“Why Should I Trust You?” Explaining the Predictions of Any Classifier

Marco Tulio Ribeiro
Computer Science & Engineering
University of Washington
Seattle, WA 98105, USA
marcotcr@cs.washington.edu

Sameer Singh
Computer Science & Engineering
University of Washington
Seattle, WA 98105, USA
sameer@cs.washington.edu

Carlos Guestrin
Computer Science & Engineering
University of Washington
Seattle, WA 98105, USA
guestrin@cs.washington.edu

Abstract

Despite widespread adoption in numerous applications, machine learning models remain mostly black boxes; it is completely opaque to the user why predictions are made. Understanding the reasons behind predictions is, however, quite important as it can lead to effective hyper-parameter tuning, feature engineering, and model selection. More importantly, such an understanding is crucial in developing *trust* between the user and the system. In particular, there are two separate (but related) types of trust: whether the end-user trusts a specific prediction (e.g. a doctor deciding if a model prediction should be acted upon), or whether a practitioner trusts a model enough to deploy it ’in the wild’.

In this work, we formalize the characteristics of good *explainers* that can explain why individual machine learning predictions are made. Using these characteristics as a guiding principle, we propose a novel explanation technique that is able to explain *any* machine learning models in an interpretable and faithful manner. We demonstrate that accurate explanations are good indicators of trust, both at an individual prediction level, as well as to compare different models. We further present example explanations for different domains (text and images) and classifiers that illustrate the usefulness and flexibility of the explanation method.

Desired Characteristics for Explanation Systems

(1) **Model Agnostic**: The system should explain predictions of *any* classifier, i.e. treat them as black boxes. Apart from explaining accurate, state-of-art models, this also provides flexibility to explain future classifiers.
(2) **Local fidelity**: In order to provide trust, explanations must be faithful, i.e. they must correspond to how the model actually behaves in the vicinity of the prediction.
(3) **Interpretability**: The explanations must be interpretable - that is, they must take human cognitive limitations into account [14].
(4) **Explanation selection**: In order to evaluate a model, examining a lot of explanations is often not possible, and thus the system should present explanations only for a representative subset.
(5) **Anytime**: The system should allow a trade-off between fidelity and computation time, i.e. give immediate explanation (at the cost of fidelity), or give a faithful explanation if the user can wait.

Introduction

Machine learning is at the core of many recent advances. Unfortunately, the important role of humans is an oft-overlooked aspect in the field. Whether humans are directly using the machine learning classifiers as tools, or are deploying models into products that need to be shipped, a vital concern remains: *if the users do not trust the model or their predictions, they will not use it*. It is important to differentiate between two different (but related) definitions of trust: (1) *trusting a prediction*, i.e. whether a user trusts an individual prediction sufficiently to take some action based on it, and (2) *trusting a model*, i.e. whether the user trusts a model to behave in reasonable ways if deployed.

Trust in Individual Predictions

Determining trust in individual predictions is an important problem when the model is being used for real-world actions. When using machine learning for medical diagnosis [5] or to detect terrorism activity, for example, the potential effects of mistakes are catastrophic - and thus the predictions cannot be acted upon on blind faith. Even when the stakes are lower, as in product or movie recommendations, the user needs to trust the prediction enough to spend money or time on it. One popular way to address the opacity of the models is to use alternate models that are more interpretable [5, 10, 11, 20] and provide insight into why predictions were made; unfortunately they do so at the cost of accuracy and flexibility. It is thus crucial to be able to explain predictions of state-of-the-art machine learning models (such as random forests, neural networks, and SVMs) that are functionally black boxes, in order to aid users in ascertaining their trust in them.

Trust in the Model

Apart from trusting individual predictions, there is also a need to evaluate the model as a whole before deploying it "in the wild". To make this decision, the users need to be confident that the model will perform well on the real-world data, on the metrics of interest. Currently, models are evaluated using metrics such as accuracy on an available validation dataset. However, real-world data is often significantly different, and further, the evaluation metric may not correlate well with the overall goal of the product. Instead, we are interested in approaches that incorporate humans in the loop to utilize prior knowledge. For example, some tools [1, 15] allow practitioners to look at measures of performance such as confusion tables, dataset visualizations, or explore specific predictions such as very confident mistakes. While these are definitely helpful in building trust in a model, we argue that they are not sufficient. Problems such as overestimating models' accuracy [15], feedback loops [18], leakage [8], dataset shift [4], or mismatch between metrics such as accuracy and business metrics (e.g. "user happiness") can be easily overlooked if one does not have an understanding of what the model is doing for particular instances, but become obvious once explanations are provided. Noticing such problems by just looking at raw data is often not possible. Furthermore, as datasets grow in size, it becomes important to guide users by suggesting the instances they should be inspecting.

An Explanation System for Classifier Predictions

Trust (both in predictions and the model as a whole) is thus a fundamental issue for human-centered machine learning, and explaining individual predictions is a significant component for providing trust. In the sidebar, we outline a number of desired characteristics from an explainer, and in the remainder of this paper we describe the local, interpretable model-agnostic explanation (LIME) system designed to meet these criteria. For individual predictions, the system is faithful and model-agnostic, while being interpretable and anytime in nature. We further propose an approach

Intelligibility

Even *interpretable* models may not be intelligible to humans. For example, an additive model or feature gradients [2] for hundreds of features is too complex for humans to comprehend. Furthermore, an assumption often made when using interpretable models is that features themselves are easily interpretable, which is often not the case. In text, for example, the features may contain linguistic structures or word embeddings that are not intuitive for most users. Similarly, for images, the features may include spectral transformations or raw pixels, both which are not directly amenable to human understanding. Instead, here we allow each instance to be accompanied by an **intelligible representation** that consists of natural components for the task, e.g. the list of words for text documents, and a collection of *super-pixels* for images. Interpretable models that use this representation as features and are restricted in their complexity will thus provide intelligible explanations.

to select representative explanations in order for a user to evaluate the whole model from a subset of instances. We evaluate trust on simulated tasks, and demonstrate that our explanations are able to help users ascertain trust in individual predictions and select between competing classifiers significantly better than existing approaches.

Explaining predictions

Current work for explaining relies either on using *interpretable* models (such as decision trees [7] or additive models [5]) or on approximating black-box model predictions with *interpretable* models globally [2, 17]. Since the family of interpretable models is quite restrictive, they are unable to provide a faithful reproduction of the classifier, and thus the utility of such explanations is not clear. On the other hand, we aim to explain individual predictions, i.e. the much more feasible task of approximating the classifier locally in the neighborhood of the prediction, instead of globally on all predictions. We call our method **Local Interpretable Model-agnostic Explanations (LIME)**.

The overall goal of LIME is to identify an interpretable model over the intelligible representation (see sidebar) that is locally faithful to the classifier. Formally, the explanation model needs to explain why the prediction $f(x)$ is made, where $x \in \mathbb{R}^d$ is the instance, $f : \mathbb{R}^d \rightarrow [0, 1]$ denotes the black-box classifier, and the prediction $f(x)$ is the probability that x belongs to a certain class. Further, let $\Pi_x(z)$ be the proximity of an instance z to x (used to define locality around x). We use x' to denote the intelligible representation of x, and let G be a family of explanations (simple, interpretable models that use x' as features). Let $\Omega_x(g)$ be a measure of *complexity* (as opposed to *interpretability*) of the explanation $g \in G$ in explaining instance x. Finally, let $\mathcal{L}(f, g, \Pi_x)$ be a measure of how unfaithful g is in approximating f in the locality defined by Π_x. We define the explanation model $\xi(x)$ as:

$$\xi(x) = \underset{g \in G}{\arg\min} \quad \mathcal{L}(f, g, \Pi_x) + \Omega_x(g)$$

Concretely, we use linear classifiers as G, and set $\Omega_x(g) = \infty$ if the number of non-zero weights in g is greater than some budget K (set to 10 unless specified otherwise), i.e. each explanation will be a set of at most K elements of x' with associated weights. We use the locally weighted square loss as $\mathcal{L}$:

$$\mathcal{L}(f, g, \Pi_x) = \sum_{z \in \mathbb{R}^d} \Pi_x(z) \left(f(z) - g(z')\right)^2$$

This summation is approximated by sampling randomly in the vicinity of x. Finally, we let $\Pi_x(z)$ be the exponential kernel applied on the cosine distance between x' and z'. This optimization thus reduces to identifying a linear model that uses K features over the samples weighted by $\Pi_x(z)$, which we approximate by first selecting K features with Lasso (using the regularization path [6]) and then learning with weighted least squares estimation.

Illustrative examples

Here we present two example explanations to show the flexibility and utility of LIME. First we study text classification - specifically, explaining predictions of a logistic regression classifier trained on unigrams to differentiate “Christianity” from “Atheism” (on a subset of the 20 newsgroup dataset). Although this classifier achieves 93% held-out accuracy, and one would be tempted to trust it based on this, the explanation for an instance (Figure 1) shows that predictions are made for quite arbitrary reasons (words “1993”, “rutgers” and “athos” have no connection to either Christianity or Atheism). After examining a few more explanations, it is clear that this dataset has serious issues (which would not be evident by studying the raw data or predictions), and that this classifier, or held-out evaluation, cannot be trusted.

From: gt7122b@prism.gatech.edu (Randal Lee Nicholas Mandock)
Subject: Re: Why do people become atheists?

In article <May.11.02.36.27.1993.28065@athos.rutgers.edu> biz@soil.princeton.edu writes:
>Who is the "atheist's prayer" being said to?
My roommate, the atheist, says "to anyone out there who might be listening."

Figure 1: Explaining a document classification prediction (logistic regression on 20 Newsgroups, $K = 5$). The classifier *correctly* predicts the class 'Christianity' with $p = 0.59$. In the explanation, words indicative (according to the model) of 'Christianity' are green, while 'Atheism' words are red.

(a) Original Image **(b)** Explaining "Electric guitar" **(c)** Explaining "Acoustic guitar" **(d)** Explaining "Labrador retriever"

Figure 2: Explaining an image classification prediction made by Google's Inception network. The top 3 classes predicted are "Electric Guitar" ($p = 0.32$), "Acoustic guitar" ($p = 0.24$) and "Labrador Retriever" ($p = 0.21$)

For the task of image classification, we explain the prediction of Google's pre-trained Inception neural network [19] on an arbitrary image (Figure 2a). Figures 2b, 2c, 2d show the super-pixels with positive weights as explanations for the top 3 predicted classes. What the neural network picks up on for each of the class is very natural to humans - Figure 2b in particular provides insight as to why acoustic guitar was predicted to be electric: due to the fretboard.

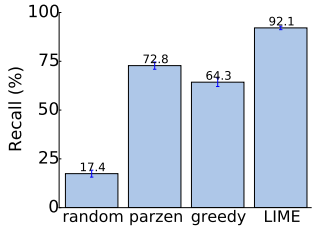
Explaining models

Although an explanation of a single prediction provides some understanding into the reliability of the classifier to the user, it is not sufficient to evaluate the model as a whole. Even though explanations of multiple instances can be quite insightful, these instances need to be selected judiciously, since users may not have ample time to examine explanations of a large number of instances. In this section, we propose such a selection technique that picks a diverse, representative set of explanations to show the user.

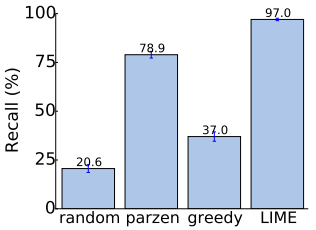
Formally, if g_i are the weights of the explanation of prediction x_i (most of which are zero), X is the set of instances ($|X| = n$), and k the number of instances to select, we define the importance of feature j as $w_j = \sqrt{\sum_{i=1}^{n} |g_{ij}|}$. Intuitively, we should pick explanations that cover all the important features. On the other hand, the set of explanations must not be redundant in the features they show the users. Thus we define an objective that, for any set of instances $V, |V| \leq k$, computes the total importance of the features that appear at least once in V, i.e.

$$\max_{V,|V|\leq k} \sum_{j=1}^{m} \mathbb{1}_{[\exists g_i \in V : g_{ij} > 0]} w_j$$

This function is submodular and monotone, and thus a greedy algorithm offers a constant-factor approximation guarantee of $1 - 1/e$ to the optimum [9]. We use this greedy approximation in all of our experiments.



(a) Sparse Logistic Regression



(b) Decision Tree

Figure 3: Recall on true explanations for transparent classifiers on the book dataset.

	LR	DT	NN	RF
Books				
Random	14.6	14.9	14.8	14.7
Parzen	84.0	92.2	87.6	94.3
Greedy	53.7	62.7	47.4	45.0
Ours	**96.6**	**97.8**	**94.5**	**96.2**
Dvds				
Random	14.2	14.6	14.3	14.5
Parzen	87.0	91.9	81.7	94.2
Greedy	52.4	66.1	58.1	46.6
Ours	**96.6**	**97.9**	**91.8**	**96.1**

Table 1: Averaged F1 score on trusted predictions.

Evaluation

There are a number of natural questions that arise for evaluating whether explanations are useful as insights into predictions and classifiers: (1) Are the explanations faithful to the model, (2) Can the explanations aid users in ascertaining trust in predictions, and (3) Are the explanations useful for evaluating the model as a whole. We will present preliminary, synthetic experiments to address these questions.

Experiment Setup

We use two sentiment analysis datasets (*books* and *dvds*, 2000 instances each) where the task is to classify product reviews as positive or negative [3]. We train decision trees (**DT**), logistic regression with L2 regularization (**LR**), and nearest neighbors (**NN**), all trained on bag of words as features. We also include random forests (with 1000 trees) trained with the mean word embedding [13] (**RF**), a setting that is quite difficult to interpret. We use the implementations and default parameters of scikit-learn [16] for all methods, unless noted otherwise. We divide each dataset into training (1600 instances) and testing (400 instances).

Along with our proposed approach (**LIME**), we also evaluate **parzen** [2], for which we take the 10 features with the highest absolute gradients. We set the hyper-parameters for parzen and LIME using cross validation. We also evaluate a **greedy** procedure (similar to [12]) in which we greedily remove features that contribute the most until the class changes (with a maximum of 10 features), and a **random** procedure that picks 10 random features as an explanation.

Experiment 1: Are explanations faithful to the model?

We can measure faithfulness of explanations on classifiers that are by themselves interpretable (sparse logistic regression and decision trees). For each prediction on the test set, we generate explanations and compute how many of the truly important features were recovered by the explanations. We report the average recall on the true explanations for the book dataset in Figure 3 (dvd dataset results are similar). Only LIME provides a consistently high recall for both logistic regression and decision trees. Greedy is worse on decision trees, as changing any one feature at a time often does not have an effect on the prediction.

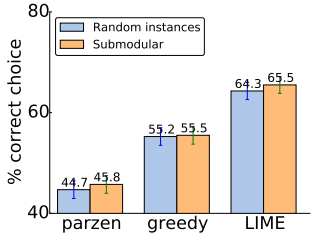
Experiment 2: Trusting individual predictions

In order to evaluate trust in predictions, we first randomly select 25% of the features to be "untrustworthy", i.e. assume that the users can identify such features and would not want to trust them. We thus develop datasets of "trustworthiness" by labeling predictions a black box classifier on the test set as "untrustworthy" if the prediction changes when untrustworthy features are removed from the instance, and "trustworthy" otherwise. In order to simulate users, we assume that users deem predictions untrustworthy from LIME and parzen explanations if the prediction from the linear approximation changes when all untrustworthy features that appear in the explanations are removed. For greedy and random, the prediction is mistrusted if any untrustworthy features are present in the explanation.

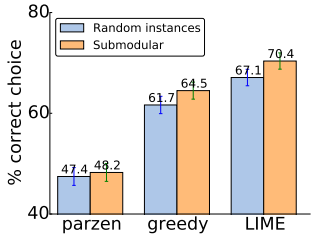
Using this setup, we report the F1 for each explanation method, averaged over 100 runs, in Table 1. The results indicate that LIME dominates others (all results are significant at $p = 0.01$) on both datasets, and for all of the black box models. The other methods either achieve a lower recall (i.e. they mistrust predictions more than they should) or lower precision (i.e. they trust too many predictions).

Experiment 3: Trusting models

In this experiment, we evaluate the utility of explanations for model selection, simulating the case where a human has to decide between two competing models with similar accuracy on validation data. For this purpose, we add 10 artificial "noisy" features as follows. On training and vali-



(a) Book dataset



(b) Dvd dataset

Figure 4: Choosing between two classifiers based on trust, after seeing $k = 10$ examples.

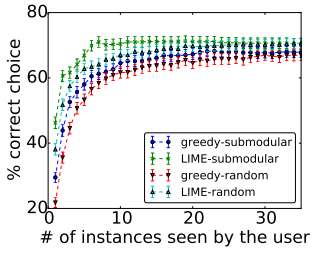


Figure 5: Varying the number of instances (k) for the dvd dataset.

dation sets ($80/20$ split of the original training data), each artificial feature appears in 10% of the examples in one class, and 20% of the examples of the other, while on the test instances, each artificial feature appears in 10% of the examples in each class. This recreates the situation where the models use both the features that are informative in the real world, and ones that are noisy and introduce spurious correlations. We also create pairs of classifiers to compare by randomly training two random forest classifiers with 30 trees until their validation accuracy is within 0.1% of each other, but their test accuracy is different by at least 5%.

The goal of this experiment is to evaluate whether a user can identify the better classifier based on the features that appear in a subset of k explanations. The simulated human marks the set of artificial features that appear in the k explanations as untrustworthy, following which we evaluate how many total predictions in the validation set should be trusted (as in the previous section, treating only marked features as untrustworthy). Then, we select the classifier with fewer untrustworthy predictions, and compare this choice to the classifier with higher test set accuracy.

We run this experiment for 800 trials for each dataset, and present results for $k = 10$ in Figure 4. We see that LIME is consistently more accurate that greedy and parzen, the latter being no better than random (50%). We also show the accuracy as k varies in Figure 5, where the submodular procedure is more helpful than random on the dvd dataset. Unfortunately submodular is not substantially better than random selection on the books dataset; we will investigate the reasons in future work.

Discussion

In this paper, we argue that trust is critical for human interaction with machine learning systems. Without an understanding of what a model is doing, human interaction becomes a leap of faith, or is based on some aggregate statistic (e.g. accuracy) that may not correspond to real world performance (like the example in Figure 1). In many scenarios, the lack of understanding causes machine learning not to be trusted - and thus not used at all. Thus, a human centered perspective on machine learning must include the notion of trust for both individual predictions and models.

We contend that being able to explain classifier predictions is a crucial task for building trust in machine learning. We outline the characteristics of a good explanation system, most importantly the ability to explain any black-box classifier and providing interpretable yet faithful descriptions. Along with proposing a modular and extensible approach called LIME that meets many of these criteria, we also design novel synthetic experiments that separately evaluate the various features of explanation systems.

Motivated by these ideas, there are a number of avenues we would like to explore in the future. Although the synthetic experiments show promise both as an evaluation platform for explanations, and for the promise of LIME as a explanation tool, it is severely handicapped by the lack of human experiments. We are also interested in exploring a broader spectrum of interpretable explanations, and investigate the kinds of explanations that are conducive to building trust in machine learning systems. Finally, we are interested in evaluation on real-world and larger datasets, where selecting an appropriate subset of instances is critical.

References

[1] Saleema Amershi, Max Chickering, Steven M. Drucker, Bongshin Lee, Patrice Simard, and Jina Suh. 2015. ModelTracker: Redesigning Performance Analysis Tools for Machine Learning. (April 2015).

[2] David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert Müller. 2010. How to Explain Individual Classification Decisions. *Journal of Machine Learning Research* 11 (Aug. 2010), 1803–1831.

[3] John Blitzer, Mark Dredze, and Fernando Pereira. Biographies, Bollywood, Boom-boxes and Blenders: Domain Adaptation for Sentiment Classification. In *Association for Computational Linguistics*.

[4] J. Quiñonero Candela, M. Sugiyama, A. Schwaighofer, and N. D. Lawrence. 2009. *Dataset Shift in Machine Learning*. MIT Press. (Editors).

[5] Rich Caruana, Yin Lou, Johannes Gehrke, Paul Koch, Marc Sturm, and Noemie Elhadad. 2015. Intelligible Models for HealthCare: Predicting Pneumonia Risk and Hospital 30-day Readmission. In *International Conference on Knowledge Discovery and Data Mining (KDD)*.

[6] Bradley Efron, Trevor Hastie, Iain Johnstone, and Robert Tibshirani. 2004. Least angle regression. *Annals of Statistics* 32 (2004), 407–499.

[7] Alex A. Freitas. 2014. Comprehensible Classification Models: A Position Paper. *SIGKDD Explor. Newsl.* 15, 1 (March 2014), 1–10.

[8] Shachar Kaufman, Saharon Rosset, and Claudia Perlich. 2011. Leakage in Data Mining: Formulation, Detection, and Avoidance. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '11)*. ACM, New York, NY, USA, 556–563.

[9] Andreas Krause and Daniel Golovin. 2014. Submodular Function Maximization. In *Tractability: Practical Approaches to Hard Problems*. Cambridge University Press.

[10] Su-In Lee, Honglak Lee, Pieter Abbeel, and Andrew Y. Ng. 2006. Efficient L1 Regularized Logistic Regression. In *Proceedings, The Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference, July 16-20, 2006, Boston, Massachusetts, USA*. 401–408.

[11] Benjamin Letham, Cynthia Rudin, Tyler H. McCormick, and David Madigan. 2015. Interpretable classifiers using rules and Bayesian analysis: Building a better stroke prediction model. *Annals of Applied Statistics* (2015). accepted.

[12] David Martens and Foster Provost. 2014. Explaining Data-driven Document Classifications. *MIS Q.* 38, 1 (March 2014), 73–100.

[13] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *Advances in Neural Information Processing Systems 26*, C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger (Eds.). Curran Associates, Inc., 3111–3119.

[14] George Miller. 1956. The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information. (1956).

[15] Kayur Patel, Naomi Bancroft, Steven M. Drucker, James Fogarty, Andrew J. Ko, and James Landay. 2010. Gestalt: Integrated Support for Implementation and Analysis in Machine Learning. In *Proceedings of the 23Nd Annual ACM Symposium on User Interface Software and Technology (UIST '10)*. ACM, New York, NY, USA, 37–46.

[16] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[17] Ivan Sanchez, Tim Rocktaschel, Sebastian Riedel, and Sameer Singh. 2015. Towards Extracting Faithful and Descriptive Representations of Latent Variable Models. In *AAAI Spring Syposium on Knowledge Representation and Reasoning (KRR): Integrating Symbolic and Neural Approaches*.

[18] D. Sculley, Gary Holt, Daniel Golovin, Eugene Davydov, Todd Phillips, Dietmar Ebner, Vinay Chaudhary, Michael Young, and Jean-Franç Crespo. 2015. Hidden Technical Debt in Machine Learning Systems. In *Advances in Neural Information Processing Systems 28*, C. Cortes, N.D. Lawrence, D.D. Lee, M. Sugiyama, R. Garnett, and R. Garnett (Eds.). Curran Associates, Inc., 2494–2502.

[19] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going Deeper with Convolutions. In *CVPR 2015*.

[20] Fulton Wang and Cynthia Rudin. 2015. Falling Rule Lists. In *Proceedings of Artificial Intelligence and Statistics (AISTATS)*.