Notes: Interpretable Machine Learning

Pablo Carrera Flórez de Quiñones

June 2019

Contents

1 Interpretability

Definitions:

- Interpretability is the degree to which a human can understand the cause of a decision.
- Interpretability is the degree to which a human can consistently predict the model's result.

1.1 Importance of Interpretability

If a machine learning model performs well, why do not we just trust the model and ignore why it made a certain decision?

- The problem is that a single metric, such as classification accuracy, is an incomplete description of most real-world tasks.
- The need for interpretability arises from an incompleteness in problem formalization, which means that for certain problems or tasks it is not enough to get the prediction (the what). The model must also explain how it came to the prediction (the why), because a correct prediction only partially solves your original problem.

If you can ensure that the machine learning model can explain decisions, you can also check the following traits more easily:

- Fairness: Ensuring that predictions are unbiased and do not implicitly or explicitly discriminate against protected groups.
- Privacy: Ensuring that sensitive information in the data is protected.
- Reliability or Robustness: Ensuring that small changes in the input do not lead to large changes in the prediction.
- Causality: Check that only causal relationships are picked up.
- Trust: It is easier for humans to trust a system that explains its decisions.

1.2 Taxonomy of Interpretability Methods

- Result of the interpretation method
  - Feature summary statistic
  - Feature summary visualization
  - Model internals
  - Data point
  - Intrinsically interpretable model
- Intrinsic or post hoc?
  - Intrinsic: by restricting the complexity of the machine learning model.
  - Post hoc: by applying methods that analyze the model after training.
- Model-specific or model-agnostic?
  - Model-specific interpretation tools are limited to specific model classes.

  - Model-agnostic tools can be used on any machine learning model and are applied after the model has been trained (post hoc). By definition, these methods cannot have access to model internals such as weights or structural information.

- Local or global?
  - Local: the interpretation method explain an individual prediction.
  - Global: the interpretation method explain the entire model behavior. This level of interpretability is about understanding how the model makes decisions, based on a holistic view of its features and each of the learned components such as weights, other parameters, and structures.

1.3 Evaluation of Interpretability

Three main levels for the evaluation of interpretability:

- Application level evaluation (real task): Put the explanation into the product and have it tested by the end user. A good baseline for this is always how good a human would be at explaining the same decision.
- Human level evaluation (simple task): is a simplified application level evaluation. The difference is that these experiments are not carried out with the domain experts, but with laypersons.
- Function level evaluation (proxy task): does not require humans. This works best when the class of model used has already been evaluated by someone else in a human level evaluation.

We want to explain the predictions of a machine learning model. To achieve this, we rely on some explanation method, which is an algorithm that generates explanations. Properties of Explanation Methods:

- Expressive Power: is the "language" or structure of the explanations the method is able to generate.
- Translucency: describes how much the explanation method relies on looking into the machine learning model, like its parameters. The advantage of high translucency is that the method can rely on more information to generate explanations. The advantage of low translucency is that the explanation method is more portable.
- Portability: describes the range of machine learning models with which the explanation method can be used.
- Algorithmic Complexity: describes the computational complexity of the method that generates the explanation.

An explanation usually relates the feature values of an instance to its model prediction in a humanly understandable way. Properties of Individual Explanations:

- Accuracy: How well does an explanation predict unseen data?
- Fidelity: How well does the explanation approximate the prediction of the black box model?
- Consistency: How much does an explanation differ between models that have been trained on the same task and that produce similar predictions? ("Rashomon Effect")
- Stability: How similar are the explanations for similar instances?

- Comprehensibility: How well do humans understand the explanations?
- Certainty: Does the explanation reflect the certainty of the machine learning model?
- Degree of Importance: How well does the explanation reflect the importance of features or parts of the explanation?
- Novelty: Does the explanation reflect whether a data instance to be explained comes from a "new" region far removed from the distribution of training data?
- Representativeness: How many instances does an explanation cover?

What makes a good explanation?:

- Explanations are contrastive: humans usually do not ask why a certain prediction was made, but why this prediction was made instead of another prediction (counterfactual explanation).
- Explanations are selective: people do not expect explanations that cover the actual and complete list of causes of an event. We are used to selecting one or two causes from a variety of possible causes as the explanation.
- Explanations are social: they are part of a conversation or interaction between the explainer and the receiver of the explanation. The social context determines the content and nature of the explanations.
- Explanations focus on the abnormal: people focus more on abnormal causes to explain events. These are causes that had a small probability but nevertheless happened. The elimination of these abnormal causes would have greatly changed the outcome (counterfactual explanation).
- Explanations are truthful: good explanations prove to be true in reality.
- Good explanations are consistent with prior beliefs of the explainee: humans tend to ignore information that is inconsistent with their prior beliefs. This effect is called confirmation bias. Explanations are not spared by this kind of bias.
- Good explanations are general and probable: a cause that can explain many events is very general and could be considered a good explanation. References

https://arxiv.org/abs/1606.03490 https://arxiv.org/abs/1702.08608 https://arxiv.org/abs/1706.07269

2 Model-agnostic Methods

Separating the explanations from the machine learning model has some advantages [1]:

- **Model flexibility**: for most real-world applications, it is necessary to train models that are accurate for the task, irrespective of how complex or uninterpretable the underlying mechanism may be. In model-agnostic interpretability, the model is treated as a black box, the separation of explanations from the model free us it to be as flexible as necessary for the task, enabling the use of any machine learning approach.
- **Explanation flexibility**: different kinds of explanations meet different information needs. By keeping the model separate from explanations, one is able to tailor the model explanation for the information need, while keeping the model fixed, that is, the same model can be explained with different types of explanations, and different degrees of interpretability for each type of explanation.
- **Representation flexibility**: in domains such as images, audio and text, many of the features used to represent instances in state-of-the-art solutions are themselves not interpretable. While an interpretable model trained on such features is still uninterpretable, model-agnostic approaches can generate explanations using different features than the one used by the underlying model.
- **Lower cost to switch**: switching models is not an uncommon operation in machine learning pipelines. If one uses model-agnostic explanations, switching the underlying model for a new one is trivial, while the way in which the explanations are presented is maintained.
- **Comparing two models**: when deploying machine learning in the real world, a system designed often has to decide between one or more contenders. With model-agnostic explanations, the models being compared can be explained using the same techniques and representations.

2.1 Partial Dependence Plot (PDP)

Consider the subvector X_S of $l < p$ of the input variables $X = (X_1, ..., X_p)$, indexed by $S \in 1, ..., p$. Let C be the complement set, with $S \cup C = 1, ..., p$. A general function $f(X)$ will in principle depend on all the input variables, so one way to define the average, or partial, dependence of $f(X)$ on X_S is

$$f_S(x_S) = \mathbb{E}_{X_C}[f(x_S, x_C)] = \int_{X_C} f(x_S, x_C)\mathbb{P}(x_C)dx_C$$

in such a way that each subset of predictors S has its own partial dependence f_S, which gives the average value of f when x_S is fixed and x_C varies over its marginal distribution $\mathbb{P}(x_C)$. These partial dependence functions can be estimated by

$$\hat{f}_S(x_S) = \frac{1}{N}\sum_{i=1}^{N} f(x_S, x_{C,i})$$

where $x_{C,i}, ..., x_{C,N}$ are the values of X_C occurring in the training data. Viewing plots (PDP) of the partial dependence approximations on selected variables can help to provide a qualitative description of its properties.

This is a visualization tool in the sense that, if $\hat{f}_S$ is evaluated at the x_S observed in the data, we will end with a set of N ordered pairs $\{x_{S,l}, \hat{f}_{S,l}\}_{l=1}^{N}$, where $\hat{f}_{S,l}$ refers to the estimated partial dependence function evaluated at the l-th coordinate of x_S, denoted as $x_{S,l}$. Then for a one or two dimensional x_S we can plot these N $x_{S,l}$ versus their associated $\hat{f}_{S,l}$, usually with a previous

discretization of x_s, conventionally joined by lines, as can be seen in Figure 1. The resulting graphi, which is called a Partial Dependence Plot (PDP), displays the average value of $\hat{f}$ as a function of x_S

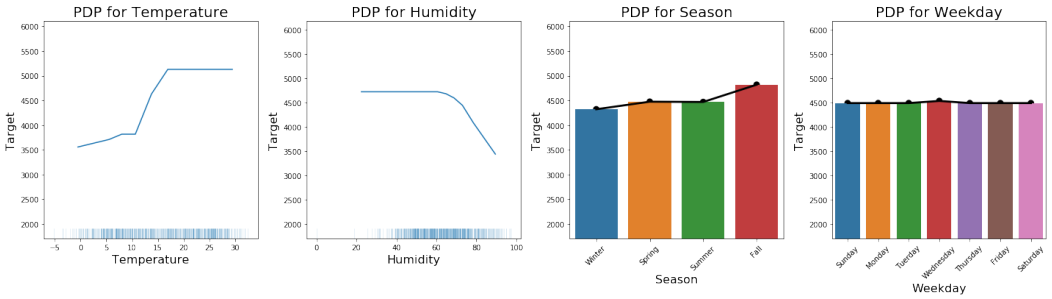

Figure 1: *Example of PDPs for the Bike Sharing dataset. We show both, numerical and categorical variables. It can be seen that the temperature and the humidity have intuitive effects over the number of bikes shared but the season and the weekday effect doesn't have an intuitive interpretation.*

This approach was developed in [2] and in [3], with the following observations:

- The computation of partial dependence plots is intuitive and easy to implement. Also the interpretation is clear, the partial dependence plot shows how the average prediction in your dataset changes when the j-th feature is changed.
- Partial dependence functions represent the effect of X_S on $f(X)$ after accounting for the effects of the other variables X_C, they are not the effect of X_S on $f(X)$ ignoring the effects of X_C.
- The realistic maximum number of features in a partial dependence function is two.
- Some PDP do not show the feature distribution, omitting the distribution can be misleading, because you might overinterpret regions with almost no data. However this can be easily solved by adding a rugosity plot on the axes.
- It is assumed that the feature(s) for which the partial dependence is computed are not correlated with other features. If this assumption is violated, the averages calculated for the partial dependence plot will include data points that are very unlikely or even impossible.
- Heterogeneous effects might be hidden because PDPs only show the average marginal effects.

2.2 Individual Conditional Expectation (ICE)

Consider the observations $(x_{S,i}, x_{C,i})_{i=1}^{N}$, and the estimated response function f. For each $x_{C,i}$ of the N observed and fixed values of x_C, a curve $f_{S,i}$ is plotted against observed values $x_{S,i}$ of x_S. Therefore, at each coordinate x_S is fixed and the x_C varies across N observations. Each curve defines the conditional relationship between x_S and f at fixed values of x_C. Thus, the ICE algorithm gives the user insight into the several variants of conditional relationships estimated by the black box model. Note that the PDP curve is the average of the N ICE curves.

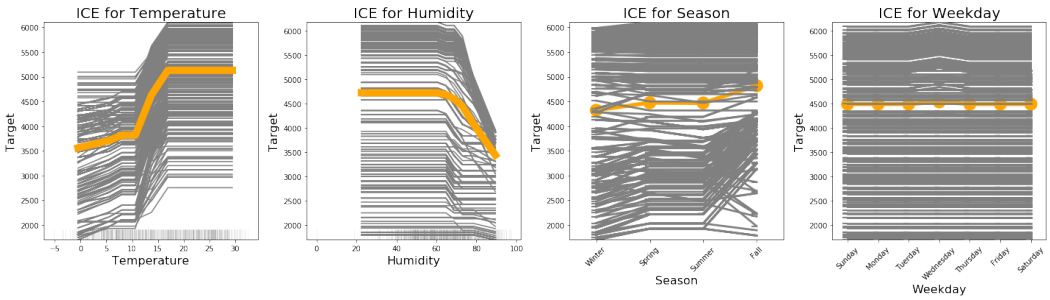

Figure 2: *Example of ICEs for the Bike Sharing dataset. We show both, numerical and categorical variables. We can see that, despite the trend in predictions is practically the same in all samples, some of the seem to bee more affected by a change of the dependent variable than anothers.*

When the curves have a wide range of intercepts and are consequently stacked on each other, heterogeneity in the model can be difficult to discern. In such cases the centered ICE plot (c-ICE), which removes level effects, is useful. For constructing a c-ICE choose a location $x*$ in the range of x_S and join all prediction lines at that point. Choosing that $x*$ as the minimum or the maximum observed value results in the most interpretable plots. For each curve $f_{S,i}$ in the ICE plot, the corresponding c-ICE curve is given by

$$f_{S,i}^c = f_{S,i} - \mathbb{I}f(x^*, x_{C,i})$$

where the point $(x*, f(x*, x_{C,i}))$ acts as a base case for each curve. If x^* is the minimum value of x_S, for example, this ensure that all curves originate at 0, thus removing the differences in level due to different $x_{C,i}$. At maximum x_S value, each centered curve's level reflects the cumulative effect of x_S on f relative to the base case.

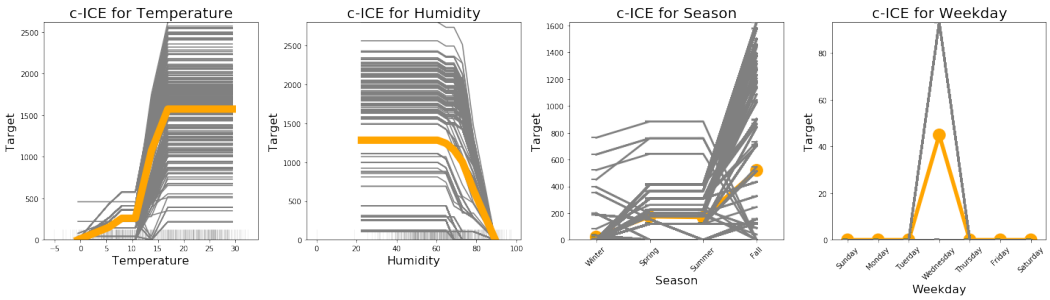

Figure 3: *Example of c-ICEs for the Bike Sharing dataset. We show both, numerical and categorical variables. We can see that, with this setup, the cumulative effects of the change in predictions due to changes in a variable can be interpreted more intuitively.*

This approach was were developed in [4] with the following observations:

- ICE curves are more intuitive to understand than PDP curves, one line represents the predictions for one instance if we vary the feature of interest.
- ICE curves can uncover heterogeneous relationships.
- ICE curves can only display one feature meaningfully and if many ICE curves are drawn, the plot can become overcrowded. Also, it might not be easy to see the average.
- If the feature of interest is correlated with the other features, then some points in the lines might be invalid data points according to the joint feature distribution.

2.3 Accumulated Local Effects (ALE)

Accumulated local effects describe how features influence the prediction of a machine learning model on average. ALE plots are a faster and unbiased alternative to partial dependence plots (PDPs).

$$f_S(X_S) = \mathbb{E}_{X_C}[f(X_S, X_C)] = \int_{X_C} f(X_S, X_C)\mathbb{P}(X_C)dX_C$$

This approach was were developed in [5] with the following observations:

2.4 Feature Interaction

.

2.5 Permutation Feature Importance

.

2.6 Global Surrogate

.

2.7 Local Surrogate (LIME)

. [6]

2.8 Scoped Rules (Anchors)

.

2.9 Shapley Values

.

2.10 Shapley Additive Explanations (SHAP)

3 Example-Based Explanations

References

[1] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. *Model-Agnostic Interpretability of Machine Learning*. 2016. arXiv: 1606.05386 [stat.ML].

[2] Jerome H. Friedman. "Greedy function approximation: A gradient boosting machine". In: *Ann. Statist.* 29 (2001), pp. 1189–1232. DOI: 10.1214/aos/1013203451.

[3] Trevor Hastie, Robert Tibshirani, and Jerome H. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer series in statistics. Springer, 2009. ISBN: 9780387848846.

[4] Alex Goldstein et al. *Peeking Inside the Black Box: Visualizing Statistical Learning with Plots of Individual Conditional Expectation*. 2013. arXiv: 1309.6392 [stat.AP].

[5] Daniel W. Apley and Jingyu Zhu. *Visualizing the Effects of Predictor Variables in Black Box Supervised Learning Models*. 2016. arXiv: 1612.08468 [stat.ME].

[6] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. *"Why Should I Trust You?": Explaining the Predictions of Any Classifier*. 2016. arXiv: 1602.04938 [cs.LG].

[7] Christoph Molnar. *Interpretable Machine Learning. A Guide for Making Black Box Models Explainable*. https://christophm.github.io/interpretable-ml-book/. 2019.