

Title: *Guidelines for using sigQC for systematic evaluation of gene signatures*

Authors: Andrew Dhawan¹, Alessandro Barberis¹, Wei-Chen Cheng¹, Enric Domingo¹, Catharine West², Tim Maughan¹, Jacob G. Scott³, Adrian L. Harris¹ and Francesca M. Buffa^{1,*}

¹ Department of Oncology, University of Oxford, Oxford, United Kingdom

² University of Manchester, Manchester, United Kingdom

³ Translational Hematology and Oncology Research, Cleveland Clinic, Cleveland, Ohio, USA

* Corresponding author email: francesca.buffa@oncology.ox.ac.uk

Keywords: gene signature, signature quality control, gene set,

Abstract

With the increase in next generation sequencing generating large amounts of genomic data, gene expression signatures are becoming critically important tools to interpret these data, and are poised to make a large impact on diagnosis, management and prognosis for a number of diseases. Increasingly, it is becoming crucial to establish whether the expression patterns and statistical properties of a set of genes, or signature, are conserved across datasets. Conversely, it is increasingly necessary to compare independent datasets with respect to the expression of established signatures reflecting their clinical or biological characteristics. In this work, we introduce the first protocol, sigQC, which enables a streamlined, systematic approach for the evaluation of gene signatures across different, independent, datasets. To facilitate accessibility, we implemented the protocol in an R package (<https://cran.r-project.org/web/packages/sigQC/>) designed for users with modest computational skills. SigQC has been adopted by us and collaborators in several basic biology and biomarker studies. The emphasis is in showing the basic but critical quality control steps involved in the generation and application of a clinically and biologically useful, transportable gene signature, including evaluating its expression, variability and structure. It begins with evaluating signature genes' expression and variability, then evaluates statistical properties of the distribution of their expression, and then computes various signature scoring metrics, and gives empirical estimates for the significance of each of these metrics. We demonstrate the application of this protocol, showing how the outputs created from sigQC may be used for the evaluation of gene signatures on large-scale gene expression datasets.

Introduction

Gene signatures, over the past decade have revolutionised our understanding of disease, pathogenesis, and clinical response^{1, 2, 3}. While there are many definitions of what a gene signature constitutes, here we define a gene signature as a gene set, whose mRNA expression shows coherent patterns representative of a particular biological state, process, or outcome⁴. That is, we differentiate gene signatures as used in this work from signatures such as mutational signatures⁵ and multi-gene mutation panels⁶. The use of gene signatures in the clinic has become a massive force driving healthcare forwards towards personalised medicine, and in doing so, has led to a large development effort. These gene signatures are derived by an ever-increasing arsenal of methodologies, spanning approaches such as supervised⁷ and unsupervised clustering⁸, seed-based approaches^{9, 10} and other machine-learning techniques¹¹. However, in many cases these signatures remain limited by narrow use cases, or display a general ability in predictive power not specific to any particular state of disease, with even random gene signatures capable of significantly separating groups of breast cancer patients with favourable and unfavourable outcomes¹².

The pertinent application of gene signatures to a vast array of clinical data depends critically upon the ability of the signature to perform robustly over a wide range of possible confounders, noise, and inter-platform differences for gene expression profiling, and therefore requires validation in independent datasets⁴. Moreover, a central problem that limits the applicability of gene signatures to narrow use cases is the difficulty in summarising the expression of a disparate set of genes into a robust and transferable single score for each sample. There are many methods to achieve this, but many of these methodologies are dataset or technology specific, and therefore limit the utility of a particular gene signature.

In order to ensure that the influence of such factors is reduced, we show, within this work, a battery of tests and validation criteria that empower the user to determine whether, for a given gene signature, the signature's statistical properties are conserved across datasets, and whether the signature could be summarised into a single score. Of note, we present this technique as a quality control protocol, to be used before applying a previously derived gene signature on a new dataset, and we see this as necessarily separate from a protocol assessing the methodology used for signature derivation. Among the many methods of gene signature derivation, each has its advantages and disadvantages in different scenarios, and evaluation of these require deep, domain-specific understanding to navigate. As such, this tool is designed to produce metrics that will assist in determining whether an existing signature is able to generalise to a new dataset. Conversely, our approach also enables one to evaluate whether different datasets are similar with respect to a given biological signature. Thus, given the increasing number of independent studies becoming available for the same biological condition, this is an important step aiding in the comparison between studies, and assessing reproducibility of results.

An R package *sigQC* was developed to standardise and simplify the quality control metrics used to evaluate gene signature behaviour across datasets, whilst providing key information to assess the process of summarising a gene signature into a single score, using general,

dataset-independent metrics (see e.g. Buffa et al.⁹, Winter et al.¹⁰, or Maseiro et al.¹³). In other words, *sigQC* has been designed to provide a standardised assessment of the statistical properties of a gene signature in order to inform its use and further application. This tool has been designed over multiple iterations, with different aspects used in practice and evolving visualisations. Discussion of each of the metrics, and the core statistical principles of this tool can be found in multiple references^{9, 10, 13, 14} in different forms, along with earlier visualisations, and a more current iteration in Dhawan et al.¹⁵. As such, this is a tool designed to provide the user with actionable quality control metrics, such as identifying poorly performing genes from an existing signature to enable generalisation, but we caution users that this tool is not designed to solve all domain-specific quality control issues. It does not provide hard cutoff values to use with signature refinement, but does provide a broad range of statistics that can facilitate such refinement; it provides information about the components of a signature in a given dataset, and allows for the comparison of quality control metrics between datasets and signatures. For example, during the process of signature refinement, different iterations of signatures can be tested using this protocol, to check for improvement in significance or value of the metrics considered. In addition, this protocol has been designed without a preconceived gene signature type. That is, any set of genes on any dataset can be evaluated using *sigQC* for its statistical and metagene-like properties. Thus, our protocol serves as a general tool for gene sets in datasets to assess their ability to function as a metagene, which can subsequently be assessed for predictive or prognostic value, for instance.

To illustrate this tool in this work, we show how *sigQC* can be used to evaluate the properties of a published gene expression signature of breast cancer metastasis when applied to clinical samples in an independent dataset, and an example of quality assessment for this signature where gene expression has been measured by two different technologies, RNA-seq and microarrays. We also consider a gene signature comprised of a random set of genes to highlight the differences in performance between a ‘highly performing’ signature which possesses statistical characteristics that generalize across datasets and technologies, and whose expression can be summarised into a simple, single, compact score; with respect to a signature which does not generalize and is not well-described by a simple single summary score. Such a signature would potentially require dataset-specific methods to define a summary score (a ‘poorly performing’ signature). We show how the outputs of the quality control plots change in the presence of this highly performing signature and the example of poorly performing signature, and how such outputs may be used to test or refine a signature to ensure its cross-platform applicability.

Overview of the procedure

Conceptually, this protocol is designed to ensure that gene signatures are derived with characteristics suitable for clinical utility, and to elicit those properties that pertain to broader application. Whilst the protocol can be used to evaluate the statistical properties of any set of genes, it is particularly useful in those cases where the signature has been assembled so that the genes have co-ordinated expression with respect to a given phenotype. Examples of such signatures are metagenes, frequently summarized as single-value score and used to rank clinical or biological samples based on a given phenotype,

and signatures that have been generated for, or simply are to be used for, enrichment analyses (such as those available through MSigDB⁴).

This procedure begins with an evaluation of the signature genes' expression on the given dataset, and then considers the variability in the genes' expression across samples within the given dataset. Next, the statistical properties of the expression of the gene set is considered across all samples; such as normality of the distribution and skewness, and how well the genes of the gene set correlate with each other. Following this, the 'score' of the gene set across each of the samples of the dataset is computed, using a variety of scoring metrics, such as the mean, median, and first principal component, as well as more advanced and specialized gene signature metrics, and how these relate to each other for the given gene set. Then, through visualizations produced from *sigQC*, a search for any underlying structure in the dataset or in the gene set is conducted, to check for differentially acting subgroups or subsets of genes potentially affecting the gene signature's function. Finally, from each of these metrics, random resampling of gene sets of equivalent length, or permuting the names of the gene sets is carried out, to ascertain the statistical significance of each of the findings for the various metrics from the previous steps of the protocol. This can be repeated over multiple gene sets and datasets, thereby enabling the rapid, efficient, and standardised approach to gene signature quality control.

During the evaluation of a signature in a dataset, we have identified four key features that must be accounted for: i) signature technical transportability, ii) signature biological integrity, iii) signature suitability and iv) dataset suitability. These are reflected by the metrics in the *sigQC* protocol.

Signature transportability refers to the use of a gene signature across datasets produced by different technologies, such as RNA-seq vs. microarrays, which quantify genes differently, though they may originate from the same sample. The importance of this is underscored by the fact that over the previous decade, most gene signatures have been developed using DNA microarray technology, i.e. a collection of DNA 'probe' sequences attached to a solid surface, but most gene expression quantification at present is done by next-generation RNA sequencing. This is further complicated by the fact that microarrays themselves comprise a range of technical methodologies (e.g. spotting, in-situ synthesis) and may have different output characteristics (e.g. one-channel vs. two-channel detection)¹⁶. Moreover, presently, RNA sequencing is increasing in popularity and decreasing in cost, providing a wealth of genomic data, quantified in yet a different manner, and represents the current trajectory forward in the technological development of new gene signatures¹⁷. All of this variability between technologies must be taken into account, and the behaviour of a gene signature should show consistency across datasets generated by these technologies. In *sigQC*, we address this issue by initially producing plots of basic characteristics of the signature genes - expression and variability, which are key elements that may vary as technical platforms change.

Secondly, given datasets generated using the same technology, a signature's ability to represent a biological phenomenon in a general, reproducible behaviour in a specific context, should be ensured, before moving on to wider application. To study the degree to

which a signature is able to represent this heterogeneity, in *sigQC* we produce plots describing the distribution of signature scores across the datasets and with covariates. *sigQC* also includes an analysis of modality, and additionally produces clustered heatmaps of signature gene expression, to identify dataset or signature subsets showing distinct patterns of expression.

Lastly, in the case of multiple signatures and multiple datasets for the same phenotype or biological process being captured, the signature under primary consideration should be the most suitable for both the dataset and the level of generalisability desired. Further, in those cases where a single score summarization of the signature is desirable, it is important to critically ask and thoroughly assess if a dataset possesses the properties facilitating the summary of signature gene expression into a single score, before proceeding with further analysis. In *sigQC*, for easy comparison across dataset and gene signature combinations, quality control metrics are summarised numerically and plotted on a radar plot. These values are scaled to ensure comparability between the different cases considered, and this allows for the efficient comparison of many gene signatures across many datasets.

Importantly, there are two overarching aspects to this protocol, the first being the tests of the properties of the genes comprising the signature itself, and the second being the properties of the dataset as it pertains to the signature genes. A flowchart of the procedure is depicted in Figure 1.

The evaluation of the genes composing the signature is primarily to determine whether signature genes are expressed and varying, and whether they co-operate to give a strong, coherent signal across the samples. For a discussion of this principle in practise, see for example the discussion in Harris et al., 2015¹⁴. In addition, as important as the signature itself, are the statistical properties of the dataset to which it is applied. Thus, within this protocol, we describe how a search for structured subcomponents of a gene signature or dataset may be done, to discover whether there are subsets of genes or samples that could benefit from subsetting as a distinct class. For instance, consider the search for clusters in Figure 3 of Buffa et al., 2010⁹. Finally, the *sigQC* package includes commands for random resampling, and evaluation of a set of negative controls, using both random resampling and gene name permutation, to reveal an understanding of the null distributions for each of the metrics we consider in evaluating signature quality, examples of which we have discussed in Winter et al.¹⁰ and Buffa et al.⁹.

Application of the method

Here, we depict a motivating example of the protocol, as implemented through the R package *sigQC* to evaluate a published gene signature for breast cancer metastasis, and a random gene signature on a RNA-seq dataset from clinical breast tumour samples, downloaded through the Firebrowse portal as part of the Cancer Genome Atlas project (TCGA)^{19, 20}. The gene signature used is a set of upregulated genes in breast cancer metastasis, taken from Van't Veer et al.¹⁸. We also present the analysis of output data for the use case of signature transportability across microarray and RNA-seq datasets for the metastasis signature in Box 1.

Experimental design

Here, we highlight the critical steps of our protocol, as well as those where analysis of gene signature quality occurs.

(Un)-certainty in signature gene annotation (Step 2)

Prior to the testing of a gene signature, as a pre-evaluation step, we propose ensuring compatibility between a gene signature and the dataset intended for use. In particular, because of a number of different annotation conventions for genes, compatibility between the genes of a signature derived from one annotation of the genome, should be able to be mapped to a matching annotation of the genome, without significant loss of content or specificity. Several tools have been developed to accomplish this task, one widely used example is BioMart²¹. Because such mappings are generally not bijective, it is critical to ensure that there is reasonable representation of all genes in a signature among the annotation used in a dataset of interest, as this uncertainty can detract from the functional ability of a given gene signature.

Evaluation of signature gene expression (Step 8)

A critical first step in the evaluation of the validity of a gene signature on a dataset is to ensure that the genes of the signature are expressed at a detectable level across the samples being considered, or at least in a sufficiently large subset of them. As a general rule, if genes within a signature are being used to differentiate biological or clinical groups, the expression value used to differentiate must be above a noise threshold. This threshold is context dependent. For example, in cases where lowly expressed genes may be key elements distinguishing biological or clinical states, a lower expression threshold is required, and this needs to be reflected by assays designed to greater specificity. In such cases, *sigQC* can aid by providing an indication on whether the minimal requirements of the assay are met. Additionally, a gene consistently not expressed within a gene signature across a whole dataset, contributes little to the overall use of the signature as a classifier, but the fact that the gene is not expressed or not varying might be very informative on the biology of the whole dataset. Thus, *sigQC* first evaluates the expression of all genes in the signature, and presents the proportion of samples expressing each gene at supra-threshold level, as well as the proportion of all samples that have gene expression recorded as non-NA value for each signature gene. The threshold for expression may be user-specified for each dataset, depending on the biological question asked, and on the technical characteristics of the dataset, such as the platform used. To aid this inspection, a graphical representation of this in the form of a bar chart and density plot showing the proportion of samples expressing each gene above a particular threshold is returned.

We note here also that each dataset considered by *sigQC* must consist of a numeric expression matrix with at least 2 samples each, pre-normalised, and log-transformed per gene signature requirements, and pre-standardised, if needed. The genes of interest should

be present in the dataset and not reported primarily as NA values (i.e. if it is not reported or not quantified for a technical reason). If a single dataset with subcomponents affected by batch effects is included, the data should be batch-corrected before use. However, if there are batch effects between multiple datasets, these do not need to be corrected before use, as *sigQC* is designed to test multiple independent datasets.

Evaluation of gene expression variability (Step 9)

In addition to having non-zero expression across a number of samples, signature genes that function well as classifiers should vary above the noise threshold across samples. As such, we propose an evaluation step involving the comparison of a standardised metric of variance, the coefficient of variation, among the genes of the signature, to all genes recorded in the assay. To facilitate inspection, this functionality is provided both as tables and as a scatter plot visualization of mean versus standard deviation of all genes, and their associated quantiles for mean and standard deviation, overlaid with the same scatter plot for all signature genes.

Summarisation of signature gene expression to a single score (Steps 10-12, 17, 18)

Across many domains of application for gene signatures, it is often desired to determine whether a signature can be summarised into a single ‘score’ to enable the comparison between biological or clinical samples. The purpose of such a score is to encapsulate information from the entire signature, but to not be swayed by outliers in the signature genes, which may detract from its performance. Such summarization must be applied carefully as it may result in artifacts and erroneous conclusions if the signature genes did not have the requisite statistical properties for the score to be robust. To assess the suitability of such summarisation for a given signature, *sigQC* compares different score metrics; namely, a ‘mean score’, a ‘median score’ and a principal component score, ‘PCA score’. Other metrics, or more advanced combinations of these metrics, could be used. However, we propose the evaluation of these three metrics as they provide crucial basic information that can be transferred to more complex summarisation methods.

A mean score, namely a score based on the mean expression of the signature genes in each sample, is attractive for its simplicity. However, by using this score we implicitly assume that the mean is a fair summary for the distribution of the expression of all signature genes. This is not the case, for example, if the signature is not compact, as big expression changes could occur in opposite directions for different groups of genes without affecting the mean expression. Another case where a mean score would not be appropriate is in the case of skewed distributions. In such cases, outliers, namely a small number of genes expressed at significantly higher or lower levels than the other genes in one or more samples, could heavily shift the mean score, and the subsequent ranking of the samples.

A median score, namely the median of the expression of the signature genes in each sample, is attractive for its robustness to outliers. This is a non-parametric score providing us with an indication of how the midpoint of the distribution (the point dividing the population of genes in half based on their expression) changes across samples. The advantage of such a

score is that the median expression is usually fairly stable, and it moves significantly only if the expression of a substantial proportion of the genes changes coherently. However, similarly to the mean score, it may not be appropriate where the signature is not compact, as big expression changes could occur in opposite directions for different genes without changing the median, and it can miss subtle changes in expression of subsets of genes which might be biologically or clinically important.

Finally, principal component analysis (PCA) projects a set of observations of possibly correlated expression values into a new gene expression space of linearly uncorrelated variables, the principal components (PCs). The transformation is defined such that the first PC accounts for as much of the expression variability as possible, or, in other words, has the largest possible variance. Then, each of the following PCs has in turn the highest possible variance, under the constraint that it is uncorrelated to the preceding PCs. By considering in each sample the magnitude of the first PC projection as a score, we guarantee that our score represents the change in the direction of the largest variation of expression. However, we also implicitly assume that this variation is sufficient to be a meaningful representation of the biology of the gene signature, which is a property of the input dataset. When this is not the case, the first PC fails to represent a large proportion of the variance, either further PCs should be considered, or more complex summarizations approaches should be used.

sigQC computes summarization scores, providing their values and distribution, and asks whether the order of the samples is conserved when different scores are used to rank them. Score values are compared using Spearman correlation, which is the correlation of the samples' ranks. A high correlation between a mean and median score indicates that outliers, if present, have a contained effect. A high correlation between the median and PCA scores, indicates that the expression of the signature genes is changing in a coherent fashion (the signature is compact), and that these changes are well-represented by changes in the midpoint of the distribution. [For each of these scoring metrics, *sigQC* also analyses the statistical properties of the distribution of scores across the samples in each dataset. Namely, it computes Q-Q plots testing normality of the distribution, and for the evaluation of modality \(e.g. in identifying subgroups of samples\), uses the mclust R package to compute the Bayes Information Criterion \(BIC\) for the likelihood of different Gaussian mixture models.](#)

We have also built into *sigQC* three further signature scoring metrics, which have been developed for signatures such as metagenes or those taken from repositories such as MSigDB⁴. These metrics are the GSVA algorithm (gene set variation analysis)²³, the ssGSEA algorithm (single sample gene set enrichment analysis)²⁴, and the PLAGE algorithm (pathway level analysis of gene expression)²⁵. GSVA functions on the basis of calculating an enrichment score for a given gene signature (or gene set), and then computing the relative activity of these genes across samples, summarised into a score value for each sample²³. ssGSEA calculates a gene set enrichment score, based on the GSEA approach for each sample, and this enrichment score is taken as the sample score²⁴. PLAGE relies on first standardising the data using z-scores, and then uses the singular

value decomposition in order to determine the weighting of each sample's expression on the given gene signature, as its score²⁵.

A high degree of correlation between all of the above metrics gives a first indication that the signature has favourable properties to a single-score summarization, and the relative ranking of the samples is robust when different summary metrics are chosen amongst the ones presented. *sigQC* provides this information for all of the above metrics, and it returns the calculated scores for each signature/s and dataset/s considered, facilitating their application and use in further evaluation studies.

Effects of data standardization (Step 13)

A subsequent issue with the application of gene signatures is the effect of data standardisation, as a given signature may be applied on a set of data standardised in a particular way for biomarker discovery purposes, but for application purposes, the data is often re-standardised in a different way. To account for this, we offer that the gene signature metrics and summarisation provided by *sigQC*, and illustrated in the following sections, should be compared using non-standardised data and standardised data. In this way, the effect of gene expression standardisation on signatures which had been originally developed using absolute expression rather than standardised expression can be established. Likewise, it can be determined whether the information carried in the standardised expression is at risk of being lost when using non-standardised data.

Evaluation of signature compactness (Step 14)

A compact gene signature is one that contains genes with high levels of pairwise correlation, or 'intra-signature correlation', among themselves. Often, this is the implicit assumption underlying the application of a signature. For example, gene set enrichment analysis²² and related methodologies, widely used both in basic and clinical research, generate biological hypotheses on a given dataset based on the co-ordinated behaviour of gene sets representative of a specific biological phenotype. In such cases, the gene signature with components acting in a co-ordinated manner ensures that the signature has more likely captured the biological phenotype of interest, and that summary scoring metrics will not have significant outliers detracting from the other genes of the signature. While this is often done as a step in the derivation of the gene signature, we pose that it should also be verified as a property that holds true in the testing of a gene signature on validation datasets prior to application. In doing so, this ensures that a key metagene property holds in the new dataset. Additionally, this can inform whether a given metagene is suitable for an application different to that originally envisaged, which is often the case with gene set enrichment approaches, and can provide an indication for the need of further refinement or *de-novo* derivation. For example, a metagene derived for a disease subtype might not behave as a compact set of genes when applied to another subtype of the same disease. However, understanding whether, or to what extent, the metagene behaviour is conserved can assist in the design of further studies. This verification step reveals signature genes or

subsets of genes displaying a discordant behavior in a new dataset, and also those genes or subsets of genes that do maintain a compact behavior across a number of normal and disease conditions. To test the level of intra-signature correlation among signature genes, in the *sigQC* package, the intra-signature correlation matrix is provided for the user to inspect, and a heatmap of correlation coefficients is created which compares the correlation of every gene with every other gene in the signature. When multiple datasets are considered together, *sigQC* takes each correlation value of each signature gene pair in each dataset, and uses the rank product statistical test to determine whether a particular gene pair is poorly correlated more than expected due to chance, across multiple datasets.

Searching for structure in signature gene expression (Steps 15, 16)

Signature structure can be thought of as an underlying set of components comprising the signature, that tend to cluster together in terms of either gene co-expression, or groups of similar phenotype. Like the evaluation of signature compactness, structure of the signature and datasets are both taken into account during the development of a gene signature, but should also be verified in new datasets to ensure a similar pattern of gene signature expression. Furthermore, a change in the structure of a gene expression signature in a new dataset or context can reveal important insight into different technological issues or biological aspects, generating new hypotheses that can be tested.

Structure can be evaluated using various techniques; here we propose PCA (introduced in the section above) and hierarchical clustering (using the *hclust* function in R) for their easier visual interpretability with respect to other methods. This initial qualitative assessment is useful to prompt the need for further more advanced analyses of the signature structure. For example, this can be used to assess the level of redundancy present in a signature; that is whether different subgroups of genes carry similar information. Depending on the application, redundancy might be a sought-after, or an unwanted characteristic. Conversely, understanding whether independent subcomponents of a signature exist is an important part of evaluating a gene signature, as such components may signal biologically distinct sets of genes or samples within the datasets considered.

Comparison of multiple signatures and datasets (Optional step)

The *sigQC* package has been designed with an extensible framework, and can be used for the evaluation of multiple signatures and datasets at once. A summary plot produced by the package displays a host of metrics summarizing the previous steps on a single radar plot. This visualization facilitates comparison of various metrics of multiple signatures on multiple datasets at once, with a single graphic image. Using this, the quality of various signatures, and the reasons for differences in quality, can be rapidly assessed over multiple datasets in a comprehensive manner. Files including raw data of all the statistics and summary scores computed by *sigQC* are also provided as output, for further analyses.

Evaluation of null distribution of gene signature quality control metrics (Optional step)

Each of the metrics presented on the summary radar plot is computed for a given gene signature on a particular dataset, but to gain a greater understanding of the significance of these values for a given signature, it is critical to consider the underlying null distribution from which each of these metrics/statistics arise. Thus, for each dataset and gene signature combination, random resampling is performed to evaluate the underlying null distribution (or *negative control*) for the statistics observed. Namely, the distribution which would be observed under the assumption that there was no effect (e.g. the genes in a given signature were not correlated, or scores were not correlated). We do this using two different widely used approaches.

The first approach, *random resampling*, considers random gene signatures of the same length as the signature to be evaluated. For each dataset and for each of the fourteen metrics considered, all statistics are computed for each of the random signatures and their distributions (the null distributions) are provided to the user in forms of boxplots. Immediately, this gives an evaluation of the significance of the quality control metrics. When using this technique, we caution the user that because we are selecting genes at random from the entire distribution of *all* genes, for gene signatures defined from more restricted subsets of genes, we have potentially inflated the significance by including irrelevant genes in the process of resampling. Thus, for a more nuanced calculation of these p values, we propose that it is important to consider the set of genes from which the signature was originally derived.

The second approach considers *permutation resampling*. Namely, instead of resampling from a random set of genes, resampling is done by randomly exchanging labels of the signature genes for each sample in each dataset. This provides a potentially stronger estimation of the null for some metrics, such as the intra-signature correlation of signature genes and the PCA, and has been previously used for similar analyses, such as gene set enrichment²⁶.

Lastly, as a further evaluation of signature metric significance, we [advise](#) the use of a *positive control* gene signature when available. That is, if there is a gene signature that has already been derived as a metagene on the given dataset, then this can be simply added to the list of gene signatures to be tested on a given dataset, and compared to the signature of interest to understand further how the metrics differ. While this positive control signature may not always be available, when it is, it can be used as an important tool to better understand signature performance on the *sigQC* metrics.

Comparison with other methods of signature quality control

To our knowledge, no generally adopted methods of gene signature quality control exist in the literature, though some methods have been suggested for specific purposes. For example, a generally adopted technique for the validation of significant prognostic ability of a signature is to resample random gene lists of the same length as the gene signature, and determine their prognostic ability¹². This resampling approach is one of the techniques adopted by *sigQC* to provide the null distributions of our metrics. Interestingly, such an analysis has also shown that a cutoff of $p = 0.05$ may be too lenient when aiming to predict

a specific clinical outcome with a gene signature, as many randomly selected gene signatures can also prognosticate with statistical significance¹². Furthermore, characteristics such as coherence, uniqueness, robustness, and transferability have been previously suggested in the context of signature evaluation²⁷, providing strong support for some of the *sigQC* metrics. However, their application has been limited to a small subset of gene signatures, namely those adopting PCA during the signature generation phase. Finally, consensus classification, addressing uncertainty arising from normalization and other data-processing steps, has also been proposed as an option to both evaluate and improve gene signature performance²⁸. These methods solve existing issues related to gene signature development and validation, but primarily test specific aspects of the gene signature's performance, without enabling a broader assessment of its basic statistical properties across different contexts and datasets, nor an evaluation the qualities of the signature genes themselves.

Limitations

The protocol presented through *sigQC* is limited by the fact that the applicability of a gene signature to a broader setting can never be entirely determined, and so there may be characteristics, intrinsic to a signature or signature types, that enable it to pass all of the proposed quality control measures, without performing well in its intended sense. More succinctly, because *sigQC* does not account for the wide range of outcomes that gene signatures have been designed to predict, we are limited to a solution which is highly general, and not domain-specific. That is, while *sigQC* provides metrics useful to the initial quality control for a gene signature and dataset, this does not optimise based on what is being predicted. For instance, *sigQC* does not take into account covariate adjustment when predicting specific outcomes, such as survival. Such a limitation will almost certainly occur, given the diversity of methodologies of gene signature generation and the wide variety of outcomes predicted, and to address this, we caution users of this method that it provides a set of conditions which are important to check, but are not fully sufficient for the determination of gene signature applicability. Undoubtedly, because of the nature of gene signatures, this limitation will be present in any broadly-scoped quality control methodology, as there may always be cases for which such a quality control methodology may not detect a poorly-performing signature.

A second limitation of *sigQC* relates to the number of signature scoring metrics possible and in use today. For simplicity and usability, *sigQC* supports basic primary summary metrics such as mean, median, and first principal component of the expression of the signature genes. While these are strong, commonly used, and easily generalisable summary metrics, and they form the building blocks of many more complex metrics, others have been proposed that may show differences with what is used in *sigQC*. Alternative scoring metrics that have been proposed in the literature include many linear modelling approaches (for example, Knudsen et al.²⁹), S-scoring^{30,31}, averaged z-scores³², and Pearson correlation based-scores³³. The metrics we employ in *sigQC* translate well to the averaged z-scores, as we perform a comparison of standardised data to unstandardised data. Additionally, by testing intra-signature correlation using the Spearman correlation coefficient, we are able to capture non-linear relationships not seen by Pearson coefficient-based signatures, at the

cost of potentially increased noise from these non-linear relationships. S-scoring is based on a linear combination of z-scores, and combines the approaches of standardising the dataset with the directionality and flexibility of a linear model³⁰. Thus, like a linear model, this scoring system, while it may be more flexible for defining dataset specific scores, often does not translate easily to new datasets or technologies. These methods are not tested explicitly in the current version of *sigQC*, however the metrics provided by *sigQC* provide a broad statistical assessment of the genes in a given signature across datasets and technologies, information which can be used to design more context-specific scoring techniques.

Materials

Equipment

Hardware:

- Personal computer, capable of running R version 3.3.0 or higher

Software:

- R version $\geq 3.3.0$, available to install from <https://www.r-project.org/>
- [Bioconductor compatible with R version; installation instructions available from https://www.bioconductor.org/install/](https://www.bioconductor.org/install/)
- *sigQC* package, available to download from <https://cran.r-project.org/web/packages/sigQC/index.html>
- The following R packages are required as dependencies: MASS, lattice, KernSmooth, cluster, nnet, class, gridGraphics, biclust, gplots, ComplexHeatmap, RankProd, fmsb, moments, grDevices, graphics, stats, utils, and mclust, and can be installed from the CRAN repositories, or are installed automatically when *sigQC* is installed.
- ImageMagick is required as a software dependency, and is available to install from: <http://imagemagick.org/script/download.php>

Equipment setup

R software installation:

- Download and install the latest version of R from <https://www.r-project.org/>, or the freely available RStudio from <https://www.rstudio.com/>.

sigQC installation:

- To install the *sigQC* package, execute the following command in R or RStudio
install.packages("sigQC")

ImageMagick installation:

- To install ImageMagick, visit <http://imagemagick.org/script/download.php> and download the appropriate software for your operating system (Unix, Windows, or MacOS)

- Follow the detailed instruction sets for your operating system listed at <http://imagemagick.org/script/download.php>

Input formats and usage:

The primary user-accessible function of *sigQC*, *make_all_plots*, expects a number of inputs, the format of each of which is defined in Table 1 as well as the package documentation. Further, once installed and with all data loaded into the appropriate variables, use of the package is accomplished with the following commands in R or RStudio:

```
library("sigQC")
```

```
make_all_plots(gene_sigs_list, mRNA_expr_matrix, names_sigs,  
names_datasets, covariates, thresholds, out_dir, showResults, origin,  
doNegativeControl, numResampling)
```

Downloading of sample data and code:

Sample randomly generated data and code can be found in the package vignette example that is available for download with the package at <https://cran.r-project.org/web/packages/sigQC/index.html>. The datasets and scripts used to generate all figures in this manuscript can be found for download at Zenodo: <https://doi.org/10.5281/zenodo.1319848> (DOI: 10.5281/zenodo.1319848).

Procedure

Prepare the input data (Timing: 1-5min, variable depending on input):

1. Load the input data as lists of numeric expression matrices of at least 2 samples each, pre-normalised, and log-transformed per gene signature requirements, and pre-standardised, if needed. Ensure that genes of interest are present in the dataset and not reported primarily as NA values. If a single dataset with subcomponents affected by batch effects is included, ensure data is batch-corrected before use.
2. Annotate the genes of the signatures to be tested in a manner consistent with the input data.

```
#load TCGA data
```

```
brca_rnaseq <-  
read.table('BRCA_TCGA.txt', stringsAsFactors=F, quote="", header=T, sep='\t')
```

```
#set row names
row.names(brca_rnaseq) <- brca_rnaseq[,1]
brca_rnaseq <- brca_rnaseq[,-1]
```

Optional

- A. Compute any specific expression thresholds (other than the global median, which is the default value the package uses as an expression cutoff).
- B. Identify any additional annotation data to be used alongside the expression heatmaps, and load this into the appropriate matrices with colour descriptors as specified in the package documentation.

Create the input variables (Timing: 1-5min, variable depending on input):

3. Prepare the gene signatures list: The gene signatures considered should each be k x 1 sized character matrices for a signature of length k genes. The individual elements of these matrices should be the gene names (and should be consistent with the naming convention for genes as named in the row names of the expression matrices). Save these matrices into a single R list variable, such that each matrix is one element of the list of gene signatures, and is named to describe the gene signature contained in this matrix.

```
gene_sig_list <- list()
```

```
#load metastasis signature
```

```
genes <- read.table(paste0('sigs/brca_met_symbols.txt'), header=F,
stringsAsFactors=F, colClasses = "character")
```

```
gene_sig_list[['Metastasis_Sig']] <- as.matrix(genes)
```

```
#generate random signature
```

```
random_sig <-
rownames(brca_rnaseq)[sample(1:length(rownames(brca_rnaseq)),57,replace
=F)]
```

```
gene_sig_list[['Random_Sig']] <- as.matrix(random_sig)
```

4. Prepare the gene expression list: The gene expression matrices considered should be matrices with rows as the genes and columns as the individual samples. The row names of these matrices should be the gene names, and naming conventions consistent with those allowable for R object naming are permitted. These are the same gene names that will be displayed on the produced plots. Save each gene

expression matrix for each dataset considered as an element of a single list variable in R, with each element of this list set as one of the gene expression matrices, and named to describe this dataset.

```
data_list <- list()  
data_list[['BRCA_RNA_Seq']] <- brca_rnaseq
```

5. Set the output directory variable as a string for a file path that is reachable from the current directory in the *out_dir* variable. If no value is set, this defaults to the temporary directory given by R in *tempdir()*.

```
out_dir = 'sigQC_sample'
```

Optional input variables:

- A. If the datasets have been derived from different labs and experimental setups, set the *origins* parameter to indicate this. This is only used in the computation of the rank product statistic when comparing intra-signature correlation of genes across datasets, to identify consistently poorly correlated signature genes, to account for batch effects between datasets. Set this as a vector of numbers or characters, with each element indicating numerically the origin of a dataset, in the same order as they appear in the dataset list input variable.
- B. If alternative names (other than those used for list indexing) are desired for the plots produced by *sigQC*, there is the option to set the *names_sigs* and *names_datasets* variables, which are vectors containing the desired names of the signatures and datasets (ordered in the same way as the list variables they represent).
- C. During the plotting of heatmaps showing the expression of the signature genes across samples in the various datasets, if it is desired to have annotation rows at the top of the heatmaps, indicating sample characteristics, set the *covariates* input variable. Create a list with one sub-list element per dataset. Within each of these sub-lists, assign two matrix elements named 'annotations' and 'colours,' describing the annotation values for each of the samples and the associated colours to be used in the plotting. For further information about this, we refer the user to the documentation for the ComplexHeatmap R package, as this is the same *covariates* variable as used in this package.
- D. If it is desired to study expression above a particular threshold (e.g. a noise value), set the *thresholds* variable (with threshold values for each dataset in the same order as they appear in the list of datasets). If this value is not set, it is defaulted to the median expression of all genes across all samples in each dataset.
- E. To compare to the null distributions via resampling for random sets of genes and permutations of the gene signature labels, set the *doNegativeControl* variable to TRUE, otherwise set this to FALSE.

- I. Set the *numResampling* variable to the number of resampling runs to be done if the *doNegativeControl* variable is true.
- F. To see results in the R graphics windows as they are created, the *showResults* parameter can be set to TRUE (default), otherwise it can be set to FALSE.

Run the *sigQC* package (Timing: 5 min - hours, variable depending on input):

- 6. With the input data pre-processed and in the appropriate variables as described above, run the principal function of the *sigQC* package with the following command:

```
library("sigQC")
```

```
make_all_plots(gene_sigs_list, mRNA_expr_matrix, names_sigs,  
names_datasets, covariates, thresholds, out_dir, showResults, origin,  
doNegativeControl, numResampling)
```

- 7. Check the output log file when the package has completed running. The file 'log.log' in the output directory is a text file that summarises the run, and reports any errors that may have occurred if they are not printed to the console. Consult this if any issues are encountered and for troubleshooting purposes.

TROUBLESHOOTING

PAUSEPOINT – the following steps in the protocol are data analysis steps and the procedure may be paused prior to analysis and resumed at any time.

Analysis of expression (Timing: 5-10min, variable depending on input):

- 8. Evaluate the expression of signature genes across samples in the datasets, done by analysis of the plots *sig_expr_*.pdf*, as shown in Figure S1A-C. These describe the proportion of samples with supra-threshold expression of each signature gene, and the proportion of samples with non-NA values, identifying non-expressed signature components.

Analysis of variability (Timing: 5-10min, variable depending on input):

- 9. Analyse variability of signature genes by loading the file '*sig_mean_vs_sd.pdf*,' an example of which for sample datasets and gene signatures is shown in Figure 2. These plots describe the mean and standard deviation of expression of all genes reported (in grey) versus all signature genes (in red), with corresponding dashed lines over the plots describing the 10th, 25th, 50th, 75th and 90th percentiles of both mean and standard deviation. This facilitates the easy identification of those signature genes, which are not variable or expressed among the samples, as well as a global evaluation of signature behaviour across samples of a dataset.

Analysis of co-correlation of scoring metrics (Timing: 5-10min, variable depending on input):

10. Analyse the relationship between various scoring metrics by loading the files called 'sig_compare_metrics_*.pdf'. This enables visualisation of the correlation of mean, median and first principal component (PCA1) as scoring metrics across the samples for each signature across each dataset, depicted in Figure 3. This also shows a principal components analysis (PCA) scree plot, which describes the proportion of the variance attributable to each principal component, reflecting whether the first principal component represents a reasonable scoring summary metric for the gene signature.
11. To analyse analogous plots for the GSVA, ssGSEA, and PLAGE gene signature scoring metrics, load the files called 'sig_compare_ES_metrics_*.pdf'. As these are enrichment scoring metrics, these are best used in concert with the quality control for signatures based on enrichment analysis results, such as those from MSigDB⁴.
12. Load the files called 'scoring_metrics_corr_*.pdf,' to observe the co-correlation between all of the six scoring metrics considered (mean, median, PCA1, GSVA, ssGSEA, and PLAGE), as depicted in the bottom row of Figure 3. This is of particular use for signatures derived based on enrichment analyses.

Analysis of data standardisation effects (Timing: 5-10min, variable depending on input):

13. Analyse data standardisation effects by loading the output file called 'sig_standardisation_comp.pdf', an example of which is presented in Figure S2. This provides the comparison of median of gene signature expression on the raw data provided versus the median of the gene signature expression on the z-transformed (standardised to zero mean and unit variance) dataset, for each sample in each dataset and each gene signature under consideration.

Analysis of signature compactness (Timing: 5-10min, variable depending on input):

14. Load the files produced in the output directory called 'sig_autocor_hmaps.pdf' and 'sig_autocor_dens.pdf' to view the plots in heatmap and kernel density estimate form for the correlation of each signature genes' expression with the expression of every other signature gene, providing an analysis of signature compactness. These are shown for sample data in Figure 4 where it can be seen that as expected, the breast cancer metastasis signature shows a high degree of intra-signature correlation, and the random set of genes does not.
 - A. When there is more than one dataset analysed for each signature, load the files 'sig_autocor_rankProd_*.pdf' (not shown for this analysis). These plots represent the output of the BioConductor RankProd package for the evaluation of signature genes whose median intra-signature correlation with all other genes consistently ranks low with the other signature genes. This finds genes with consistently poor intra-signature correlation with the other genes of the gene signature, particularly useful when refining a given signature for optimal performance across a number of different datasets (e.g. multiple clinical cohorts, or clinical data and cell line data).

Analysis of signature structure (Timing: 5-10min, variable depending on input):

15. Load the plots in the output directory named 'sig_eval_struct_clustering_*.pdf', which expression heatmaps for each of the gene signatures considered, clustered with hierarchical clustering, based on each dataset in turn, and run over each signature and each dataset present. An example of such a plot is shown in Figure 5, where the different expression profiles of the random gene signature and the metastasis gene signature across patients can be seen.
16. Check for the presence of biclusters (subsets of samples and genes acting differently than all others), by looking for and loading, if present, the plots named 'sig_eval_bivariate_clustering.pdf' in the output directory. These describe the biclusters of sample groups and signature elements if these are found (not shown for example considered analysis). The particular details of the biclustering outputs and algorithm used may be found in the R Package documentation. If no biclusters are found, these plots are not created.
17. Assess the signature and dataset structure through an analysis of the distributions of signature scores produced by each of the mean, median, and first principal component. In the files called 'sig_qq_plots_*.pdf', analyse the Q-Q plots against the normal distribution, to determine how close to normally distributed the signature scores are across the datasets, for each of these three metrics.
18. Assess the modality of the score distributions for each metric across each dataset. The files called 'sig_gaussian_mixture_model_*.pdf' provide plots of the modality on the x axis for each type of considered Gaussian mixture model, and the corresponding BIC on the y-axis. The peak values on these plots show the most likely modality of the distribution of gene signature scores, assuming an underlying Gaussian mixture model, showing how the scores are distributed among the datasets considered.

Optional: Comparison of multiple signatures:

- A. Compare the quality metrics of multiple gene signatures by loading the file entitled 'sig_radarplot.pdf'. This plot evaluates each gene signature and dataset combination across a number of metrics, described in detail in the Supplementary Table S1. A sample of this plot, for the metastasis gene signature and the random gene signature on the TCGA breast cancer dataset is shown Figure 6.

Optional: Analysis of null distributions of QC metrics (Timing: several minutes-hours)

- A. Load the file 'boxplot_metrics.pdf', in the negative_control subfolder of the results output to analyse the distributions of each of the fourteen quality metrics reported in the radar plot for each signature and dataset combination. These distributions are generated for the number of repeats as specified by the input parameter numResampling, with default set to 50. The values for the gene signature and dataset combination in question are shown in red overlaid with the other points in grey, giving a sense of significance of each of the metrics, as shown in Figure 7.

Optional: Analysis of raw data:

- A. Load the tab-delimited text files in subfolders of the output directory for re-analysis of signature properties. *sigQC* produces subfolders for tables of mean expression and standard deviation of expression for all signature elements, tables of the mean, median, and first PCA of each sample, tables of the median and z-transformed median for each sample, intra-signature correlation matrices for all signature elements, tables of proportion of expression above threshold and proportion of NA expression for all signature elements, as well as the table of values plotted for each signature and dataset in the summary radar plot. These raw data can be used for further custom analysis pipelines or re-plotting.

Timing

The timing of *sigQC* functions varies, depending on the number of datasets and signatures analysed, from a few minutes (for the examples shown here) to hours (for concomitant analysis of several datasets and signatures, and high number of replicate resampling).

Troubleshooting**Step 3:**

Issues may be experienced if the ‘ImageMagick’ dependency is not installed on the user's system (particularly for Windows systems). To install this dependency, please follow directions at: <http://imagemagick.org/script/download.php>.

Issues may be experienced with input data not conforming to the format required by *sigQC*. If this occurs, the package will alert the user with an error message describing the nature of the discrepancy. For example, common errors may include the following:

- Gene signatures must be formatted as a list of matrices, of dimension k rows by 1 column, for a signature of length k genes. Inputting a single list as a vector will cause an error to the program.
- Datasets must also be formatted as lists of matrices, such that genes are the rownames of the dataset, and samples are organised by columns of the dataset.
- Gene signatures and datasets must be annotated in the same way, as if the names of the genes of a signature are not found in a dataset, the computation will not continue.
- Care must be taken to ensure that NA-valued genes are removed as optimally as possible, as if there are too many values in the expression matrix for the gene signature are NA, calculations dependent upon singular value decomposition (e.g. principal components analysis) cannot be carried out.

Font warnings:

Warnings may be produced if R is unable to scale text characters either in the gene signature or dataset name appropriately during the plotting step. These can be alleviated by changing the affected characters.

Graphical output errors:

Errors may be experienced with graphical output in certain cases; sometimes due to dependencies that *sigQC* relies on. When these errors have occurred, the log file will often contain a string similar to: `'editThisGrob(grob, specs) : slot 'vp' not found'`. The output file produced may be called `'RPlots.pdf,'` indicating that there has been a failure to appropriately plot one of the *sigQC* results. To troubleshoot this, we recommend first ensuring that 'ImageMagick' is installed on your system as described above, and then that the RStudio plots pane window is maximised (even if not immediately viewing plots).

Anticipated results

Here, we provide an explanation of the results generated in the example described above. In particular, we describe the figures produced in steps 4-11 of the above procedure. Output plots are produced and stored in the directory set in the variable `out_dir`, and raw data used to produce these plots is stored in the subfolders within `out_dir`, as tab-delimited text or .csv files.

Analysis of expression and variability (Steps 4-5):

In the case of a signature performing well on a given dataset, as defined by *sigQC* metrics, the genes of the signature will be highly expressed and highly variable, as evidenced by the plots of expression and variability in Figures 2 and S1. As shown in Figure 2, the red dots, corresponding to the genes of the signatures are enriched higher-expression and higher-variability regions of the plot for the metastasis signature, as compared to the random gene signature.

Analysis of scoring metrics and standardisation (Steps 6-7):

The next steps in the protocol are the evaluation of the correlation between different scoring metrics, and whether standardisation preserves scoring metrics' assigned samples' ranks, or order, within the dataset. In the case of a highly-performing signature on a dataset, as evidenced by the case of the metastasis signature, as seen in Figures 3 and S2, each of the scoring metrics is correlated with the other, as well as the median metric between standardised and un-standardised data. This is not observed for the random gene signature, as might be expected.

Analysis of intra-signature correlation (Step 8):

Within a *sigQC* highly-performing gene signature, each of the genes of the signature should be acting coherently; that is, each gene should be increasing or decreasing in expression in a concordant manner with biological phenotype. This is an important characteristic, as discussed in the presentation of the protocol above, for signatures to be summarised in to a single score (e.g. metagenes), or to be used in gene set enrichment analysis approaches. We quantify this by providing intra-signature correlation values for the gene signature. As shown in Figure 4, it can be seen that for a highly-performing

signature, the metastasis signature, there is a significant intra-signature correlation between each of the genes, which is not seen for the random gene signature.

Analysis of signature structure (Step 9):

As further evaluation step, we ask the question of whether there are subgroups of patients with markedly different expressions of some of the genes in the signature, discordant with other genes of the signature. In other words, we assess whether the signature genes act in a similar manner to capture a biological phenotype. A coherent signature is evidenced by the lack of biclusters and obvious visual subclusters of patients. This is the case for both signatures considered in this example in Figure 5. Moreover, the analysis of distribution of the scoring metrics should show normality if the dataset is expected to represent a complete range of samples, or multi-modality if multiple subgroups are thought to be present.

Global analysis of metrics and their significance (Steps 10-11):

In order to effectively compare signatures across a range of metrics, we designed the radar plot depicted in Figure 6, to show, on a scaled plot, the various means of comparing statistical properties of gene signatures across datasets. [Pseudocode describing the computation of each of these metrics is provided within Supplementary Section S3](#). As can be seen, over nearly all metrics, the metastasis gene signature outperforms the random gene signature, as might be expected. However, to fully appreciate the magnitude of these differences, an understanding of the null distribution of each of these metrics is required, which is shown in Figure 7, from Step 11. This shows that over nearly every metric, the highly-performing gene signature for breast cancer metastasis is highly significant, whereas the random gene signature does not show significance across many of the metrics considered, thereby facilitating the rapid identification of a highly-performing versus a poorly-performing gene signature.

Box 1: Example evaluation of signature translatability cross-platform

As an example to highlight the utility of *sigQC* in determining the translatability of a signature across different sequencing platforms, we consider here the outputs of the package in comparing cross-platform performance, encapsulating a very significant batch effect. In particular, we consider the same breast cancer metastasis signature¹⁸ taken from MSigDb, on each of an RNA-seq dataset (TCGA), and a microarray generated dataset (GEO Series GSE3494). An initial step is to generate a signature annotated for each of the platforms, which is done through the use of BioMart, enabling the conversion of gene symbols into Affymetrix U133A probe IDs for use with the microarray dataset. Subsequently, running *sigQC* on each of these datasets and converted signatures individually gives the underlying data needed to generate the following radar plots, which can be used, in conjunction with the plots of negative control, to determine both the differences and the significance of these differences of the metrics reported by the radar plot. In this example, we observe that there is a high concordance between the outputs of the radar plots in both case, as well as high significance of many of the metrics in both cases, suggesting that this signature is highly applicable cross-platforms.

Acknowledgements

This work was funded by Cancer Research UK (F.M.B., A.L.H., A.B., W-C.C., A.D.) and the Medical Research Council (T.M., E.D.). The authors are also grateful for the support of the Clarendon Fund to A.D.

Author contributions

F.M.B. conceived the idea and designed the study. A.D., A.B., W-C.C., J.S. and F.M.B. contributed statistics and data visualization. A.D. performed analyses. A.D., A.B. and W-C.C. wrote and debugged code. A.B. and F.M.B. supervised the implementation. All authors contributed application cases and interpretation of data. A.D. and F.M.B. wrote the manuscript with contribution from all other authors.

Competing financial interests

The authors declare no competing financial interests.

Figures

Figure 1:

Title: [sigQC protocol flowchart](#)

Legend: Flowchart of steps involved in the proposed *sigQC* protocol and sample output plots produced by the *sigQC* R package. Example outputs for a metastasis signature¹⁸ on a clinical breast cancer dataset from the Cancer Genome Atlas Project^{19, 20} are shown.

[These examples are shown in greater detail in the ensuing figures.](#)

Figure 2:

Title: [Evaluation of the signature gene expression and its variability](#)

Legend: Expression of signature gene expression and variability across datasets for RNA-seq breast cancer for the metastasis signature (left) and a random gene signature (right).

[Horizontal axis represents the mean expression value for the gene and the vertical axis represents the standard deviation for the gene. Gray dots are representative of non-signature genes in the dataset, and red dots are representative of the signature genes in the dataset. Dotted lines are indicative of the 10th, 25th, 50th, 75th, and 90th percentile intervals along the horizontal and vertical axes. In general, it can be observed that the genes of the metastasis signature are more highly expressed and variable than those of the random gene signature, supporting that this signature may have greater clinical utility.](#)

Figure 3:

Title: [Comparison of scoring metrics summarising signature gene expression](#)

Legend: Comparison of scoring metrics for a metastasis gene signature (left) and a set of random genes (right) in the TCGA breast cancer RNA-seq dataset. [From top to bottom, the plots show correlation of mean and median scores, correlation of PCA1 and mean scores, correlation of PCA1 and median scores, a PCA scree plot, and a heatmap showing the correlations between all six gene signature scoring metrics considered \(mean, median, PCA1, ssGSEA, GSVA, and PLAGE\). Solid red line on the scatterplots of the first three rows are representative of the distribution of scores using the median \(first row\), mean \(second row\), and PCA1 \(third row\), across all samples of the dataset. Scoring metric correlations are greater for the metastasis signature than the random signature, suggesting the metastasis signature is a better choice for a gene signature in this dataset, as it can be summarised in a consistent manner. The PCA scree plot \(4th row\) shows the proportion of variance carried by each principal component, and for the metastasis signature carries a higher proportion than for the random signature, suggesting that indeed, the first principal component is a faithful representative of the expression values of the metastasis signature, more so than the random gene signature, in this dataset.](#)

Figure 4:

Title: [Analysis of correlation between signature genes \(intra-signature correlation\)](#)

Legend: Intra-signature correlation of signature genes across datasets for a metastasis gene signature (top left) and a random gene signature (top right), with heatmaps represented in density plot form (bottom) for the TCGA breast cancer RNA-seq dataset.

[The heatmaps and comparison density plots show that, in general, the genes of the metastasis signature have a higher intra-signature correlation than that of the random](#)

gene signature. In other words, the metastasis gene signature behaves more like a metagene in this dataset, as compared to the random gene signature.

Figure 5:

Title: Searching for signature structure

Legend: Hierarchical clustering of signature gene expression for the metastasis signature (left) and the random signature (right) over the TCGA breast cancer dataset. This groups together those samples (columns) with similar expression across the signatures' genes, and it groups together those genes (rows) with similar expression values in the different samples. SigQC uses the *hclust* function in R with the default parameters, namely Euclidean distance as similarity metric and the complete linkage method for the construction of the tree. Of note, expression can be considered as standardized for each gene, or as absolute values, as shown here. The former emphasises the relative changes in expression between samples, the latter allows to quickly identify clusters of genes characterized by similarly high or low expression across sample. In the example, the random signature reveals a large cluster of genes with very low signal, as expected due to the random sampling, whilst the metastasis signature only shows one gene expressed at these levels, confirming its suitability to capture the dataset expression signal. Neither signature shows different behaviour among different subsets of the dataset.

Figure 6:

Title: Summary radar plot for quality control metrics

Legend: Radar plot showing summary of gene signature quality control metrics for a metastasis signature (solid line) and a random set of genes (dashed line), as calculated for the TCGA breast cancer RNA-seq dataset. Among nearly all of the 14 metrics considered, the metastasis signature outperforms the random gene signature, and encompasses a much greater area proportion of the radar plot (0.39 vs. 0.16), confirming it is a better quality signature for this dataset. Each of the metrics plotted is described in detail in Table S1.

Figure 7:

Title: Analysis of statistical significance of quality control metrics

Legend: Box and scatter plots depicting the null distributions of each of the metrics measured on the radar plot for the metastasis signature (left) and a random signature (right), for $N = 50$ resampling runs using random gene signatures of the same length. For each of the 14 metrics considered, random resampling of gene signatures of the same length from the TCGA breast cancer dataset used were considered. From each of these resamplings, the metrics were re-computed, and their values are shown by the clear circles, with representative boxplots for their distributions overlaid. The red points indicate the values for the gene signatures under consideration, thereby providing context for the significance for the value of each of these metrics. Among nearly all metrics, the metastasis gene signature is highly significant, whereas the random gene signature is not, suggesting the better performance of the metastasis signature.

Tables

Table 1:

Title: sigQC input variables

Legend: Description of input variables to sigQC function make_all_plots().

Variable name	Default value	Description
gene_sigs_list	None	A list of gene signature matrices, representing the gene signatures to be tested.
mRNA_expr_matrix ^{[L][SEP]}	None	A list of expression matrices, one for each dataset.
names_sigs	NULL	The names of the gene signatures (e.g. Hypoxia,Invasiveness), one name per each signature in gene_sigs_list ^{[L][SEP]}
names_datasets	NULL	The names of the different datasets contained in mRNA_expr_matrix ^{[L][SEP]}
covariates	NULL	A list containing a sub-list of ‘annotations’ and ‘colours’ which contains the annotation matrix for the given dataset and the associated colours with which to plot in the expression heatmap ^{[L][SEP]}
thresholds	NULL	A list of thresholds to be considered for each dataset, default is median of the dataset. A gene is considered expressed if above the threshold, non-expressed otherwise. One threshold per dataset, in the same order as the dataset list.
out_dir ^{[L][SEP]}	tempdir()	A path to the directory where the resulting output files are written ^{[L][SEP]} .
showResults	TRUE	Tells if open dialog boxes showing the computed results. Default is TRUE ^{[L][SEP]}

origin	NULL	Tells if datasets have come from different labs/experiments/machines. Is a vector of characters, with same character representing same origin. Default is assumption that all datasets come from the same source.
doNegativeControl	TRUE	Logical, tells the function if negative and permutation controls must be computed.
numResampling	50	Integer for the number of resamplings while computing negative and permutation controls.

Data availability statement:

All data that has been used in this publication has been made available through Zenodo, available at: <https://doi.org/10.5281/zenodo.1319848>, DOI: 10.5281/zenodo.1319848.

Code availability statement:

All code that comprises the sigQC R package is available for download and installation from the CRAN repository at: <https://CRAN.R-project.org/package=sigQC>.

All scripts used to create the figures within this manuscript can be downloaded through Zenodo, available at <https://doi.org/10.5281/zenodo.1319848>, DOI 10.5281/zenodo.1319848.

References

- [1] Marc J Van De Vijver, Yudong D He, Laura J Van't Veer, Hongyue Dai, Augustinus AM Hart, Dorien W Voskuil, George J Schreiber, Johannes L Peterse, Chris Roberts, Matthew J Marton, et al. A gene-expression signature as a predictor of survival in breast cancer. *New England Journal of Medicine*, 347(25):1999–2009, 2002. ^[1]_[SEP]
- [2] Rui Liu, Xinhao Wang, Grace Y Chen, Piero Dalerba, Austin Gurney, Timothy Hoey, Gavin Sherlock, John Lewicki, Kerby Shedden, and Michael F Clarke. The prognostic role of a gene signature from tumorigenic breast-cancer cells. *New England Journal of Medicine*, 356(3):217–226, 2007. ^[1]_[SEP]
- [3] Lauren Averett Byers, Lixia Diao, Jing Wang, Pierre Saintigny, Luc Girard, Michael Peyton, Li Shen, Youhong Fan, Uma Giri, Praveen K Tumula, et al. An epithelial–mesenchymal transition gene signature predicts resistance to EGFR and PI3K inhibitors and identifies Axl as a therapeutic target for overcoming EGFR inhibitor resistance. *Clinical Cancer Research*, 19(1):279–290, 2013. ^[1]_[SEP]
- [4] Arthur Liberzon, Chet Birger, Helga Thorvaldsdóttir, Mahmoud Ghandi, Jill P Mesirov, and Pablo Tamayo. The molecular signatures database hallmark gene set collection. *Cell Systems*, 1(6):417–425, 2015. ^[1]_[SEP]
- [5] Ludmil B Alexandrov, Serena Nik-Zainal, David C Wedge, Peter J Campbell, and Michael R Stratton. Deciphering signatures of mutational processes operative in human cancer. *Cell Reports*, 3(1):246–259, 2013. ^[1]_[SEP]
- [6] Rashmi Kanagal-Shamanna, Bryce P Portier, Rajesh R Singh, Mark J Routbort, Kenneth D Aldape, Brian A Handal, Hamed Rahimi, Neelima G Reddy, Bedia A Barkoh, Bal M Mishra, et al. Next-generation sequencing-based multi-gene mutation profiling of solid tumors using fine needle aspiration samples: promises and challenges for routine clinical diagnostics. *Modern Pathology*, 27(2):314, 2014. ^[1]_[SEP]
- [7] Margaret A Shipp, Ken N Ross, Pablo Tamayo, Andrew P Weng, Jeffery L Kutok, Ricardo CT Aguiar, Michelle Gaasenbeek, Michael Angelo, Michael Reich, Geraldine S Pinkus, et al. Diffuse large B-cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning. *Nature Medicine*, 8(1):68–74, 2002. ^[1]_[SEP]
- [8] Aiguo Li, Jennifer Walling, Susie Ahn, Yuri Kotliarov, Qin Su, Martha Quezado, J Carl Oberholtzer, John Park, Jean C Zenklusen, and Howard A Fine. Unsupervised analysis of transcriptomic profiles reveals six glioma subtypes. *Cancer Research*, 69(5):2091–2099, 2009. ^[1]_[SEP]
- [9] FM Buffa, AL Harris, CM West, and CJ Miller. Large meta-analysis of multiple cancers reveals a ^[1]_[SEP] common, compact and highly prognostic hypoxia metagene. *British Journal of Cancer*, 102(2):428–435, 2010.

- [10] Stuart C Winter, Francesca M Buffa, Priyamal Silva, Crispin Miller, Helen R Valentine, Helen Turley, Ketan A Shah, Graham J Cox, Rogan J Corbridge, Jarrod J Homer, et al. Relation of a hypoxia metagene derived from head and neck cancer to prognosis of multiple cancers. *Cancer Research*, 67(7):3441–3449, 2007. ^[1]_{SEP}
- [11] Konstantina Kourou, Themis P Exarchos, Konstantinos P Exarchos, Michalis V Karamouzis, and Dimitrios I Fotiadis. Machine learning applications in cancer prognosis and prediction. *Computational and Structural Biotechnology Journal*, 13:8–17, 2015. ^[1]_{SEP}
- [12] David Venet, Jacques E Dumont, and Vincent Detours. Most random gene expression signatures are significantly associated with breast cancer outcome. *PLoS Computational Biology*, 7(10):e1002240, 2011. ^[1]_{SEP}
- [13] Massimo Masiero, Filipa Costa Simões, Hee Dong Han, Cameron Snell, Tessa Peterkin, Esther Bridges, Lingegowda S Mangala, Sherry Yen-Yao Wu, Sunila Pradeep, Demin Li, et al. A core human primary tumor angiogenesis signature identifies the endothelial orphan receptor ELTD1 as a key regulator of angiogenesis. *Cancer Cell*, 24(2):229–241, 2013. ^[1]_{SEP}
- [14] B H L Harris, A Barberis, Catharine M L West, and F M Buffa. Gene expression signatures as biomarkers of tumour hypoxia. *Clinical Oncology*, 27(10):547–560, 2015. ^[1]_{SEP}
- [15] Andrew Dhawan, Jacob G Scott, Adrian L Harris, and Francesca M Buffa. Pan-cancer characterisation of microRNA with hallmarks of cancer reveals role of microRNA-mediated downregulation of tumour suppressor genes. *bioRxiv*, page 238675, 2018. ^[1]_{SEP}
- [16] Almut Schulze and Julian Downward. Navigating gene expression using microarrays-a technology review. *Nature Cell Biology*, 3(8):E190–E195, 2001. ^[1]_{SEP}
- [17] Zhong Wang, Mark Gerstein, and Michael Snyder. RNA-Seq: a revolutionary tool for transcriptomics. *Nature Reviews Genetics*, 10(1):57–63, 2009. ^[1]_{SEP}
- [18] Laura J Van’t Veer, Hongyue Dai, Marc J Van De Vijver, Yudong D He, Augustinus A M Hart, Mao Mao, Hans L Peterse, Karin Van Der Kooy, Matthew J Marton, Anke T Witteveen, et al. Gene expression profiling predicts clinical outcome of breast cancer. *Nature*, 415(6871):530–536, 2002. ^[1]_{SEP}
- [19] Cancer Genome Atlas Network et al. Comprehensive molecular portraits of human breast tumours. *Nature*, 490(7418):61–70, 2012. ^[1]_{SEP}
- [20] Analysis-ready standardized TCGA data from Broad GDAC Firehose 2016_01_28 run. *Broad Institute TCGA Genome Data Analysis Center*. ^[1]_{SEP}

- [21] Durinck, Steffen and Moreau, Yves and Kasprzyk, Arek and Davis, Sean and De Moor, Bart and Brazma, Alvis and Huber, Wolfgang. BioMart and Bioconductor: a powerful link between biological databases and microarray data analysis. *Bioinformatics* 21(16):3439–3440, 2005.
- [22] A Subramanian, P Tamayo, VK Mootha, Sayan Mukherjee, BL Ebert, MA Gillette, A Paulovich, SL Pomeroy, TR Golub, ES Lander, and JP Mesirov. Gene set enrichment analysis: A knowledge- based approach for interpreting genome-wide expression profiles. *PNAS*, 102(43):15545–15550, 2005. ^[L]_[SEP]
- [23] Sonja Hänzelmann, Robert Castelo, and Justin Guinney. Gsva: gene set variation analysis for microarray and rna-seq data. *BMC Bioinformatics*, 14(1):7, 2013. ^[L]_[SEP]
- [24] David A Barbie, Pablo Tamayo, Jesse S Boehm, So Young Kim, Susan E Moody, Ian F Dunn, Anna C Schinzel, Peter Sandy, Etienne Meylan, Claudia Scholl, et al. Systematic rna interference reveals that oncogenic kras-driven cancers require tbk1. *Nature*, 462(7269):108, 2009. ^[L]_[SEP]
- [25] John Tomfohr, Jun Lu, and Thomas B Kepler. Pathway level analysis of gene expression using singular value decomposition. *BMC Bioinformatics*, 6(1):225, 2005. ^[L]_[SEP]
- [26] Bradley Efron and Robert Tibshirani. On testing the significance of sets of genes. *The Annals of Applied Statistics*, pages 107–129, 2007. ^[L]_[SEP]
- [27] Anders E Berglund, Eric A Welsh, and Steven A Eschrich. Characteristics and Validation Techniques for PCA-Based Gene-Expression Signatures. *International Journal of Genomics*, 2017, 2017. ^[L]_[SEP]
- [28] Natalie S Fox, Maud H W Starmans, Syed Haider, Philippe Lambin, and Paul C Boutros. Ensemble analyses improve signatures of tumour hypoxia and reveal inter-platform differences. *BMC Bioinformatics*, 15(1):170, 2014. ^[L]_[SEP]
- [29] Steen Knudsen, Thomas Jensen, Anker Hansen, Wiktor Mazin, Justin Lindemann, Irene Kuter, Naomi Laing, and Elizabeth Anderson. Development and validation of a gene expression score that predicts response to fulvestrant in breast cancer patients. *PLoS One*, 9(2):e87415, 2014. ^[L]_[SEP]
- [30] Hung-I Harry Chen, Tzu-Hung Hsiao, Yidong Chen, and Charles Keller. S-score: A novel scoring method of gene signatures for molecular classification. In *Genomic Signal Processing and Statistics (GENSIPS), 2011 IEEE International Workshop on*, pages 154–157. IEEE, 2011. ^[L]_[SEP]
- [31] Tzu-Hung Hsiao, Hung-I Harry Chen, Jo-Yang Lu, Pei-Ying Lin, Charles Keller, Sarah Comerford, Gail E Tomlinson, and Yidong Chen. Utilizing signature-score to identify oncogenic pathways of cholangiocarcinoma. *Translational Cancer Research*,

2(1):6, 2013. ^[1]_[SEP]

[32] Hiromichi Ebi, Shuta Tomida, Toshiyuki Takeuchi, Chinatsu Arima, Takahiko Sato, Tetsuya Mitsudomi, Yasushi Yatabe, Hirotaka Osada, and Takashi Takahashi. Relationship of deregulated signaling converging onto mTOR with prognosis and classification of lung adenocarcinoma shown by two independent in silico analyses. *Cancer Research*, 69(9):4027–4035, 2009. ^[1]_[SEP]

[33] Don L Gibbons, Wei Lin, Chad J Creighton, Shuling Zheng, Dror Berel, Yanan Yang, Maria Gabriela Raso, Diane D Liu, Ignacio I Wistuba, Guillermina Lozano, et al. Expression signatures of metastatic capacity in a genetic mouse model of lung adenocarcinoma. *PloS One*, 4(4):e5401, 2009. ^[1]_[SEP]

Supplementary information

S1: Radar plot metrics

In Table S1 we provide a description of the metrics plotted on the arms of the radar plot.

Table S1: Description of metrics defining components of summary radar plot.

Metric Abbreviation	Metric Description	Metric Calculation
Relative Med. SD	Relative median standard deviation of signature genes as compared median standard deviation of all genes.	Consider the standard deviation of all signature elements' expression across all samples, then consider the median of this list, α . Similarly consider the median of the standard deviation of all reported genes across all samples, β . Value considered is $ \alpha/(\alpha+\beta) $, where $ \cdot $ represents the absolute value. ^{[1][SEP]}
$\rho_{\text{Med.}, \text{Z-Med.}}$	Absolute correlation coefficient of median of signature genes and median of signature genes on z- transformed dataset.	Absolute value of Spearman correlation coefficient between median and z- median of signature elements, used as scoring metrics across samples.
$\rho_{\text{Mean}, \text{PCA1}}$	Absolute correlation coefficient of mean and first principal component of signature genes.	Absolute value of Spearman correlation coefficient between mean and first principal component of signature elements, used as scoring metrics across samples. ^{[1][SEP]}
$\rho_{\text{PCA1}, \text{Med.}}$	Absolute correlation coefficient of first principal component and median of signature genes.	Absolute value of Spearman correlation coefficient between first principal component and z-median of signature elements, used as scoring metrics across samples.
$\rho_{\text{Mean}, \text{Med.}}$	Absolute correlation coefficient of mean and median of signature genes. ^{[1][SEP]}	Absolute value of Spearman correlation coefficient between mean and median of signature elements, used as scoring metrics across samples.
Intra-sig. Corr.	Median of intra-signature correlation	Median of list of all correlation coefficients for each signature element

	values for all signature genes.	with every other signature element. ^{[1][SEP]}
Prop. Expressed	Median proportion of samples expressing signature genes above threshold.	Median value of list of proportions of samples expressing each signature element above threshold for each signature element. Threshold is defined as median of expression of all genes, if not user-specified.
Non-NA Prop.	Median over all samples expressing each element as non-NA. ^{[1][SEP]}	Median value of list of proportions of samples which have expression not recorded as NA, for each signature element.
Coef. Of Var.	Median coefficient of variation of all signature genes, relative to the median coefficient of variation of all genes.	Consider the coefficient of variation of all signature elements across all samples, then consider the median of this list, α . Similarly consider the median of the coefficient of variation of all reported genes across all samples, β . Value considered is $ \alpha/(\alpha+\beta) $, where $ \cdot $ represents the absolute value.
$\sigma_{\geq 50\%}$	Proportion of signature genes in the top 50% of all varying genes.	This is the proportion of signature elements that have coefficients of variation in the top 50% of all coefficients of variation for all genes.
$\sigma_{\geq 25\%}$	Proportion of signature genes in the top 25% of all varying genes.	This is the proportion of signature elements that have coefficients of variation in the top 25% of all coefficients of variation for all genes.
$\sigma_{\geq 10\%}$	Proportion of signature genes in the top 10% of all varying genes.	This is the proportion of signature elements that have coefficients of variation in the top 10% of all coefficients of variation for all genes. ^{[1][SEP]}
Skewness	Relative skew of distribution of signature gene expression over all	Consider the skewness of the distribution for the mean expression of all signature elements across all samples, α . Similarly consider the

	samples compared with skewness of overall expression distribution for all genes.	skewness of the distribution for the mean expression of all genes across all samples, β . Value considered is $ \alpha /(\alpha + \beta)$, where $ \cdot $ represents the absolute value.
σ_{PCA1}	Proportion of gene signature score taken by median, by first principal component.	This is the proportion of the variance of gene signature score that is explained by the first principal component of the expression of the signature genes taken across all samples.

S2: Supplementary figures:

Figure S1:

Title: Measures of expression of signature genes across TCGA breast cancer dataset

Legend: Expression of signature genes across the TCGA breast cancer RNA-seq dataset for the metastasis gene signature (top) and a random set of genes (bottom), shown as (a) a barplot for the proportion of samples expressing a gene above the median, (b) a density plot showing the same information as the barplots in (a), and (c) a plot of the proportion of samples showing NA expression for each of the genes of the signature.

Figure S2:

Title: Assessment of standardisation of dataset values on gene signature score

Legend: Comparison of median and z-transformed median of signature gene expression across the RNA-seq breast cancer dataset for the metastasis gene signature (left) and the random set of genes (right).

S3: Pseudocode for radar plot metrics

We define an m -dimensional array, $e = [e_1, \dots, e_m]$ as the gene expression data relative to a single sample, such that e_k is the expression value of gene k in the given sample. In this way, we may define the full dataset as the bi-dimensional matrix $E = [e_1, \dots, e_n]$, where n is the number of samples and e_{ij} is the expression value of gene i in the j -th sample.

Similarly, we denote by $E = [e_1, \dots, e_m]^t$ the same matrix, where e_k is an n -dimensional array containing the expression data of a single gene across all n samples and $(.)^t$ indicates the transpose of a matrix. Finally, we denote by $R = [r_1, \dots, r_n]$ the reduced gene expression matrix containing only the expression of the genes included in the assessed signature so that $r_k = [r_1, \dots, r_l]$, where $1 \leq m$.

Ratio of Med. SD

1. Compute the standard deviation (σ_1) of each signature gene across all samples
Denote by α the median of the standard deviations
2. For every gene, compute the standard deviation (σ_2) across all samples
3. Denote by β the median of the standard deviations
4. Return the absolute value of $\alpha/(\alpha + \beta)$

Pseudocode

```
 $R = [r_1, \dots, r_n] = [r_1, \dots, r_l]^t$   
 $\sigma_1 = [\sigma_{11}, \dots, \sigma_{1l}] = l\text{-dim array}$   
 $\sigma_2 = [\sigma_{21}, \dots, \sigma_{2n}] = m\text{-dim array}$   
for  $i = 1; i \leq l; i = i + 1$  {  
   $\sigma_1(i) = \text{standard deviation}(r_i)$   
}  
  
for  $j = 1; j \leq m; j = j + 1$  {  
   $\sigma_2(j) = \text{standard deviation}(e_j)$   
}  
  
 $\alpha = \text{median}(\sigma_1)$   
 $\beta = \text{median}(\sigma_2)$   
return  $|\alpha / (\alpha + \beta)|$ 
```

Med., Z-Med. Score Cor.

1. Compute the median of each signature gene across all samples
2. Normalise the input matrix using the z score
3. Compute the median of each signature gene in the normalised matrix across all samples
4. Compute the Spearman correlation between the 2 median arrays
5. Return the absolute value of the Spearman correlation coefficient

Pseudocode

```
 $R = [r_1, \dots, r_n] = [r_1, \dots, r_l]^t$   
 $med, med_z, \mu, \sigma = l\text{-dim arrays}$   
 $Z = [z_1, \dots, z_n] = [z_1, \dots, z_l]^t = \text{normalised matrix}$   
for  $i = 1; i \leq l; i = i + 1$  {  
     $med(i) = \text{median}(r_i)$   
     $\mu(i) = \text{mean}(r_i)$   
     $\sigma(i) = \text{standard deviation}(r_i)$   
}  
  
for  $i = 1; i \leq l; i = i + 1$  {  
    for  $j = 1; j \leq n; j = j + 1$  {  
         $Z(i,j) = (r_{ij} - \mu(i)) / \sigma(i)$   
    }  
}  
  
for  $i = 1; i \leq l; i = i + 1$  {  
     $med_z(i) = \text{median}(z_i)$   
}  
 $\rho = \text{correlation}(med, med_z)$   
return  $|\rho|$ 
```

Mean, PCA1 Score Cor.

1. Compute the mean of each signature gene across all samples
2. Compute the first principal component (PCA1) of each signature gene across all samples
3. Compute the Spearman correlation between the mean and PCA1 arrays
4. Return the absolute value of the Spearman correlation coefficient

Pseudocode

```
 $R = [r_1, \dots, r_n] = [r_1, \dots, r_l]^t$   
 $\mu, PCA1 = l\text{-dim arrays}$   
  
for  $i = 1; i \leq l; i = i + 1$  {  
     $\mu(i) = \text{mean}(r_i)$ 
```



```

    PCA1(i) = first principal component( $r_i$ )
}
 $\rho$  = correlation( $\mu$ , PCA1)
return | $\rho$  |

```

PCA1, Z-Med. Score Cor.

1. Compute the first principal component (PCA1) of each gene across all samples
2. Normalise the input matrix using the z score
3. Compute the median for each signature gene across all samples in the normalised matrix
4. Compute the Spearman correlation between the PCA1 and median arrays
5. Return the absolute value of the Spearman correlation coefficient

Pseudocode

```

 $R = [r_1, \dots, r_n] = [r_1, \dots, r]^\text{t}$ 
PCA1, medz,  $\mu$ ,  $\sigma$  = 1-dim arrays
 $Z = [z_1, \dots, z_n] = [z_1, \dots, z]^\text{t}$  = normalised matrix

for  $i = 1$ ;  $i \leq l$ ;  $i = i + 1$  {
    PCA1(i) = first principal component( $r_i$ )
     $\mu(i) = \text{mean}(r_i)$ 
     $\sigma(i) = \text{standard deviation}(r_i)$ 
}

for  $i = 1$ ;  $i \leq l$ ;  $i = i + 1$  {
    for  $j = 1$ ;  $j \leq n$ ;  $j = j + 1$  {
         $Z(i,j) = (r_{ij} - \mu(i))/\sigma(i)$ 
    }
}

for  $i = 1$ ;  $i \leq l$ ;  $i = i + 1$  {
    medz(i) = median( $z_i$ )
}

 $\rho$  = correlation(PCA1, medz)
return | $\rho$  |

```

Mean, Med. Score Cor.

1. Compute the mean of the signature genes for each sample
2. Compute the median of the signature genes for each sample

3. Compute the Spearman correlation of the mean and median arrays
4. Return the absolute value of the Spearman correlation coefficient

Pseudocode

```

 $R = [r_1, \dots, r_n]$ 
 $\mu, med = n\text{-dim arrays}$ 
for  $j = 1; j \leq n; j = j + 1$  {
     $\mu = mean(r_j)$ 
     $med = median(r_j)$ 
}
 $\rho = correlation(\mu, med)$ 
return  $|\rho|$ 

```

Intra-sig. Corr.

1. Compute the intra-signature correlation of the reduced gene expression matrix
2. Return the absolute value of median of all correlations coefficients

Pseudocode

```

 $R = [r_1, \dots, r_n] = [r_1, \dots, r_l]^t$ 
 $A = l \times l \text{ matrix}$ 
for  $i = 1; i \leq l; i = i + 1$  {
    for  $j = 1; j \leq l; j = j + 1$  {
         $A(i,j) = correlation(r_i, r_j)$ 
    }
}
return  $|median(A)|$ 

```

Med. Prop. Expressed

1. Compute the median of the dataset
2. For each gene, check if expression is greater than median
3. For each gene, count the proportion over all samples
4. Return the median over the array of proportions

Pseudocode

```

 $E = [e_1, \dots, e_n], R = [r_1, \dots, r_n] = [r_1, \dots, r_l]^t$ 

 $prop = m\text{-dim array}$ 
 $med = median(E)$ 

```

```

C = l x n zeros matrix
for i = 1; i ≤ l; i = i + 1 {
    for j = 1; j ≤ n; j = j + 1{
        if (rij > med){
            C(i,j) = 1
        }
    }
}
prop(i) = count (C(i))/n
}
return median(prop)

```

Med. non-NA Prop

1. Count the number of times each gene in the signature is expressed over all samples
2. For each gene, compute the expression proportion over all samples
3. Return the median over the array of proportions

Pseudocode

```

R = [r1, ..., rn] = [r1, ..., r]t
prop = m-dim array
C = l x n zeros matrix
for i = 1; i ≤ l; i = i + 1 {
    for j = 1; j ≤ n; j = j + 1{
        if (rij ≠ NA){
            C (i,j) = 1
        }
    }
}
prop(i) = count (C(i))/n
}
return median(prop)

```

Coef. of Var. Ratio

1. Compute the standard deviation (σ) for each signature gene across all samples
2. Compute the mean (μ) for each gene across all samples
3. Compute the coefficient of variation ($c_{v1} = \sigma / \mu$) for each signature gene across all samples
4. Denote by α the median of the coefficients of variation
5. For each gene, compute the coefficient of variation (c_{v2}) across all signature genes
6. Denote by β the median of all c_{v2}
7. Return the absolute value of $\alpha / (\alpha + \beta)$

Pseudocode

```

 $E = [e_1, \dots, e_n] = [e_1, \dots, e_m]^t$ ,  $R = [r_1, \dots, r_n] = [r_1, \dots, r]^t$ 
 $c_{v1} = l\text{-dim arrays}$ 
 $c_{v2} = n\text{-dim arrays}$ 
for  $i = 1; i \leq l; i = i + 1$  {
     $c_{v1}(i) = \text{standard deviation}(r_i) / \text{mean}(r_i)$ 
}
 $\alpha = \text{median}(c_{v1})$ 
for  $j = 1; j \leq m; j = j + 1$  {
     $c_{v2}(j) = \text{standard deviation}(e_j) / \text{mean}(e_j)$ 
}
 $\beta = \text{median}(c_{v2})$ 
return  $|\alpha / (\alpha + \beta)|$ 

```

Prop in top 50% var.

1. Compute the standard deviation (σ) for each gene across all samples
2. Compute the mean (μ) for each gene across all samples
3. Compute the coefficient of variation ($c_v = \sigma / \mu$) for each gene across all samples
4. Rank the c_v
5. Return the proportion of signature genes with c_v in the top 50% of the rank

Pseudocode

```

 $E = [e_1, \dots, e_n] = [e_1, \dots, e_m]^t$ 
 $c_v = m\text{-dim arrays}$ 
 $c = l\text{-dim zero array}$ 
for  $i = 1; i \leq m; i = i + 1$  {
     $c_v(i) = \text{standard deviation}(e_i) / \text{mean}(e_i)$ 
}

```

```

 $q = \text{quantile}_{0.5}(c_v)$ 
for  $i = 1; i \leq l; i = i + 1$  {
    if ( $c_v(i) \geq q$ ) {
         $c(i) = 1$ 
    }
}
return count( $c$ )/ $l$ 

```

Prop in top 25% var.

1. Compute the standard deviation (σ) for each gene across all samples
2. Compute the mean (μ) for each gene across all samples
3. Compute the coefficient of variation ($c_v = \sigma/\mu$) for each gene across all samples
4. Rank the c_v
5. Return the proportion of signature genes with c_v in the top 25% of the rank

Pseudocode

```

 $E = [e_1, \dots, e_n] = [e_1, \dots, e_m]^t$ 
 $c_v = m\text{-dim arrays}$ 
 $c = l\text{-dim zero array}$ 
for  $i = 1; i \leq m; i = i + 1$  {
     $c_v(i) = \text{standard deviation}(e_i) / \text{mean}(e_i)$ 
}
 $q = \text{quantile}_{0.75}(c_v)$ 
for  $i = 1; i \leq l; i = i + 1$  {
    if ( $c_v(i) \geq q$ ) {
         $c(i) = 1$ 
    }
}
return count( $c$ )/ $l$ 

```

Prop in top 10% var.

1. Compute the standard deviation (σ) for each gene across all samples
2. Compute the mean (μ) for each gene across all samples
3. Compute the coefficient of variation ($c_v = \sigma/\mu$) for each gene across all samples
4. Rank the c_v
5. Return the proportion of signature genes with c_v in the top 10% of the rank

Pseudocode

```

 $E=[e_1, \dots, e_n]=[e_1, \dots, e_m]^t$ 
 $c_v = m\text{-dim arrays}$ 
 $c = l\text{-dim zero array}$ 
for  $i = 1; i \leq m; i = i + 1$  {
     $c_v(i) = \text{standard deviation}(e_i) / \text{mean}(e_i)$ 
}
 $q = \text{quantile}_{0.90}(c_v)$ 
for  $i = 1; i \leq l; i = i + 1$  {
    if ( $c_v(i) \geq q$ ) {
         $c(i) = 1$ 
    }
}
return  $\text{count}(c)/l$ 

```

Skew Ratio

1. Compute the skewness (α) of mean of each signature gene, across all samples
2. Compute the skewness (β) of mean of each gene, across all samples
3. Return $|\alpha| / (|\alpha| + |\beta|)$

Pseudocode

```

 $E=[e_1, \dots, e_n]=[e_1, \dots, e_m]^t$ ,  $R = [r_1, \dots, r_n] = [r_1, \dots, r]^t$ 
 $\mu = m\text{-dim array}$ 
for  $i = 1; i \leq m; i = i + 1$  {
     $\mu = \text{mean}(e_i)$ 
}
 $\alpha = \text{skewness}[\mu(r_i)]$ 
 $\beta = \text{skewness}[\mu(e_j)]$ 
return  $|\alpha| / (|\alpha| + |\beta|)$ 

```

Prop Var by PCA1

1. Compute the principal component of every signature gene across all samples
2. Return proportion of variance explained by first principal component

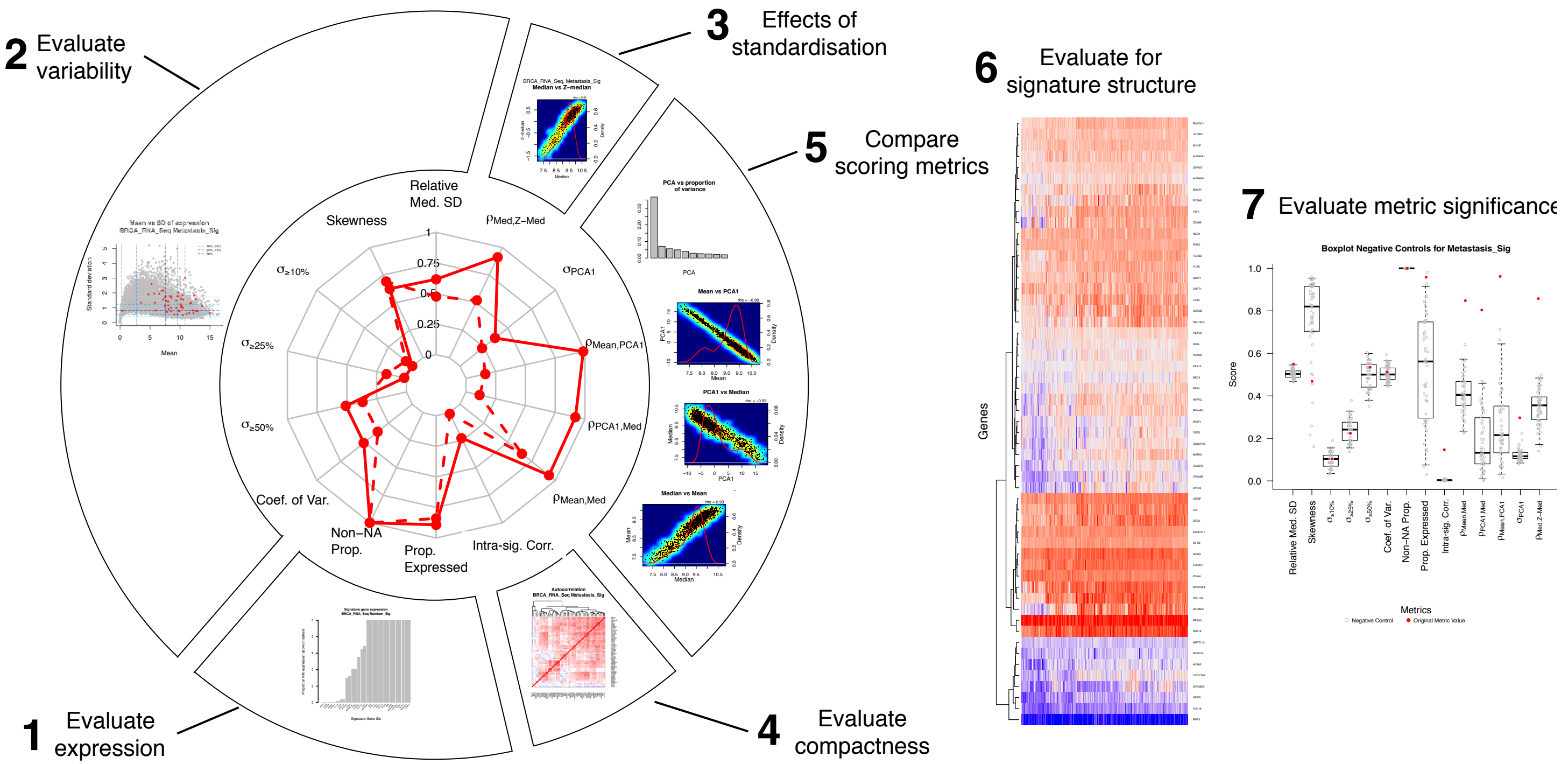
Pseudocode

```

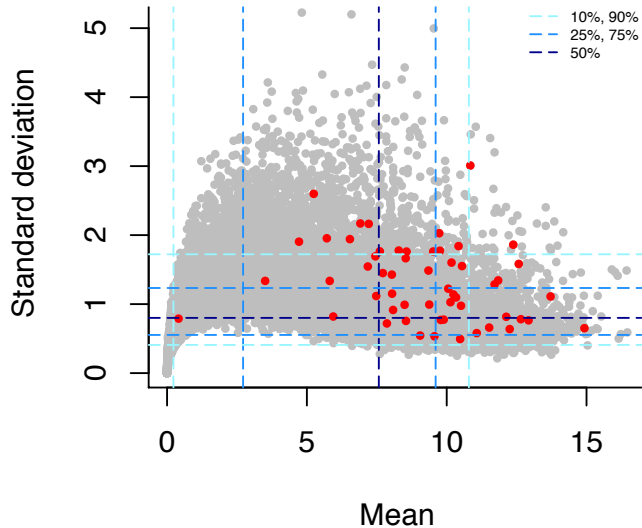
 $R = [r_1, \dots, r_n] = [r_1, \dots, r]^t$ 
PCA1 = 1-dim arrays

```

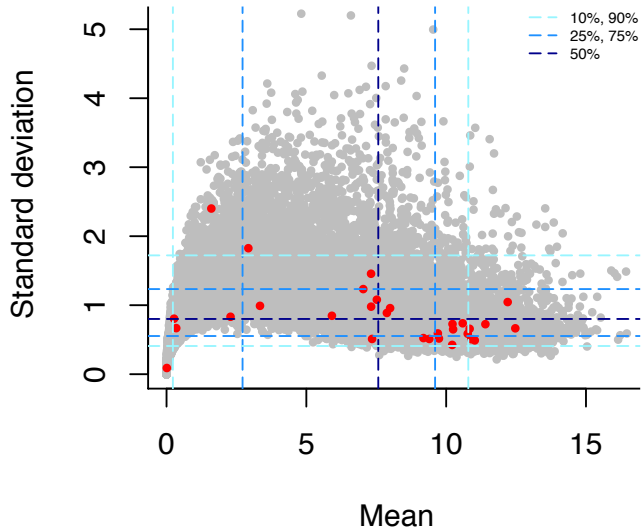
```
for  $i = 1; i \leq l; i = i + 1$  {  
     $PCA1(i) = \text{first principal component}(r_i)$   
}  
return variance_prop( $PCA1$ )
```



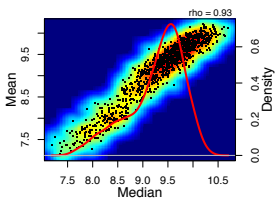
Mean vs SD of expression
BRCA_RNA_Seq Metastasis_Sig



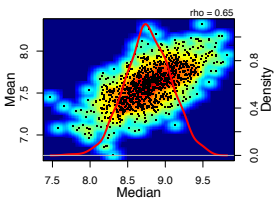
Mean vs SD of expression
BRCA_RNA_Seq Random_Sig



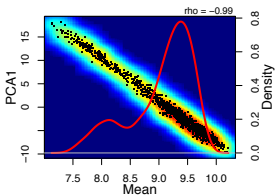
BRCA_RNA_Seq Metastasis_Sig
Median vs Mean



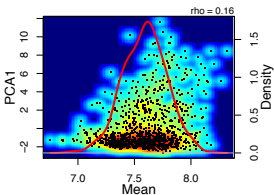
BRCA_RNA_Seq Random_Sig
Median vs Mean



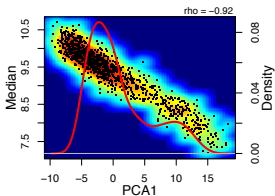
Mean vs PCA1



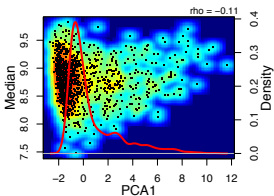
Mean vs PCA1



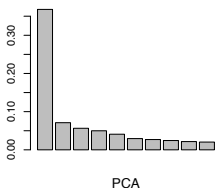
PCA1 vs Median



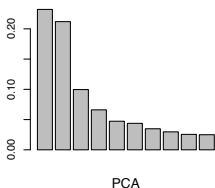
PCA1 vs Median



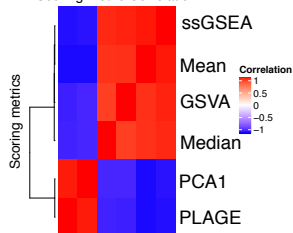
PCA vs proportion
of variance



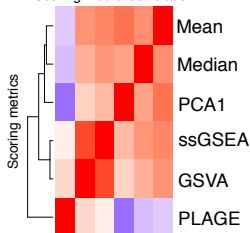
PCA vs proportion
of variance

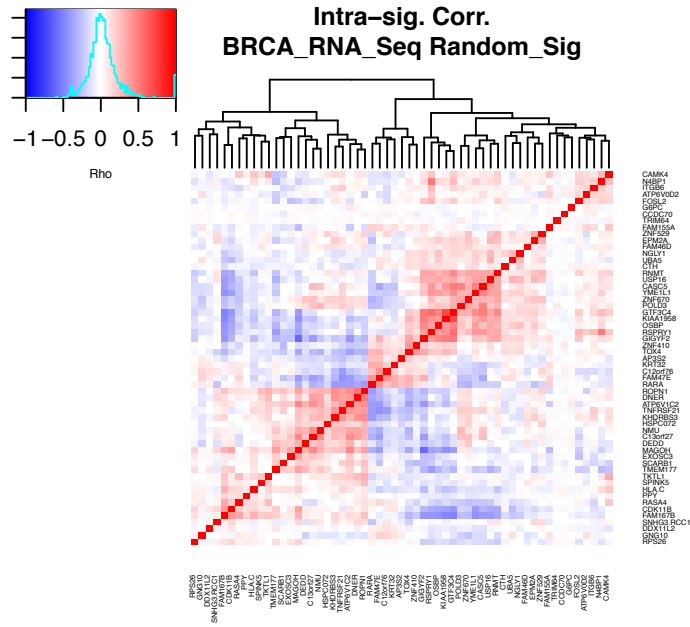
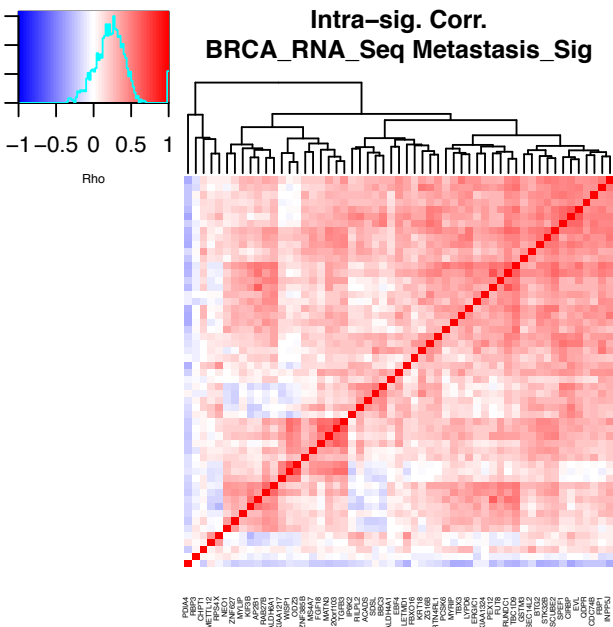


Scoring Metric Correlation

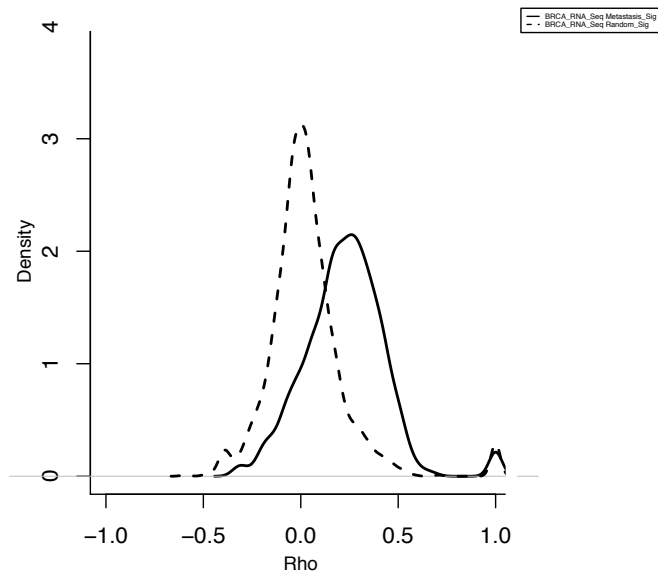


Scoring Metric Correlation

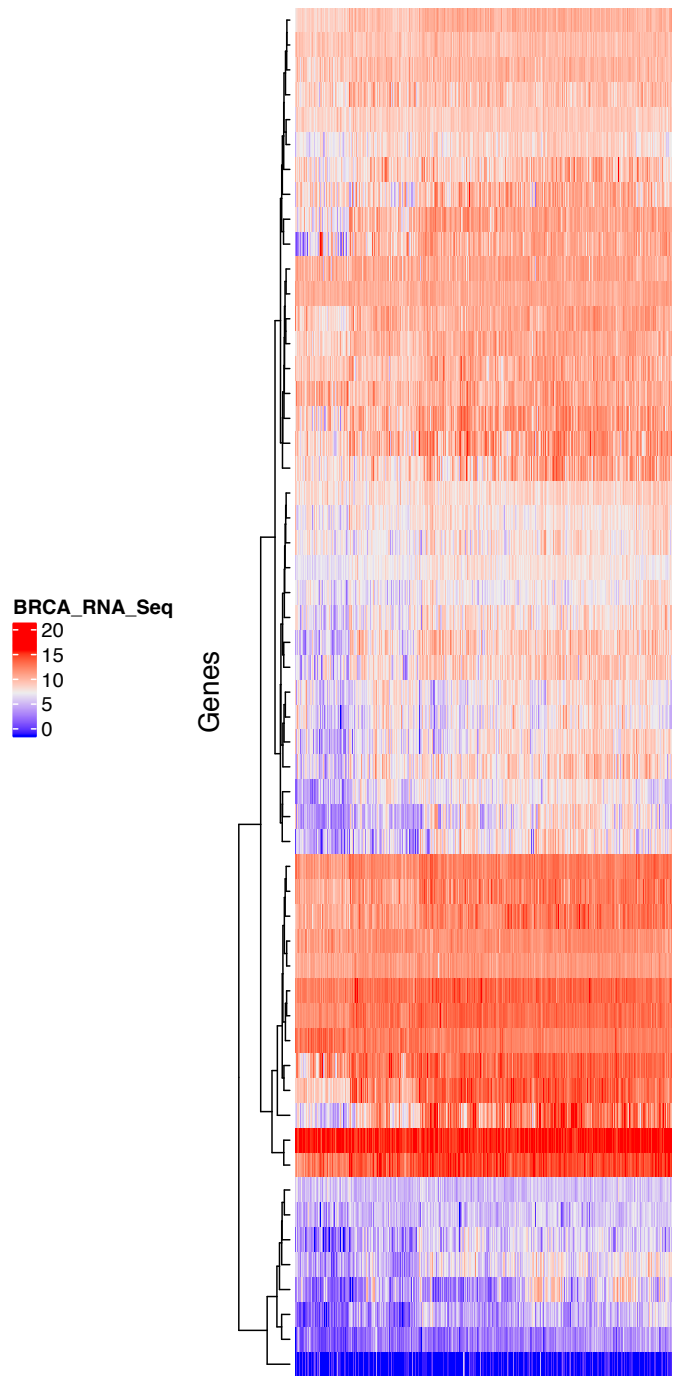




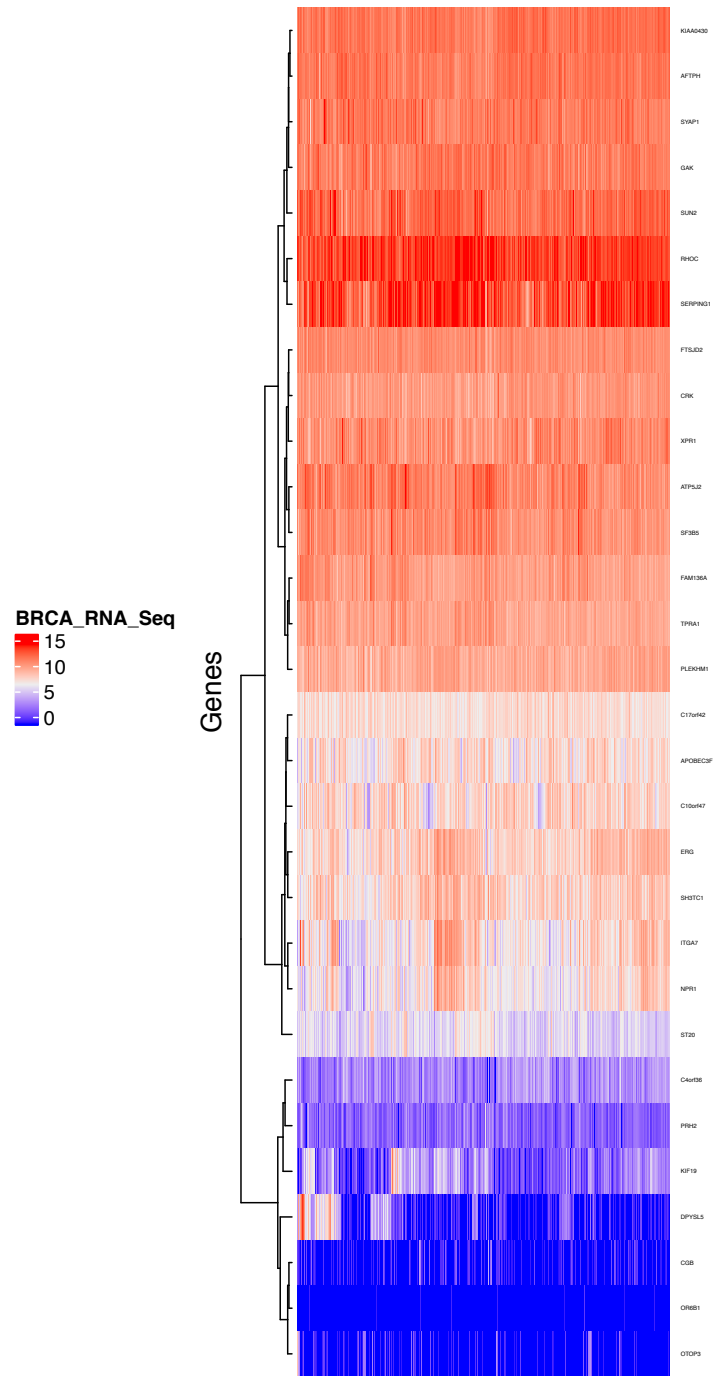
Intra-sig. Corr. Density

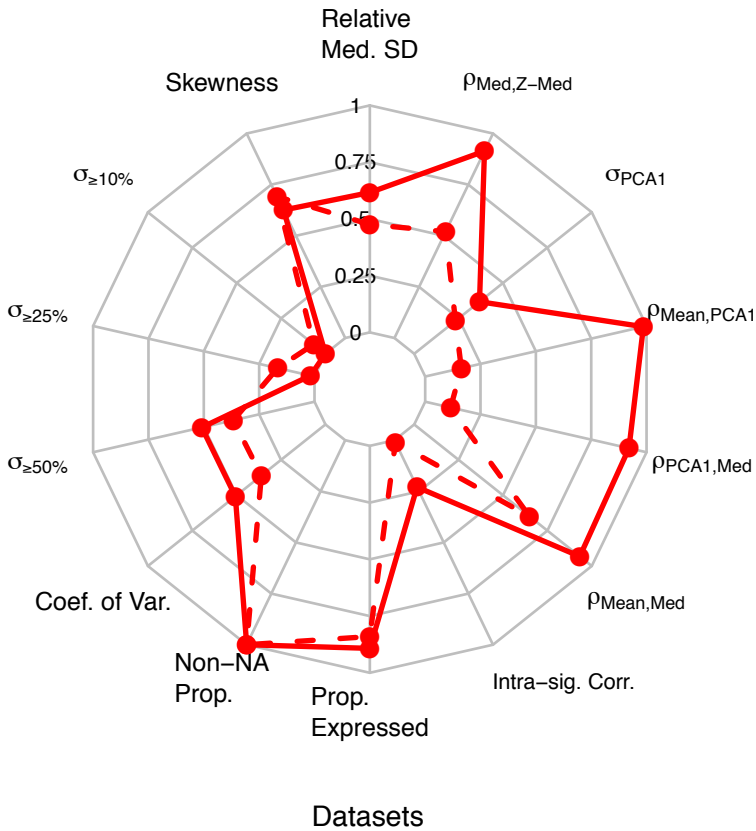


BRCA_RNA_Seq

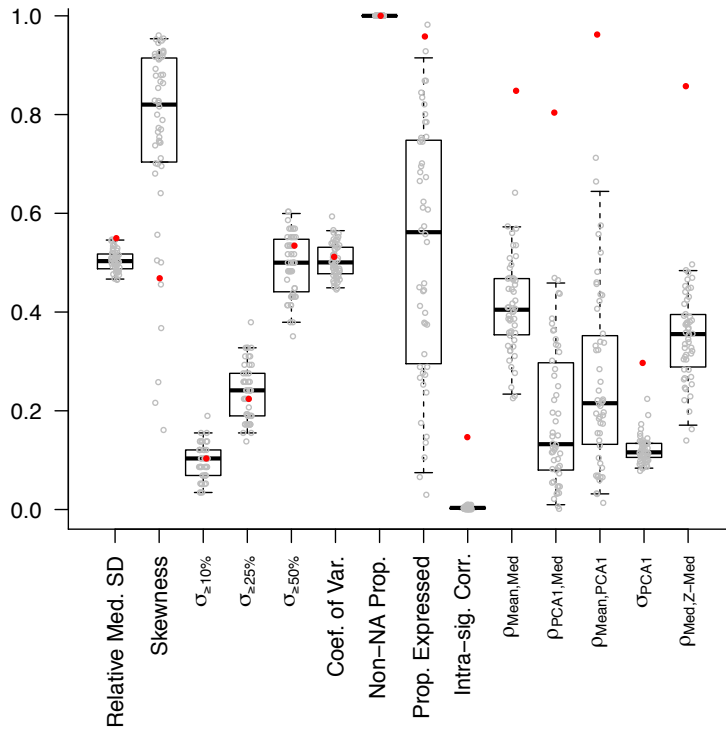


BRCA_RNA_Seq

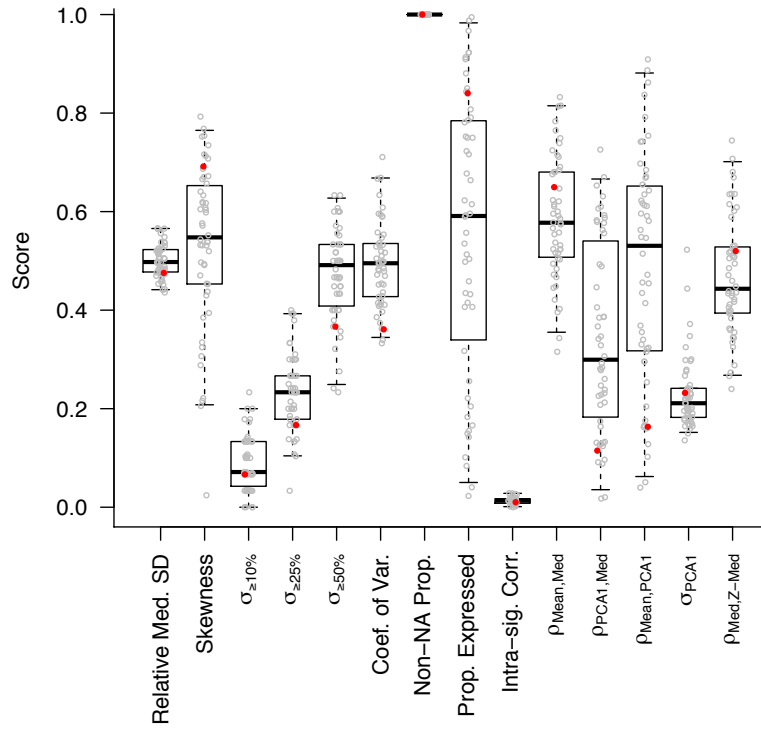


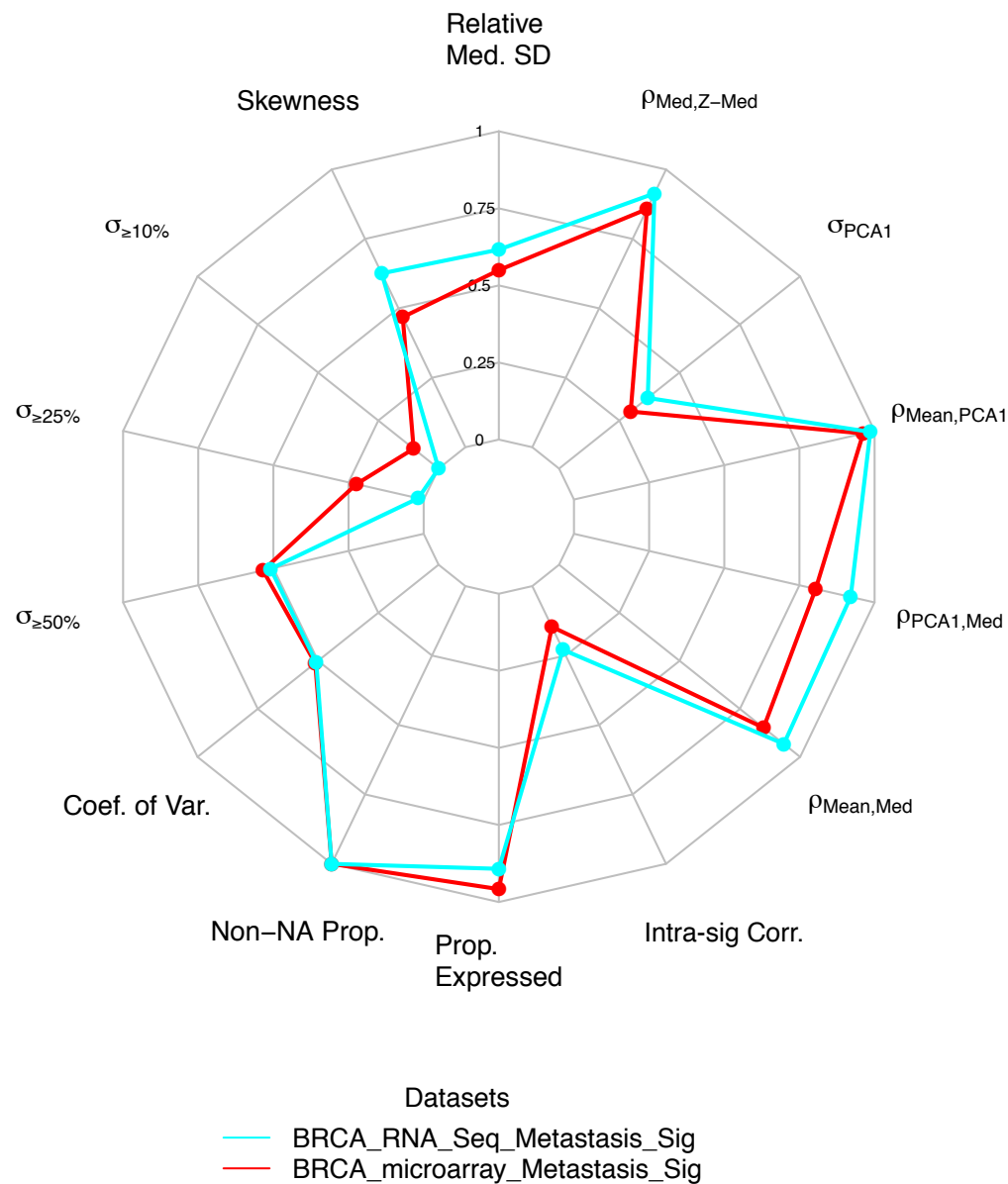


Boxplot Negative Controls for Metastasis_Sig

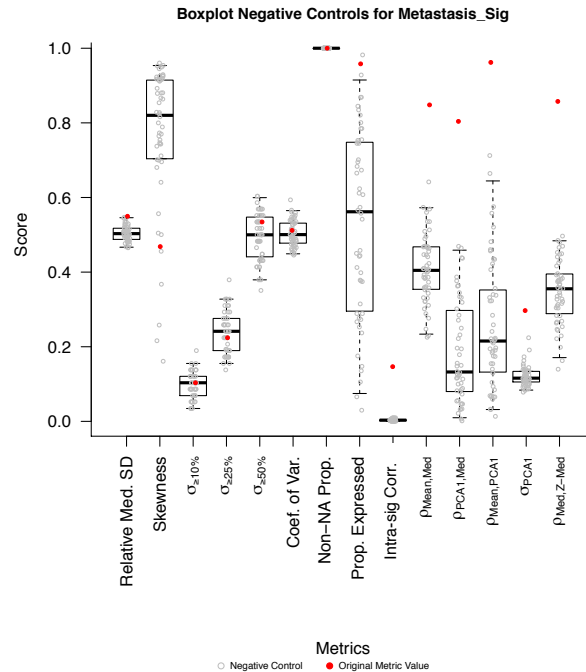


Boxplot Negative Controls for Random_Sig





Microarray



RNA-Seq

