Manifold:
A Model-Agnostic Framework for Interpretation and Diagnosis of Machine Learning Models

Jiawei Zhang, Yang Wang, Piero Molino, Lezhi Li and David S. Ebert

Presented by: Atefeh Mahdavi
What is the problem?
Recent technical breakthroughs in the machine learning field have led to highly improved accuracies and utilization in many scenarios, including sophisticated pattern recognition tasks.
Two major challenges:
First, the complexity of the models being designed and adopted has significantly increased to the point that it is difficult for model developers to explain why and how the model works.

Second, model developers often lack solid reasoning or evidence to guide their development and debugging due to the hidden mechanisms of the models, making this iterative process more time-consuming and error-prone.
What has been done?
Visual and interactive interfaces have proved to be effective in terms of enabling users to integrate domain knowledge in the process of interpreting and diagnosing these complex models.

Typical solutions include visualizing the internal structure or intermediate states of the model to enhance the understanding and interpretation, evaluating and analyzing the performance of models or algorithms, and interactively improving the models at different development stages.
But,
The focus of these approaches has been mostly restricted to a specific model type or task type (i.e., classification tasks), lacking the ability to extend to more complex industry-level use scenarios where the size and the complexity of both the model and the task increase.
So?
An interactive framework called Manifold to address these problems of integrating, evaluating and debugging multiple machine learning models.

The design process of the framework has been guided by three major phases that are typically involved in diagnosing and comparing machine learning models: inspection (hypothesis), explanation (reasoning), and refinement (verification).
What is the motivation?
1. Debugging coding errors in the model
2. Understanding strengths and weaknesses of one model both in isolation and in comparison with other models
3. Model comparison and ensembling
4. Incorporating insights gathered through inspection and performance analysis into model iterations
Equip users with tools to make models more transparent and easy-to-understand; ML visualization, an emerging domain, solves this problem.

Instead of inspecting models, users inspect individual data points, by:

- Identifying the data segments that make a model perform well or poorly, and how this data affects performance between models.
- Assessing the aggregate feature characteristics of these data segments to identify the reasons for certain model behaviors. This approach facilitates model-agnosticism, a particularly useful feature when it comes to identifying opportunities for model ensembling.
How?

Through two main visual components.
First, a novel scatterplot-based visual technique that provides a comparative visual summary of the diversity and complementarity of the model pairs, and allows the users to effectively inspect symptom data instances and make hypotheses accordingly. The technique consists of multiple encoding schemes that are flexible and adaptable to various task types such as classification or regression.
Second, a tabular view for the users to visually discriminate features extracted from symptom instances and identify which features are more influential in the models’ outcome, thus providing explanations for the hypotheses generated earlier on. These explanations can then be incorporated into a new iteration of the model development in order to validate and refine the model.
Manifold is model-agnostic, in the sense that it does not need access to the internal logic of the model and only relies on the input instances and the output results, allowing the framework to support a broad range of model types, as long as they target the same machine learning task and have a consistent format of input and output.
What are the analysis phases that are commonly involved in the model diagnosis?
Inspection (Hypothesis)
Inspection (Hypothesis) is the entry of the analysis process when the user designs a model and attempts to investigate and compare the model outcome with other existing ones.

Accuracy, precision/recall, and receiver operating characteristic curve (ROC)

1. The user has reasonable knowledge of the instances (i.e., feature distribution, ground truth) prior to the analysis process.
2. The user filters a subset of instances that have some features in common,
3. The user identifies a subset where the results generated by the model are erroneous or suspicious, for example, the instances where the new model has low accuracy while others have high accuracy. We define this type of subset as a symptom set since it is representative of a potential fault within the model. The symptom set is of particular interest to the user during the diagnosis process.
Explanation (Reasoning)
1. After selecting a symptom set, in this phase, the user attempts to explain her hypotheses.
2. The user may access the detailed information at the instance level or the feature level that can potentially explain the symptom. Comparative analysis is intensively involved in this phase.
3. An explanation may not necessarily have a causal relationship with the symptom. The user can generate multiple explanations relevant to a symptom. A verification phase is required to validate the explanation.
Refinement (Verification)
In this phase, the user attempts to verify the explanations generated from the previous phase through encoding the knowledge extracted from the explanation into the model and testing the performance. This verification process may be both expensive and challenging.
How does this framework work?
For visually inspecting potential issues within the model:

**Model Comparison View**

For comparing feature distributions and generating explanations for the issue

**Feature Interpretation View**
Case studies
Multi-Class Classification
• Spooky author identification dataset from Kaggle
• The dataset contains a set of excerpts from horror stories written by three authors (Edgar Allan Poe (EAP), Mary Shelley (MWS), and HP Lovecraft (HPL)).
• The task is to predict the author given a specific excerpt.
• Developed 12 models.
• Picked 2 of them based on the overall good performance.
Feature Attribution
Contributing features that lead to the discrepancy between data segments.
Regression
- Bike Sharing Demand dataset from Kaggle
- Build models to predict the demand, i.e., the total number of bikes rented with the Root Mean Squared Logarithmic Error (RMSLE) as the evaluation metric
- Early exploratory data analysis revealed a strong correlation between features casual, registered and the prediction target count (corr = 0.67, 0.98, respectively).
- Five commonly used regression models with the default hyperparameters

<table>
<thead>
<tr>
<th>Feature</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>datetime</td>
<td>datetime</td>
<td>hourly date + time stamp</td>
</tr>
<tr>
<td>season</td>
<td>categorical</td>
<td>spring, summer, fall, winter</td>
</tr>
<tr>
<td>weather</td>
<td>categorical</td>
<td>clear, mist + cloudy, light snow, heavy rain</td>
</tr>
<tr>
<td>holiday</td>
<td>boolean</td>
<td>whether the day is considered a holiday</td>
</tr>
<tr>
<td>workingday</td>
<td>boolean</td>
<td>whether the day is non weekend nor holiday</td>
</tr>
<tr>
<td>temp</td>
<td>numerical</td>
<td>temperature in Celsius</td>
</tr>
<tr>
<td>atemp</td>
<td>numerical</td>
<td>&quot;feels like&quot; temperature in Celsius</td>
</tr>
<tr>
<td>humidity</td>
<td>numerical</td>
<td>relative humidity</td>
</tr>
<tr>
<td>windspeed</td>
<td>numerical</td>
<td>wind speed</td>
</tr>
<tr>
<td>casual</td>
<td>numerical</td>
<td>number of on-registered user rentals initiated</td>
</tr>
<tr>
<td>registered</td>
<td>numerical</td>
<td>number of registered user rentals initiated</td>
</tr>
<tr>
<td>count</td>
<td>numerical</td>
<td>number of total rentals</td>
</tr>
</tbody>
</table>

Table 3. Feature table of the bike sharing demand prediction dataset.
Between Model Comparison
An overview of model performance over classes.
Feature Attribution
Contributing features that lead to the discrepancy between data segments.

* A larger coordinate indicates a larger residual value
Domain Expert Feedback
Commonly adopted performance measures such as F-measure, ROC (or AUC), and confusion matrices are usually oriented towards a single model at a coarse-grained level.

When multiple models were involved in the analysis, they often had to investigate different models independently and then combine or compare the results to form a comprehensive understanding.

Visual confusion when initially presented to them without training.
Discussion
• Many existing visual analytics solutions focus on interpreting the internal working mechanism of a specific model type.
• While common approaches enable a comprehensive understanding of one model, it is usually not straightforward to transfer these insights to other models due to their varied working mechanisms.
• A higher-level scope, in the sense that we do not attempt to investigate or improve a specific model to a great extent.
• Allowing the end users to easily load a model into Manifold to have an initial understanding of its performance without needing prior knowledge about the model.
Questions?