A MODEL EXPLANATION SYSTEM

Ryan Turner
Northrop Grumman Corporation

ABSTRACT
We propose a new methodology for explaining the predictions of black box classifiers. We use the motivating paradigm that predictive performance is of primary importance but human analysts (e.g., in fraud detection) desire a classifier’s predictions to be augmented with useful explanations. To be truly general and principled, we derive a scoring system for finding explanations based on formal requirements. In this system, the explanations are assumed to take the form of simple logical statements. We derive an efficient Monte Carlo algorithm to find explanations for black box classifiers with finite sample guarantees. The methodology is then applied to interesting examples in facial recognition and credit data.

1. INTRODUCTION

We propose a general model explanation system (MES) for “explaining” the output of black box classifiers. In this introduction we use the motivating example of a classifier trained to detect fraud in a credit card transaction history. The key aspect is that we provide explanations applicable to a single prediction, rather than provide an interpretable set of parameters. We focus on explaining positive predictions (alerts). However, the presented methodology is symmetrically applicable to negative predictions.

In many classification applications, but especially in fraud detection, there is an expectation of false positives. Alerts are given to a human analyst before any further action is taken. Such problems are sometimes referred to as “anomaly detection.” Analysts often insist on understanding “why” there was an alert, since an opaque alert makes it difficult for them to proceed. Analogous scenarios occur in computer vision, credit risk, spam detection, etc.

Furthermore, the MES framework is useful for model criticism. In the world of generative models, practitioners often generate synthetic data from a trained model to get an idea of “what the model is doing” [1]. Our MES framework augments such tools. As an added benefit, MES is applicable to completely nonprobabilistic black boxes that only provide hard labels.

Example: In the context of credit card fraud we may have feature vectors $x$ containing the number of online transactions, the geographic distance traveled for in-person transactions, the number of novel merchants, and so on. A simple example explanation is: “Today, there were two in-person transactions in the USA, followed by $1700 in country X.” MES would output “($x_i \geq 2) \land (x_j \geq 1700)$” for the appropriate features $i$ and $j$. The former could be generated via a natural language generation (NLG) module.

1.1. Explanation vs. interpretability

We adopt the paradigm where prediction accuracy is of paramount importance, but explanation is also important. Therefore, we are not willing to give up any predictive accuracy for explanation. Both machine learning and statistics have a long history of building models that are “interpretable”: such as, (small) decision trees [2] and sparse linear models [3]. MES augments black box predictions with explanations, as the best prediction system may not be “interpretable.”

Historically, this dilemma has created two distinct approaches: 1) the “interpretable” (small) models approach, common in scientific discovery/bioinformatics, and 2) the accuracy-focused approach, common in computer vision with methods including deep learning, k-NNs, and support vector machines (SVMs). The downside of the interpretable approach is seen in machine learning competitions, where the winning methods are typically nonparametric (big), or have a very large number of parameters (e.g., deep networks).

MES has elements of both approaches. We do not aim to summarize how the model “works in general” (see [4, 5, 6]), but only seek explanations of individual cases. Although the distinction is subtle, explanation is a much easier task than explaining an entire model. MES is the first method to utilize this weaker requirement to augment black boxes with explanations without affecting accuracy.

We graphically depict MES on a simple example in Fig. 1. The rest of this paper is organized as follows: In Section 2, we formalize some basic desiderata on how an explanation system should work. In Sections 3–4 we derive a Monte Carlo (MC) algorithm (with finite sample guarantees) that works with arbitrary input densities and black box classifiers. We show enlightening results on the workings of a face recognition system and credit risk classifier in Sections 5–6.
2. FORMAL SETUP

Consider a black box binary classifier \( f \) that takes a feature vector \( x \in \mathcal{X} = \mathbb{R}^d \) and provides a binary label: \( f \in \mathcal{X} \rightarrow \{0, 1\} \). In the introductory examples, explanations are Boolean statements about the feature vector. In effect, an explanation \( E \) is a function from \( \mathcal{X} \) to \( \{0, 1\} \). The mapping \( E^* \in \mathcal{X} \rightarrow \mathcal{E} \) finds the best explanation from the set of possible explanations \( \mathcal{E} \subset \mathcal{X} \rightarrow \{0, 1\} \). We also define that \( E \) contains a default “null explanation” \( E_0(x) := 1 \).

2.1. Desired properties

What remains is to describe some desiderata on \( E^* \). However, we must first define some key terms: **Eligibility:** An explanation is eligible if it increases the probability that the classifier alerts: Explanation \( E \in \mathcal{E} \) is eligible if \( P(f(x) = 1|E(x) = 1) \geq P(f(x) = 1) \), where \( x \sim p \). Note that \( f \) and \( E \) are deterministic functions of the input \( x \); we are marginalizing over the inputs \( p(x) \). Also, the null explanation \( E_0 \) is always eligible under this definition. **Generality:** An explanation’s generality \( G \in \mathcal{E} \rightarrow \{0, 1\} \) is the probability of it being true (among possible cases): \( G(E) := P(E(x) = 1|f(x) = 1) \). **Accuracy:** An explanation’s accuracy \( A \in \mathcal{E} \rightarrow [0, 1] \) is the probability the classifier alerts given the explanation is true: \( A(E) := P(f(x) = 1|E(x) = 1) \). Eligibility is now definable in terms of accuracy, \( A \) cannot be worse than \( E_0: A(E) \geq A(E_0) \). **Validity:** An explanation \( E \) is valid-at-\( x \) if it is eligible and true-at-\( x \) \((E(x) = 1) \).

To summarize, eligibility, generality, and accuracy are properties of \( E \) that do not depend on \( x \), although they do depend on the marginal \( p(x) \). Whether an explanation \( E \) is true or valid requires knowledge of the test input \( x \). In a classification context, accuracy \( A \) and generality \( G \) are analogous to precision and recall, respectively.

We contend that a sensible \( E^*(x) \) mapping returns the most preferable explanation that is valid-at-\( x \); and the following two properties: 1) If two valid-at-\( x \) explanations have equal accuracy then the one with more generality is preferred: If \( E_1 \) \& \( E_2 \) have equal generality then the one with more accuracy is preferred: 2) If two valid-at-\( x \) explanations have equal generality then the one with more accuracy is preferred: If \( A(E_1) > A(E_2) \) then \( E_2 \prec E_1 \). Requiring generality is useful since useless explanations such as \( x_i \in [a, b] \) may have different accuracies. Conversely, explanations like \( x_i \in \mathbb{R} \) have generality 1, but no accuracy. Hence, a good explanation system trades off the need for both accuracy and generality.

These requirements are naturally encoded in a scoring function: 

\[
E^*(x) = \arg\max_{E \in \mathcal{E}} S(A(E), G(E)) \quad \text{s.t.} \quad E(x) = 1, \quad \partial S/\partial A > 0, \quad \partial S/\partial G > 0, \quad S \in [0, 1]^2 \rightarrow \mathbb{R},
\]

where \( \mathcal{E} \subset \mathcal{E} \) is the set of eligible explanations; and we have put gradient conditions on \( S \). Note that (1) only cares about the relationship between \( E \) and \( f \); we do not care if \( E(x) \) directly predicts the data.

We describe three examples for \( S \) that obey (1) and are normalized on \([0, 1]\) for eligible explanations: mutual information \( S = \operatorname{MI}(E; f) = H(f) - \operatorname{H}(f|E) \), correlation \( S = \operatorname{Corr}(E, f) \), and covariance \( S = \operatorname{Cov}(E, f) \). Being normalized, regardless of the constant background rate \( P(f) \), means the score \( S \) is suitable as an explanation quality score. Although mutual information is intuitively appealing, covariance will provide greater ease of analysis in Section 3.

---

1. Notational warning: an explanation is \( E \); an expectation is \( E \).
2. An explanation is either sufficiently simple to be in \( \mathcal{E} \) or not. There is no other metric of “explanation simplicity.”
3. We write \( P(f) \) instead of \( P(f(x) = 1) \) for brevity. We write \( S(E) \in \mathcal{E} \rightarrow \mathbb{R} \) for the score of an explanation \( E \), and \( S(a) \) for the scores of explanations parameterized by a threshold \( a \in \mathbb{R} \).

The setup in (1) provides three appealing derived properties: 1) Any valid-at-\( x \) explanation \( E \), not independent of \( f \), is preferable to the null explanation \( E_0 \). 2) If the decision rule \( f \) is in \( \mathcal{E} \), then it is preferable to any other explanation: If \( f \in \mathcal{E} \) then \( E^*(x) = f \) for all \( x \) that alert. 3) Based on the definition of eligibility, any explanation \( E \neq E_0 \) eligible for explaining \( f(x) = 1 \) would be ineligible in the converse problem of trying to explain \( f(x) = 0 \).

2.2. Comparison with decision trees

Although it may appear that we are merely “reinventing decision trees,” there are key differences between MES and a decision tree. At a high level, MES is explaining why a decision is made on a single input, while decision trees aim to make the entire model easy to understand. More precisely, although we use simple decision functions in \( \mathcal{E} \), the \( E^* \) mapping may be arbitrarily complex. By contrast, simple decision trees cannot match any black box classifier exactly. If the decision tree is not the top performing model, we must sacrifice performance for interpretability. MES augments any model with an explanation, alleviating any need to sacrifice performance.

2.3. Dependence on input distribution

Discriminative models, even when probabilistic, do not learn the input feature distribution, which we refer to as \( p \). However, they can be augmented by learning an input feature distribution “on the side.” An interesting twist of MES is that it does depend on the input distribution even when explaining discriminative models. We contend that this dependence is unavoidable since knowing the correlations in the input space is essential to making sensible explanations. For example, heuristics based on gradients with respect to input features can be tricked when features on differing scales are correlated.

3. SCORE ESTIMATION WITH BLACK BOX MODELS

In this section we use simple Monte Carlo to approximate the optimization in (1) with black box models. We merely require the classifier \( f \) be queryable at an arbitrary input \( x \) and that we can obtain samples from the input density \( p(x) \). We allow for general explanation functions of the form \( g_i \in \mathcal{X} \rightarrow \mathbb{R} : \)

\[
E = \bigcup_{i=1}^{M} \{I[g_i(x) \leq a], \forall a \in \mathbb{R} \}.
\]

(2)

Explanations of the form \( I[g_i(x) \geq a] \) are obtainable by including \( g_i = -g_i \in \mathcal{E} \). The **axis aligned** explanations from Fig. 1 are recovered using \( g_i(x) = \pm x_i \), yielding \( E = \bigcup_{i=1}^{d} I[x_i \leq a], \forall a \in \mathbb{R} \).

Alternatively, we may have a predefined set of linear decision functions that are reasonable explanations: \( g_i(x) = w_i^\top x + b_i \). We could also use a distance function \( d \) to “class archetypes” \( \{\alpha_i \in \mathbb{R}^d\}_{i=1}^{M} : g_i(x) = d(x, \alpha_i) \).

Although we can optimize an MC estimate of any scoring function, we use a scoring function equivalent to the covariance. This allows us to provide worst-case guarantees about the closeness of the MC explanation to the true optimal explanation. We use:

\[
S(E) = E \rightarrow \mathbb{R} = P(E|f = 1) - P(E|f = 0)
\]

\[
= G(E) (1 - (P(f)^{-1} - 1)^{-1}(A(E)^{-1} - 1))
\]

(3)

(4)

We note that \( S(E) \) is equivalent to the covariance, i.e., \( S(E) \propto P(E|f) - P(E) \propto \operatorname{Cov}(E, f) \), where we have utilized that \( E \) and \( f \) are binary variables in the algebraic manipulations. The reader can also verify that (3) satisfies the gradient requirements in (1) when \( E \) is eligible \( (P(f) \prec A(E)) \).
We now describe the general MC procedure used to optimize explanations. We only consider a single explanation function \(g_i\), for the time being and drop the index \(i\) for brevity. Defining \(z := g(x)\),

\[
S(E) = P(z \leq a|f = 1) - P(z \leq a|f = 0)
\]

\[
= \text{CDF}_a(f|a = 1) - \text{CDF}_a(f|a = 0) := F(a) - H(a).
\]

We refer to \(S(E)\) as the Kolmogorov score as \(\max_a |S(E)|\) is known as the Kolmogorov distance between \(p(z|f = 1)\) and \(p(z|f = 0)\). Reassuringly, \(E\) is eligible iff \(S(E) \geq 0\), which is shown utilizing that \(E\) and \(f\) are binary variables and Bayes’ rule.

We approximate (3) using \(S_n(\hat{a}) := F_n(\hat{a}) - H_n(\hat{a})\), where \(F_n\) and \(H_n\) are empirical CDFs of \(F\) and \(H\) from \(n\) MC samples. This is easily done with rejection sampling. Next, we provide an upper confidence bound (UCB) on the suboptimality \(e\) of the approximate explanation threshold \(\hat{a}\) vs the exact threshold \(a^*\):

\[
e := S(a^*) - S(\hat{a}) \geq 0,
\]

\[
a^* = \arg\max_a S(a), \quad \hat{a} = \arg\max_a S_n(a).
\]

Utilizing \(S_n(a^*) - S_n(\hat{a}) \leq 0\), we bound (6) as:

\[
e = S_n(a^*) - S_n(\hat{a}) \leq S_n(a^*) - S_n(\hat{a}) + (S_n(\hat{a}) - S(\hat{a}))
\]

\[
\leq (S(a^*) - S_n(a^*)) + (S_n(\hat{a}) - S(\hat{a})) \leq 2 \max_a |S_n(a) - S(a)|
\]

\[
\leq 2 \max_a |F_n(a) - F(a)| + 2 \max_a |H_n(a) - H(a)|.
\]

Next, we use the union bound and the DKW inequality [8] to show:

\[
P(e \geq \epsilon) \leq P(\max_a |F_n(a) - F(a)| \geq \epsilon/4)
\]

\[
+ P(\max_a |H_n(a) - H(a)| \geq \epsilon/4)
\]

\[
\leq 4 \exp(-2n(\epsilon/4)^2) = 4 \exp(-n\epsilon^2/8) =: \delta_0.
\]

This bounds the suboptimality when searching on a single explanation function \(g_i\). When searching over \(M\) different explanation functions we use a multiple comparison correction of \(\delta_0 = \delta/M\). This allows reuse of MC samples for searches on each explanation function. To obtain score suboptimality \(e\) with confidence \(\delta\) we need

\[
n = \lceil 8 \log(4M/\delta)/\epsilon^2 \rceil
\]

MC samples. Since \(M\) is inside the log of (10), using a simple multiple comparison correction only adds a small penalty in \(n\).

3.1. Implementation

The optimization to find the best explanation is done as follows: For each explanation function \(g_i\), we utilize the output of a precomputation phase to efficiently find the optimal threshold \(\hat{a}\) and its corresponding score. We then compare the optimized scores for each explanation function \(g_i\) and report the function \(g_i\) (and corresponding threshold \(\hat{a}\)) with the highest score.

Our precomputation phase is based on finding the cumulative maximum w.r.t. \(a\) of the estimated score function \(S_n\). More precisely, \(\hat{a} = \arg\max_{a \in [g(x),\infty)} F_n(a) - H_n(a)\),

\[
\hat{a} = \arg\max_{a \in [g(x),\infty)} F_n(a) - H_n(a), \quad (11)
\]

where the max is a tiebreaker so that \(\hat{a}\) equals the largest \(a\) of the set returned by the argmax. Eq. (11) allows us to put MES precomputation in the form of Algo. 1. The computations to find \(A_{1:M}\) are informally thought of as the best optimum so far scanning from \(+\infty\) backwards. The variables \(H_n, F_n, \hat{S}_n\) and \(A_i\) are all piecewise constant functions and efficiently represented in a computer exactly. After precomputation, we efficiently find the explanation for a test point \(x\) using Algo. 2. No extra MC samples are required for Algo. 2.

4. EXTENDING TO LARGER EXPLANATION SPACES

In Section 3 we developed the machinery for jointly choosing among \(M\) explanation functions \(g_{1:M}\) and a scalar threshold parameter \(\alpha \in \mathbb{R}\). In this section we propose extended MES, which maximizes the score \(S\) with respect to some continuous free parameters \(\theta\) of the explanation \(g\). For instance, Section 3 mentions using linear decision functions as explanations. In this section we assume explanations of the general form:

\[
\mathcal{E} = \{k(g(x; \theta) \leq a) \quad \forall a \in \mathbb{R}, \forall \theta\},
\]

where \(g\) is now parameterized by \(\theta\) rather than a discrete index \(i\). In the case of linear explanations \(\theta = w \in \mathbb{R}^D\). We now have to optimize the score (1) with respect to a free vector parameter \(\theta\). To do this efficiently we put the objective in the form of an expected loss. This enables us to employ learning theoretic results that replace the optimization with a convex surrogate.

First, we find it convenient to rewrite the explanations as:

\[
\mathbb{I}[g(x; \theta) \leq a] = \mathbb{I}[0 \leq a - g(x; \theta)] = u(a - g(x; \theta)) = u(\tilde{g}(x; \tilde{\theta}))
\]

\[
\tilde{g}(x; \tilde{\theta}) := a - g(x; \theta), \quad \tilde{\theta}^T := [\theta^T \quad a],
\]

where \(u(\cdot)\) is the unit (Heaviside) step function. Given a test input \(x\), where \(f(x) = 1\), we find the best explanation \(E\) using

\[
E^*(x) = \arg\min_{E \in \mathcal{E}} \mathbb{E}_x[E(x')]|f(x')] - \mathbb{E}_x[E(x' - f(x))|f(x)]
\]

s.t. \(E(x) = 1\), \(x' \sim p\),

where \(f\) and \(\neg f\) are a shorthand for \(f(x') = 1\) and \(f(x') = 0\), respectively. Note that the \(\arg\max\) became an \(\arg\min\) by swapping the order of subtraction. Since the explanation space \(\mathcal{E}\) is now parameterized by \(\tilde{\theta}\), (15) is equivalent to:

\[
\theta^*(x) = \arg\min_{\theta \in \mathcal{E}_x} [\mathbb{E}_x[u(\tilde{g}(x'; \tilde{\theta}))] - \mathbb{E}_x[u(\tilde{g}(x' - f(x); \tilde{\theta}))]]
\]

s.t. \(\tilde{g}(x; \tilde{\theta}) \geq 0\),

where \(\theta^*(x) = \tilde{\theta}\) are the best parameters \(\tilde{\theta}\) for explaining \(x\).
Manipulating different forms of the zero-one loss gets us closer to putting (16) in an expected loss form:

\[
E_x'[u(\hat{g}(\mathbf{x}; \Theta))|f] - E_x'[u(\hat{g}(\mathbf{x}; \Theta))|\neg f] = E_x'[u(\hat{g}(\mathbf{x}; \Theta))|f] - E_x'[u(\hat{g}(\mathbf{x}; \Theta))|\neg f] - 1
\]

where we set \( x := 2f - 1 \in \{-1, 1\} \) and \( f \) is the zero-one loss: \( \ell(x) := (1 - x) \). In (18) we utilize that \( |u(f) - f| = \ell(2f - 1) = \ell(yf) \) are alternative forms of the zero-one loss for some prediction \( f \in \mathbb{R} \). Finally, by defining a “class rebalanced” version of \( p \), we achieve the expected loss formulation:

\[
p'(\mathbf{x}, y) := p(\mathbf{x}|2f(\mathbf{x}) - 1 = y) \frac{1}{2}\{y \in \{-1, 1\}\},
\]

\[
\theta^*(\mathbf{x}) = \arg\min_{\theta} \mathbb{E}_{p'}[\ell(y \hat{g}(\mathbf{x}; \Theta))] \text{ s.t. } \hat{g}(\mathbf{x}; \Theta) \geq 0. \tag{19}
\]

Although this objective can be estimated with MC samples from \( p' \), the resulting function is multivariate and discontinuous. This makes direct optimization of (19) problematic. However, [9] showed objectives in the form of (19) can be solved by replacing \( \ell \) with a convex surrogate loss \( \phi \in \mathbb{R} \rightarrow \mathbb{R}^+ \) such as the hinge loss or log-logistic:

\[
\theta^*(\mathbf{x}) = \arg\min_{\theta} \mathbb{E}_{p'}[\phi(y \hat{g}(\mathbf{x}; \Theta))] \text{ s.t. } \hat{g}(\mathbf{x}; \Theta) \geq 0. \tag{20}
\]

If we take a large number of MC samples, the resulting parameter estimates have asymptotically minimal risk.

4.1. Simplifying the constraint

Although it is possible to solve for \( \theta^*(\mathbf{x}) \) by optimizing (20) directly, we take the “poor man’s” approach of putting the constraint \( \hat{g}(\mathbf{x}; \Theta) \geq 0 \) in the objective. This has the practical advantage of allowing us to use existing (highly optimized) software modules. We modify our objective as follows using \( \gamma \in (0, 0.5) \):

\[
\theta^*(\mathbf{x}) = \arg\min_{\theta} \mathbb{E}_{p'}[\ell(y \hat{g}(\mathbf{x}; \Theta)) + (1 - \gamma)\ell(\hat{g}(\mathbf{x}; \Theta))]
\]

\[
= \arg\min_{\theta} \mathbb{E}_{p'}[\ell(y \hat{g}(\mathbf{x}; \Theta))],
\]

\[
p''(\mathbf{x}, y) := (1 - \gamma)\mathbb{I}(y = 1)\delta_{\mathbf{x}}(\mathbf{x}') + \gamma p'(\mathbf{x}', y),
\]

where \( \delta_{\mathbf{x}}(\cdot) \) is a Dirac delta centered at \( \mathbf{x} \). Now applying the surrogate loss function \( \phi \) we optimize:

\[
\theta^*(\mathbf{x}) = \arg\min_{\theta} \mathbb{E}_{p''} \phi(y_i \hat{g}(\mathbf{x}_i; \Theta)), (\mathbf{x}_i, y_i) \sim p'',
\]

which matches the form found in [9].

4.2. Linear explanation classes

In the case of linear explanations we have \( \hat{g}(\mathbf{x}; \Theta) = \Theta^\top \mathbf{x} \), where we have defined \( \mathbf{x}^\top := [\mathbf{x} \mathbf{1}^\top] \). This puts (23) in the form

\[
\theta^*(\mathbf{x}) = \arg\min_{\theta} \mathbb{E}_{p''} \phi(y_i \Theta^\top \mathbf{x}_i), (\mathbf{x}_i, y_i) \sim p''.
\]

When \( \phi \) is the log-logistic we find \( \Theta \) by applying logistic regression to MC samples \( D := (\mathbf{x}_1:n, y_1:n) \). Likewise, when \( \phi \) is the hinge loss we use a linear SVM.

We now proceed as follows: We pass an \( M \) point subset of the data through \( \theta^*(\mathbf{x}) \) to yield \( \{\hat{\Theta}_1:n, D\} \). We then finish by placing \( \{\hat{\Theta}_1:n\} \) back in the form of (12), utilizing (14):

\[
\mathcal{E} = \bigcup_{i=1}^{M} \{\mathbb{I}[\mathbf{w}_1:n^\top \mathbf{x} + b_i \leq a], \forall a \in \mathbb{R}\},
\]

\[
\mathbf{w}_i = -[\hat{\Theta}_1:1:D]/||[\hat{\Theta}_1:1:D]||, \quad b_i = -[\hat{\Theta}_1:D+1:||[\hat{\Theta}_1:1:D]||],
\]

where \( || \cdot || \) is the \( L_2 \) norm. Although normalizing \( \mathbf{w} \) and \( b \) is not strictly necessary, their scale is not identifiable, and are thus normalized for uniformity. The \( b \) component is likewise optional due to its unidentifiability with \( a \).

Extended MES is based upon a two-phase approach. We first find the parameters for our explanations \( g_{1:M} \) using Algo. 3. Since the methods of Section 3 have finite sample guarantees, the output of Algo. 3 is passed to Algos. 1 and 2 to provide the final explanations.
The explanation becomes: $x_E \cdot x = \sum x_H > a$. Thus we set the right image to be $x_E := C^T w$. Then we set the far right image to be $x_H := x_E \odot x$. Thus, by setting $\alpha = \sum x_H - a$, the explanation is false: $E(x_E) = 0$. In the cases shown, this is achieved with $\alpha = 2$. The white balance quoted is $\sum x_H / \dim x \times 100\%$, where $\dim x = 1,850$ is the number of pixels.

**Algorithm 3 Extended MES**

```
input data subset $X \in \mathcal{X}^n$, $n$, classifier $f$, input density $p$
repeat
  $x \leftarrow$ random point from $X$
  $D \leftarrow n$ samples from $p''$ (see (22)) using $f$, $x$, and $p$
  Set $\theta''$ by fitting linear SVM (or logistic reg.) to $D$
  Delete from $X$ points correctly classified by SVM
  Compute $w$ and $b$ by (25) and append to list $L$
until $X$ empty
output parameter list $L$ (used for $g_{1:M}$)
```

**Table 1.** Empirical performance of MES vs. heuristics. The confidence intervals are derived from a bootstrap analysis with $10^4$ bootstrap replications. Additionally, we performed a one-sided paired bootstrap hypothesis test on the mean performance difference versus MES. All of the MES improvements are significant at $p < 10^{-3}$, except for (*) with $p = 0.04$ and (†).

<table>
<thead>
<tr>
<th>method</th>
<th>accuracy $A$</th>
<th>generality $G$</th>
<th>eligible</th>
<th>score $S$</th>
</tr>
</thead>
<tbody>
<tr>
<td>MES</td>
<td>0.78±0.04</td>
<td>0.94±0.03</td>
<td>1.00±0.00</td>
<td>0.84±0.04</td>
</tr>
<tr>
<td>grad</td>
<td>*0.73±0.06</td>
<td>0.51±0.06</td>
<td>0.91±0.04</td>
<td>0.43±0.08</td>
</tr>
<tr>
<td>local lin.</td>
<td>0.51±0.05</td>
<td>0.88±0.03</td>
<td>1.00±0.01</td>
<td>0.44±0.03</td>
</tr>
<tr>
<td>SV</td>
<td>†1.00±0.00</td>
<td>0.06±0.03</td>
<td>1.00±0.00</td>
<td>0.06±0.03</td>
</tr>
</tbody>
</table>

5. FACE RECOGNITION EXAMPLE

We now demonstrate MES on the scikit-learn demo “Faces recognition example using eigenfaces and SVMs.” The faces are reduced to dimension $D = 150$ from $50 \times 37 = 1,850$ using PCA. Then 966 training examples are plugged into a (Gaussian kernel) multiclass SVM for classifying the faces as one of seven political figures. When explaining a classification of face $k$ (e.g., Bush) we convert the SVM to a binary black box, informally as $f(x) = I\{\text{SVM}(x) = k\}$.

Throughout this paper, we use $\epsilon = 0.025$ and $\delta = 0.05$ implying $n = 129,099$. Induced from the assumptions of PCA, we use a standard multivariate Gaussian for the input density $p(x)$.

In Fig. 2, we use axis aligned explanations and MES to explain why the SVM classifies Hugo Chavez as George W Bush. In the far right image we see the classifier is utilizing the black arc under his teeth and the dark area around his (reader’s left) right eye. In most training photos of Bush, he has an open (dark) mouth and a lot of shading above his eyes. After subtracting the (normalized) eigenface from the original, the SVM correctly predicts the face as Chavez. The corrected face (far left) has a lighter right eye and whiter smile.

We are also able to find interesting explanations using the linear explanations from Section 4. In Fig. 3 we show a correct prediction of Colin Powell, and use MES to shed light on the responsible elements of the images. Extended MES allows the explanation faces on the right in Fig. 3 to be any image, not just an eigenface as in Fig. 2.

Fig. 3 shows that the SVM is “picking up” on the dark shading on the left side of Powell’s chin, shading below his left eye, and a wide area for the dark pixels of his nostrils and nasolabial folds (smile lines). Indeed, in many training images of Powell the lighting is to his right. MES has uncovered the high relevance that the classifier places on these non-obvious features.

We also provide quantitative results on the quality of the explanations in Table 1. We compare with three standard heuristics applicable as explanations for SVMs: 1) the nearest support vector (SV) in the same class, e.g., $E(x) = I\{\arg\min_j d(x, SV_j) = i\}$, 2) a local linearization of the confidence score of the SVM as a linear explanation, and 3) which input dimension (eigenface) gives the largest gradient w.r.t. the SVM confidence score. We assess the mean empirical accuracy, generality, proportion eligible, and score of the explanations estimated using the test set examples.

MES explanations are always eligible and dominate the heuristics in terms of score. Also, MES provides significant advantages in terms of generality and accuracy, simultaneously, over local linear and grad. Although the SV heuristic has high accuracy, those explanations are only applicable to very small pieces of the input space; they thus have low generality, albeit with high accuracy. We further emphasize that all of these heuristics are only applicable to certain models while MES can just as easily be applied to any black box.
To further show the generality of MES, we use it on the UCI German credit data set. The data consists of 1000 individual loan applications including ground truth on whether the individual was a bad credit risk. Twenty features are present in the data (6 categorical, 7 ordinal, and 7 numeric). The data is randomly split into 750 training and 250 test points. The categorical features are encoded in a 1-of-k encoding scheme and use unary encoding for the ordinal features. This yields a total of 48 possible features.

We chose to apply MES to $L_1$ logistic regression (LR) as it was the top performing model after an extensive comparison including cross-validated SVMs and decision trees. For the input distribution, we use the empirical distribution on the discrete features and then fit cross-validated SVMs and decision trees. For the input distribution, the top performing model after an extensive comparison including the predictively optimal model on 22.4% of the test points. It would be problematic to try to use the constrained model to explain a prediction of the full model when it is making an opposing prediction. The same story applies to the small decision tree, where after crossvalidating max depth, merely considers an individual high risk if they have no checking account and low risk otherwise. By using the full LR model and MES we have coherently achieved the best performance and more enlightening explanations than alternative methods.

### 7. CONCLUSIONS

We have presented a general framework for explaining black box models. The framework alleviates the tension between performance and interpretability. We described an MC algorithm (with accuracy guarantees) for black box classifiers. The methodology is useful for model criticism: It was used to explain, and then correct, why a classifier from a standard face recognition demo misclassified a seemingly standard test input. Furthermore, it is useful for human interface: The credit risk example has demonstrated that our methodology provides useful explanations in an important application.

### 8. REFERENCES


