy in the 1000 Genomes dataset. A, Result for all individuals in the UKB. B, Result for British people in the UKB.

On the other hand, HLA*LA never called 99:01 for loci DRB3 and DRB4. This property of HLA*LA was mentioned in the tutorial for HLA*LA ("Since HLA*LA does not estimate the copy number for these alleles, it can end up with calls for genes that are not present") [170]. This meant that HLA*LA was not designed for the inference of DRB1 paralogues (including DRB3, DRB4 and DRB5 genes). After excluding individuals for whom the 99:01 allele was called by HLA*IMP:02 for loci DRB3 or DRB4, the agreement between the two algorithms significantly improved (Figure 6.10B).

In addition to the paralogues of DRB1, there was also an issue with the locus DQA1. We found that in many cases where the two methods gave different inference results, HLA*IMP:02 called the 03:01 allele, while HLA*LA didn't. The exact reason behind this disagreement is still unclear. After excluding individuals for whom the 03:01 allele was called by HLA*IMP:02, the agreement between the two algorithms again significantly improved (Figure 6.10B).

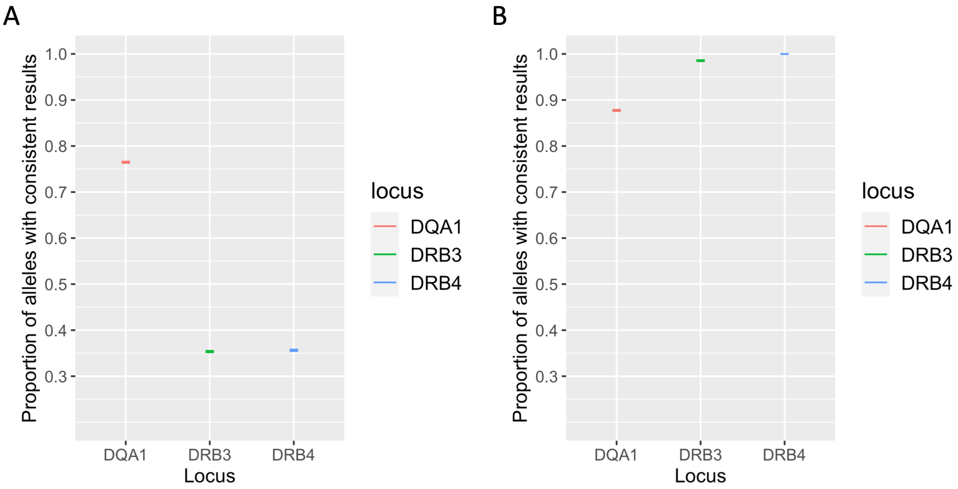


Figure 6.10: Agreement between HLA*LA and HLA*IMP:02 for three HLA loci before and after correction of potential technical issues. For HLA loci DRB3 and DRB4, the initial results were corrected by excluding the results of individuals for whom the 99:01 allele were called by HLA*IMP:02. For HLA*DQA1, results of individuals for whom the 03:01 allele were called by HLA*IMP:02 were removed. A, initial results. B, corrected results.

6.4 Discussion

In this chapter, we focused on the validation and application of an HLA typing algorithm - HLA*LA. Extensive validation for the algorithm was done with the 1000 Genomes dataset, since for people in this dataset we have both their NGS and SBT data. After validation, we applied the algorithm to the WES data in the UKB, and assessed the reliability of the typing result given by HLA*LA using the imputation result given by HLA*LA based on genotyping data.

Based on the validation result, it was found that the typing accuracy was much higher with WGS data than with WES data. This finding was in accordance with the result obtained with a small-scale validation according to the original paper for HLA*LA [130]. The typing accuracy was quite different for different HLA loci, indicating different levels of complexity for the typing at these loci. In the future, further technical improvements might be dedicated to specific HLA loci (such as DRB1). Furthermore, it was found that some common alleles also had unsatisfactory typing accuracy. In the future, it will be important to focus on these alleles and figure out if there is space for further improvement, since common alleles are ideal candidates for association studies in biobanks. Differences in the typing accuracy for different ethnicity groups were also observed. For British people, the typing accuracy was high for almost all loci, even with WES data. However, for some small populations in 1000 Genomes dataset, like the Chinese and Japanese subgroup and the Mexican subgroup, the typing accuracy for some loci (for example, the HLA*A locus) was quite low (less than 85 %). This reminds us to be cautious in applying HLA*LA to certain populations.

To assess the reliability of the typing result for individuals in the UKB, we compared the results given by HLA*LA and HLA*IMP:02. In theory, the NGS data should contain more information about the genetic variations, thus it should have the potential for a higher inference accuracy. Overall, the results given by the two algorithms should have a high level of agreement, since the performance of HLA*IMP:02 had already been validated. Notably, the two algorithms output inferred HLA alleles at different resolutions. The matching between the G-resolution (HLA*LA) and four-digits resolution (HLA*IMP:02) is not trivial, since one G-allele can correspond to multiple four-digits alleles, and one four-digit allele can also correspond to more than one G-group alleles. The difference in the resolution of the output alleles was also one of the motivations for us to do inference with HLA*LA in spite of the availability of imputed alleles for all people in UKB. Results suggested the agreement between the two methods was high, although a few technical issues were identified. In the future, it is expected that both the typing and imputation algorithms will continue to play important roles, as was discussed by Dilthey in his review [111], since the inference result obtained with NGS data can be used to further improve the accuracy of imputation algorithms, which is less expensive and more practical to be applied to many large population cohorts. Besides, the large amount of genotyping data available at present also awaits to be fully utilized.

Overall, results in this chapter suggest that HLA*LA is a reliable typing algorithm with the potential to be applied to large biobank datasets. Specifically, for British people, the typing with both WES and WGS data had satisfying accuracy for most alleles (including many rare alleles). With alleles known to have unsatisfying accuracy excluded, the performance of HLA*LA with British people can be further improved. Considering the fact that most individuals in the UKB are British, it should be reliable to use the inference result given by HLA*LA as the input for other studies. At present, the WES data for more than 450,000 individuals in the UKB has already been released and used in association studies [171–173], which has paved the way for HLA typing at an unprecedented large scale. In addition, the efforts of finishing the whole genome sequencing for all individuals in the UKB are also ongoing, and the WGS data for 200,000 individuals has also been released and utilized for research [174, 175]. In the near future, the high-quality inferred HLA alleles will be available for almost everyone in the UKB, which will undoubtedly give us the opportunity to make new discoveries for the HLA gene complex.

7
Conclusion

Contents

7.1 Studying multi-morbidity with topic models

In recent years, there have been several studies which utilized standard topics models (including LDA and NMF) to study people's major multi-morbidity patterns and the pleiotropic effects of genomic loci [88–91]. In these studies, the topic modelling framework was proved to have unique strength in revealing the major multi-morbidity clusters and their associated causal factors. However, there is still room for further improvement in terms of the analytic framework and the methodology.

In this thesis, I developed a new topic model based on the mean-parameterized Bayesian non-negative binary matrix factorization, named "treeLFA", which also incorporates a hierarchical prior for topics based on the ICD-10 ontology. Meanwhile, I proposed reliable solutions to a few technical issues related to the application of topic models. With both simulated and real-world data, it was shown that treeLFA had unique advantages compared to standard LDA. By applying it on datasets constructed using the HES data in UKB, topics of common diseases and peoples' corresponding topic weights were inferred, and various downstream analyses were explored. New discoveries that cannot be obtained by the standard single disease analysis were made using the new analytic framework developed in this thesis.

7.1.1 Strengths of treeLFA

According to the results in Chapter 4, one of the major advantages of treeLFA is that it provides additional power for downstream genetic association studies. This might result from the inference of the empty topic, whose weight is always negatively correlated with all other non-empty disease topics. Meanwhile, the empty topic also makes other disease topics mutually uncorrelated, which is a favorable situation for the downstream association studies. This situation cannot be achieved for LDA. I tried a few methods to mimic the existence of an empty topic for LDA (detailed results not shown in previous chapters), but without success.

In addition to the empty topic, there are also always a few dense topics (with a large number of active codes) inferred by treeLFA. These dense topics usually have a substantial numbers of associated loci, and some of them are not found by single code GWAS. These topics also play indispensable roles in the prediction of risks for single disease codes (relevant results not shown in Section 4.4.2). Similar dense topics are also inferred by LDA, but have many fewer associated loci.

In summary, one of the major strength of treeLFA is that it better captures the empty topic and the dense topics, which are not only biologically meaningful but also pave the way for some downstream analyses.

7.1.2 Limitations of treeLFA

The major limitation of treeLFA is the long computational time required to perform the inference, which makes it difficult to scale treeLFA to very large datasets. There are several reasons for this limitation:

Model configuration

treeLFA is based on non-negative binary matrix factorization instead of LDA, which means it models up to hundreds of binary diseases variables for each individual. By contrast, LDA only models a few diseases variables for each individual. This difference in the size of the input data is huge, so does the time needed to sample topic assignments ($\boldsymbol{Z}$) for disease variables.

Hyperparameter learning

The choice of hyperparameter $\boldsymbol{\alpha}$ has a large influence on the inference result given by treeLFA. This is largely due to the existence of the empty topic. On both UKB datasets, it can be seen that the empty topic has much larger weight in $\boldsymbol{\alpha}$ than all the other disease topics. The ratio of weights in $\boldsymbol{\alpha}$ for the empty topic and a disease topic needs to be optimized. We employed a Gibbs-EM algorithm to optimize $\boldsymbol{\alpha}$ based on multiple samples of $\boldsymbol{Z}$ generated by the Gibbs sampler, which is quite time consuming. Unlike treeLFA, LDA relies much less on the optimization of $\boldsymbol{\alpha}$ (the relevant analyses were done, but results were not shown in Section 4.2.4), which makes it faster than treeLFA.

On the UKB data, we optimized $\boldsymbol{\alpha}$ for 40 disease topics. In the future, if we are to apply treeLFA on the phenotypic data in other biobanks, a smart initialization for the optimization of $\boldsymbol{\alpha}$ can be used based on the result obtained on UKB.

Inference algorithm

treeLFA uses Gibbs sampling for inference, which is slower than other deterministic algorithms like Variational Bayes (VB). VB is another mainstream inference algorithm for topic models, which is known to converge faster than MCMC algorithms and does not spend time on collecting multiple posterior samples of hidden variables [176].

In general, it is feasible to run treeLFA on a biobank-scale dataset to analyze hundreds of diseases. In a realistic research setting, with parallel implementation of treeLFA it takes more than two weeks to finish the training of treeLFA on the top-436 UKB dataset, which is quite slow but still acceptable. Notably, diseases in the top-436 dataset have prevalences as small as 0.001, which corresponds to around 500 cases in the full UKB dataset. At present it may not be meaningful to study very rare diseases in biobanks using topic models, since it would be difficult to find stable multi-morbidity patterns for them. All in all, this means that treeFLA is capable of analysing most currently available biobank datasets, though there is certainly space for further technical improvement.

7.1.3 Configurations of topic models for phenotypic data

treeLFA and LDA have a lot in common in terms of the model configuration, since they are all based on the idea to assign diseases with different topics. The major difference is that treeLFA is aimed for binary data, and is based on NBMF. Both configurations have their own advantages: the inference for LDA is much faster, while treeLFA better characterizes dense topics and the empty topic, and also de-correlate the weights for disease topics.

Notably, there are also other options for the configuration of topic models to study multi-morbidity. For instance, Singliar *et al.* proposed the “Noisy-OR Component Analysis” framework [177], which aims to model binary observable variables with a smaller numbers of binary latent variables (topics). In this case, each topic is still a vector of probabilities that parameterize the Bernoulli distributions for all the observable binary variables. But unlike LDA and BNMF, topic weights are also binary, denoting which topics are active for individuals. If multiple topics are active for an individual, and a disease has non-zero probabilities in more than one of the active topics, then the overall probability for a disease to occur will be calculated via a noisy-or model, which can be expressed as below:

$$\boldsymbol{P(W_{dl} \mid \theta_d) = [1 - \prod_{k=1}^{K}(1 - \phi_{kl})^{\theta_{dk}}]^{W_{dl}} \cdot [\prod_{k=1}^{K}(1 - \phi_{kl})^{\theta_{dk}}]^{(1-W_{dl})}}$$

In the above equation, $\boldsymbol{W_{dl}}$ is the disease variable $\boldsymbol{l}$ for person $\boldsymbol{d}$; $\boldsymbol{\theta_d}$ is the binary indicator vector for $\boldsymbol{K}$ topics for individual $\boldsymbol{d}$; $\boldsymbol{\phi_{kl}}$ is the probability of disease $\boldsymbol{l}$ in topic $\boldsymbol{k}$. The overall probability for a disease to occur is calculated as one minus the probability that the disease does not occur through any of the active topics.

An important feature of the noisy-OR component analysis is that the topics are binary traits. As a result, the downstream GWAS will be the standard case-control study, instead of GWAS on continuous traits. This makes the comparison between topic-GWAS and single disease GWAS more straightforward, especially for the effect sizes of variants. For the topic-GWAS we carried out, attempts were also made to compare the effect sizes given by the two types of GWAS, but thus far convincing results have not been obtained (relevant results not presented in the previous chapters), since the topic-GWAS and single code GWAS are in essence different (linear regression for continuous topics weights and logistic regression for single codes). Besides, like treeLFA, weights of different topics for the same individual inferred by the noisy-OR component analysis will also be independent, since topic weights are modelled with multiple independent Bernoulli distributions, instead of a single Dirichlet distribution which implicitly assumes weak negative correlation among them. Last but not the least, the assumption of binary topic weights also makes biological sense. As is known, it is likely that along the trajectory of the pathological mechanisms of diseases, at some points there are thresholds to be reached for the activation of the downstream cascades. The binary topic weights adopted in the noisy-OR component analysis can model this process quite well.

In summary, there are different frameworks for the topic modelling of binary variables, and each of them have its advantages and disadvantages. treeLFA (NBMF) is one of the available methods, and has been proved to have its unique value. In the future, it will be interesting to try other frameworks, since they may discover different hidden structure from the data and give us insights for the underlying biology from different perspectives.

7.1.4 Priors for topics models

A distinguishing feature for treeLFA is the use of a hierarchical prior for the structure of topics. Results on both simulated data and UKB data suggest that it is a reasonable choice to use this prior, and it helps with the inference. However, its influence on the inference result is not as large as what we expected when we began this project. On the top-100 UKB dataset, treeLFA and flatLFA give almost identical results, in terms of both the inferred topics themselves and the results for the downstream analyses. On the top-436 dataset, the topics inferred by treeLFA and flatLFA are significantly different, and treeLFA chains always have larger predictive likelihood than flatLFA chains, though the difference is not very large. This result verifies our findings on the simulated data, that the prior for topics plays important role when the training data is small, which is equivalent to the inclusion of more rare diseases.

From a technical perspective, it is easy to update flatLFA to treeLFA, especially when Gibbs sampling is used to do the inference. With other algorithms (such as the Variational Bayes) used for inference, it may be difficult to incorporate the prior by running a Markov process on the hierarchical structure of disease codes. Fortunately, there are also other mathematical tools to construct a hierarchical prior for topics, as was introduced in other relevant works [100].

Overall, the use of the hierarchical prior for topics is beneficial, yet its effect is much smaller than that brought by changing the model configuration. From my perspective, results in this thesis suggest that if possible, an informative prior should always be incorporated. However, if this is in conflict with other adjustments to the model, and given that the informative prior has limited influence on the final results, it will also be reasonable to consider leaving out the use of the prior and try other ideas first.

7.2 Genetic analyses for topics

7.2.1 topic-GWAS on topic weights as continuous traits

In Chapter 4 and 5, on the two UKB datasets the GWAS results given by single code GWAS and topic-GWAS are compared. A noteworthy observation is that topic-GWAS found many fewer significant loci than single code GWAS. This result indicates that most genomic loci are specific to single diseases, instead of multi-morbidity clusters. Therefore, at present topic-GWAS should be a complement to the standard single disease GWAS. Moreover, this also suggests that there are limited numbers of loci play key role in pathways related to multi-morbidity clusters. In the future, these topic-associated loci's function should be investigated to further understand the mechanisms of multi-morbidity clusters.

The topic-associated loci that cannot be identified by single code GWAS proves that topic-GWAS brings additional power for discovery. These loci in general have smaller effect sizes than the loci associated with both topics and single codes, thus they are also harder to be found by single code GWAS. Different topics have different profiles for topic-associated loci. Notably, topics centered on metabolic diseases and immune diseases tend to have more loci uniquely found by topic-GWAS. Besides, dense topics and the empty also have a substantial numbers of loci only found by topic-GWAS. The gene set enrichment analysis shows that many of the loci associated with the empty topic and dense topics were found to be associated with behaviours which influence people's general health conditions (such as regular exercise and having religious belief) by other studies. It is possible that the empty topic and dense topics are correlated with other traits (such as various human behaviours or social-economic status), and thus also have relevant associated loci. At present, what the empty topic and dense topics truly are and what do they represent are still not clear, and requires further investigation.

During the progression of my research, I once expected to learn both topics with single active codes and topics with multiple active codes at the same time. The motivation was that some diseases may have distinctive pathological mechanisms, therefore it will not make sense to put them into multi-morbidity clusters with other unrelated diseases. With simulated data, it was proved that both LDA (relevant result was not shown in previous chapters) and treeLFA can learn topics with single active disease code. However, this relies on the proper setting of the hyperparameters. For LDA, the concentration parameter of the Dirichlet prior for topics controls the sparsity and smoothness of the inferred topics; for treeLFA, the transition probabilities of the Markov process on the hierarchical structure of diseases control the sparsity and structure of topics. Ideally, the model should be able to automatically figure out if single code topics should be learnt. But in reality, this has not been achieved, regardless the use of different hyperparameters. In the future, this may possibly be attained by other topic models with different configurations.

7.2.2 Genetic analyses other than topic-GWAS

In Chapter 4 I explored genetic analyses based on the inferred topics and topic weights other than the topic-GWAS. Specifically, an univariate multi-trait GWAS method is applied, which uses summary statistics given by the single disease GWAS for top active codes in a topic as input. There are two main reasons in choosing this method. Firstly, topic-GWAS is not powerful in finding variants which have heterogeneous effects on different active codes in a topic. For instance, some loci may be associated with only a small fraction of the active diseases in a topic, and with topic-GWAS these association signals will be diluted by other active diseases in the same topic. Meanwhile, there can also be a lot of loci which are associated with active diseases in different topics. However, these loci are not the focus of this study, since the primary aim of this study is to find major multi-morbidity clusters and understand their underlying mechanism. Secondly, with topics inferred from UKB, it is also desirable to find a way to generalize the inference result (topics) given by treeLFA to other types of existing datasets, such as cohort datasets constructed for the study of single diseases. Overall, there are a wide variety of multi-trait GWAS methods developed in recent years, as was discussed in Chapter 1. Different methods are based on different ideas, and are suited for different research purposes. Depending on the specific research goal, specific multi-trait GWAS method may have unique strength and advantage.

7.3 Further directions

7.3.1 Mechanisms of multi-morbidity clusters

In this thesis, a substantial number of variants are found to be associated with people's topic weights. However, to further dissect the mechanisms of these associations, more analyses still remain to be done.

There are multiple possible models for the association and causal structure between topic-associated loci and active diseases in the topic: First, a variant G might be directly associated with disease A in the topic, and disease A is causal to disease B in the same topic. As a result, variant G will be indirectly associated with disease B. Secondly, the association between variant G and topic T might only be driven by disease A in the topic, which may has very large prevalence in the training dataset. This means variant G is not associated with other active diseases in topic T. Thirdly, it is also possible that variant G is associated with an upstream pathway that is shared by both disease A and B. These are only a few but not all reasonable hypotheses. To figure out which one reveals the true biological mechanism, specifically designed statistical tests are required. Fortunately, some genetic tools such as Mendelian randomization and latent causal variable model [178] can be applied to provide us with more evidence.

For genetic variations in the HLA complex, it would be more ideal to use alleles instead of single variants for association study due to the complex LD structure in this region. In Chapter 6, it is validated that the state-of-art algorithm - HLA*LA gives accurate inference for most of the alleles for common HLA loci, especially for British people. For the top-436 UKB dataset, a large fraction of the topic-associated loci are located in the HLA complex (about 16 %). In the future, with the availability of NGS data for most individuals in UKB, the dissection of association signals in the HLA complex will undoubtedly be much improved.

In addition to genetic variants, it is also important to take into account the non-genetic factors. There are already tools designed to accomplish this type of analysis, such as "PHESANT" [179], which can be used to scan through the raw phenome data in UKB to find variables associated with any trait of interest. Last but not least, there can also be significant interaction between different variables. For instance, there may be interactions between different causal genetic and non-genetic variables, or interactions between different diseases that are active in the same topic.

All in all, in this thesis I mainly focus on developing and testing the new model, and figuring out what is the best way to apply it on the real world data. This is merely the beginning of the whole research pathway. To truly advance our understanding of any specific multi-morbidity cluster, various hypotheses need to be proposed and tested, followed by functional analyses as well as the final experimental validation.

7.3.2 Subgroups of common diseases

In Chapter 3, I explore defining different subgroups for patients who were diagnosed with the same disease using their comorbidity information. For the few exemplary diseases presented in Chapter 3, the subgroup-GWAS results indicate that different subgroups of a disease may have different profiles for the genetic associations, in terms of both P-values and effect sizes for the significant variants. However, there are also unanswered questions. For instance, there can be very strong multi-morbidity pattern for certain subgroups of a disease, such as angina. In these subgroups, in addition to the disease being studied (angina), most of the patients are also diagnosed with other active diseases in the same multi-morbidity cluster (such as hyperlipidemia, hypertension and chronic ischemic heart disease). As a result, it would be difficult to differentiate which disease is the major driving force for the association found by the subgroup-GWAS. To tackle this problem, analysis of the genetic correlation between different subgroups of different diseases might be helpful.

Another more straightforward way to define subgroups for a disease is to directly use people's topic assignments for this disease. According to the inference result given by treeLFA, most diseases are active in more than one topic, which means the same disease diagnosed for different people can be assigned with different topics. In the future, this method can also be tried. However, it would be reasonable to expect that it would give similar result as the method in current use.

For most of the diseases we studied, among all the subgroups there is usually a dominant one with quite strong co-morbidity pattern. This subgroup should receive specific attention in the future, since there might be unique and important biological pathways involved. In addition, for diseases with relatively low prevalence, it would be beneficial to carry out meta-analysis using multiple cohorts in the future, based on the preliminary results obtained from UKB. Most importantly, clinical profiles of different subgroups, including patient's demographic characteristics and clinical manifestations, lab tests, response to treatment and prognosis should also be thoroughly studied, such that a more complete picture can be depicted for different subgroups of a diseases.

7.3.3 More advanced topic models

It is important to emphasize that topics of diseases are inferred by treeLFA purely based on the co-occurrence data for diseases, without making use of other relevant information (such as the genotype data). In other words, unlike methods such as treeWAS and PCHAT (introduced in Chapter 1), treeLFA is not aimed at maximizing the power for discovery of associations or dissecting the causal structure. In spite of this, the NBMF framework for treeLFA is a basic framework that can be upgraded easily by plugging in other modules which have different functions, like what have been done for LDA in the past decade.

For example, one natural extension to the basic LDA is the supervised LDA [82], which is developed by introducing additional observable variables (labels of documents). These additional variables usually have dependencies with the topic weight variables. For instance, documents with different labels may have different Dirichlet priors for their topic weights. In the context of text mining, these additional variables can be documents' authors, ratings or years of publications. For the study of multi-morbidity, these labels can be common covariates such as age and sex, whose impacts on the genetic risks of common diseases have been investigated by multiple studies [180, 181]. Notably, the complex relationships between covariates and topics/topic weights can be modelled using various mathematical functions [182]. In addition to the supervised topic models, there are many other interesting extensions to the basic topic models, such as the correlated topic models [158], or its updated version, the "Pachinko Allocation" (PAM) [183]. They all aim to find the correlation structure between topics and/or individual words. In summary, these models can be employed to solve various specific topic modelling problems, as long as these problems make biological sense.

7.4 Conclusions

In this thesis, my main research goal was to study multi-morbidity clusters using topic models and the diagnosis data in biobanks. I developed a new topic model named "treeLFA" based on BNMF, and incorporated a prior for topics constructed by running a Markov process on the hierarchical structure of diseases specified by a medical ontology. Meanwhile, I came up with reliable solutions to multiple technical issues related to topic model, including learning of hyperparameters, model selection and combining results of multiple posterior samples from multiple Gibbs chains. With simulated data, I proved that treeLFA could accurately infer topics of binary variables, and it outperformed both LDA and flatLFA (treeLFA with non-informative prior for topics) when the training data was small and/or most individuals were mixture of multiple non-trivial topics. Besides, treeLFA was also robust to wrongly specified tree strudture of diseases and number of topics.

With empirical analysis of the HES data from the UK Biobank, topics of common diseases were inferred, and were found to be in alignment with our medical knowledge. Among the inferred topics, there were always an empty topic, a few dense topics with a large number of active codes, and many sparse topics which focused on codes from 1-3 ICD-10 chapters. Notably, the empty topic and the dense topics were found to have a strong genetic basis, and were better characterized by treeLFA, rather than LDA. With topic-GWAS, loci associated with individual's topic weights were identified. In general, the number of topic-associated loci was much smaller than the number of single disease associated loci, yet topic-GWAS still provided additional power for finding topic-associated loci with relatively small effect sizes and cannot be found by the standard GWAS. Through the training of treeLFA models with different number of topics, I confirmed that the inferred topics and their associations were stable and robust to the number of topics set for the model, which implied a convenient model selection strategy. Lastly, a few other exploratory analyses were done, including subtyping diseases according to patients' comorbidity patterns, constructing PRS for single diseases using the topic-GWAS results and performing genetic analyses other than the topic-GWAS. All these analyses gave promising preliminary results that can be further explored in the future.

In summary, topic modelling of the high-dimensional and sparse diagnosis data in biobanks provides a reliable and valuable analytic framework for the study of multi-morbidity. Combined with analyses making use of other types of omics data in biobanks, better prediction of risk and further understanding of the mechanisms of diseases are all attainable.

Appendices

A
CPASSOC: a univariate multi-trait GWAS method

CPASSOC is a general multi-traits GWAS approach to integrate the association evidence for multiple correlated traits of different types (continuous, binary, etc) from one or more studies [48]. It takes the summary statistics of single trait GWAS as input, and defines new summary statistics to measure the evidence of association between one variant and multiple traits, based on the assumption that the T-statistics of all traits jointly follow a multivariate normal distribution. The effects of the variant on traits can be either homogeneous or heterogeneous, and two different summary statistics ($\boldsymbol{S_{Hom}}$ and $\boldsymbol{S_{Het}}$) are defined accordingly.

The test statistics $\boldsymbol{S_{Hom}}$ defined below follows a $\boldsymbol{\chi^2}$ distribution with one degree of freedom:

$$\boldsymbol{S_{Hom}} = \frac{\boldsymbol{(e^T(RW)^{-1}T)(e^T(RW)^{-1}T)^T}}{\boldsymbol{e^T(WRW)^{-1}e}}$$

$\boldsymbol{T}$: a vector of T-statistics for all the correlated traits.
$\boldsymbol{R}$: the correlation matrix for all the traits.
$\boldsymbol{e = (1, ..., 1)}$
$\boldsymbol{W}$: a diagonal matrix of weights for individual test statistics. $\boldsymbol{W_k = \sqrt{n_k}}$. $\boldsymbol{W_k}$ is the $\boldsymbol{k^{th}}$ diagonal entry in matrix $\boldsymbol{W}$, and $\boldsymbol{n_k}$ is the size of the $\boldsymbol{k^{th}}$ cohort.

$\boldsymbol{S_{Hom}}$ is most powerful when variant's effect is homogeneous to all traits, which is also the reason it is named using $\boldsymbol{Hom}$ as the subscript. For heterogeneous effect, a more powerful statistics $\boldsymbol{S_{Het}}$ is defined as below:

$$\boldsymbol{S_\tau} = \frac{\boldsymbol{(e(R(\tau)W(\tau))^{-1}T(\tau))(e^{T(\tau)}(R(\tau)W(\tau))^{-1}T(\tau))^T}}{\boldsymbol{e^TW(\tau)^{-1}R(\tau)^{-1}W(\tau)^{-1}e}}$$

$$\boldsymbol{S_{Het} = \max_{\tau > 0}(S_\tau)}$$

$\boldsymbol{T(\tau)}$: a subvector of $\boldsymbol{T}$ in which all entries are larger than $\boldsymbol{\tau}$.
$\boldsymbol{R(\tau)}$: a submatrix of $\boldsymbol{R}$.
$\boldsymbol{W(\tau)}$: a submatrix of $\boldsymbol{W}$. It is noteworthy that $\boldsymbol{W_k = \sqrt{n} \times sign(T_k)}$, in which signed weights ensure that with opposite effects for different traits, the evidence of association will still be added up, instead of being cancelled with each other.

B

Top active codes in the 40 topics inferred from the top-436 UKB dataset.

Topic	Active codes	Topic name
1	I10,N17,J18,E87,N18,E11,I25,D64,E78,N39, B96,I48,K29,I50,I20,I95,A41,J44,L03,J22, D50,J90,K59,B95,E86,A09,K52,K92,E66,M19, I73,K76,K57,J96,K44,M79,J45,I51,F32,I12, F17,F10,L97,M54,H26,K21,K80,J98,K63,M25, M10,E10,N19,E83,K31,M13,I21,K62,L89,M81, K74,K22,E16,B37,E03,E14,N28,D69,G47,I67, I44,N40,I77,K20,K70,M17,I26,I84,I70,M06	inf-blo-met-cir
2	K29,K44,K21,I10,F32,J45,K57,N39,M19,E03, K62,K52,M54,K59,M79,I84,M13,K58,K80,F41, N81,E78,K30,M81,E66,K63,M47,K92,A09,D64, N95,K31,N92,M25,D50,M17,B96,F17,K20,J44, G43,E11,G56,M06,J22,M20,N32,M75,K22,D12	inf-dig-imm
3	M19,I10,M17,M25,M13,M54,K21,E78,M47,M23, K44,E66,M15,M79,J45,K29,M16,M75,K57,G56, F32,E11,M51,M65,M20,M06,E03,K30,I84,N39, I25,I20,D64,M81,H26,K62,K52,M48,K80	imm
4	C78,C18,C77,I10,K56,D64,K63,C79,K59,K57, K66,C20,J90,K91,N39,K62,J18,K52,E87,N17, D12,B96,K43,K29,E86,A41,C19,K92,K83,A09, C25,K44,E78,E11,K76,D50,C80,K80,I26,J98, K21,D37,I95,K31,C16,C15,D70	inf-neo-dig
5	C79,C78,D70,C77,J18,A41,J90,C34,I10,D64, J22,K59,C50,E87,N17,N39,A09,C80,J98,K52, I26,C85,B96,E86,E83,C83,D69,B37,I48,I95, C61,M54,E78,F17,M79	inf-neo-blo-res

6	I10,I25,E78,K29,I20,K44,K57,K21,E11,K62, K63,I84,D12,K22,D50,J45,D64,K20,K92,J44, K31,I48,M19,K25,N40,I21,M13,E66,F17,K30, K52	dig
7	I10,I63,E78,G81,I67,N39,I69,G40,F32,I48, J18,K59,E87,H53,B96,G45,J22,G93,F03,G30, E11,I64,I61,I25,I95,F17,I60,F05,I65,F41, G31,G20,N17,F10	men-ner
8	J44,J18,F17,F32,J45,I10,F10,J22,F41,J43, J98,J47,J96,E87,J90,E78,K29,M81,K21,M54, J84,C34	men-res
9	I10,I25,I48,I50,E78,I51,I20,I44,I34,I21, I08,I47,I35,J90,J18,I49,I42,E11,I45,I95, N17,J22,N18,J44,E87,E66	met-cir-res
10	I10,N32,N40,N39,E78,C67,N30,N35,N13,E11, N20,C61,K57,N28,D41,B96,K40,N18	gen
11	E11,I10,E78,E10,H26,H36,E14,I25,H25,E66, I20,E03,H35,G56,N18	met-ner-eye
12	M51,M54,G55,M47,M48,I10,M79,M43,M50,M25, M19,M16	ner-imm
13	D25,N83,N92,N80,N73,N94,N84,D27,N85,J45	gen
14	L03,I10,L40,B95,L02,M07,F32,M79,J45,E11, E66,F17,M17	ski-imm
15	M19,M20,M25,I10,M75,M16,M13,G56,J45,M06, M65	imm
16	N95,N84,N85,I10,D25,N81,N92,E03,C50	gen
17	H26,H25,I10,H35,H40,H33,H52,M13,E78,H43	eye
18	I25,I10,E78,I20,I21,E11,I48	cir
19	K52,K51,K62,K63,K50,K57,I84,A09,K59,D12, K56,K58	dig
20	KN40, N32, I10, C61, K40, N41, N39, N42, N35	dig
21	C44, D22, L98, L57, I10, L82, C43, D23, L90	dig
22	K80,I10,K81,K82,K85,K66,K29,K83	dig
23	N39,N81,N32,I10,N30	gen
24	C44,D22,L57,L98,L82,I10,C43,D23,L90	neo-ski
25	N40,N32,I10,C61,K40,N39,N41,N42,N35	gen
26	I10,E78,I48,E11,M16	met
27	N20,I10,N23,N13,D35,E21	met-gen
28	J34,J33,J32,J45,J31	res
29	H72,H91,H90,H65,I10,H66,H61	ear
30	M23,M17,I10,M25	imm
31	C50,C77,D05,I10	neo-gen
32	I84,K62,K57,K60,K92	dig
33	N92,D25,N84,N93,N94	gen
34	I80,M79,I83,I26	blo-cir
35	K02,K08,K01,K04,K05	dig
36	N87,N84,N95,N92	inf-gen
37	H02,H04,H00	eye
38	H26,H33,H25	eye
39	K40,D17	dig
40		empty

Table B.1: Codes with an un-normalized probability of at least 0.2 in each topic inferred by treeLFA from the top-436 UKB dataset are shown (active codes are not defined using normalized probability to reduce the total number of top active codes in topics). Top active codes in the same topic are shown in an descending order of their probabilities. Topics have the same order and names as those in 5.4.

References

[1] *MS Windows NT Kernel Description.* https://acmedsci.ac.uk/file-download/82222577.

[2] Noe Garin et al. "Global multimorbidity patterns: a cross-sectional, population-based, multi-country study". In: *Journals of Gerontology Series A: Biomedical Sciences and Medical Sciences* 71.2 (2016), pp. 205–214.

[3] Martin Fortin et al. "A systematic review of prevalence studies on multimorbidity: toward a more uniform methodology". In: *The Annals of Family Medicine* 10.2 (2012), pp. 142–151.

[4] Concepció Violan et al. "Prevalence, determinants and patterns of multimorbidity in primary care: a systematic review of observational studies". In: *PloS one* 9.7 (2014), e102149.

[5] Aine Ryan et al. "Multimorbidity and functional decline in community-dwelling adults: a systematic review". In: *Health and quality of life outcomes* 13.1 (2015), pp. 1–13.

[6] Bhautesh Dinesh Jani et al. "Relationship between multimorbidity, demographic factors and mortality: findings from the UK Biobank cohort". In: *BMC medicine* 17.1 (2019), pp. 1–13.

[7] Frances S Mair and Carl R May. *Thinking about the burden of treatment.* 2014.

[8] Thomas Lehnert et al. "Health care utilization and costs of elderly persons with multiple chronic conditions". In: *Medical Care Research and Review* 68.4 (2011), pp. 387–420.

[9] Alexandra Prados-Torres et al. "Multimorbidity patterns: a systematic review". In: *Journal of clinical epidemiology* 67.3 (2014), pp. 254–266.

[10] Concepció Violan et al. "Prevalence, determinants and patterns of multimorbidity in primary care: a systematic review of observational studies". In: *PloS one* 9.7 (2014), e102149.

[11] Marina Guisado-Clavero et al. "Multimorbidity patterns in the elderly: a prospective cohort study with cluster analysis". In: *BMC geriatrics* 18.1 (2018), pp. 1–11.

[12] César A Hidalgo et al. "A dynamic network approach for the study of human phenotypes". In: *PLoS computational biology* 5.4 (2009), e1000353.

[13] Mengfei Guo et al. "Analysis of disease comorbidity patterns in a large-scale China population". In: *BMC medical genomics* 12.12 (2019), pp. 1–10.

[14] A Amell et al. "Disease networks identify specific conditions and pleiotropy influencing multimorbidity in the general population". In: *Scientific reports* 8.1 (2018), pp. 1–16.

[15] Guiying Dong et al. “A global overview of genetically interpretable comorbidities among common diseases in UK Biobank”. In: *medRxiv* (2021).

[16] Michael C O’Donovan and Michael J Owen. “The implications of the shared genetics of psychiatric disorders”. In: *Nature medicine* 22.11 (2016), pp. 1214–1219.

[17] Abdullah S Al-Goblan, Mohammed A Al-Alfi, and Muhammad Z Khan. “Mechanism linking diabetes mellitus and obesity”. In: *Diabetes, metabolic syndrome and obesity: targets and therapy* 7 (2014), p. 587.

[18] Wouter Van Rheenen et al. “Genetic correlations of polygenic disease traits: from theory to practice”. In: *Nature Reviews Genetics* 20.10 (2019), pp. 567–581.

[19] Briana Mezuk et al. “Depression and type 2 diabetes over the lifespan: a meta-analysis”. In: *Diabetes care* 31.12 (2008), pp. 2383–2390.

[20] Naomi R Wray et al. “Genome-wide association analyses identify 44 risk variants and refine the genetic architecture of major depression”. In: *Nature genetics* 50.5 (2018), pp. 668–681.

[21] Saskia P Hagenaars et al. “Genetic comorbidity between major depression and cardio-metabolic traits, stratified by age at onset of major depression”. In: *American Journal of Medical Genetics Part B: Neuropsychiatric Genetics* 183.6 (2020), pp. 309–330.

[22] Steven C Bagley et al. “Constraints on biological mechanism from disease comorbidity using electronic medical records and database of genetic variants”. In: *PLoS computational biology* 12.4 (2016), e1004885.

[23] Elena Dıaz-Santiago et al. “Phenotype-genotype comorbidity analysis of patients with rare disorders provides insight into their pathological and molecular bases”. In: *PLoS genetics* 16.10 (2020), e1009054.

[24] Emil Uffelmann et al. “Genome-wide association studies”. In: *Nature Reviews Methods Primers* 1.1 (2021), pp. 1–21.

[25] Brooke N Wolford, Cristen J Willer, and Ida Surakka. “Electronic health records: the next wave of complex disease genetics”. In: *Human molecular genetics* 27.R1 (2018), R14–R21.

[26] Jodell E Linder et al. “The Role of Electronic Health Records in Advancing Genomic Medicine”. In: *Annual Review of Genomics and Human Genetics* 22 (2021).

[27] Clare Bycroft et al. “The UK Biobank resource with deep phenotyping and genomic data”. In: *Nature* 562.7726 (2018), pp. 203–209.

[28] Zhengming Chen et al. “China Kadoorie Biobank of 0.5 million people: survey methods, baseline characteristics and long-term follow-up”. In: *International journal of epidemiology* 40.6 (2011), pp. 1652–1666.

[29] Masahiro Kanai et al. “Genetic analysis of quantitative traits in the Japanese population links cell types to complex human diseases”. In: *Nature genetics* 50.3 (2018), pp. 390–400.

[30] Masahiro Kanai et al. “Genetic analysis of quantitative traits in the Japanese population links cell types to complex human diseases”. In: *Nature genetics* 50.3 (2018), pp. 390–400.

[31] William S Bush, Matthew T Oetjens, and Dana C Crawford. “Unravelling the human genome–phenome relationship using phenome-wide association studies”. In: *Nature Reviews Genetics* 17.3 (2016), pp. 129–145.

[32] Lijuan Wang et al. “Methodology in phenome-wide association studies: a systematic review”. In: *Journal of Medical Genetics* 58.11 (2021), pp. 720–728.

[33] Joshua C Denny et al. “PheWAS: demonstrating the feasibility of a phenome-wide scan to discover gene–disease associations”. In: *Bioinformatics* 26.9 (2010), pp. 1205–1210.

[34] Joshua C Denny et al. “Systematic comparison of phenome-wide association study of electronic medical record data and genome-wide association study data”. In: *Nature biotechnology* 31.12 (2013), pp. 1102–1111.

[35] Sarah A Pendergrass et al. “Phenome-wide association study (PheWAS) for detection of pleiotropy within the Population Architecture using Genomics and Epidemiology (PAGE) Network”. In: *PLoS genetics* 9.1 (2013), e1003087.

[36] Robert M Cronin et al. “Phenome-wide association studies demonstrating pleiotropy of genetic variants within FTO with and without adjustment for body mass index”. In: *Frontiers in genetics* 5 (2014), p. 250.

[37] Jixia Liu et al. “Phenome-wide association study maps new diseases to the human major histocompatibility complex region”. In: *Journal of medical genetics* 53.10 (2016), pp. 681–689.

[38] Jason H Karnes et al. “Phenome-wide scanning identifies multiple diseases and disease severity phenotypes associated with HLA variants”. In: *Science Translational Medicine* 9.389 (2017), eaai8708.

[39] Kyoko Watanabe et al. “A global overview of pleiotropy and genetic architecture in complex traits”. In: *Nature genetics* 51.9 (2019), pp. 1339–1348.

[40] Joseph K Pickrell et al. “Detection and interpretation of shared genetic influences on 42 human traits”. In: *Nature genetics* 48.7 (2016), pp. 709–717.

[41] Nadia Solovieff et al. “Pleiotropy in complex traits: challenges and strategies”. In: *Nature Reviews Genetics* 14.7 (2013), pp. 483–495.

[42] Peter M Visscher et al. “10 years of GWAS discovery: biology, function, and translation”. In: *The American Journal of Human Genetics* 101.1 (2017), pp. 5–22.

[43] Hanna Julienne et al. “Multitrait GWAS to connect disease variants and biological mechanisms”. In: *PLoS genetics* 17.8 (2021), e1009713.

[44] Miriam S Udler et al. “Type 2 diabetes genetic loci informed by multi-trait associations point to disease mechanisms and subtypes: a soft clustering analysis”. In: *PLoS medicine* 15.9 (2018), e1002654.

[45] Sophie Hackinger and Eleftheria Zeggini. “Statistical methods to detect pleiotropy in human complex traits”. In: *Open biology* 7.11 (2017), p. 170125.

[46] Chris Cotsapas et al. “Pervasive sharing of genetic effects in autoimmune disease”. In: *PLoS genetics* 7.8 (2011), e1002254.

[47] Samsiddhi Bhattacharjee et al. “A subset-based approach improves power and interpretation for the combined analysis of genetic association studies of heterogeneous traits”. In: *The American Journal of Human Genetics* 90.5 (2012), pp. 821–835.

[48] Xiaofeng Zhu et al. “Meta-analysis of correlated traits via summary statistics from GWASs with an application in hypertension”. In: *The American Journal of Human Genetics* 96.1 (2015), pp. 21–36.

[49] Sophie Van der Sluis, Danielle Posthuma, and Conor V Dolan. “TATES: efficient multivariate genotype-phenotype analysis for genome-wide association studies”. In: *PLoS genetics* 9.1 (2013), e1003235.

[50] Patrick Turley et al. “Multi-trait analysis of genome-wide association summary statistics using MTAG”. In: *Nature genetics* 50.2 (2018), pp. 229–237.

[51] Holly Trochet et al. “Bayesian meta-analysis across genome-wide association studies of diverse phenotypes”. In: *Genetic epidemiology* 43.5 (2019), pp. 532–547.

[52] Arunabha Majumdar et al. “An efficient Bayesian meta-analysis approach for studying cross-phenotype genetic associations”. In: *PLoS genetics* 14.2 (2018), e1007139.

[53] Arthur Korte et al. “A mixed-model approach for genome-wide association studies of correlated traits in structured populations”. In: *Nature genetics* 44.9 (2012), pp. 1066–1071.

[54] Xiang Zhou and Matthew Stephens. “Efficient multivariate linear mixed model algorithms for genome-wide association studies”. In: *Nature methods* 11.4 (2014), pp. 407–409.

[55] Nicholas A Furlotte and Eleazar Eskin. “Efficient multiple-trait association and estimation of genetic correlation using the matrix-variate linear mixed model”. In: *Genetics* 200.1 (2015), pp. 59–68.

[56] Jong Wha J Joo et al. “Efficient and accurate multiple-phenotype regression method for high dimensional data considering population structure”. In: *Genetics* 204.4 (2016), pp. 1379–1390.

[57] Jianfeng Liu et al. “Bivariate association analyses for the mixture of continuous and binary traits with the use of extended generalized estimating equations”. In: *Genetic Epidemiology: The Official Publication of the International Genetic Epidemiology Society* 33.3 (2009), pp. 217–227.

[58] James A Hanley et al. “Statistical analysis of correlated data using generalized estimating equations: an orientation”. In: *American journal of epidemiology* 157.4 (2003), pp. 364–375.

[59] Jonathan Marchini et al. “A new multipoint method for genome-wide association studies by imputation of genotypes”. In: *Nature genetics* 39.7 (2007), pp. 906–913.

[60] Stephen W Hartley and Paola Sebastiani. “PleioGRiP: genetic risk prediction with pleiotropy”. In: *Bioinformatics* 29.8 (2013), pp. 1086–1088.

[61] Matthew Stephens. “A unified framework for association analysis with multiple related phenotypes”. In: *PloS one* 8.7 (2013), e65245.

[62] Paul F O’Reilly et al. “MultiPhen: joint model of multiple phenotypes can increase discovery in GWAS”. In: *PloS one* 7.5 (2012), e34861.

[63] Qiong Yang and Yuanjia Wang. “Methods for analyzing multivariate phenotypes in genetic association studies”. In: *Journal of probability and statistics* 2012 (2012).

[64] Luke Jostins and Gilean McVean. “Trinculo: Bayesian and frequentist multinomial logistic regression for genome-wide association studies of multi-category phenotypes”. In: *Bioinformatics* 32.12 (2016), pp. 1898–1900.

[65] Daniel J Schaid et al. “Statistical methods for testing genetic pleiotropy”. In: *Genetics* 204.2 (2016), pp. 483–497.

[66] Tessel E Galesloot et al. “A comparison of multivariate genome-wide association methods”. In: *PloS one* 9.4 (2014), e95923.

[67] Heather F Porter and Paul F O’Reilly. “Multivariate simulation framework reveals performance of multi-trait GWAS methods”. In: *Scientific reports* 7.1 (2017), pp. 1–12.

[68] Christy L Avery et al. “A phenomics-based strategy identifies loci on APOC1, BRAP, and PLCG1 associated with metabolic syndrome phenotype domains”. In: *PLoS genetics* 7.10 (2011), e1002322.

[69] David Karasik et al. “Genome-wide association of an integrated osteoporosis-related phenotype: Is there evidence for pleiotropic genes?” In: *Journal of Bone and Mineral Research* 27.2 (2012), pp. 319–330.

[70] L-N He et al. “Genomewide linkage scan for combined obesity phenotypes using principal component analysis”. In: *Annals of human genetics* 72.3 (2008), pp. 319–326.

[71] Hugues Aschard et al. “Maximizing the power of principal-component analysis of correlated phenotypes in genome-wide association studies”. In: *The American Journal of Human Genetics* 94.5 (2014), pp. 662–676.

[72] Lambertus Klei et al. “Pleiotropy and principal components of heritability combine to increase power for association analysis”. In: *Genetic Epidemiology: The Official Publication of the International Genetic Epidemiology Society* 32.1 (2008), pp. 9–19.

[73] Manuel AR Ferreira and Shaun M Purcell. “A multivariate test of association”. In: *Bioinformatics* 25.1 (2009), pp. 132–133.

[74] Clara S Tang and Manuel AR Ferreira. “A gene-based test of association using canonical correlation analysis”. In: *Bioinformatics* 28.6 (2012), pp. 845–850.

[75] Jose A Seoane et al. “Canonical correlation analysis for gene-based pleiotropy discovery”. In: *PLoS computational biology* 10.10 (2014), e1003876.

[76] Scott Deerwester et al. “Indexing by latent semantic analysis”. In: *Journal of the American society for information science* 41.6 (1990), pp. 391–407.

[77] Daniel D Lee and H Sebastian Seung. “Learning the parts of objects by non-negative matrix factorization”. In: *Nature* 401.6755 (1999), pp. 788–791.

[78] Thomas Hofmann. “Unsupervised learning by probabilistic latent semantic analysis”. In: *Machine learning* 42.1 (2001), pp. 177–196.

[79] David M Blei, Andrew Y Ng, and Michael I Jordan. “Latent dirichlet allocation”. In: *the Journal of machine Learning research* 3 (2003), pp. 993–1022.

[80] Jonathan K Pritchard, Matthew Stephens, and Peter Donnelly. “Inference of population structure using multilocus genotype data”. In: *Genetics* 155.2 (2000), pp. 945–959.

[81] David M Blei and John D Lafferty. “A correlated topic model of science”. In: *The annals of applied statistics* 1.1 (2007), pp. 17–35.

[82] David M Blei and Jon D McAuliffe. “Supervised topic models”. In: *arXiv preprint arXiv:1003.0783* (2010).

[83] Alberto Lumbreras, Louis Filstroff, and Cédric Févotte. “Bayesian mean-parameterized nonnegative binary matrix factorization”. In: *Data Mining and Knowledge Discovery* 34.6 (2020), pp. 1898–1935.

[84] Lin Liu et al. “An overview of topic modeling and its current applications in bioinformatics”. In: *SpringerPlus* 5.1 (2016), pp. 1–22.

[85] Mikyung Lee et al. “Of text and gene–using text mining methods to uncover hidden knowledge in toxicogenomics”. In: *BMC systems biology* 8.1 (2014), pp. 1–11.

[86] Manuele Bicego et al. “Investigating topic models’ capabilities in expression microarray data classification”. In: *IEEE/ACM transactions on computational biology and bioinformatics* 9.6 (2012), pp. 1831–1836.

[87] Yue Li et al. “Inferring multimodal latent topics from electronic health records”. In: *Nature communications* 11.1 (2020), pp. 1–17.

[88] Thomas H McCoy et al. “Efficient genome-wide association in biobanks using topic modeling identifies multiple novel disease loci”. In: *Molecular Medicine* 23.1 (2017), pp. 285–294.

[89] TH McCoy et al. “Polygenic loading for major depression is associated with specific medical comorbidity”. In: *Translational psychiatry* 7.9 (2017), e1238–e1238.

[90] Thomas H McCoy, Amelia M Pellegrini, and Roy H Perlis. “Using phenome-wide association to investigate the function of a schizophrenia risk locus at SLC39A8”. In: *Translational psychiatry* 9.1 (2019), pp. 1–6.

[91] Juan Zhao et al. “Using topic modeling via non-negative matrix factorization to identify relationships between genetic variants and disease phenotypes: A case study of Lipoprotein (a)(LPA)”. In: *PloS one* 14.2 (2019), e0212112.

[92] Hanna M Wallach et al. “Evaluation methods for topic models”. In: *Proceedings of the 26th annual international conference on machine learning*. 2009, pp. 1105–1112.

[93] Mariflor Vega-Carrasco et al. “Modelling Grocery Retail Topic Distributions: Evaluation, Interpretability and Stability”. In: *arXiv preprint arXiv:2005.10125* (2020).

[94] Linzi Xing, Michael J Paul, and Giuseppe Carenini. “Evaluating Topic Quality with Posterior Variability”. In: *arXiv preprint arXiv:1909.03524* (2019).

[95] Linzi Xing and Michael J Paul. “Diagnosing and improving topic models by analyzing posterior variability”. In: *Thirty-Second AAAI Conference on Artificial Intelligence*. 2018.

[96] Sergei Koltcov, Olessia Koltsova, and Sergey Nikolenko. “Latent dirichlet allocation: stability and applications to studies of user-generated content”. In: *Proceedings of the 2014 ACM conference on Web science*. 2014, pp. 161–165.

[97] Juan Cao et al. “A density-based method for adaptive LDA model selection”. In: *Neurocomputing* 72.7-9 (2009), pp. 1775–1781.

[98] Yee Whye Teh. *Dirichlet Process*. 2010.

[99] David Andrzejewski, Xiaojin Zhu, and Mark Craven. “Incorporating domain knowledge into topic modeling via Dirichlet forest priors”. In: *Proceedings of the 26th annual international conference on machine learning*. 2009, pp. 25–32.

[100] Yuening Hu et al. “Interactive topic modeling”. In: *Machine learning* 95.3 (2014), pp. 423–469.

[101] World Health Organization et al. *ICD-10: international statistical classification of diseases and related health problems: tenth revision*. World Health Organization, 2004.

[102] Edward Choi et al. “GRAM: graph-based attention model for healthcare representation learning”. In: *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*. 2017, pp. 787–795.

[103] Cheng Li et al. “Hierarchical Bayesian nonparametric models for knowledge discovery from electronic medical records”. In: *Knowledge-Based Systems* 99 (2016), pp. 168–182.

[104] Adrian Cortes et al. “Bayesian analysis of genetic association across tree-structured routine healthcare data in the UK Biobank”. In: *Nature genetics* 49.9 (2017), pp. 1311–1318.

[105] Thomas Minka. *Estimating a Dirichlet distribution*. 2000.

[106] Hanna M Wallach, David M Mimno, and Andrew McCallum. “Rethinking LDA: Why priors matter”. In: *Advances in neural information processing systems*. 2009, pp. 1973–1981.

[107] Hanna Megan Wallach. “Structured topic models for language”. PhD thesis. University of Cambridge Cambridge, UK, 2008.

[108] *Histocompatibility complex*. https://ghr.nlm.nih.gov/primer/genefamily/hla.

[109] Amy E Kennedy, Umut Ozbek, and Mehmet T Dorak. “What has GWAS done for HLA and disease associations?” In: *International journal of immunogenetics* 44.5 (2017), pp. 195–211.

[110] Calliope A Dendrou et al. “HLA variation and disease”. In: *Nature Reviews Immunology* 18.5 (2018), p. 325.

[111] Alexander T Dilthey. “State-of-the-art genome inference in the human MHC”. In: *The International Journal of Biochemistry & Cell Biology* 131 (2021), p. 105882.

[112] TH Lam et al. “Population-specific recombination sites within the human MHC region”. In: *Heredity* 111.2 (2013), pp. 131–138.

[113] Idan Alter et al. “HLA class I haplotype diversity is consistent with selection for frequent existing haplotypes”. In: *PLoS computational biology* 13.8 (2017), e1005693.

[114] James Robinson et al. “Ipd-imgt/hla database”. In: *Nucleic acids research* 48.D1 (2020), pp. D948–D955.

[115] Jason H Karnes et al. “Phenome-wide scanning identifies multiple diseases and disease severity phenotypes associated with HLA variants”. In: *Science translational medicine* 9.389 (2017), eaai8708.

[116] Philip Deitiker and M Zouhair Atassi. “MHC genes linked to autoimmune disease”. In: *Critical Reviews™ in Immunology* 35.3 (2015).

[117] Jenefer M Blackwell, Sarra E Jamieson, and David Burgner. “HLA and infectious diseases”. In: *Clinical microbiology reviews* 22.2 (2009), pp. 370–385.

[118] Aleksandar Senev et al. “Clinical importance of extended second field high-resolution HLA genotyping for kidney transplantation”. In: *American Journal of Transplantation* 20.12 (2020), pp. 3367–3378.

[119] Patricia T Illing, Anthony W Purcell, and James McCluskey. “The role of HLA genes in pharmacogenomics: unravelling HLA associated adverse drug reactions”. In: *Immunogenetics* 69.8 (2017), pp. 617–630.

[120] Francesco Sabbatino et al. “Role of human leukocyte antigen system as a predictive biomarker for checkpoint-based immunotherapy in cancer patients”. In: *International Journal of Molecular Sciences* 21.19 (2020), p. 7295.

[121] YM Mosaad. “Clinical role of human leukocyte antigen in health and disease”. In: *Scandinavian journal of immunology* 82.4 (2015), pp. 283–306.

[122] Carolyn Katovich Hurley. “Naming HLA diversity: a review of HLA nomenclature”. In: *Human immunology* 82.7 (2021), pp. 457–465.

[123] Lee Ann Baxter-Lowe. “The changing landscape of HLA typing: Understanding how and when HLA typing data can be used with confidence from bench to bedside”. In: *Human Immunology* (2021).

[124] Alexander T Dilthey et al. “HLA* IMP—an integrated framework for imputing classical HLA alleles from SNP genotypes”. In: *Bioinformatics* 27.7 (2011), pp. 968–972.

[125] Alexander Dilthey et al. “Multi-population classical HLA type imputation”. In: *PLoS computational biology* 9.2 (2013), e1002877.

[126] Xiuwen Zheng et al. “HIBAG—HLA genotype imputation with attribute bagging”. In: *The pharmacogenomics journal* 14.2 (2014), pp. 192–200.

[127] Allan Motyer et al. “Practical use of methods for imputation of HLA alleles from SNP genotype data”. In: *BioRxiv* (2016), p. 091009.

[128] Tatsuhiko Naito et al. “A deep learning method for HLA imputation and trans-ethnic MHC fine-mapping of type 1 diabetes”. In: *Nature communications* 12.1 (2021), pp. 1–14.

[129] Alexander Dilthey et al. “Improved genome inference in the MHC using a population reference graph”. In: *Nature genetics* 47.6 (2015), pp. 682–688.

[130] Alexander T Dilthey et al. “High-accuracy HLA type inference from whole-genome sequencing data using population reference graphs”. In: *PLoS computational biology* 12.10 (2016), e1005151.

[131] Alexander T Dilthey et al. "HLA* LA—HLA typing from linearly projected graph alignments". In: *Bioinformatics* (2019).

[132] Daehwan Kim et al. "Graph-based genome alignment and genotyping with HISAT2 and HISAT-genotype". In: *Nature biotechnology* 37.8 (2019), pp. 907–915.

[133] Chao Xie et al. "Fast and accurate HLA typing from short-read next-generation sequence data with xHLA". In: *Proceedings of the National Academy of Sciences* 114.30 (2017), pp. 8059–8064.

[134] Chang Liu. "A long road/read to rapid high-resolution HLA typing: The nanopore perspective". In: *Human immunology* 82.7 (2021), pp. 488–495.

[135] Jieming Chen et al. "In silico tools for accurate HLA and KIR inference from clinical sequencing data empower immunogenetics on individual-patient and population scales". In: *Briefings in bioinformatics* 22.3 (2021), bbaa223.

[136] Thomas L Griffiths and Mark Steyvers. "Finding scientific topics". In: *Proceedings of the National academy of Sciences* 101.suppl 1 (2004), pp. 5228–5235.

[137] Quentin F Gronau et al. "A tutorial on bridge sampling". In: *Journal of mathematical psychology* 81 (2017), pp. 80–97.

[138] Andrew Kachites McCallum. "Mallet: A machine learning for language toolkit". In: *http://mallet. cs. umass. edu* (2002).

[139] Hanna Megan Wallach. "Structured topic models for language". PhD thesis. University of Cambridge Cambridge, UK, 2008.

[140] Xinyu Que et al. "Scalable community detection with the louvain algorithm". In: *2015 IEEE International Parallel and Distributed Processing Symposium*. IEEE. 2015, pp. 28–37.

[141] *MS Windows NT Kernel Description*. https://icd.who.int/browse10/2016/en. Accessed: 2010-09-30.

[142] Robert H Eckel, Scott M Grundy, and Paul Z Zimmet. "The metabolic syndrome". In: *The lancet* 365.9468 (2005), pp. 1415–1428.

[143] Marc-Andre Cornier et al. "The metabolic syndrome". In: *Endocrine reviews* 29.7 (2008), pp. 777–822.

[144] Annegret Kuhn and Aysche Landmann. "The classification and diagnosis of cutaneous lupus erythematosus". In: *Journal of autoimmunity* 48 (2014), pp. 14–19.

[145] Akihisa Imagawa et al. "A novel subtype of type 1 diabetes mellitus characterized by a rapid onset and an absence of diabetes-related antibodies". In: *New England journal of medicine* 342.5 (2000), pp. 301–307.

[146] Tin Nguyen et al. "A novel approach for data integration and disease subtyping". In: *Genome research* 27.12 (2017), pp. 2025–2039.

[147] Lisa Bastarache. "Using Phecodes for research with the electronic health record: from PheWAS to PheRS". In: *Annual Review of Biomedical Data Science* 4 (2021), pp. 1–19.

[148] Brendan K Bulik-Sullivan et al. "LD Score regression distinguishes confounding from polygenicity in genome-wide association studies". In: *Nature genetics* 47.3 (2015), pp. 291–295.

[149] Randall J Pruim et al. “LocusZoom: regional visualization of genome-wide association scan results”. In: *Bioinformatics* 26.18 (2010), pp. 2336–2337.

[150] *The GWAS catalog*. https://www.ebi.ac.uk/gwas/.

[151] Danielle Welter et al. “The NHGRI GWAS Catalog, a curated resource of SNP-trait associations”. In: *Nucleic acids research* 42.D1 (2014), pp. D1001–D1006.

[152] Alan P Boyle et al. “Annotation of functional variation in personal genomes using RegulomeDB”. In: *Genome research* 22.9 (2012), pp. 1790–1797.

[153] Shengcheng Dong and Alan P Boyle. “Predicting functional variants in enhancer and promoter elements using RegulomeDB”. In: *Human mutation* 40.9 (2019), pp. 1292–1298.

[154] *chromhmm15*. https://egg2.wustl.edu/roadmap/web_portal/chr_state_learning.html.

[155] Kyoko Watanabe et al. “Functional mapping and annotation of genetic associations with FUMA”. In: *Nature communications* 8.1 (2017), pp. 1–11.

[156] Shing Wan Choi, Timothy Shin-Heng Mak, and Paul F O’Reilly. “Tutorial: a guide to performing polygenic risk score analyses”. In: *Nature Protocols* 15.9 (2020), pp. 2759–2772.

[157] Jack Euesden, Cathryn M Lewis, and Paul F O’Reilly. “PRSice: polygenic risk score software”. In: *Bioinformatics* 31.9 (2015), pp. 1466–1468.

[158] David Blei and John Lafferty. “Correlated topic models”. In: *Advances in neural information processing systems* 18 (2006), p. 147.

[159] Ana Márquez et al. “Meta-analysis of Immunochip data of four autoimmune diseases reveals novel single-disease and cross-phenotype associations”. In: *Genome medicine* 10.1 (2018), pp. 1–13.

[160] Xiaocan Jia et al. “Identification of 67 pleiotropic genes associated with seven autoimmune/autoinflammatory diseases using multivariate statistical analysis”. In: *Frontiers in Immunology* 11 (2020), p. 30.

[161] David González-Serna et al. “A cross-disease meta-GWAS identifies four new susceptibility loci shared between systemic sclerosis and Crohn’s disease”. In: *Scientific reports* 10.1 (2020), pp. 1–11.

[162] Fatou K Ndiaye et al. “The expression of genes in top obesity-associated loci is enriched in insula and substantia nigra brain regions involved in addiction and reward”. In: *International Journal of Obesity* 44.2 (2020), pp. 539–543.

[163] Pascal N Timshel, Jonatan J Thompson, and Tune H Pers. “Genetic mapping of etiologic brain cell types for obesity”. In: *Elife* 9 (2020), e55851.

[164] Zhenyao Ye et al. “Meta-analysis of transcriptome-wide association studies across 13 brain tissues identified novel clusters of genes associated with nicotine addiction”. In: *Genes* 13.1 (2021), p. 37.

[165] Cristopher V Van Hout et al. “Whole exome sequencing and characterization of coding variation in 49,960 individuals in the UK Biobank”. In: *bioRxiv* (2019), p. 572347.

[166] Clare Bycroft et al. “Genome-wide genetic data on˜ 500,000 UK Biobank participants”. In: *BioRxiv* (2017), p. 166298.

[167] Xiangqun Zheng-Bradley et al. “Alignment of 1000 Genomes Project reads to reference assembly GRCh38”. In: *GigaScience* 6.7 (2017), gix038.

[168] Pierre-Antoine Gourraud et al. “HLA diversity in the 1000 genomes dataset”. In: *PloS one* 9.7 (2014), e97282.

[169] Shuji Kawaguchi et al. “HLA-HD: An accurate HLA typing algorithm for next-generation sequencing data”. In: *Human mutation* 38.7 (2017), pp. 788–797.

[170] *HLA*LA*. https://github.com/DiltheyLab/HLA-LA.

[171] Joshua D Backman et al. “Exome sequencing and analysis of 454,787 UK Biobank participants”. In: *Nature* 599.7886 (2021), pp. 628–634.

[172] Ruoyu Tian et al. “Whole exome sequencing in the UK Biobank reveals risk gene SLC2A1 and biological insights for major depressive disorder”. In: *medRxiv* (2021).

[173] Aoife McMahon et al. “Sequencing-based genome-wide association studies reporting standards”. In: *Cell genomics* 1.1 (2021), p. 100005.

[174] Jocelyn Kaiser. “200,000 whole genomes made available for biomedical studies”. In: *Science* 374.6571 (2021), pp. 1036–1036.

[175] Bjarni V Halldorsson et al. “The sequences of 150,119 genomes in the UK biobank”. In: *bioRxiv* (2021).

[176] Arthur Asuncion et al. “On smoothing and inference for topic models”. In: *arXiv preprint arXiv:1205.2662* (2012).

[177] Tomáš Šingliar and Miloš Hauskrecht. “Noisy-or component analysis and its application to link analysis”. In: *The Journal of Machine Learning Research* 7 (2006), pp. 2189–2213.

[178] Luke J O’Connor and Alkes L Price. “Distinguishing genetic correlation from causation across 52 diseases and complex traits”. In: *Nature genetics* 50.12 (2018), pp. 1728–1734.

[179] Louise AC Millard et al. “Software Application Profile: PHESANT: a tool for performing automated phenome scans in UK Biobank”. In: *International journal of epidemiology* (2017).

[180] Xilin Jiang, Chris Holmes, and Gil McVean. “The impact of age on genetic risk for common diseases”. In: *PLoS genetics* 17.8 (2021), e1009723.

[181] Elena Bernabeu et al. “Sex differences in genetic architecture in the UK Biobank”. In: *Nature genetics* 53.9 (2021), pp. 1283–1289.

[182] David M Mimno and Andrew McCallum. “Topic models conditioned on arbitrary features with Dirichlet-multinomial regression.” In: *UAI*. Vol. 24. Citeseer. 2008, pp. 411–418.

[183] Wei Li and Andrew McCallum. “Pachinko allocation: DAG-structured mixture models of topic correlations”. In: *Proceedings of the 23rd international conference on Machine learning*. 2006, pp. 577–584.