

# HypoML: Visual Analysis for Hypothesis-based Evaluation of Machine Learning Models

Qianwen Wang, William Alexander, Jack Pegg, Huamin Qu, and Min Chen, *Member, IEEE*

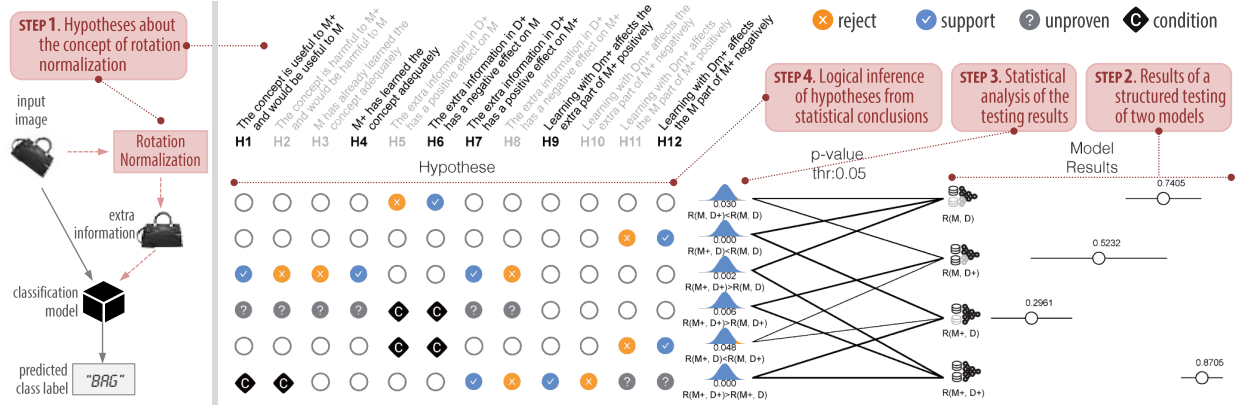


Fig. 1. There are many invariance problems in machine learning. Some are easy and others are hard. For example, one may wish to know if a classification model is rotation-invariant. If the model is not so, one may use another model that can detect the rotation angle of an object or perform some rotation regularization. The detected rotation angle or regularized image is a piece of extra information about a “concept” that the original model may or may not know. With HypoML, one can conduct a set of structured tests, obtained automated statistical and logical analysis of the results, and visualize the conclusions about the hypotheses related to the concept.

**Abstract**—In this paper, we present a visual analytics tool for enabling hypothesis-based evaluation of machine learning (ML) models. We describe a novel ML-testing framework that combines the traditional statistical hypothesis testing (commonly used in empirical research) with logical reasoning about the conclusions of multiple hypotheses. The framework defines a controlled configuration for testing a number of hypotheses as to whether and how some extra information about a “concept” or “feature” may benefit or hinder an ML model. Because reasoning multiple hypotheses is not always straightforward, we provide HypoML as a visual analysis tool, with which, the multi-thread testing results are first transformed to analytical results using statistical and logical inferences, and then to a visual representation for rapid observation of the conclusions and the logical flow between the testing results and hypotheses. We have applied HypoML to a number of hypothesized concepts, demonstrating the intuitive and explainable nature of the visual analysis.

**Index Terms**—Visual analytics, model-developmental visualization, machine learning, neural network, hypothesis test, HypoML.

## 1 INTRODUCTION

In computer vision, data mining, and machine learning (ML), a *feature* is a measurable variable that characterizes a particular kind of property or attribute of a data object (e.g., an image, a time series, a multivariate record, etc.). Many technical solutions in these fields heavily rely on model-developers’ knowledge about various features and include human-centric feature engineering as a critical process in a model development workflow [1, 8]. On the other hand, from an AI perspective, it is desirable for ML to minimize the dependence on the human knowledge of potentially useful features [20]. Technical solutions, such as deep learning, aims to achieve this objective.

Despite of the success of deep learning and other ML methods without the need for humans to specify features, there have been some

concerns about whether the learned “useful” features contribute towards undesirable biases and may actually undermine model performance [16, 41]. The concerns can be about whether the selection of some variation types (e.g., variables such as colors, patterns, dates, etc.) or some variation ranges (e.g., shades of yellow, circles, summer time, etc.). Inevitably, model-developers have been interested in what features may have been learned by an ML model and how these features influence the model performance. This interest is especially high with respect to deep neural networks (NN) for computer vision tasks. A class of visualization techniques, such as neuron activation plot, filter plot, gradient ascent plot [32], Deconvolution [39], and their variants, have been widely used by developers to observe neurons and analyze their learned features. Since a NN typically consists of a huge number of neurons, the visual observation may encounter several obstacles, including time demand for viewing all neurons that may reveal some features, subjectivity and memory limitation of an observer, and uncertainty about the semantic meaning of an observed feature. More importantly, while most model-developers have a non-trivial amount of knowledge about features that are potentially useful or harmful, their initiatives are limited to searching for patterns in many thousands of neuron-based plots and speculating if a feature has been learned.

In this work, we propose a new visual analytics approach that enables model-developers to use their knowledge and initiatives in hypothesising and evaluating if any feature may be useful or harmful, if such a feature is learned by a model, and how it may affect a learned model.

- Qianwen Wang and Huamin Qu are with Hong Kong University of Science and Technology, Hong Kong, China. Emails: qwangbb@connect.ust.hk, huamin@cse.ust.hk.
- William Alexander, Jack Pegg, and Min Chen are with University of Oxford, UK. E-mails: william.alexander@wadhams.ox.ac.uk, jack.p3gg@gmail.com, min.chen@oerc.ox.ac.uk

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxx/TVCG.201x.xxxxxxx

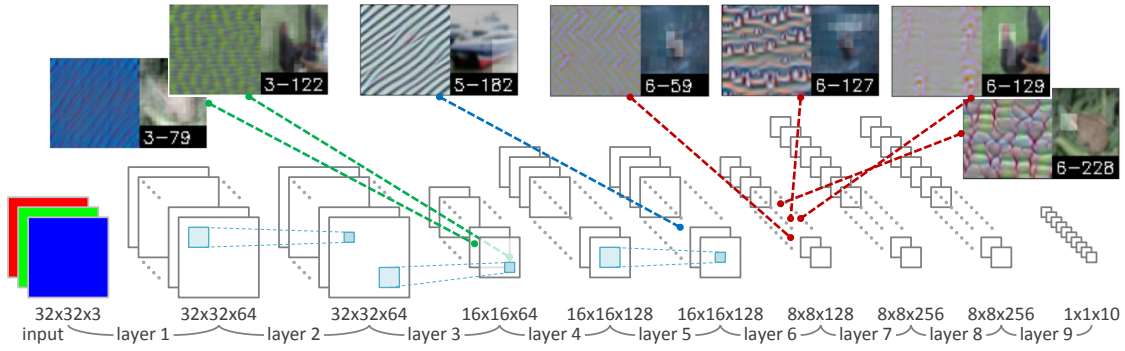


Fig. 2. Gradient ascent [32] can help model-developers observe the pattern that a specified neuron has learned. However, even a small CNN has a huge number of neurons waiting to be inspected, while many patterns shown are not semantically interpretable. Meanwhile, model-developers are often unable to determine whether a pattern is useful or harmful. The CNN and gradient ascent plots shown are from an experiment by the authors.

In particular, we outline a framework for testing such hypotheses systematically, and describe the underlying statistical and logical analysis for inferring conclusions about multiple hypotheses from multiple sets of testing results. Because many model-developers may not be familiar with or remember the underlying statistical and logical analysis, we develop a visual analytics tool, HypoML, for carrying out analysis as well as for depicting the flow of inference (Fig. 1), facilitating rapid observation of the conclusions and the logical flow between the testing data and hypotheses. [HypoML itself is independent of ML models or input data types, though we report only its application to CNN models for image classification in this work.](#) We have made HypoML available as open-source software, a demo is available at <https://hypoml.bitbucket.io/> and the source code is available at <https://bitbucket.org/hypoml/hypoml.bitbucket.io>.

The term “feature” typically implies a piece of information contained in the original input data. Since HypoML can also be used to test a hypothesis about a piece of information that is not included in the original data, we will use the term “concept-based hypotheses” to describe what to be tested with HypoML.

## 2 RELATED WORK

While machine learning (ML) has an important role to play in visualization and visual analytics [9], almost every aspect of ML processes can benefit from visualization as shown by a recently established ontology VIS4ML [29]. In general, when model-developers observe some phenomena in an ML process, such as its training and testing data, results, the inner states of a model, and the provenance of the learning process, they acquire new information to inform their various decisions that affect the ML process. As demonstrated quantitatively by Tam et al. [34], a model-developer can contribute a huge amount of knowledge (measured in bits) to an ML process through the use of visualization. This work focuses on the evaluation stage of ML workflows, aiming to enable ML developers, who are able to hypothesize about a model intelligently, to evaluate their hypotheses rigorously using a VA tool.

Methods for evaluating ML models fall into two main classes: *black-box analysis* and *white-box analysis*. Here we focus our review of the previous works on model evaluation that feature visualization techniques. More comprehensive surveys on using visualization for ML can be found in the works of Zhang and Zhu [42] and Hohman et al. [12].

Black-box analysis enables users to investigate and evaluate ML models without knowing the internal working mechanism. Statistic metrics (e.g., accuracy, recall), ROC curve, and confusion matrices are widely-used black-box analysis and have commonly been provided as built-in functions in ML environments. To aid the aggregated statistical analysis, researchers recently proposed visualization techniques to support black-box evaluation of ML models [2, 18, 27, 40]. For example, Squares [27] juxtaposes a set of histograms to present an instance-level visualization for models in multi-class classification problems. Manifold [40] employs a scatterplot-based visual technique to assist in the comparison between multiclass classifiers. [Among black-box analysis, what-if analysis \[18, 36\] is the most relevant to our study.](#)

[What-if analysis examines hypotheses about how perturbations to inputs affect the ML model outputs, and has been supported by visualization tools, e.g., What-if tool \[36\], Prospector \[18\], and GAMUT \[11\]. These tools allow users to drill down into specific input data points, manipulate their feature values, and examine the effect of such manipulations. Our work focuses on statistically-meaningful “what-if” analysis.](#)

White-box analysis, on the contrary, opens the black box and displays the internal states of ML models. A number of visualization tools have been proposed to support white-box analysis of different ML models, including MLP [26], CNNs [14, 21, 23, 25, 26], deep generative models [15, 22, 35], and RNNs [24, 33]. Although these tools have utilized some of the most sophisticated visual representations and have assisted model-developers in evaluating, understanding, and explaining their models, comprehending a huge number of high-dimensional internal states is naturally challenging for humans.

In addition, researchers proposed techniques to summarize information about internal variables and present the summary information visually. Especially in computer vision, saliency-based methods, such as CAM [43], Grad-CAM [30], and guided back propagation [32], identify discriminative regions in the input image and thus highlight important features for a certain prediction. However, these saliency-based methods can only offer explanations for specific predictions but cannot confirm whether or not a concept has been learned. To offer instance-independent explanation, Yosinski et al. employed gradient ascent plots [38] to depict the patterns that an individual neuron has learned. Fig. 2 illustrates a small selection of gradient ascent plots being observed in conjunction with a CNN. However, even for such a simple model, there are a huge number of neurons, it is impossible for model-developers to conduct a full examination. Moreover, the depicted pattern would provide largely a hunch, but not a proof whether a certain concept is useful or not to the classification task. Perhaps the most relevant to our work is TCVA [17], which learns human-friendly concepts from an already trained model and conducts hypothesis testing. However, TCVA requires a time-consuming process to label the concept across the whole dataset.

In this work, we propose a novel ML-testing framework that combines black-box and white-box analysis. Whether an ML model has learned a concept or feature is a typical “internal problem” that is to be investigated using white-box analysis. The new framework allows model-developers to investigate such “internal problems” in a manner of black-box analysis.

## 3 CONCEPT-BASED TESTING OF ML MODELS

Let  $M$  be an ML model that transforms an input data  $d \in D$  to an output decision. A *concept*  $\xi$  is a variable that is not explicitly defined in  $D$ , but is hypothesized by an ML model-developer to be related to the quality of the output decision. Fig. 3 shows several examples of concepts. We can observe that some concepts may be extracted from the original data objects using known techniques, while it may be almost impossible to infer some other concepts from the data objects.

As long as  $M$  has a finite number of constructs (e.g., neurons or

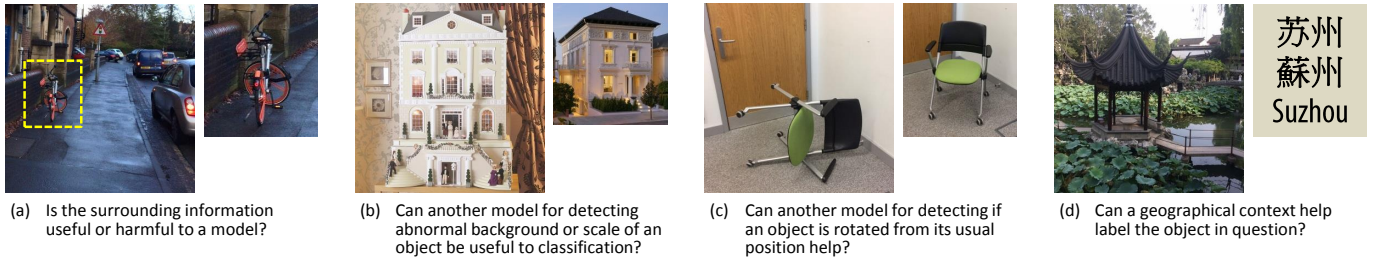


Fig. 3. During the development of ML models, model-developers usually have many hypotheses about whether certain extra information (e.g., location, date, time, etc.) or certain preprocessing methods (e.g., cropping, histogram normalization, etc.) would help a model.

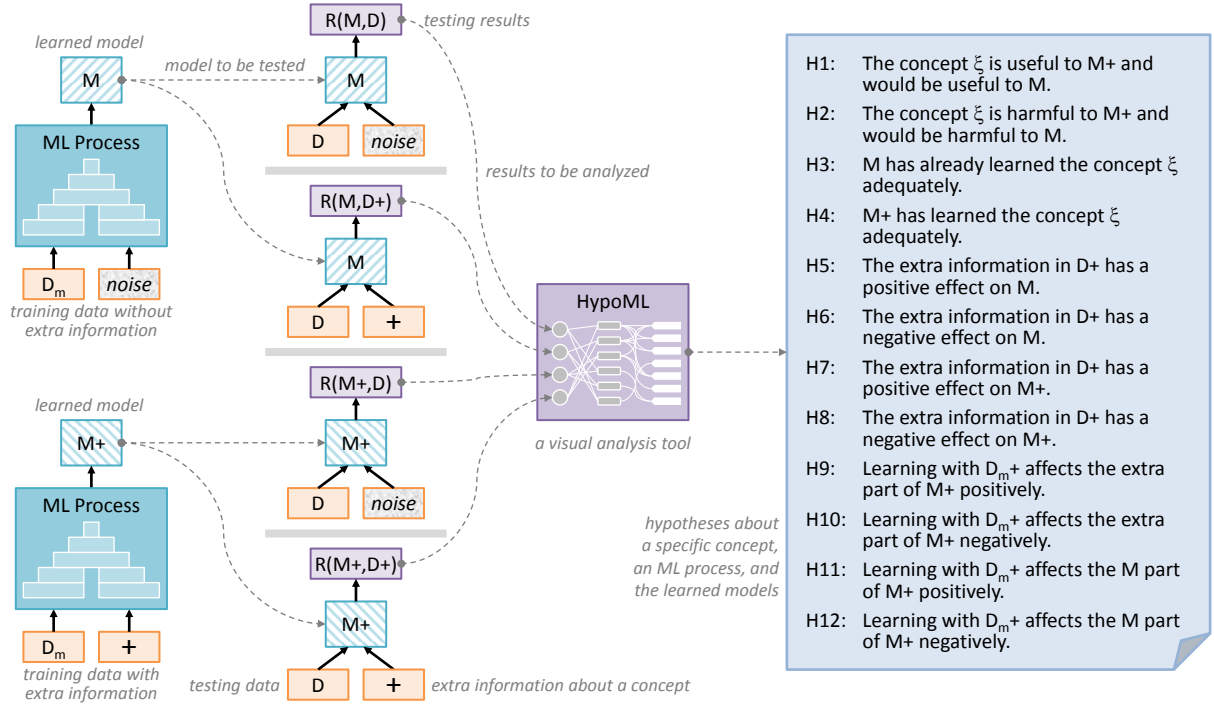


Fig. 4. An illustration of the structured testing method proposed in this work. The concept to be tested is encoded as extra information to accompany the original data. Two models,  $M+$  and  $M$ , are trained with and without extra information. Both models are then tested using two different types of testing data, one with extra information and one without. The four sets of results are then analyzed by the HypoML tool against 12 hypotheses. HypoML presents the analytical conclusions using visualization as shown in Fig. 1.

tree nodes) or receives input data with finite informative dimensions, there will always be some concepts that  $M$  cannot learn. Inevitably, most model-developers will ponder about some concepts in relation to a learned model  $M$ . Considering the examples in Fig. 3, one may ask:

- Would having an extended field of view be useful for recognizing an object captured from a less ideal viewing angle?
- Would another model for detecting an anomalous background or some scale inconsistency be useful to differentiate a toy from a real building?
- Would another model that is able to detect an object in an unusual position and estimate the rotation angle be useful to the recognition of the object?
- Would having additional information about a geographical context improve the accuracy of building recognition?

One can easily imagine many other questions about different types of extra information, such as different meta-data, multiple data capture modalities, and various pre-processing techniques. All these questions are essentially hypotheses. Just as in psychology, healthcare, social science, and many other disciplines, one can conduct experiments to evaluate such hypotheses. Indeed, one can test ML models against many thousands of data objects in comparison with tens of stimuli in typical empirical studies. However, since an ML experiment typically

features many uncontrolled variables, having a lot of testing data does not imply that the testing result will be statistically meaningful.

Because model testing is a routine operation in ML, it is desirable to establish a structured method such that many model-developers can adopt the same method and produce comparable testing results. As the above definition of *concept* is relatively broad, developers of different ML models in various applications can benefit from open source or commercial software for supporting such a structured testing method.

Fig. 4 illustrates the framework for concept-based hypothesis testing. Given an ML process and a training and testing dataset, a model-developer is interested to know how some extra information about a concept may affect the ML process and the learned model. The framework thus requires the developer to invoke two ML processes that receive two pieces of input data. As shown on the left of Fig. 4, both processes take the original training data  $D_m$  as one piece of the data. For the other piece, one process takes random noise as its input, while the other takes extra data about a concept (denoted by the sign “+”). In general, the extra data for training  $M+$  can be in any form, as long as there is a matching form of noise for training  $M$ . In practice, for testing a specific model  $M$ , it is cost-effective to fix the data structure of the “+” part, allowing  $M$  and  $M+$  to have the same fixed model structure for testing many concepts. Such examples will be given in Section 6.

The framework then requires the model-developer to test each model



with two runs using the exactly the same ML process and configuration. As illustrated in the middle column of Fig. 4, one testing run uses testing data  $D$  that does not have extra data, while the other run uses testing data  $D+$  that include extra data. The two runs with  $M$  thus produce two sets of results,  $R_{M,D}$  and  $R_{M,D+}$ , while the two runs with  $M+$  produce  $R_{M+,D}$  and  $R_{M+,D+}$ . Because evaluating an ML model typically involves testing many thousands of data objects, some computational analysis of the four sets of results will be necessary.

HypoML is designed to automate the computational analysis. In particular, it provides statistical and logical analysis for evaluating a set of hypotheses. The statistical analysis is based on the well-established method for hypothesis testing, while the logical analysis is formulated in this work for reasoning about the intertwining relationships between 12 hypotheses and 6 statistical conclusions drawn from different pairs of results. To assist users in understanding such complex relationships, HypoML provides a purposely-designed visual representation, which enables users to trace the conclusion of each hypothesis to related statistical analysis and to the corresponding testing results.

The 12 hypotheses are listed on the right of Fig. 4. While the first a few are what many model-developers may hypothesize, the others are the “side products”, about which the tests may possibly offer some inference. The primary goal of HypoML is to enable model-developers to evaluate their hypotheses rather than for a novice audience. For those model-developers who know their models and ML processes, it is not difficult to interpret these 12 hypotheses and selectively pay attention to a subset relevant to a particular model or a testing experiment.

The first two hypotheses,  $H_1$  and  $H_2$ , are about whether the concept concerned is useful (or harmful) to  $M+$ , and would be useful (or harmful) to  $M$ . Although the conclusions for these two hypotheses cannot in principle be both positive, each can also be inconclusive. We thus follow the convention of hypothesis testing by listing them as separate hypotheses, each can be independently confirmed, rejected, or unproven (inconclusive). We also anticipate that more testing and analysis methods may be developed in the future, which may support or reject those apparently-paired hypotheses asymmetrically. Having separate hypotheses will not hinder such advancement.

$H_3$  hypothesizes that model  $M$  has already learned the concept adequately, while  $H_4$  hypothesizes that model  $M+$  has learned the concept adequately. For  $H_3$ , the adverb “adequately” implies that the concept can be learned by a model, such as  $M$ , without the need for any extra data about the concept. For  $H_4$ , the adverb “adequately” implies that  $M+$  would perform worse without the extra data of the concept. When the conclusion is in favor of  $M+$ , it suggests a potential for improving  $M$ , but should be interpreted that  $M+$  has a better structure.

As shown in Fig. 4, model  $M$  is trained with the original data and random noise for the  $+$  part. Although the  $+$  part provides an extra signal channel, noise is not expected to provide extra information. Hence,  $M$  is not expected to be affected by the noise signal. In other words, there should be no significant difference between the testing results  $R_{M,D+}$  and  $R_{M,D}$ . However, as a scientific exercise, one cannot take this assumption for granted due to some unexpected configuration or implementation errors.  $H_5$  and  $H_6$  are thus designed to examine if  $M$  is affected positively or negatively by the extra data during testing.

$M+$  is trained using the original data and the  $+$  part encoding the concept being tested.  $H_7$  and  $H_8$  are designed to test whether the extra information in the  $+$  part has a positive or negative effect on  $M+$ .

Meanwhile, as long as any part of an ML model can learn, directly or indirectly, from the information in both the original data and the  $+$  part, learning from one part of the data could be affected by the information in another part. Assume the model template for  $M$  and  $M+$  is constructed based on an original model template  $\hat{m}$  without the extra data input. For  $M+$ , we define its  $M$  part as all components of the model (e.g., neurons and connections in NNs) belonging to  $\hat{m}$ , and its  $+$  part as those added components (i.e., not in  $\hat{m}$ ).  $H_9$ ,  $H_{10}$ ,  $H_{11}$ , and  $H_{12}$  are for investigating the trade-off between the two parts of  $M+$  in the development of its intelligence. Depending on the design of the model template or architecture, the  $M$  and  $+$  parts of  $M+$  can be quite separated as well as rather integrated. When the two parts are more integrated, one should consider two parts as functional units rather

than geometric or topological regions. For developers who know the structures of their models, it will not be difficult for them to interpret these hypotheses correctly. Similarly, we separate  $H_9$  from  $H_{10}$ , and separate  $H_{11}$  from  $H_{12}$  because of the inconclusive state in each case.

Evaluating hypotheses  $H_9$ ,  $H_{10}$ ,  $H_{11}$ , and  $H_{12}$  can also aid investigation into different strategies for *information fusion* in ML, ranging from multi-model decisions at higher layers (e.g., [10, 13]) to multi-sensor information fusion at lower layers (e.g., [5, 28]). We will show an example of instigating different fusion strategies in Section 6.

## 4 STATISTICAL AND LOGICAL REASONING OF HYPOTHESES

As shown in Fig. 4, HypoML receives four sets of results, namely  $R_{M,D}$ ,  $R_{M,D+}$ ,  $R_{M+,D}$ , and  $R_{M+,D+}$ . Each set of results is a list of tuples, each of which consists of

- **id** — the unique identifier of a data object. The data object may be an image, a feature vector, a multivariate data record, or a more complex data record.
- **ground truth** — a ground truth label, which can be a nominal value, an integer, a real number, a range, or a data record of a more complex data type (e.g., a time series).
- **ML label** — a label generated by an ML model. The label must be of the same data type as ground truth.
- **ML uncertainty** — an optional value indicating the uncertainty estimated by an ML model equipped with a self-assessment capacity.
- **correctness** — This is a value in the range of  $[0, 1]$  with 1 indicating absolutely correct, and 0 indicating absolutely incorrect. The value is mostly computed based on ground truth and ML label using a user-defined function such as accuracy, recall, precision.
- **correctness with uncertainty** — This is used by the statistical analysis and is defined as  $\text{ML uncertainty} \times \text{correctness}$ .

Given two sets of results,  $R_a$  and  $R_b$ , we assume that the tuples in the two lists are paired, i.e., the id entries are in the same order exactly. We can compare  $R_a$  and  $R_b$  with their accuracy, i.e., the average of correctness with uncertainty. As testing in ML often shows small variations of accuracy, it is necessary to measure the statistical significance, providing the probabilities of Type I and Type II errors. Although paired, one-tail  $t$ -test can be used for this purpose, HypoML uses paired, two-tail  $t$ -test in order to maintain consistent interpretation of  $p$ -values and prevent setting an overgenerous threshold  $p$ -value by mistake.

Let us introduce the following notation to denote the possible outcomes of the statistical analysis between  $R_a$  and  $R_b$ .

- $R_a \overset{\sim}{\ll} R_b$  — significantly lower than
- $R_a \overset{\sim}{\gg} R_b$  — significantly higher than
- $R_a \overset{\sim}{\approx} R_b$  — insignificant higher or lower than
- $R_a \overset{\sim}{\approx} R_b$  —  $R_a \overset{\sim}{\approx} R_b$  or  $R_a \approx R_b$ , but not  $R_a \overset{\sim}{\ll} R_b$ .
- $R_a \overset{\sim}{\approx} R_b$  —  $R_a \overset{\sim}{\approx} R_b$  or  $R_a \approx R_b$ , but not  $R_a \overset{\sim}{\gg} R_b$ .

With four sets of results, there are six pairs of statistical comparison, which are labelled as  $A_1, A_2, \dots, A_6$ . Each analytical conclusion  $A_i$  may support or reject some of the 12 hypotheses  $H_1, H_2, \dots, H_{12}$ , but not all. For example the analysis  $A_1$ , which compares  $R_{M+,D+}$  and  $R_{M,D}$ , can inform the evaluation of  $H_1$  and  $H_2$ . If  $R_{M+,D+}$  is statistically better than  $R_{M,D}$ , i.e.,  $R_{M+,D+} \overset{\sim}{\gg} R_{M,D}$ , we can draw a conclusion that  $A_1$  supports  $H_1$  and rejects  $H_2$ . If  $R_{M+,D+} \overset{\sim}{\approx} R_{M,D}$ ,  $A_1$  supports  $H_2$  and rejects  $H_1$ . If  $R_{M+,D+} \approx R_{M,D}$ ,  $A_1$  returns an unproven (inconclusive) verdict about  $H_1$  and  $H_2$ .

We can observe that  $A_1$  can also inform the evaluation of  $H_3$ ,  $H_4$ ,  $H_7$ , and  $H_8$ .  $A_2$ , which compares  $R_{M+,D+}$  and  $R_{M,D+}$ , can inform the evaluation of  $H_1$ ,  $H_2$ ,  $H_3$ ,  $H_4$ ,  $H_7$ , and  $H_8$ , but it can only do so subject to that some other hypotheses have already been confirmed or rejected. Table 1 summaries the relations between the six sets of statistical analysis  $A_1, A_2, \dots, A_6$  and the 12 hypotheses  $H_1, H_2, \dots, H_{12}$ . Since the analysis of  $A_i$  and  $H_j$  depends only on the four sets of results, HypoML can be used to test any ML model with any input data type.

Clearly, reasoning about these relations is time consuming and error prone. In order to support the frequent analytical tasks of the developers



Table 1. The relations between statistical analysis and hypotheses.

Analysis	Condition	Hypothesis
$A_1: R_{M+,D+} \text{ v. } R_{M,D}$	$H_5, H_6$ $H_1, H_2$	$H_1, H_2, H_3, H_4, H_7, H_8$
$A_2: R_{M+,D+} \text{ v. } R_{M,D+}$		$H_1, H_2, H_3, H_4, H_7, H_8$
$A_3: R_{M+,D+} \text{ v. } R_{M+,D}$		$H_7, H_8, H_9, H_{10}, H_{11}, H_{12}$
$A_4: R_{M+,D} \text{ v. } R_{M,D}$	$H_5, H_6$	$H_{11}, H_{12}$
$A_5: R_{M+,D} \text{ v. } R_{M,D+}$		$H_{11}, H_{12}$
$A_6: R_{M,D+} \text{ v. } R_{M,D}$		$H_5, H_6$

in testing their ML models, we formulated a set of logical inference rules following careful reasoning about the causal relations among different conclusions of statistical analysis and different hypotheses. This allows HypoML to provide automated logical analysis as well as statistical analysis. To help describe the logical analysis, we employ some additional notations. They are:

- $\top(S)$  — The statement  $S$  is true.
- $\perp(S)$  — The statement  $S$  is false.
- $\ast(S)$  — The statement  $S$  is unproven.
- $\wedge$  — Logical conjunction.
- $\vee$  — Logical (inclusive) disjunction.

Appendix A provides detailed explanations of the six sets of logical influence rules. For the self-containment of the main text, we list these rules with brief explanations highlighting the major considerations.

The statistical analysis  $A_1$  reflects the conventional comparison between  $R_{M+,D+}$  and  $R_{M,D}$ . If the  $t$ -test shows  $R_{M+,D+} > R_{M,D}$  is statistically meaningful, we can infer that  $H_1, H_4, H_7$  are confirmed, and  $H_2, H_3, H_8$  are rejected. If the  $t$ -test shows  $R_{M+,D+} < R_{M,D}$  is statistically meaningful, we can infer the opposite conclusions for  $H_1, H_2, H_3$ , and  $H_4$ , while  $H_7$  and  $H_8$  are inconclusive. To simplify our logical formulae further, we will not explicitly list those inconclusive hypotheses. With this simplification, we can now express the inferences as a combined set of logical rules:

$A_1$ : Comparing  $R_{M+,D+}$  and  $R_{M,D}$  may conclude:

- $R_{M+,D+} \gtrsim R_{M,D} \implies \top(H_1) \wedge \top(H_4) \wedge \top(H_7) \wedge \perp(H_2) \wedge \perp(H_3) \wedge \perp(H_8)$ . This reads as  $H_1, H_4$ , and  $H_7$  are all true, and  $H_2, H_3$ , and  $H_8$  are all false.
- $R_{M+,D+} \lesssim R_{M,D} \implies \top(H_2) \wedge \top(H_4) \wedge \perp(H_1) \wedge \perp(H_3)$ .

Analysis  $A_2$  cannot draw conclusions about  $H_5$  and  $H_6$ , but its conclusion may depend on them. In general, there is a common-sense assumption that neither  $H_5$  nor  $H_6$  is likely to be true.

$A_2$ : Comparing  $R_{M+,D+}$  and  $R_{M,D+}$  may conclude:

- $R_{M+,D+} \gtrsim R_{M,D+} \implies$   
 (i) if  $\perp(H_6)$  then  $\top(H_1) \wedge \top(H_4) \wedge \top(H_7) \wedge \perp(H_2) \wedge \perp(H_3) \wedge \perp(H_8)$ ; or  
 (ii) if  $\ast(H_6)$  then  $\top(H_1) \wedge \top(H_4) \wedge \top(H_7) \wedge \perp(H_2) \wedge \perp(H_3) \wedge \perp(H_8)$ ; or  
 (iii) if  $\top(H_6)$ . This offers an explanation but it is against a common-sense assumption that  $H_6$  is unlikely to be true, and should be treated cautiously.
- $R_{M+,D+} \lesssim R_{M,D+} \implies$   
 (i) if  $\perp(H_5)$  then  $\top(H_2) \wedge \top(H_4) \wedge \top(H_8) \wedge \perp(H_1) \wedge \perp(H_3) \wedge \perp(H_7)$ ; or  
 (ii) if  $\ast(H_5)$  then  $\top(H_2) \wedge \top(H_4) \wedge \top(H_8) \wedge \perp(H_1) \wedge \perp(H_3) \wedge \perp(H_7)$ ; or  
 (iii) if  $\top(H_5)$ . This offers an explanation but it is against a common-sense assumption that  $H_5$  is unlikely to be true, and should be treated cautiously.

Because analysis  $A_3$  does not compare  $M+$  with  $M$ , the conclusion is limited to the context of  $M+$ . Mathematically, it is possible for  $A_3$  to conclude that the concept is useful in the context of  $M+$ , while  $A_1$  or  $A_2$  concludes that the concept is harmful or is neither useful nor harmful. Considering this limitation, it is unsafe for this analysis to draw a conclusion about  $H_1$  and  $H_2$ . Meanwhile the analysis depends on the conclusions of  $H_1$  and  $H_2$  in a small way.

$A_3$ : Comparing  $R_{M+,D+}$  and  $R_{M+,D}$  may conclude:

- $R_{M+,D+} \gtrsim R_{M+,D} \implies$   
 (i) if  $\top(H_1)$ , then  $\top(H_7) \wedge \top(H_9) \wedge \perp(H_8) \wedge \perp(H_{10})$ ; or  
 (ii) if  $\ast(H_1)$ , then  $\top(H_9) \wedge \perp(H_{10})$ ; or  
 (iii) if  $\perp(H_1)$ , then  $\top(H_{12}) \wedge \perp(H_{11})$ .
- $R_{M+,D+} \lesssim R_{M+,D} \implies$   
 (i) if  $\top(H_2)$ , then  $\top(H_8) \wedge \top(H_{10}) \wedge \perp(H_7) \wedge \perp(H_9)$ ; or  
 (ii) if  $\ast(H_2)$ , then  $\top(H_{10}) \wedge \perp(H_9)$ ; or  
 (iii) if  $\perp(H_2)$ , then  $\top(H_{10}) \wedge \perp(H_9)$ . This conclusion is against a common-sense assumption that a useful concept normally should not affect the extra part of  $M+$  negatively, and should be treated cautiously.

Analysis  $A_4$  is relatively easy to reason, and it is useful for investigating if the part of model  $M+$  for handling the original data  $D$  becomes less capable due to the training with extra information.

$A_4$ : Comparing  $R_{M+,D}$  and  $R_{M,D}$  may conclude:

- $R_{M+,D} \gtrsim R_{M,D} \implies \top(H_{11}) \wedge \perp(H_{12})$ .
- $R_{M+,D} \lesssim R_{M,D} \implies \top(H_{12}) \wedge \perp(H_{11})$ .

$A_5$  cannot draw conclusions about  $H_5$  and  $H_6$ , but its conclusion may depend on them. In general, there is a common-sense assumption that neither  $H_5$  nor  $H_6$  is true.

$A_5$ : Comparing  $R_{M+,D}$  and  $R_{M,D+}$  may conclude:

- $R_{M+,D} \gtrsim R_{M,D+} \implies$   
 (i) if  $\perp(H_6)$  then  $\top(H_{11}) \wedge \perp(H_{12})$ ; or  
 (ii) if  $\ast(H_6)$  then  $\top(H_{11}) \wedge \perp(H_{12})$ ; or  
 (iii) if  $\top(H_6)$ . This offers an explanation but it is against a common-sense assumption that  $H_6$  is unlikely to be true, and should be treated cautiously.
- $R_{M+,D} \lesssim R_{M,D+} \implies$   
 (i) if  $\perp(H_5)$  then  $\top(H_{12}) \wedge \perp(H_{11})$ ; or  
 (ii) if  $\ast(H_5)$  then  $\top(H_{12}) \wedge \perp(H_{11})$ ; or  
 (iii) if  $\top(H_5)$ . This offers an explanation but it is against a common-sense assumption that  $H_6$  is unlikely to be true, and should be treated cautiously.

Analysis  $A_6$  is the only comparison that may inform the evaluation of  $H_5$  or  $H_6$ . In general, there is a common-sense assumption that neither  $H_5$  nor  $H_6$  is true if the model template or architecture was correctly defined, the correct ML method was followed, and the correct ML process was executed. When  $H_5$  or  $H_6$  is confirmed, it usually suggests some imperfection of the model template or learning process. Therefore the conclusions of  $A_6$  should not be interpreted as their face values. However, the evaluation of  $H_5$  nor  $H_6$  is necessary since  $A_2$  and  $A_5$  depend on them.

$A_6$ : Comparing  $R_{M,D+}$  and  $R_{M,D}$  may conclude:

- $R_{M,D+} \gtrsim R_{M,D} \implies \top(H_5) \wedge \perp(H_6)$ ;
- $R_{M,D+} \lesssim R_{M,D} \implies \top(H_6) \wedge \perp(H_5)$ .

Because of the dependency among the six sets of analysis, the computation of the logical inference must follow an appropriate order, which is summarized as follows:

- STEP 0: Initialize the indicator of each hypothesis to 0.  
 STEP 1: Compute the six comparative values, i.e.,  $A_1, A_2, \dots, A_6$ , in terms of  $\gtrsim, \gtrless, \approx$ , based on statistical analysis.  
 STEP 2: Compute the logical inference (i.e., in terms of  $\top, \perp, \ast$ ) based on  $A_1, A_4, A_6$ . For each true statement, i.e.,  $\top(H_i)$ , add +1 to the indicator of  $H_i$ . For each false statement, i.e.,  $\perp(S)$ , add -1 to the indicator of  $H_i$ .  
 STEP 3: Compute the indicators based on  $A_2, A_5$ .  
 STEP 4: Compute the indicators based on  $A_3$ .  
 STEP 5: Then display each indicator based on positive or negative values. HypoML displays each hypothesis according to its indicator in three states:  $>0$  (confirmed),  $0$  (unproven),  $<0$  (rejected).

Clearly, given  $A_1 \sim A_6$ , it is not feasible for a user to perform rigorous reasoning defined by these logical inference rules. Hence it is necessary to introduce visualization to aid the automated analysis.

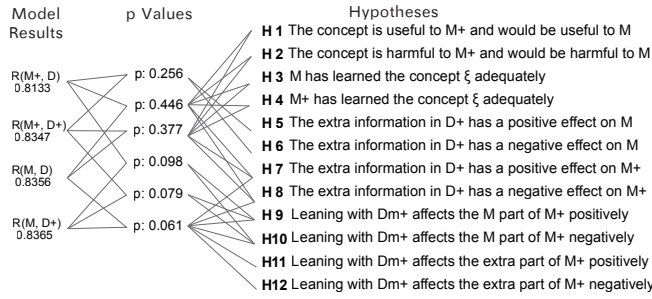


Fig. 5. The analytical workflow from testing results to statistical analysis and then logical inference of hypothesis. As a basic visual design, it has a number of shortcomings.

## 5 VISUAL ANALYSIS OF HYPOTHESES

Fig. 5 shows a typical workflow of the proposed hypotheses testing. To start with, model-developers conduct experiments and obtain four sets of results, i.e.,  $R_{M,D}$ ,  $R_{M,D+}$ ,  $R_{M+,D}$ , and  $R_{M+,D+}$ . HypoML then performs six sets of statistical analysis by comparing each pair of the results. Based on the statistical analysis, HypoML makes logical inference about the twelve hypotheses, deciding whether a hypothesis should be supported, rejected, or inconclusive.

It is helpful for model-developers to make quick observation about the analysis and conclusions. It will also be useful for the model-developers to convey the outcomes of the test to other stakeholders, such as fellow model-developers and sometimes users of the ML models being evaluated. It can be difficult for some model-developers and many of ML users to remember and reason the complicated relationships among experiment results, statistic and logical analysis, and multiple hypotheses. Therefore, an effective visual representation is necessary. The bipartite graph shown in Fig. 5 is a straightforward solution but it exhibits several shortcomings that hinder efficient information acquisition and effective information dissemination.

One main shortcoming is the cluttered links between the six statistical comparisons and the twelve hypotheses (i.e., middle of Fig. 5). These links have no obvious or memorable structures and are difficult to track by eye. One can add additional visual encoding to these links to depict three types of conclusions (i.e., reject, support, unproven) and conditional dependency. However, such encoding would further worsen the cluttering of the bipartite graph. To address this issue, we designed a matrix-based visualization for HypoML as shown in Fig. 6(a), where four types of icons (a2) are introduced to indicate reject, support, unproven, and conditional dependency.

The second shortcoming is that displaying those numerical values in Fig. 5 incurs a fair amount of cognitive load upon users for visualization tasks other than value retrieving. Since it is necessary to display them for the value retrieving task, we introduce additional visual encoding to ease other tasks. With the four accuracy values, one visualization task is to compare them. Using positions can significantly reduce the cognitive load for such a task [7]. As shown in Fig. 6(c), we use the positions of circles to indicate the accuracy values and the lines to indicate the 95% confidence interval.

With the six  $p$ -values, one visualization task is to identify those clearly below or above the threshold. Another is to use a specific value as a pivot point when tracking from hypotheses to testing results or vice versa. We decided to add a glyph to each  $p$ -value to ease these tasks as the visual encoding can aid memory recall [4]. As shown in Fig. 6(b1), we considered several alternative designs. With one design option, the area of a circle is used to encode the level of statistical significance, i.e., the inverse of a  $p$ -value. The lower the  $p$ -value, the more significant the difference and the larger the circle. However, the initial user feedback suggested that this design was “confusing” due to the reverse encoding. With another design option, the  $p$ -value is encoded using the area of an orange circle, which is inside a large blue circle of a fixed size. While this design enables direct observation of statistical significance through the blue area as well as the  $p$ -value through the orange area, it was found to be “unintuitive” for those who

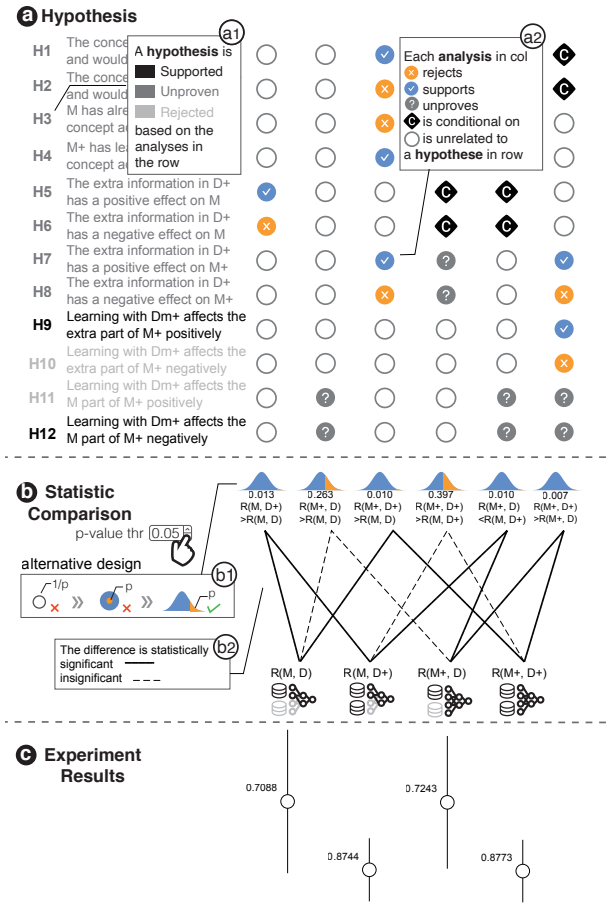


Fig. 6. The vertical version of HypoML interface depicts the three phases of a hypothesis test: (a) at the top, *conclusions of the 12 hypotheses*, which allow users to decide whether to track back to the statistical inference and experiment results for detailed reasoning; (b) in the middle, *six sets of statistic analysis* for comparing model testing results, where each  $p$ -value is doubly encoded using glyph patterns (b1) and link styles (b2); (c) at the bottom, four sets of model testing results. A horizontal version, which is more suitable for wide-screen monitors, is shown in Fig. 1.

were unfamiliar with the definition of  $p$ -value. We finally settled down on the third design based on the illustration widely used for explaining statistical hypothesis testing. In this design, the whole shape represents a normal distribution and the area in orange coarsely encodes the  $p$ -value. The normal distribution curve can quickly remind users of the meaning of  $p$ -value.

The third shortcoming is that while depicting the reasoning flow from data to conclusion as in Fig. 5 correctly represents the temporal order of the computation, it would slow users down when they wish to find out the conclusions quickly. We thus reverse the order of the workflow in both the vertical and horizontal versions of the visual user interface (see Fig. 1 and Fig. 6). The horizontal design is more suitable for wide-screen displays, while the vertical design can be used on portable devices and high-resolution monitors. Users may benefit from having both designs available. In addition to these two versions, we considered another two less-integrated layouts. In Appendix B, we provide more detailed analysis of the four layouts, including independent feedback from ML specialists.

The HypoML interface was designed and developed by following an iterative design process with regular discussions with ML developers within the team and beyond. From the discussions, we discovered that most users would prefer to observe the conclusions of the hypotheses as soon as the testing results were loaded into HypoML. They could then decide whether it would be necessary to track back to the statistical comparison and experiment results for detailed reasoning. We also discovered that double encoding used for the  $p$ -value and hypotheses had enhanced users’ perception of the information and enable them

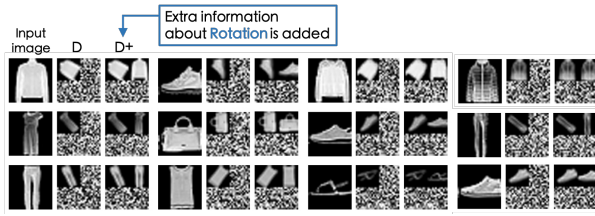


Fig. 7. Samples of the training data for testing the concept of rotation correction. For each sample, the left image shows the original object. The middle image shows the corresponding stimulus in the testing dataset  $D$ , where the object has been arbitrarily rotated. The right image shows the stimulus in the testing dataset  $D+$  where the rotated object is accompanied by an up-right view of the object.

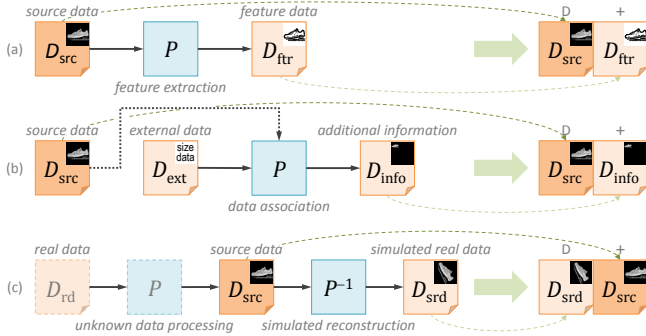


Fig. 8. Three approaches to experiment design: (a) using features extracted from the source data as the  $+$  data; (b) using extra information as the  $+$  data; and (c) assuming that the source data has already been processed and using the source data as the  $+$  data.

to switch between overview (through visual encoding) and details on demand (through numerical values) rapidly by simply changing their visual attention. While each  $p$ -value is already encoded using the glyph and numerical value, we further encode it through its links with the testing results. The link style (i.e., solid or dashed) shows whether the difference between two sets of results is statistically significant or not (Fig. 6(b2)). While the decision state of a hypothesis is already encoded using icons in the matrix, we doubly encode it using black and two grey-scale values to the levels of support to the hypothesis (Fig. 6(a1)). The black color draws users' attention quickly to those hypotheses that have been confirmed.

HypoML supports a set of interactions. Users may modify the threshold of  $p$ -value, which dynamical updates of the whole visualization and may lead to changes in the conclusions of the hypotheses. By hovering on a  $p$ -value, users can highlight the two corresponding sets of results.

## 6 USE CASES

The use cases reported in this section is primarily for testing HypoML to see if HypoML can make correct transformation from four sets of results  $R_{M,D}$ ,  $R_{M,D+}$ ,  $R_{M+,D}$ , and  $R_{M+,D+}$  to visual representations of the conclusions about 12 hypotheses. The examples shown are not intended to establish the truth about the goodness of any particular ML technique, but to demonstrate the practical uses of HypoML. If a developer suspects an ML model or a training dataset may have a shortcoming, HypoML can help the developer confirm or reject such a hypothesis. With convolution neural networks (CNN), a common wisdom is that the deeper and the larger a CNN is, more likely a concept will be learned by the CNN. When our tests show that a particular CNN model has not learned a concept adequately, it does not necessarily mean that a more complex CNN model would not be able to learn the concept either. This is indeed what testing is for in software engineering. One common goal of ML testing is to discover the shortcomings of a model, which could be due to a particular feature or a dataset.

**Experiment Design.** We used two datasets: Fashion MNIST [37] and CIFAR-10 [19]. We adopted CNN classification models from the Keras documented examples [6]. All models were specified using Keras

and Tensorflow in Python, and trained and tested using the Google Colaboratory server. In order to make testing results comparable across different tests, we used the same data template for testing each of the two models. Because the lower-layer of a CNN model depends on the template of input data, using the same data template also made the processes for testing multiple concepts more cost-effective.

One key step in the experiment design is to find extra data that encodes the testing concept. As shown in Fig. 8, HypoML supports three ways of constructing such extra data. (a) The extra data can be obtained from the source data using an external model or a pre-processing algorithm. For example, humans can often recognize objects drawn as outline sketches. One may hypothesize that outline features may be helpful, and pre-process the source dataset  $D_{src}$  to generate  $D_{ftr}$ . (b) The extra data can be obtained from an external dataset. For example, humans can estimate the size of an object being observed and the estimation can be used to ascertain a recognition decision. One may hypothesize that it may be useful to add extra size information  $D_{info}$ . (c) The extra data can be obtained, indirectly, by simulating the raw, unprocessed data. In many cases, the source data  $D_{src}$  has been manipulated using an unknown process  $P$ , e.g., rotation regularization. One may suspect the impact of  $P$  and decides to simulate the unprocessed data, yielding  $D_{srd}$ . One can test the impact of  $P$  by making  $D_{srd}$  as the  $D$  part shared by  $M$  and  $M+$  and the original  $D_{src}$  as the  $+$  part. Note that the role of  $D_{src}$  has changed.

Another key step is to combine the  $D$  and the  $+$  parts of the data. As we use a fixed data template to test many concepts in relation to a dataset, we generated data samples in a way similar to mosaic augmentation [3], a technique widely used in computer vision. As shown in Fig. 7, for the Fashion MNIST dataset, we used a template with  $2 \times 2$  placeholders. The top-left quadrant holds the source data  $D$ , while the other three quadrants hold the  $+$  data representing up to three concepts. For the CIFAR-10 dataset, we used a template with two placeholders, the left for  $D$  and the right for  $+$ . A placeholder is filled with noise if it is not used for either  $D$  or  $+$ . Note that we can use other methods, such as concatenation, to combine the  $D$  and  $+$  parts.

**Testing Concepts.** We have conducted some 15 tests. Three examples are given in this section. We detail three further examples in Appendix C, where we also demonstrate the use of experiment design (b) in Fig. 8, and compare its use with experiment design (c).

**Average Luminosity.** Let us first consider a simple hypothesis, for which the conclusion is relatively easy to anticipate for most models. Given an object in an image, its average luminosity is one of its simplest features. In general, a CNN is expected to learn features about some aggregated properties (e.g., mean, median, or mode). To test this hypothesis, we can use the experiment design in Fig. 8(a). As shown in Fig. 9, we introduced the average intensity value of an object as a single-colored square in the upper-right quadrant. We then trained two models  $M$  and  $M+$ , and tested each of them using two datasets  $D$  and  $D+$  according to the workflow in Fig. 4.

From the four sets of testing results, HypoML carries out statistical and logical analysis and displays the results. As shown in Fig. 9, the analysis indicates that most hypotheses are unproven. Although  $R_{M+,D+}$  has the highest accuracy, it is statistically insignificant to support or reject most hypotheses. This example shows the importance of statistical inference. The only hypothesis that has been confirmed is  $H_9$ , i.e., learning with  $D_m+$  affects the extra part of  $M+$  positively. However, this does not translate to a confirmation of  $H_7$  about the overall positive impact to  $M+$ . By observing the details about how  $H_9$  was confirmed, we can see that it is confirmed only within the context of  $M+$ , without involving any tests about  $M$ .

**Rotation Regularization.** In computer vision, there are many desired invariance properties for a model. Among them rotation-invariance is relative difficult to achieve. The original images in the Fashion MNIST dataset feature all fashion objects in an upright position. This naturally leads to a speculation that a trained model may not be rotation invariant. One possible way to address the need for rotation-invariance is to train a model with images featuring randomly rotated objects, which is widely employed in data augmentation techniques [31]. Since the source data



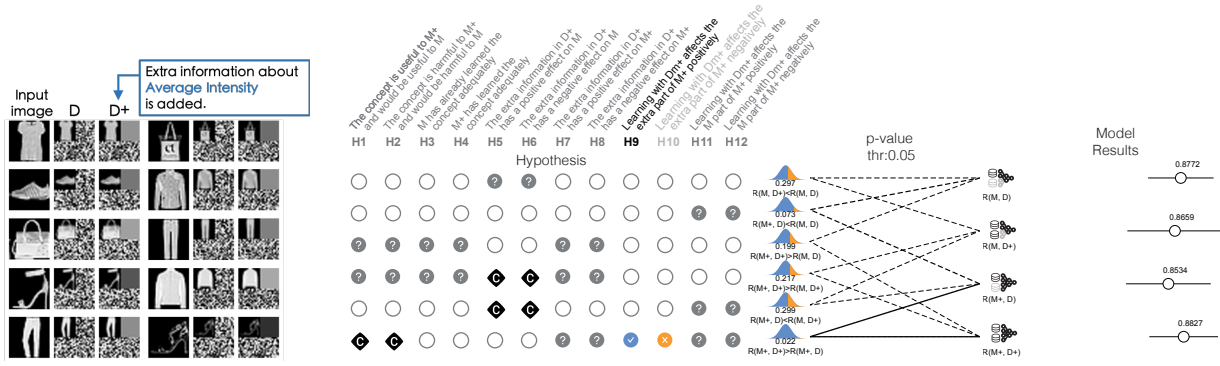


Fig. 9. Testing the combined concept of average intensity. For each sample, the stimulus in D contains an original object. The stimulus in D+ contains an extra piece of information about the average intensity of the object.

is already pre-processed with an unknown rotation-regularization  $P$ , we adopt the experiment design in Fig. 8(c), to see if  $P$  is beneficial. To simulate the real data, we applied random rotation (as  $P^{-1}$ ) to each image in the training and testing data, and placed the randomly-rotated image in the placeholder for D. As illustrated in Fig. 7, we simply reused the original upright images (as the results of  $P$ ), and placed them in one of the placeholders for +.

The testing results are shown in Fig. 1, from which we can observe that six hypotheses have been confirmed. They indicate:

- **H<sub>1</sub>**: The concept of rotation regularization is useful to M+ and would be useful to M.
- **H<sub>3</sub>**: M+ has learned from the concept of rotation regularization adequately.
- **H<sub>6</sub>**: The extra information in D+, when it is fed to M, has a negative effect on M. Although M has only learned from noise the upper-right quadrant of the stimuli, when non-noise information appears in that area, it still affects M, in a negative way.
- **H<sub>7</sub>**: The extra information in D+ (upper-right quadrant) has a positive effect on M+.
- **H<sub>9</sub>**: Learning with  $D_m+$  affects the extra part of M+ positively. This is somehow anticipated because **H<sub>1</sub>** is confirmed.
- **H<sub>12</sub>**: Learning with  $D_m+$  affects the M part of M+ negatively, that is, if the extra information is unavailable, M+ performs worse than M, which has not learned with the extra information.

While it is inspirational to train a CNN with rotation-invariance capability, this test indicates that an object detection model can benefit from the extra information generated by another model that can detect the rotation angle or perform rotation regularization.

**HSV Color Space.** In computer vision, there is an unsettled debate about which color space may be the best for the images to be processed by CNNs. Gowda and Yuan conducted a series of experiments, showing that images encoded using several color spaces may improve the accuracy of a classification model [10]. For example, they showed that mean accuracy of RGB+HSV (81.42%) is higher than that of RGB (78.89%), suggesting that HSV may be useful in addition RGB.

The comparison between RGB and RGB+HSV fits perfectly with the HypoML platform as we can place RGB in the D placeholder and HSV in the + placeholder. The comparison thus becomes a hypothesis test about whether HSV is useful as an extra concept.

Using the CIFAR-10 dataset [19], we trained models M for RGB only and M+ for RGB+HSV. Following the design outlined by Gowda and Yuan [10], we merged the network for RGB and that for HSV/noise with an ADD function at the top activation layer. As shown in Fig. 10(a), we obtained mean accuracy of  $R_{M+,D+}$  (RGB+HSV) as 81.04% and that of  $R_{M,D}$  (RGB) as 80.57%. However, the  $t$ -test shows that the comparison between  $R_{M,D}$  and  $R_{M+,D+}$  is statistically insignificant ( $p = 0.43$ ). The concept of HSV is neither useful or harmful.

We also tested other ways of integrating the two networks. For example, using a MAX function at the top activation layer, merging them at a maxpool layer in the middle of the two networks, and merging

them at a maxpool layer in the lower part near the data inputs. As shown in Figs. 10(b,c,d), none of the tests resulted in statistically insignificant  $R_{M,D} \approx R_{M+,D+}$  or  $R_{M,D} \approx R_{M+,D+}$ . Hence, with the CNN models that we used for testing RGB vs. RGB+HSV, the hypothesis that HSV may be useful in addition to RGB has not been confirmed.

**Effects of Componentization.** Recall the early discussion about interpreting **H<sub>9</sub>**, **H<sub>10</sub>**, **H<sub>11</sub>**, and **H<sub>12</sub>** in Section 3. Fig. 10 exemplifies the different ways of merging the two networks, i.e., the M part and the + part, into M+ has different impact on **H<sub>9</sub>**, **H<sub>10</sub>**, **H<sub>11</sub>**, and **H<sub>12</sub>**.

When two networks are merged at an upper layer, M+ can be seen as highly componentized. When they are merged at a lower layer, they can be seen as highly integrated. In the highly integrated case, as shown in Fig. 10(d), **H<sub>9</sub>** and **H<sub>12</sub>** were confirmed. However, the boundary between the two parts is not clearly defined, the conclusions about learning with extra information impacts the M part negatively and the + part positively should be treated with care. We can observed the changes when the merge takes place in the middle. As shown in Fig. 10(c), none of **H<sub>9</sub>**, **H<sub>10</sub>**, **H<sub>11</sub>**, and **H<sub>12</sub>** is confirmed. Interestingly, **H<sub>9</sub>** is unconfirmed as  $R_{M+,D+} \approx R_{M+,D}$ . This means that the + part of M+ did not really learn much from HSV data.

When the two networks are merged at the top activation layer, the conclusions of **H<sub>9</sub>**, **H<sub>10</sub>**, **H<sub>11</sub>**, and **H<sub>12</sub>** are much more meaningful. Because ML developers are expected to know how the two networks are merged, we believe most ML developers can interpret these hypotheses correctly in conjunction with their knowledge about their CNNs.

## 7 INDEPENDENT FEEDBACK

The VA technique proposed in this work consists of several components, which are new and are expected to take some time to be adopted by the wide ML community. These components include (i) the overall testing method, (ii) the logical inference rules, and (iii) the visual design of the HypoML tool. As shown in Section 6 and Appendix C, the author team, which includes ML specialists, have used HypoML extensively. This allowed the team to validate the correctness of (i) and (ii), and evaluate (iii). The numerical values, the computed inference results, and the visual summary of the these shown in Section 6 and Appendix C provide objective evidence to demonstrate the usefulness of this new VA technique and the HypoML tool.

To complement the objective evaluation in Section 6 and Appendix C, we sought independent feedback from four ML specialists who are not part of the author team. They include (a) a faculty member with 20 years of ML research experience, (b) a senior industrial researcher with more than five years of ML experience, (c) an industrial researcher and a doctoral student with about five years of ML experience, and (d) a doctoral student with about four years of ML experience. Appendix D details the interview procedure and questions. We organized our discussions around the following topics:

**Existing Methods for Considering Features.** Three specialists (a), (b), (c) use confusion matrix to observe mistakes made by learned models. (b) sometimes browses mistakes made by a model one by one. (c), (d) sometimes view activation or similar plots, hoping to find interesting patterns. (d) mainly use summary statistics as required by

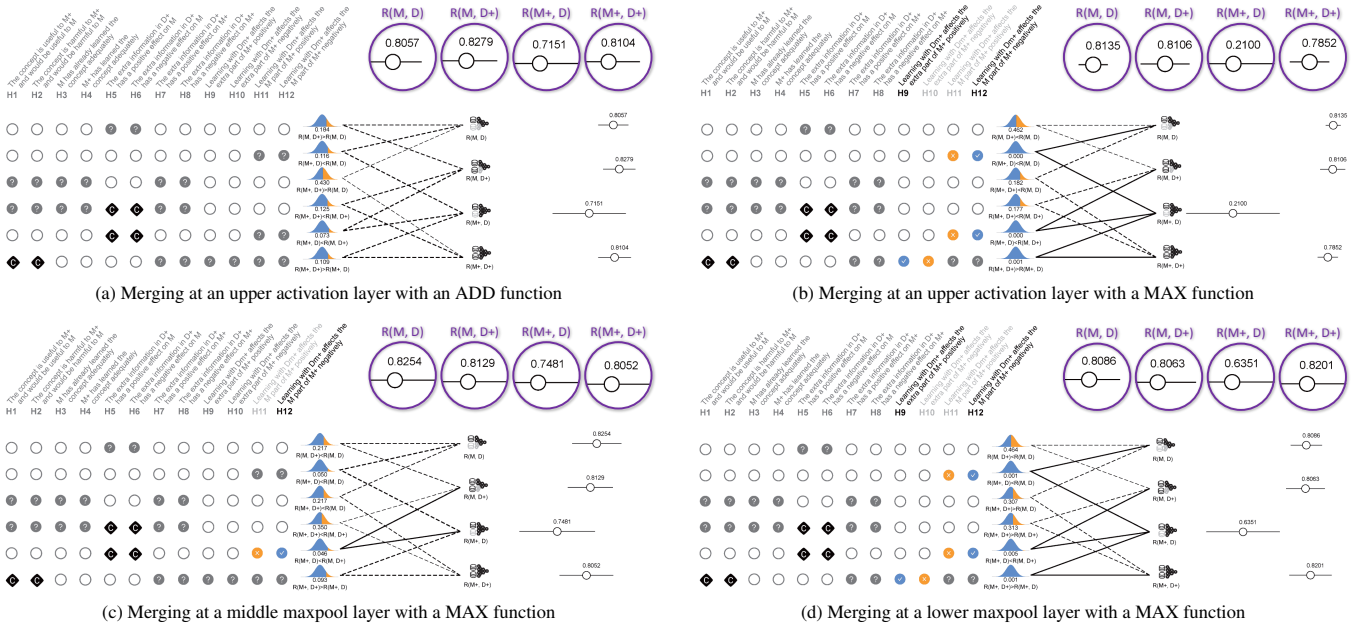


Fig. 10. Using HypoML to compare two models: M for RGB only and M+ for RGB+HSV. This enables the evaluation of the hypothesis if HSV is useful in addition to RGB, while observing the effect of different types of integration. Purple circles are zoomed-in views of the model results.

publications and competitions. ► **Reflection:** *All these methods can stimulate hypotheses about features. HypoML completes these methods by offering a structured method for evaluating hypotheses.*

**Necessity of Hypothesis Testing.** When facing a question about how much difference (e.g., accuracy 84.9% 84.8%), all four specialists stated no easy answer as the difference may be caused by some unknown facts. (b), (d) considered that it may depend on how difficult the task is, or the size of the testing data. (b) added: “For industrial problems, we do not always have big dataset, and hypothesis testing will be useful.” (c) reasoned: “According to information theory, I know hypothesis testing is needed even for big datasets.” None of the interviewees were aware of anyone using hypothesis testing to evaluate ML models. ► **Reflection:** *Most ML developers are cautious about some differences when comparing models, though some may not be aware that testing with a big dataset can still yield results that are statistically insignificant. Statistical hypothesis testing provides the subsequent logical influences with a standardized and consist basis.*

**Usefulness of Feature-based Hypothesis Testing.** All four specialists confirmed that they had hypotheses all the time, e.g., about the influence of background, size, color, position, especially when a model exhibited unexpected errors. To evaluate a hypothesis, they would make modification to the model structure, training data, or some training parameters, and then compared the performance of the old and new models. (d) commented: “For hypotheses about features, it is not always easy to change a model structure or find new training data.” (a) added: “For many medical imaging applications, we use both ‘bag of features’ and CNN models. One always has difficulties with some CNN models, and I can think about testing some CNN models against some features now. I should ask my students to download the software.” (c) concluded: “I have worked with models with image and text inputs, I know HypoML will work for these applications.” (b) reasoned: “[The workflow] from the model results to statistical comparison and then to hypothesis logic works for me.” (d) indicated: “I would like to use the visualization to show the results to non-expert model users.” ► **Reflection:** *The feedback shows that ML specialists can anticipate the potential uses of HypoML, and appreciate that designing a test for a feature will be easier than modifying a model structure and acquiring new training data. This is similar to education where examinations are used to find out any missing knowledge and inform the modification course syllabuses or teaching methods.*

**HypoML Visual Designs.** We also asked the four ML specialists to comment on the four visual designs mentioned in Section 5. In general,

they considered that they could work with all four visual designs, and there was slightly more preference for the vertical design shown in Fig. 6, followed by the horizontal design in Figs. 1 and 9. Further details of the four ML specialists’ comments can be found in Appendix B.

**Suggestions for Future Work.** (d) suggested: “I would like to test how independent or integrated two features are. I wonder if I need to change the logical influences rules.” (c) mentioned: “I am working with graph inputs. I may need to change the two parts of the inputs.” (d) indicated: “I have other types of hypotheses beyond the 12 listed.” ► **Reflection:** *It is encouraging to see that ML specialists are already thinking about extending various components of HypoML.*

## 8 CONCLUSIONS

In this paper, we propose a novel testing framework to aid the evaluation of ML models. In particular, this framework tests a set of hypotheses about a concept, checking whether extra information about the concept can benefit an ML model, and if so, how the extra information affects the model. The testing framework is underpinned by statistical analysis of the experiment results as well as logical inferences about the relations between six statistical conclusions and twelve hypotheses. Through an implementation of this framework HypoML, we demonstrate that with a purposely-designed visual representation, model-developers can visualize the conclusions about the twelve hypotheses as soon as the four sets of testing result data become available. This approach complements the traditional way of observing various plots for monitoring neuron activities, such as activation plots and gradient ascent plots. Model-developers, who observe any interesting patterns or failed to find desired patterns, can now formulate a concept-based hypothesis and carry out a structured test to evaluate their hypotheses.

We recognize that HypoML is only one of the many steps towards an ultimate goal of developing a powerful testing suite for evaluating, understanding, and explaining ML models. There is a need for further theoretical developments, including, e.g., formulating more **complex logical inference for sub-group, multi-model and multi-concept** analysis of the testing results, designing an advanced user interface for supporting detailed observation of sub-group analysis, and integrating with other visualization techniques for observing, understanding, and explaining ML models. **While HypoML allows users to answer new questions about ML models, it also demands extra computational cost for model training and testing. Our experiments showed that using lower-resolution images could reduce such cost. However, further investigation is necessary for gaining a full understanding in this respect.**

## REFERENCES

- [1] E. Alpaydin. *Introduction to Machine Learning*. The MIT Press, 2nd ed., 2010.
- [2] S. Amershi, M. Chickering, S. M. Drucker, B. Lee, P. Simard, and J. Suh. Modeltracker: Redesigning performance analysis tools for machine learning. In *Proc. 33rd Annual ACM Conference on Human Factors in Computing Systems*, pp. 337–346. ACM, Seoul, Republic of Korea, 2015.
- [3] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*, 2020.
- [4] R. Borgo, A. Abdul-Rahman, F. Mohamed, P. W. Grant, I. Reppe, L. Floridi, and M. Chen. An empirical study on using visual embellishments in visualization. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2759–2768, 2012.
- [5] H.-C. Chao, C.-Y. Chen, and F.-H. Tseng, eds. *Sensors: Special Issue on Recent Advances in Artificial Intelligence and Deep Learning for Sensor Information Fusion*, vol. 19(6). 2019.
- [6] F. Chollet et al. Keras CNN examples. [https://keras.io/examples/mnist\\_cnn/](https://keras.io/examples/mnist_cnn/), 2014. Accessed: 2019-12-03.
- [7] W. S. Cleveland and R. McGill. Graphical perception: Theory, experimentation, and application to the development of graphical methods. *Journal of the American Statistical Association*, 79(387):531–554, 1984.
- [8] J. F. Elder IV. Machine learning, neural, and statistical classification. *Journal of the American Statistical Association*, 91(433):436–439, 1996.
- [9] A. Endert, W. Ribarsky, C. Turkyay, B. L. W. Wong, I. Nabney, I. D. Blanco, and F. Rossi. The state of the art in integrating machine learning into visual analytics. *Computer Graphics Forum*, 36(8):458–486, 2017.
- [10] S. N. Gowda and C. Yuan. ColorNet: Investigating the importance of colorspace for image classification. In C. Jawahar, H. Li, G. Mori, and K. Schindler, eds., *Proc. Asian Conference on Computer Vision*, vol. LNCS, 11364. Springer, 2018.
- [11] F. Hohman, A. Head, R. Caruana, R. DeLine, and S. M. Drucker. GAMUT: A design probe to understand how data scientists understand machine learning models. In *Proc. ACM CHI conference on human factors in computing systems*, pp. 1–13, 2019.
- [12] F. Hohman, H. Park, C. Robinson, and D. H. P. Chau. SUMMIT: Scaling deep learning interpretability by visualizing activation and attribution summarizations. *IEEE Transactions on Visualization and Computer Graphics*, 26(1), 2020.
- [13] M.-X. Jiang, C. Deng, J.-S. Shan, Y.-Y. Wang, Y.-J. Jia, and X. Sun. Hierarchical multi-modal fusion FCN with attention model for RGB-D tracking. *Information Fusion*, 50:1–8, 2019.
- [14] M. Kahng, P. Y. Andrews, A. Kalro, and D. H. Chau. ActiVis: visual exploration of industry-scale deep neural network models. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):88–97, 2018.
- [15] M. Kahng, N. Thorat, D. H. Chau, F. B. Viégas, and M. Wattenberg. GAN Lab: Understanding complex deep generative models using interactive visual experimentation. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):310–320, 2019.
- [16] N. Kilbertus, M. Rojas-Carulla, G. Parascandolo, M. Hardt, D. Janzing, and B. Schölkopf. Avoiding discrimination through causal reasoning. In *Proc. 31st International Conference on Neural Information Processing Systems*, pp. 656–666. Long Beach, CA, USA, 2017.
- [17] B. Kim, M. Wattenberg, J. Gilmer, C. Cai, J. Wexler, F. Viegas, and R. Sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *Proc. 35th International Conference on Machine Learning*, 2018.
- [18] J. Krause, A. Perer, and K. Ng. Interacting with predictions: Visual inspection of black-box machine learning models. In *Proc. ACM CHI Conference on Human Factors in Computing Systems*, pp. 5686–5697. San Jose, CA, USA, 2016.
- [19] A. Krizhevsky. Learning multiple layers of features from tiny images. <https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf>, 2009.
- [20] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.
- [21] D. Liu, W. Cui, K. Jin, Y. Guo, and H. Qu. Deeptacker: Visualizing the training process of convolutional neural networks. *ACM Transactions on Intelligent Systems and Technology*, 2018.
- [22] M. Liu, J. Shi, K. Cao, J. Zhu, and S. Liu. Analyzing the training processes of deep generative models. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):77–87, 2018.
- [23] M. Liu, J. Shi, Z. Li, C. Li, J. Zhu, and S. Liu. Towards better analysis of deep convolutional neural networks. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):91–100, 2017.
- [24] Y. Ming, S. Cao, R. Zhang, Z. Li, Y. Chen, Y. Song, and H. Qu. Understanding hidden memories of recurrent neural networks. *arXiv preprint arXiv:1710.10777*, 2017.
- [25] N. Pezzotti, T. Höllt, J. Van Gemert, B. P. Lelieveldt, E. Eisemann, and A. Vilanova. Deepeyes: Progressive visual analytics for designing deep neural networks. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):98–108, 2018.
- [26] P. E. Rauber, S. G. Fadel, A. X. Falcao, and A. C. Telea. Visualizing the hidden activity of artificial neural networks. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):101–110, 2017.
- [27] D. Ren, S. Amershi, B. Lee, J. Suh, and J. D. Williams. Squares: Supporting interactive performance analysis for multiclass classifiers. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):61–70, 2017.
- [28] F. Rodrigues, I. Markou, and F. C. Pereira. Combining time-series and textual data for taxi demand prediction in event areas: A deep learning approach. *Information Fusion*, 49:120–129, 2019.
- [29] D. Sacha, M. Kraus, D. A. Keim, and M. Chen. VIS4ML: An ontology for visual analytics assisted machine learning. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):385–395, 2019.
- [30] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. Grad-CAM: Visual explanations from deep networks via gradient-based localization. In *Proc. International Conference on Computer Vision (ICCV)*, pp. 618–626. IEEE, 2017.
- [31] P. Y. Simard, D. Steinkraus, and J. Platt. Best practices for convolutional neural networks applied to visual document analysis. In *Proc. 7th International Conference on Document Analysis and Recognition*, 2003.
- [32] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller. Striving for simplicity: The all convolutional net. *arXiv 1412.6806*, 2014.
- [33] H. Strobelt, S. Gehrmann, H. Pfister, and A. M. Rush. LSTMVis: A tool for visual analysis of hidden state dynamics in recurrent neural networks. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):667–676, 2018.
- [34] G. K. L. Tam, V. Kothari, and M. Chen. An analysis of machine- and human-analytics in classification. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):71–80, 2017.
- [35] J. Wang, L. Gou, H. Yang, and H. Shen. Ganviz: A visual analytics approach to understand the adversarial game. *IEEE Transactions on Visualization and Computer Graphics*, 24(6):1905–1917, 2018.
- [36] J. Wexler, M. Pushkarna, T. Bolukbasi, M. Wattenberg, F. Viégas, and J. Wilson. The what-if tool: Interactive probing of machine learning models. *IEEE transactions on visualization and computer graphics*, 26(1):56–65, 2019.
- [37] H. Xiao, K. Rasul, and R. Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv 1708.07747*, 2017.
- [38] J. Yosinski, J. Clune, T. Fuchs, and H. Lipson. Understanding neural networks through deep visualization. In *Proc. 32nd International Conference on Machine Learning*. Lille, France, 2015.
- [39] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *Proc. 13th European Conference on Computer Vision*, pp. 818–833. Springer, 2014.
- [40] J. Zhang, Y. Wang, P. Molino, L. Li, and D. S. Ebert. Manifold: A model-agnostic framework for interpretation and diagnosis of machine learning models. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):364–373, 2019.
- [41] L. Zhang, Y. Wu, and X. Wu. Achieving non-discrimination in prediction. In *Proc. 27th International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 3097–3103. Stockholm, Sweden, 2018.
- [42] Q.-s. Zhang and S.-c. Zhu. Visual interpretability for deep learning: a survey. *Frontiers of Information Technology & Electronic Engineering*, 19(1):27–39, Jan 2018. doi: 10.1631/fitee.1700808
- [43] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning deep features for discriminative localization. In *Proc. Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2921–2929. IEEE, 2016.



# APPENDICES

## HypoML: Visual Analysis for Hypothesis-based Evaluation of Machine Learning Models

Qianwen Wang, HKUST\*, Hong Kong, China  
William Alexander, University of Oxford, UK  
Jack Pegg, University of Oxford, UK  
Huamin Qu, HKUST\*, Hong Kong, China  
Min Chen, University of Oxford, UK

\* HKUST: Hong Kong University of Science and Technology

### A FURTHER DETAILS OF LOGICAL INFERENCE RULES

In this Appendix, we describe the six sets of logical inference rules in Section 4 using plain English text. It is necessary to note, the main reason why these rules were defined mathematically is because the mathematical descriptions are concise, precise, easy to check and verify by experts. If there were errors found, it would be easy to redefine and correct. As these rules are used for automated computation, their complexity will not affect the users directly.

Before reading the text below this paragraph, readers are advised to first read Sections 3 and 4 where the mathematical notations for

- The 12 hypotheses:  $\mathbf{H}_1, \mathbf{H}_2, \dots, \mathbf{H}_{12}$ ;
- The four types of results:  $R_{M,D}, R_{M,D+}, R_{M+,D}, R_{M+,D+}$ ;
- The six sets of statistical analysis:  $A_1, A_2, \dots, A_6$ ;
- The symbols:  $\gtrsim, \lesssim, \approx, \approx, \approx, \implies, \top(S), \perp(S), \ast(S), \wedge, \vee$

are described in detail. As summarized in Table 1, the logical inference rules define the relations between the six sets of statistical analysis  $A_1, A_2, \dots, A_6$  and the 12 hypotheses  $\mathbf{H}_1, \mathbf{H}_2, \dots, \mathbf{H}_{12}$ .

In general, a hypothesis  $F$  about a concept  $\xi$  can be considered as a function  $F(\xi)$ . When a hypothesis is expressed in a sentence, all words related to the concept describe the variable  $\xi$ , while the remaining words describe the framing of the hypothesis. The latter is referred to as a *frame*, denoted as  $F()$ , which is independent of the variable  $\xi$ .

In HypoML, the hypothesis  $F(\xi)$  is evaluated as a sequence of processing and reasoning. It starts with two ML processes with two training datasets respectively, which result in two models  $M$  and  $M+$ . This is followed by four testing processes  $[M, M+] \times [D, D+]$ , resulting in four sets of testing results  $R_{M,D}, R_{M,D+}, R_{M+,D}$ , and  $R_{M+,D+}$ . The ML learning and testing processes do not have direct knowledge about the concept  $\xi$ , because  $\xi$  is encoded implicitly in the two types of input datasets.

From the above set-up of the ML training and testing processes, we found 12 hypothesis frames,  $F_i(), i = 1, 2, \dots, 12$ , that can be reasoned using the testing results. We have made an extensive effort to find as many frames as possible, but we cannot rule out the existence of other frames under this set-up. Of course, if one changes the set-up, one can certainly find new frames. Since there can be numerous designs of training and testing processes for multi-model and/or multi-concept ML, some future research will no doubt discover other useful set-ups and other meaningful hypothesis frames.

The process of statistical inference takes the four sets of testing results as the input, and produces six sets of statistical analysis  $A_1, A_2, \dots, A_6$ . This process no longer requires any knowledge about the models and datasets except the labels used to differentiate the four sets of results. It is thus mathematically-independent of the ML models, the training and testing data, and the concept  $\xi$ .

The process of logical inference takes six sets of statistical analysis as the input, and produces the conclusions about the 12 hypotheses  $F_i(\xi), i = 1, 2, \dots, 12$ . For each hypothesis frame,  $F_i()$ , there is a set of logical rules  $\mathbf{H}_i$  for transforming the six sets of statistical analysis to a conclusion. Clearly, these logical rules are mathematically-independent of the ML models, the training and testing data, and the concept  $\xi$ . Because the hypothesis frames are organized in pairs, in the following six subsections, we describe the rules for the 12 hypothesis frames.

### A.1 Reasoning $A_1$ and $\mathbf{H}_1, \mathbf{H}_2, \mathbf{H}_3, \mathbf{H}_4, \mathbf{H}_7, \mathbf{H}_8$

The statistical analysis  $A_1$  reflects the conventional comparison between  $R_{M+,D+}$  and  $R_{M,D}$ . As the simple comparison of the mean accuracy values may not be statistically meaningful,  $t$ -test provides a more reliable assessment. If the  $t$ -test confirms  $R_{M+,D+} > R_{M,D}$  is statistically meaningful according to a predefined threshold  $p$  value (typically 0.05 or 0.01), we denote this as  $R_{M+,D+} \gtrsim R_{M,D}$ . Such a statistical conclusion provides a basis for us to infer the following:

- Most likely the concept  $\xi$  featured in the  $+$  part of  $D+$  is useful. In other words, the hypothesis  $\mathbf{H}_1$  is most likely to be true.
- Most likely the model  $M+$  has learned the concept  $\xi$  adequately (enough to be better than  $M$ ). In other words, the hypothesis  $\mathbf{H}_4$  is most likely to be true.
- Most likely the extra information in the  $+$  part of  $D+$  has a positive effect on the model  $M+$ . In other words, the hypothesis  $\mathbf{H}_7$  is most likely to be true.
- Most unlikely the concept  $\xi$  featured in the  $+$  part of  $D+$  is harmful, otherwise  $M+$  would not be better than  $M$ . In other words, the hypothesis  $\mathbf{H}_2$  is most likely to be false.
- Most unlikely the model  $M$  has learned the concept  $\xi$  adequately, i.e., not adequately enough for  $M$  to be better than  $M$ . In other words, the hypothesis  $\mathbf{H}_3$  is most likely to be false.
- Most unlikely the extra information in the  $+$  part of  $D+$  has a negative effect on the model  $M+$ . In other words, the hypothesis  $\mathbf{H}_8$  is most likely to be false.

where the level of likelihood or unlikelihood depends on the predefined threshold  $p$  value. Using the mathematical notations given in Section 4, this list of inferences can be written simply as  $R_{M+,D+} \gtrsim R_{M,D} \implies \top(\mathbf{H}_1) \wedge \top(\mathbf{H}_4) \wedge \top(\mathbf{H}_7) \wedge \perp(\mathbf{H}_2) \wedge \perp(\mathbf{H}_3) \wedge \perp(\mathbf{H}_8)$ .

It is necessary to note that  $A_1$  cannot be used to draw any conclusion about  $\mathbf{H}_5, \mathbf{H}_6, \mathbf{H}_9, \mathbf{H}_{10}, \mathbf{H}_{11}$ , and  $\mathbf{H}_{12}$ . This can be expressed as  $\ast(\mathbf{H}_5) \wedge \ast(\mathbf{H}_6) \wedge \ast(\mathbf{H}_9) \wedge \ast(\mathbf{H}_{10}) \wedge \ast(\mathbf{H}_{11}) \wedge \ast(\mathbf{H}_{12})$ .

On the other hand, if the testing results show  $R_{M+,D+} < R_{M,D}$ , and the  $t$ -test confirms the “less than” is statistically meaningful, we denote this as  $R_{M+,D+} \lesssim R_{M,D}$ . Such a statistical conclusion allows us to infer the following:

- Most likely the concept  $\xi$  featured in the  $+$  part of  $D+$  is harmful. In other words, the hypothesis  $\mathbf{H}_2$  is most likely to be true.
- Most likely the model  $M+$  has learned the concept  $\xi$  adequately despite that the consequence is harmful. In other words, the hypothesis  $\mathbf{H}_4$  is most likely to be true.
- Most unlikely the concept  $\xi$  featured in the  $+$  part of  $D+$  is useful, otherwise  $M+$  would be no worse than  $M$ . In other words, the hypothesis  $\mathbf{H}_1$  is most likely to be false.
- Most unlikely the model  $M$  has learned the concept  $\xi$  adequately. In other words, the hypothesis  $\mathbf{H}_3$  is most likely to be false.

This list of inferences can be expressed as  $R_{M+,D+} \lesssim R_{M,D} \implies \top(\mathbf{H}_2) \wedge \top(\mathbf{H}_4) \wedge \perp(\mathbf{H}_1) \wedge \perp(\mathbf{H}_3)$ . Similarly, we can also express those inconclusive hypotheses using the  $\ast$  symbol.

There is a third condition, denoted as  $R_{M+,D+} \approx R_{M,D}$ , where  $R_{M+,D+} = R_{M,D}$  or the  $t$ -test shows that the comparison is not statistically meaningful despite  $R_{M+,D+} > R_{M,D}$  or  $R_{M+,D+} < R_{M,D}$ . Under this condition, all 12 hypotheses are inconclusive. To simplify our logical formulae further, we will not explicitly list those inconclusive hypotheses. With this simplification, we can now express the inferences under all three conditions as a combined set of logical rules:

$A_1$ : Comparing  $R_{M+,D+}$  and  $R_{M,D}$  may conclude:

- $R_{M+,D+} \gtrsim R_{M,D} \implies \top(\mathbf{H}_1) \wedge \top(\mathbf{H}_4) \wedge \top(\mathbf{H}_7) \wedge \perp(\mathbf{H}_2) \wedge \perp(\mathbf{H}_3) \wedge \perp(\mathbf{H}_8)$ .
- $R_{M+,D+} \lesssim R_{M,D} \implies \top(\mathbf{H}_2) \wedge \top(\mathbf{H}_4) \wedge \perp(\mathbf{H}_1) \wedge \perp(\mathbf{H}_3)$ .

It may be helpful to reassert that the above mathematical representation does not explicitly show the inconclusive hypotheses, including six other hypotheses under the condition  $\gtrsim$ , eight hypotheses under the condition  $\lesssim$ , and all 12 hypotheses under the condition  $\approx$ .

One may notice that the inferences resulting from  $R_{M+,D+} \not\approx R_{M,D}$  do not include  $H_7$  and  $H_8$ . On the surface, when  $M+$  performs worse, it might imply that the  $+$  part of  $D+$  has a negative effect on  $M+$ . However, this is not the only possible explanation. It is also possible that while the extra information in the  $+$  part of the data may still have a positive effect on  $M+$ , but curiously it also has a positive effect on  $M$ , and even made  $M$  perform better. In other words, the  $+$  part of the data has a positive effect on both  $M$  and  $M+$ , and unexpectedly, more on  $M$ . Although this scenario is less likely, it is still possible if there are some anomalies in the structure of  $M$  and  $M+$  and in their training processes. We will discuss this further in Section A.2 in relation to hypotheses  $H_5$  and  $H_6$ .

One may also notice that the hypotheses for  $H_1, H_2, H_3$ , and  $H_4$  are not symmetric under the  $\approx$  and  $\not\approx$  conditions. Under the  $\approx$  condition, we have  $\top(H_1) \wedge \top(H_4)$  with  $H_2, H_3$ , while under the  $\not\approx$  condition, we have  $\top(H_2) \wedge \top(H_4)$  with  $H_1, H_3$  inconclusive. This shows that it is necessary to separate  $H_1, H_2$  and  $H_3, H_4$ .

Some may wish to combine  $H_1$  with  $H_2$ , and  $H_7$  with  $H_8$ . Because when  $H_1$  is not false,  $H_2$  can be either true or inconclusive. The combination would require the logical conclusion of a hypothesis to have three states, i.e., “true”, “false”, and “inconclusive”. Although there are  $n$ -valued logic systems, they are not as widely-used as Boolean algebra. To our best knowledge, we are not aware of using a 3-valued logic system for hypothesis reasoning. Hence separating a tri-state hypothesis into two hypotheses adheres to the convention of hypothesis testing while enabling the formulation of logical inference rules using the widely-used Boolean algebra. This reasoning is also applicable to other hypotheses such as  $H_5$  and  $H_6, H_9$  and  $H_{10}$ , and  $H_{11}$  and  $H_{12}$ .

## A.2 Reasoning $A_2$ and $H_1, H_2, H_3, H_4, H_7, H_8$

Analysis  $A_2$  compares  $R_{M+,D+}$  with  $R_{M,D+}$ . Since model  $M$  is trained with noise from the  $+$  part of the data, some might assume that  $M$  would not be influenced by the  $+$  part of the data during the test. It is relatively common that this assumption does not hold, as it is often that the difference between  $R_{M,D+}$  and  $R_{M,D}$  is statistically significant. The impact of the noise from the  $+$  part on  $M$  depends on the structure of  $M$  and the training algorithm used. This is to be observed using hypotheses  $H_5$  and  $H_6$ , which will be discussed in conjunction with statistical analysis  $A_6$ .

Analysis  $A_2$  cannot draw conclusions about  $H_5$  and  $H_6$ , but its conclusion may depend on them. In general, there is a common-sense assumption that neither  $H_5$  nor  $H_6$  is likely to be true, unless there are some imperfections in the structure of  $M$  or its learning process. However, subject to their outcomes,  $A_2$  can be used to infer conclusions about  $H_1, H_2, H_3, H_4, H_7$ , and  $H_8$ .

The logical inferences based on  $A_2$  are organized with two levels of decomposition. It is first decomposed according to the three conditions,  $R_{M+,D+} \not\approx R_{M,D+}$ ,  $R_{M+,D+} \approx R_{M,D+}$ , and  $R_{M+,D+} \approx R_{M,D+}$  (not explicitly expressed).

Under the condition  $R_{M+,D+} \approx R_{M,D+}$ , it is further decomposed according to whether  $H_6$  is true, inconclusive, or false. If  $H_6$  is not true (i.e.,  $\perp(H_6)$  or  $\ast(H_6)$ ), the  $+$  part of the data does not have a negative impact on model  $M$ . When  $M+$  performs better than  $M$  as indicated by  $R_{M+,D+} \approx R_{M,D+}$ , one has to conclude that  $M+$  has benefited from the  $+$  part, since  $M$  was not disadvantaged. This leads to the same conclusions as in the first part of the inference rules based on  $A_1$ .

However, if  $H_6$  is true (i.e.,  $\top(H_6)$ ), we cannot draw the same set of conclusions, since the better performance of  $M+$  could possibly be because  $M$  was disadvantaged by the  $+$  part of the data. In general, many ML developers may assume that  $H_6$  is unlikely to be true, and will likely treat the conclusion  $\top(H_6)$  cautiously. With their knowledge about the structure of  $M$  and the training process, they will likely investigate the reason behind the conclusion that  $H_6$  is true.

Similarly, under the condition  $R_{M+,D+} \not\approx R_{M,D+}$ , it is further decomposed according to  $H_5$ . When  $H_5$  is not true (i.e.,  $\perp(H_5)$  or  $\ast(H_5)$ ), the  $+$  part of the data does not have a positive impact on model  $M$ . We can thus draw the same conclusions as the second part of the inference rules based on  $A_1$ . Because  $\top(H_5)$  rules out the possibility that the extra information in the  $D+$  data has a positive effect on  $M$ ,

we can safely infer conclusions for  $H_7$  and  $H_8$  under the condition  $R_{M+,D+} \not\approx R_{M,D+}$ .

Also similar to the condition  $\top(H_6)$ , when hypothesis  $H_5$  is true, it offers an explanation why  $R_{M+,D+} \not\approx R_{M,D+}$ . Since it is very unlikely that  $H_5$  is true unless there are some anomalies in the structure of  $M$  or its training process, ML developers will need to investigate the reason behind the conclusion that  $H_5$  is true.

With the two-level decomposition, the set of inference rules based on  $A_2$  can be expressed logically as:

---

$A_2$ : Comparing  $R_{M+,D+}$  and  $R_{M,D+}$  may conclude:

- $R_{M+,D+} \approx R_{M,D+} \implies$ 
    - (i) if  $\perp(H_6)$  then  $\top(H_1) \wedge \top(H_4) \wedge \top(H_7) \wedge \perp(H_2) \wedge \perp(H_3) \wedge \perp(H_8)$ ; or
    - (ii) if  $\ast(H_6)$  then  $\top(H_1) \wedge \top(H_4) \wedge \top(H_7) \wedge \perp(H_2) \wedge \perp(H_3) \wedge \perp(H_8)$ ; or
    - (iii) if  $\top(H_6)$ . This offers an explanation but it is against a common-sense assumption that  $H_6$  is unlikely to be true, and should be treated cautiously.
  - $R_{M+,D+} \not\approx R_{M,D+} \implies$ 
    - (i) if  $\perp(H_5)$  then  $\top(H_2) \wedge \top(H_4) \wedge \top(H_8) \wedge \perp(H_1) \wedge \perp(H_3) \wedge \perp(H_7)$ ; or
    - (ii) if  $\ast(H_5)$  then  $\top(H_2) \wedge \top(H_4) \wedge \top(H_8) \wedge \perp(H_1) \wedge \perp(H_3) \wedge \perp(H_7)$ ; or
    - (iii) if  $\top(H_5)$ . This offers an explanation but it is against a common-sense assumption that  $H_5$  is unlikely to be true, and should be treated cautiously.
- 

## A.3 Reasoning $A_3$ and $H_7, H_8, H_9, H_{10}, H_{11}, H_{12}$

Analysis  $A_3$  compares  $R_{M+,D+}$  with  $R_{M+,D}$ , i.e., comparing  $M+$  with  $M+$  with two different types of input data. Because analysis  $A_3$  does not compare  $M+$  with  $M$ , the conclusion is limited to the context of  $M+$ . Mathematically, it is possible for  $A_3$  to conclude that the concept is “useful” in the context of  $M+$ , while  $A_1$  or  $A_2$  concludes that the concept is harmful or is neither useful nor harmful. Considering this limitation, it is unsafe for this analysis to draw a conclusion about  $H_1$  and  $H_2$ .

Meanwhile the analysis depends on the conclusions of  $H_1$  and  $H_2$  in a small way. Hence, the inferences based on  $A_3$  are expressed with two levels of decomposition. Under the condition  $R_{M+,D+} \approx R_{M+,D}$ , we consider three situations:

- i. When  $H_1$  is true, i.e., the concept  $\xi$  is useful, we can conclude from  $R_{M+,D+} \approx R_{M+,D}$  that:
  - Most likely the extra information in the  $+$  part of  $D+$  has a positive effect on the model  $M+$ . In other words, the hypothesis  $H_7$  is most likely to be true.
  - Most likely learning with the extra information affects the  $+$  part of  $M+$  positively. In other words, the hypothesis  $H_9$  is most likely to be true.
  - Most unlikely the extra information in the  $+$  part of  $D+$  has a negative effect on the model  $M+$ . In other words, the hypothesis  $H_8$  is most likely to be false.
  - Most unlikely learning with the extra information affects the  $+$  part of  $M+$  negatively. In other words, the hypothesis  $H_{10}$  is most likely to be false.
- ii. When  $H_1$  is inconclusive, we cannot infer  $H_7$  and  $H_8$ , since we do not know if the concept  $\xi$  is useful or harmful. However we can still make the same conclusion about  $H_9$  and  $H_{10}$  as above.
- iii. When  $H_1$  is false, i.e., the concept  $\xi$  is not useful,  $R_{M+,D+} \approx R_{M+,D}$  may not seem to make sense. The only explanation for this situation is phrased in hypothesis  $H_{12}$ . In all inference rules that feature  $\perp(H_1)$ , there are also  $\top(H_2)$ . Hence the concept  $\xi$  is most likely to be harmful and normally  $R_{M+,D+}$  would be

less than  $R_{M+,D}$ . However, if there are some anomalies in the structure of  $M+$  or its training process, which have caused the  $M$  part of  $M+$  to learn “wrong” intelligence from the  $+$  part of the data. During the test for obtaining  $R_{M+,D+}$ , the concept  $\xi$  in the  $+$  part of the data is harmful,  $M+$  may appear to have “correct” intelligence. During the test for obtaining  $R_{M+,D}$ , the harmful concept is not available, and  $M+$  cannot “benefit” from its “wrong” intelligence. Therefore, if this rather curious phenomenon (i.e.,  $\perp(H_1) \wedge (R_{M+,D+} \approx R_{M+,D})$ ) does happen, we can infer the followings:

- Most likely that learning with  $D_m+$  affects the  $M$  part of  $M+$  negatively. In other words, the hypothesis  $H_{12}$  is most likely to be true.
- Most unlikely that learning with  $D_m+$  affects the  $M$  part of  $M+$  positively. In other words, the hypothesis  $H_{11}$  is most likely to be false.

Similarly we can reason about  $R_{M+,D+} \approx R_{M+,D}$  based on the three conditions of  $H_2$ . The inferences under the conditions of  $\top(H_2)$  and  $\ast(H_2)$  mirror those discussed above in conjunction with the conditions of  $\top(H_1)$  and  $\ast(H_1)$  when  $R_{M+,D+} \approx R_{M+,D}$ . There is a similar curious phenomenon, i.e.,  $\perp(H_2) \wedge (R_{M+,D+} \approx R_{M+,D})$ . One possible explanation is that there is some anomalies in the structure of  $M+$  or its training process, which have made  $+$  part of  $M+$  to learn “wrong” intelligence from the  $+$  part of the data. This explanation leads to the following inferences about  $H_9$  and  $H_{10}$ :

- Most likely that learning with the extra information affects the  $+$  part of  $M+$  negatively. In other words, the hypothesis  $H_{10}$  is most likely to be true.
- Most unlikely that learning with the extra information affects the  $+$  part of  $M+$  positively. In other words, the hypothesis  $H_9$  is most likely to be false.

One may notice that the explanation does not include hypotheses  $H_{11}$  and  $H_{12}$ . This is because as long as we have  $\top(H_{10}) \wedge \perp(H_9)$ , a condition of  $\top(H_{11}) \wedge \perp(H_{12})$  is possible but not necessary.

It is necessary to note that this explanation appears to contradict with  $\perp(H_2)$ , and hence it should be treated cautiously. Because what underlines the logical inferences is statistical analysis, such an unusual contradiction reflects borderline statistical conclusions. Nevertheless, it is better for each set of inference rules to offer relatively independent conclusions based on its statistical analysis (i.e.,  $A_3$  in this case), and show all the logical inferences through the visualization. If there is such a contradiction due to borderline statistics, it is better to show it to the users and let them make a further observation of the statistics, instead of hiding the situation from the users by forcefully making a logical conclusion one way or another.

With the two-level decomposition, the set of inference rules based on  $A_3$  can be expressed logically as:

---

$A_3$ : Comparing  $R_{M+,D+}$  and  $R_{M+,D}$  may conclude:

- $R_{M+,D+} \approx R_{M+,D} \implies$ 
    - (i) if  $\top(H_1)$ , then  $\top(H_7) \wedge \top(H_9) \wedge \perp(H_8) \wedge \perp(H_{10})$ ; or
    - (ii) if  $\ast(H_1)$ , then  $\top(H_9) \wedge \perp(H_{10})$ ; or
    - (iii) if  $\perp(H_1)$ , then  $\top(H_{12}) \wedge \perp(H_{11})$ .
  - $R_{M+,D+} \lesssim R_{M+,D} \implies$ 
    - (i) if  $\top(H_2)$ , then  $\top(H_8) \wedge \top(H_{10}) \wedge \perp(H_7) \wedge \perp(H_9)$ ; or
    - (ii) if  $\ast(H_2)$ , then  $\top(H_{10}) \wedge \perp(H_9)$ ; or
    - (iii) if  $\perp(H_2)$ , then  $\top(H_{10}) \wedge \perp(H_9)$ . This conclusion is against a common-sense assumption that a useful concept normally should not affect the extra part of  $M+$  negatively, and should be treated cautiously.
- 

The interpretation of hypotheses  $H_9$ ,  $H_{10}$ ,  $H_{11}$ , and  $H_{12}$  requires users to have some reasonable knowledge about their own models, and in particular the structure of  $M+$ , in order to be aware of how

separated or integrated the  $M$  and  $+$  parts are in  $M+$ . As we have already discussed the separation and integration of the  $M$  and  $+$  parts in  $M+$  in Section 4, and showed examples in Section 6, we will not repeat this in this appendix.

#### A.4 Reasoning $A_4$ and $H_{11}$ , $H_{12}$

Analysis  $A_4$  compares  $R_{M+,D}$  with  $R_{M,D}$ , which can reveal how model  $M+$ , which is trained with the  $+$  part of the data, may perform without that part of the data. Ideally,  $M+$  should perform at least the same as  $M$  in such a situation. However, in practice,  $M+$  often performs worse than  $M$ , suggesting that learning with an additional concept could potentially hinder some aspects of the “normal” learning process. This is phrased as hypothesis  $H_{12}$ : learning with  $D_m+$  affects the  $M$  part of  $M+$  negatively.

It is common in human education, learning an additional concept may strengthen the learning of other concepts. Hence it is reasonable to consider also the possibility of hypothesis  $H_{11}$ : learning with  $D_m+$  affects the  $M$  part of  $M+$  positively.

As we have already discussed the separation and integration of the  $M$  and  $+$  parts of  $M+$  in Section 4, and showed examples in Section 6, we will not repeat this in this appendix. Assuming that the users who have good knowledge about the structure of their models can interpret the meaning of the  $M$  and  $+$  parts correctly, analysis  $A_4$  is relatively easy to reason. If  $R_{M+,D} \approx R_{M,D}$ ,  $H_{11}$  is true and  $H_{12}$  is false. If  $R_{M+,D} \lesssim R_{M,D}$ ,  $H_{11}$  is false and  $H_{12}$  is true. If  $R_{M+,D} \approx R_{M,D}$ , both  $H_{11}$  and  $H_{12}$  are inconclusive. These are logically expressed as:

---

$A_4$ : Comparing  $R_{M+,D}$  and  $R_{M,D}$  may conclude:

- $R_{M+,D} \approx R_{M,D} \implies \top(H_{11}) \wedge \perp(H_{12})$ .
  - $R_{M+,D} \lesssim R_{M,D} \implies \top(H_{12}) \wedge \perp(H_{11})$ .
- 

#### A.5 Reasoning $A_5$ and $H_{11}$ , $H_{12}$

Analysis  $A_5$  compares  $R_{M+,D}$  with  $R_{M,D+}$ . In the two tests concerned,  $M+$  is denied by the extra information about the concept  $\xi$ , while  $M$  is fed with the extra information that it did not learn to handle. Hence we cannot use this comparison to inform us about whether the concept  $\xi$  is useful or not. Nevertheless, this comparison can help us make some observations about how the learning with and without extra information has affected the development of “intelligence” in  $M+$  and  $M$ .

Similar to  $A_2$ ,  $A_5$  cannot draw conclusions about  $H_5$  and  $H_6$ , but its conclusion may depend on them. In general, there is a common-sense assumption that neither  $H_5$  nor  $H_6$  is likely to be true unless there are some imperfections in the structure of  $M$  or its learning process. However, subject to their outcomes,  $A_5$  can be used to infer conclusions about  $H_{11}$ ,  $H_{12}$ .

The logical inferences based on  $A_5$  are expressed with two levels of decomposition. When  $R_{M+,D} \approx R_{M,D+}$ , the conditions of  $\perp(H_6)$  and  $\ast(H_6)$  indicate that the extra information may have a non-negative effect on  $M$ . Since  $M+$  performs better without the extra information,  $M+$  must have gained some advantages in its  $M$  part through the process of learning with the extra information. This leads to conclusions that most likely  $H_{11}$  is true and  $H_{12}$  is false. When  $H_6$  is true, it offers an explanation of why we have  $R_{M+,D} \approx R_{M,D+}$ . As this is not consistent with the common-sense assumption, further investigation is desirable.

When  $R_{M+,D} \lesssim R_{M,D+}$ , the conditions of  $\perp(H_5)$  and  $\ast(H_5)$  indicate that the extra information may have a non-positive effect on  $M$ . Since  $M+$  performs worse without the extra information,  $M+$  must have lost some advantages in its  $M$  part through the process of learning with the extra information. This leads to conclusions that most likely  $H_{12}$  is true and  $H_{11}$  is false. Similarly, when  $H_5$  is true, further investigation is desirable.

With the two-level decomposition, the set of inference rules based on  $A_5$  can be expressed logically as:

---

$A_5$ : Comparing  $R_{M+,D}$  and  $R_{M,D+}$  may conclude:



- $R_{M,D} \gtrsim R_{M,D+} \Rightarrow$   
 (i) if  $\perp(H_6)$  then  $\top(H_{11}) \wedge \perp(H_{12})$ ; or  
 (ii) if  $\ast(H_6)$  then  $\top(H_{11}) \wedge \perp(H_{12})$ ; or  
 (iii) if  $\top(H_6)$ . This offers an explanation but it is against a common-sense assumption that  $H_6$  is unlikely to be true, and should be treated cautiously.
- $R_{M,D} \lesssim R_{M,D+} \Rightarrow$   
 (i) if  $\perp(H_5)$  then  $\top(H_{12}) \wedge \perp(H_{11})$ ; or  
 (ii) if  $\ast(H_5)$  then  $\top(H_{12}) \wedge \perp(H_{11})$ ; or  
 (iii) if  $\top(H_5)$ . This offers an explanation but it is against a common-sense assumption that  $H_6$  is unlikely to be true, and should be treated cautiously.

## A.6 Reasoning $A_6$ and $H_5$ , $H_6$

Analysis  $A_6$  compares  $R_{M,D+}$  with  $R_{M,D}$ , which allows us to reason about the less-expected situations where the  $+$  part of the data may affect model  $M$  positively or negatively.  $H_5$  or  $H_6$  hypothesize such situations. Among the six sets of statistical analysis,  $A_6$  is the only comparison that may inform the evaluation of  $H_5$  or  $H_6$ .

As discussed previously in conjunction with  $A_2$  and  $A_5$ , in general, there is a common-sense assumption that neither  $H_5$  nor  $H_6$  is true if the model template or structure was correctly defined, the correct ML method was followed, and the correct ML process was executed. When  $H_5$  or  $H_6$  is confirmed, it usually suggests some imperfections or anomalies in the model template or the training process. Therefore the conclusions of  $A_6$  should not be interpreted as their face values. However, the evaluation of  $H_5$  and  $H_6$  is necessary since  $A_2$  and  $A_5$  depend on them.

The logical inferences from  $A_6$  are relatively straight forward. When  $R_{M,D+} \gtrsim R_{M,D}$ , most likely  $H_5$  is true and  $H_6$  is false. When  $R_{M,D+} \lesssim R_{M,D}$ , the conclusion is the other way around. When  $R_{M,D+} \approx R_{M,D}$ ,  $H_5$  and  $H_6$  are inconclusive, which is not shown explicitly in the following logical expression.

---

$A_6$ : Comparing  $R_{M,D+}$  and  $R_{M,D}$  may conclude:

- $R_{M,D+} \gtrsim R_{M,D} \Rightarrow \top(H_5) \wedge \perp(H_6)$ ;
  - $R_{M,D+} \lesssim R_{M,D} \Rightarrow \top(H_6) \wedge \perp(H_5)$ .
- 

## B DIFFERENT LAYOUTS IN VISUAL DESIGN

As discussed in Section 5, we have considered a number of visual designs. Because each visual design takes a fair amount of space, we discussed four main visual designs in this appendix. They are labeled as A, B, C, and D, and are shown in Figs. 11-14.

The author team includes three members who collectively have published more than 10 papers on the topic of using visualization in ML and two members who have been focused solely on ML. During the development of HypoML, we reasoned the relative merits of these designs based on our experience of using HypoML to test various features and concepts (see also Section 6 and Appendix C, our informal discussions with ML developers within the team and beyond, and our knowledge about the visualization tasks, the typical objectives, work patterns, and knowledge of ML developers, and the perceptual and cognitive considerations of these designs.

Following the completion of HypoML, we consulted independent ML specialists to gather their opinions about the visual designs, in conjunction with other aspects of HypoML as already reported in Section 7. When asked about the four visual designs shown in 11-14, the four specialists had different opinions. ① commented: “I cannot see much difference among these designs. If I have to choose one, it might A as it is aesthetically neat.” ② said: “Basically they are similar. I might prefer D because some elements are duplicated in A and B.” ③ observed: “Once I have understood what are displayed, I can work with any of these design. I prefer C slightly more.” ④ noted: “I prefer D as I can follow the logical flow.” In terms of viewing different parts of the visualization, ①, ②, ④ indicated that they might first pay attention to the hypotheses part, while ③ might first pay attention to the bipartite graph in the middle. ③ added: “If HypoML is for non-expert users,

some of the 12 hypotheses may be removed. If it is for ML developers, it is effortless for us to choose which is more interesting in different situations.” ► Reflection: *It is not always easy for potential users to evaluate different visual designs without some experience of using each of them in practice.*

In the list below we detail our assessment of the merits and demerits of these designs, juxtaposed with the independent feedback as reported in Section 7.

- **Design A.** It has two separate display areas, one shows the relationship between the 12 hypotheses and the six sets of statistical analysis, and another shows the four mean accuracy values and their relationship with the six sets of statistical analysis.

**Merits.** The hypotheses are written horizontally, and they are easy to read. The two separate areas enable new users to pay attention to the left area for the tasks of observing the conclusions of hypotheses, observing statistical analysis, and their relationships, and to the right area for the tasks of observing mean accuracy values, statistical analysis, and their relationships.

**Demerits.** The flow from hypotheses to mean accuracy is broken. A user normally needs to follow the flow to inspect some detailed statistical measures and analysis when a hypothesis is unexpectedly confirmed or refuted. It requires some extra cognitive load to connect the two duplicated sets of statistical analysis.

**Independent Feedback.** Some components are duplicated (i.e., the six sets of statistical analysis). Aesthetically, the design is neat. One independent ML specialist preferred this design.

- **Design B.** It takes the separation idea further by splitting the right area into two areas. Each of the six pairwise comparisons is shown directly with two mean accuracy values and a  $p$ -value without using connections as in A.

**Merits.** Each set of statistical analysis is directly associated with its two accuracy values being compared. This demands less cognitive load for tracing through the connection lines.

**Demerits.** The hypotheses are written diagonally, and are not easy to read for new users before they become accustomed to the numerical order and positional cue. The flow from hypotheses to mean accuracy is broken. A user normally needs to follow the flow to inspect some detailed statistical measures and analysis when a hypothesis is unexpectedly confirmed or refuted. It requires some extra cognitive load to connect the two duplicated sets of statistical analysis. Each of the four mean accuracy values in the top-right area connects to its three copies in bottom-right area through real numbers, demanding further cognitive load to make such connections.

**Independent Feedback.** Some components are duplicated (i.e., the six sets of statistical analysis and the four mean accuracy values. None of the four independent ML specialists preferred this design.

- **Design C.** It takes an integrated approach by showing the flow from hypotheses on the left to statistical analysis in the middle and then to the mean accuracy values on the right. It is suitable for both desktop displays and mobile devices.

**Merits.** It is easy to follow the flow continuously from the conclusions of the hypotheses to mean accuracy values and vice versa.

**Demerits.** The hypotheses are written diagonally, and are not easy to read for new users before they become accustomed to the numerical order and positional cue.

**Independent Feedback.** One independent ML specialist preferred this design.

- **Design D.** Similar to C, it takes an integrated approach by showing the flow from hypotheses on the top to statistical analysis in the middle and then to the mean accuracy values on the bottom.

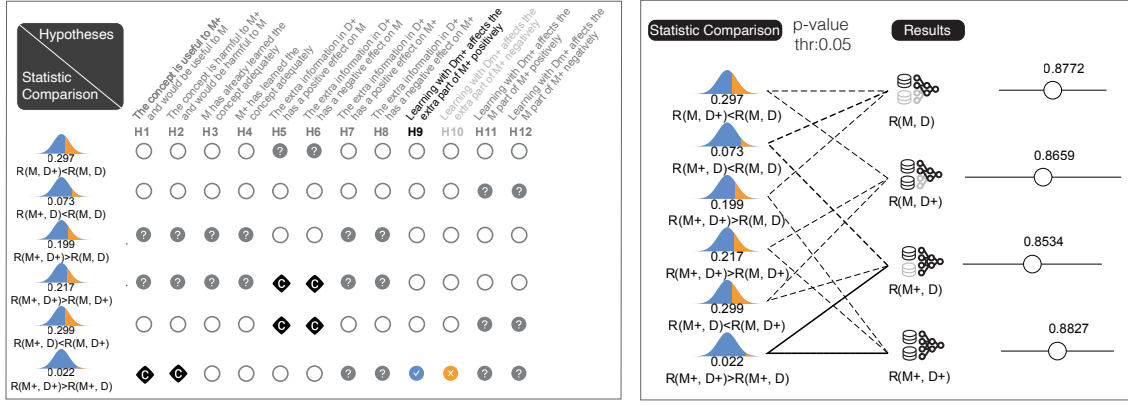


Fig. 11. Design A: Two separate display areas.



Fig. 12. Design B: Three separate display areas.

**Merits.** The hypotheses are written horizontally, and they are easy to read. It is easy to follow the flow continuously from the conclusions of the hypotheses to mean accuracy values and vice versa. The layout can be used naturally on mobile devices, such as tablets.

**Demerits.** The vertical layout requires the support of a scroll bar on a typical desktop display.

**Independent Feedback.** Two independent ML specialists preferred this design, one of whom stated the reason that one can follow the flow.

In summary, the independent feedback indicated that the potential users considered these are minor variations and did not see any of the variations would be inhibitory. The preference of the four ML specialists are distributed among A, C, and D. While the preference for A is based on an aesthetic reason, the preference for C and D is functional. The independent feedback reinforced our assessment of the visual designs.

### C ADDITIONAL TESTING EXAMPLES

In addition to the four testing examples described in Section 6, we have conducted a number of other tests. In this appendix, we briefly report three additional testing examples to demonstrate that HypoML can support hypothesis-based testing by offering integrated visual analytics with automated statistical and logical analysis as well as rapid observation through visualization.

**Relative Scaling.** When working with the Fashion MNIST dataset [37], we noticed that the fashion objects in all images are maximized within the boundary of the image. We wondered if this would introduce some weaknesses to a trained model. As humans can usually perceive the size of an everyday object fairly quickly, we hypothesized that a model that can remap a maximized object to a more realistic size may help the classification of such an object. We thus adopt the experiment design in Fig. 8(b) by introducing some external data into the source data. We measured typical sizes of fashion objects in each category and defined a relative range for the category accordingly as:

- T-shirt/top (Class 0): 0.52 – 0.68,
- Trouser (Class 1): 0.78 – 1.00,
- Pullover (Class 2): 0.60 – 0.80,
- Dress (Class 3): 0.81 – 1.00,
- Coat (Class 4): 0.69 – 0.91,
- Sandal (Class 5): 0.17 – 0.23,
- Shirt (Class 6): 0.57 – 0.73,
- Sneaker (Class 7): 0.22 – 0.28,
- Bag (Class 8): 0.11 – 0.29,
- Ankle boot (Class 9): 0.26 – 0.34.

For the additional information, we randomly selected a scaling factor within the range defined for the corresponding category, and used the factor to scale the image. As shown on the left of Fig. 15, we used

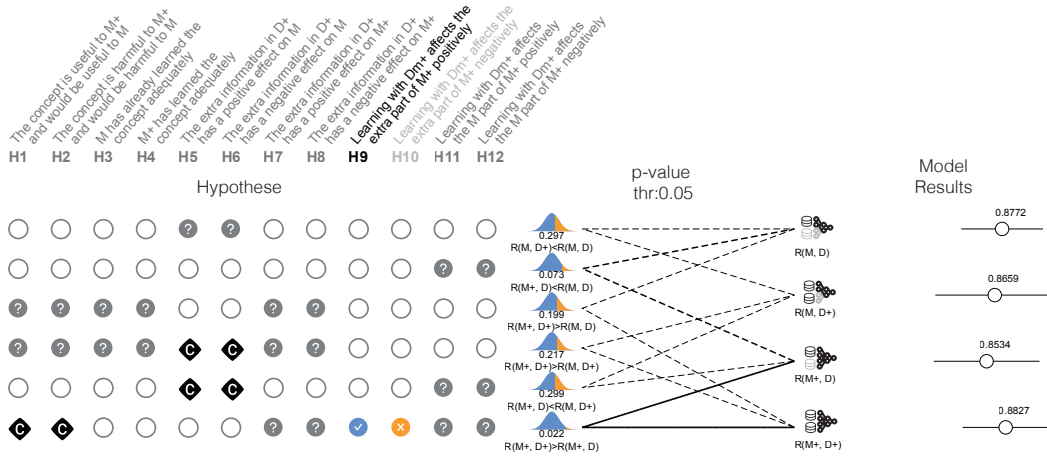


Fig. 13. Design C: An integrated design with a horizontal layout.

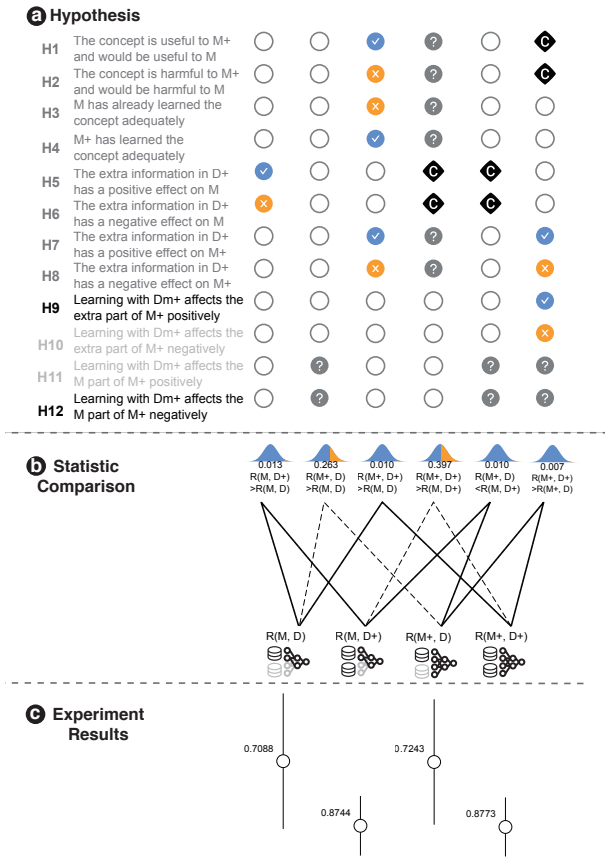


Fig. 14. Design D: An integrated design with a vertical layout.

the proportionally scaled image to reflect the size detection capability, and placed it on the top-right quadrant. As shown on the right of Fig. 15, the analytical results indicate that the concept of size detection and relative scaling is useful to M+, and would be useful to M. The conclusions about other hypotheses are more or less the same as the testing results for rotation regularization as shown in Fig. 1.

Readers may notice that a real-world object may appear in different sizes and with different rotation angles, and the source data (i.e., Fashion MNIST [37]) have regularized both the size and rotation angle of all objects. However, we used two different experiment designs for testing the two invariance hypotheses. This is because we observed that object cropping (hence size regularization) is a common pre-processing step deployed before object recognition, while rotation regularization

is less common. Meanwhile we hypothesized that the size information lost due to size regularization would be more useful than the angle information lost due to rotation regularization. Nevertheless, one can test an invariance property using different experiment designs as long as one is clear about the hypothesis concerned.

**Combined Rotation Regularization and Relative Scaling.** To demonstrate a slightly more complex design of a test, we combined the above test with the rotation regularization test (Section 6) to examine the combined effects of the two concepts. As shown in Fig. 16, we used the upper-left quadrant for the rotated object as the information present in all training and testing data. We placed rotation-regularization and relatively-scaled object at the lower-right quadrant. As perhaps expected, the test confirmed the same set of hypotheses as the two tests mentioned before.

**Lighting Variation.** Considering further about the intensity of the images, one common idealized requirement in computer vision is lighting invariance, i.e., a model can recognize the same object under different lighting conditions. We thus hypothesized that another model for normalizing the intensity of an image may help a classification model. Using the same experiment design as the rotation regularization (i.e., Fig. 8(c)), we randomly change the intensity of the original images to create a new dataset as the D dataset, featuring lighting variation. We used the source dataset as the + dataset, presupposing that the original images were the results of intensity normalization.

Fig. 17 shows that the extra information is useful to M+ (H<sub>1</sub>), and M+ has learned the concept adequately (H<sub>4</sub>). While the test confirms H<sub>7</sub> and H<sub>9</sub>, it is inconclusive about H<sub>11</sub> and H<sub>12</sub>. Interestingly, the test confirms H<sub>5</sub> unexpectedly, i.e., the extra information in D+ has a positive effect on M. This is in some way related to the failure to confirm H<sub>12</sub> as in some earlier tests. For each image in D+, the signals in the extra information (i.e., the upper-right quadrant), which in many ways is similar to those in D (i.e., upper-left quadrant). One possible explanation is the signals in the upper-right quadrant somehow strengthen the signals in the upper-left quadrant, even though M has not learned to use the extra information. As demonstrated by the last example in Section 6, the four hypotheses H<sub>9</sub>-H<sub>12</sub> are better studied using a few CNNs with different levels of network decomposition.

**Other Experiments.** We have also used HypoML to test a number of other hypotheses, including:

- What would happen if a + image is a grayscale image representing its class label?
- Using the same experiment setting, What may happen if  $x\%$  labels are incorrect. Cases of 10%, 25%, and 50% have been tested.
- Can a black image serve the same purpose as the noise image for filling an unused + placeholder?
- When there are two or more + placeholders, does the position of a + dataset matter?



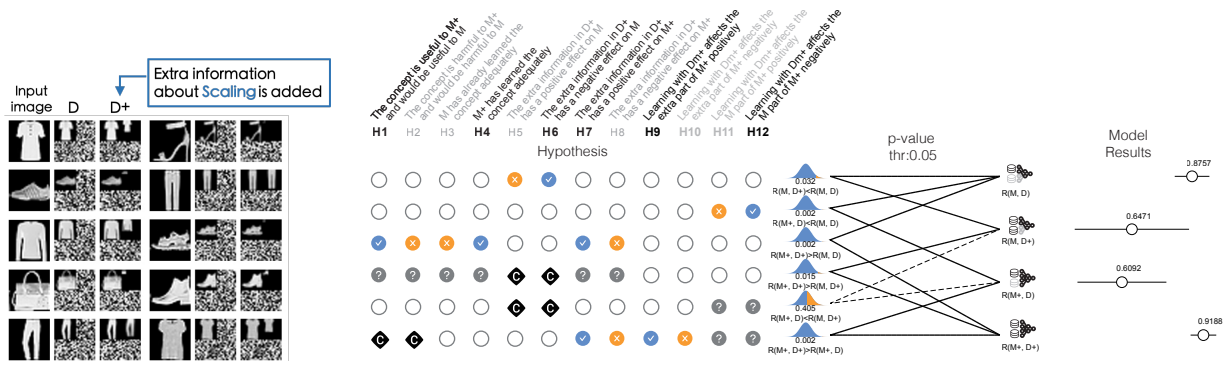


Fig. 15. Data samples and the visualization of the testing results for testing the concept of scaling correction. For each sample, the stimulus in D contains an object of a “maximized” size. The stimulus in D+ contains an extra object of a “relative” size.

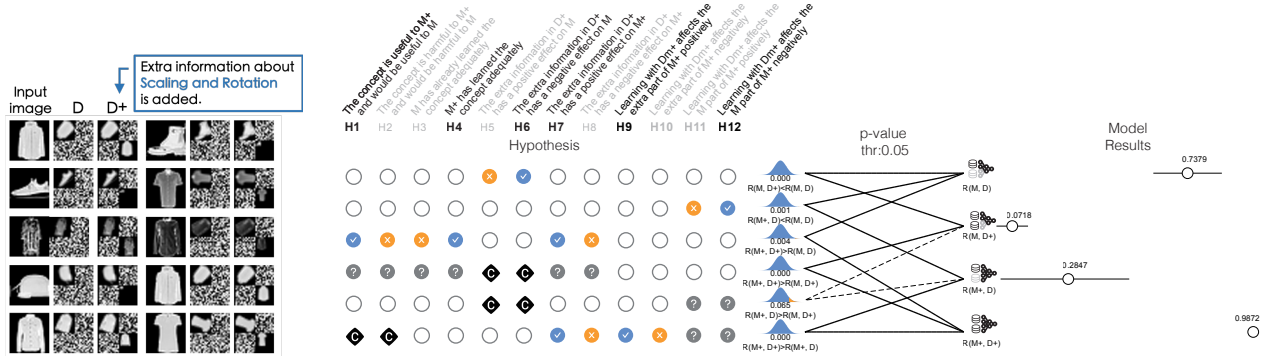


Fig. 16. Testing the combined concept of rotating and scaling correction. For each sample, the stimulus in D contains an object of a rotated and “maximized” size. The stimulus in D+ contains an extra object of a “relative” size in an up-right view.

- When a + dataset is of a uni-variable (e.g., average intensity), does the size of the + image matter? Cases of  $2 \times 2$ ,  $8 \times 8$ , and  $14 \times 14$  have been tested.
- What would happen if the images are randomly scaled in the relative scaling experiment? Cases of 10%, 25%, and 50% have been tested.
- Similar to the RGB vs RGB+HSV test, would LAB color space be different?

We are in the process of introducing HypoML into an ML education program and ask students to formulate and test their own hypotheses, such as:

- Would the concept of edge detection (blob detection, Fourier transform, Gabor filter, Hough transform, etc.) be useful?
- What would happen if a + image is a symbol representing its class label?
- Using the same experiment setting, what would happen if some of the labels are incorrect?
- Can a white image serve the same purpose as the noise image for filling an unused + placeholder?
- In the relative scaling experiment, what may happen if the scaled image is represented by a grey-scale image representing the relative size?
- What would happen if we apply experiment design (c), instead of (b), in the relative scaling experiment?
- Similar to the RGB vs RGB+HSV test, would any of other color spaces be useful?

## D INTERVIEW PROCEDURE AND QUESTIONS

**Interview Procedure.** We conducted semi-structured interviews with four ML experts individually through face-to-face or video-conference meetings. Each interview consists of two parts, and took about 30 to 45 minutes. The first author conducted two interviews at HKUST, while the last author conducted other two interviews at Oxford. No co-author of the paper took part in the interview as an interviewee.

In the first part, we first collected background information from the interviewee, including the interviewee’s experience of ML in academic and industrial settings. We then introduced the motivation and the system design of HypoML using a few figures in the paper, and the visual design of HypoML using the online demo at <http://hypoml.bitbucket.io>. We also described the use case of testing rotation regularization to help the interviewee better understand the workflow of HypoML and a method of experiment design.

In the second part of each interview, we discussed a series of semi-structured questions with the interviewee. During the process, the interviewee could freely explore the online tool, ask questions, offer an opinion, and make suggestions.

**Interview Questions.** The interview questions were prepared beforehand, and they are listed below:

### • About the general motivation:

- i. Apart from statistic summaries (e.g., accuracy), what are the common method you use to evaluate an ML model? What aspects are usually evaluated?
- ii. In your working with ML, have you ever encountered a situation where you need to evaluate whether an ML model has learned a feature?  
If yes, what features have you usually evaluated?  
Please describe these situations in detail.

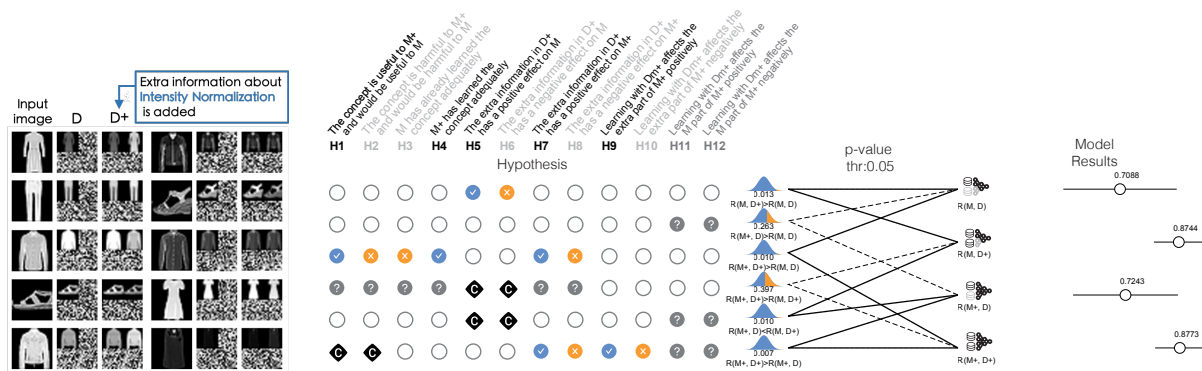


Fig. 17. Testing the concept of intensity normalization. For each sample, the stimulus in D contains an original object whose intensity has been arbitrarily re-scaled. The stimulus in D+ contains an extra object featuring the original intensity as a form of normalization.

- Please describe the method you used for the evaluation, including a) a step-by-step workflow of using this method; b) how much time and efforts the method usually cost.
  - In your working with ML, have you ever encountered a situation where you guess or hypothesize if a feature (or an extra piece of metadata) may be useful or harmful? If yes, can you give one or a few examples of such features or extra information? Please describe these situations in detail. How do you normally evaluate your guess (i.e., hypotheses), including a) a step-by-step workflow of using this method; b) how much time and efforts the method usually cost. If no, would a feature such as the rotation angle of an object or the time when a photo was taken might be useful?
  - When developing an ML model, have you ever integrated extra information (e.g., human knowledge, photo metadata) to improve the model performance? How did the extra information influence the model performance?
- 
- About the idea of Hypothesis testing
    - Are you familiar with hypothesis testing?
    - We often use statistic summary (e.g., mean accuracy) to compare and evaluate ML models. Is the conclusion always reliable? Is it possible that the difference is caused by random factors? Based on your experience, how much difference (e.g., accuracy 84.9% > 84.8%) can be considered as a solid evidence for superior performance?
    - Do you think hypothesis testing is required in the evaluation of ML models?
    - Have you ever used hypothesis testing to evaluate ML models? If yes, please describe the hypothesis and how you test it in detail.
- 
- About the experiment design
    - [We introduce the features we tested in the paper, then ask the interviewee's opinion.] Would you list some other features that you would like to test with your model?
    - [We introduce the 12 hypotheses, then ask the interviewee's opinion.]
    - [We introduce the 3-step workflow (model results, statistic comparison, hypotheses), then ask the interviewee's opinion.]
- 
- About the visualization tool