Optimized Scoring Systems: Towards Trust in Machine Learning for Healthcare and Criminal Justice

Cynthia Rudin and Berk Ustun
Duke University and Harvard University

Questions of trust in machine learning models are becoming increasingly important, as these tools are starting to be used widely for high-stakes decisions in medicine and criminal justice. Transparency of models is a key aspect affecting trust. This paper reveals that there is new technology to build transparent machine learning models that are often as accurate as black box machine learning models. These methods have had impact already in medicine and criminal justice. This work calls into question the overall need for black box models in these applications.

There has been an increasing trend in healthcare and criminal justice to leverage machine learning for high-stakes prediction problems such as detecting heart attacks (Weng *et al.* 2017), diagnosing Alzheimer's disease (Pekkala *et al.* 2017), and assessing recidivism risk (Berk & Bleich 2013, Tollenaar & van der Heijden 2013). In many of these problems, practitioners are deploying black box machine learning models that do not explain their predictions in a way that humans can understand. In some cases, model development is outsourced to private companies, who build and sell proprietary predictive models using confidential datasets, without regulatory oversight.

The lack of transparency and accountability of a predictive model can have severe consequences when it is used to make decisions that significantly affect human lives. In criminal justice, proprietary predictive models can lead to questions about due process, or may discriminate based on race or poverty status (Wexler 2017b). In 2015, for instance, Billy Ray Johnson was imprisoned based on evidence from software developed by a private company, TrueAllele, which refused to reveal how the software worked. This led to a landmark case (People v. Chubbs) where the California Appeals Court ruled that such companies were not required to reveal how their software worked. As a different example, consider the controversy surrounding the COMPAS recidivism prediction model (Northpointe 2015), which is used for several applications in the U.S. criminal justice system, but does not provide clear reasons for its predictions. COMPAS has been accused of discriminating on the basis of race (Angwin *et al.* 2016, Citron 2016), and possibly uses socioecononomic information such as how often the individual is not paid above minimum wage.

A key problem with proprietary models is that they are prone to data-entry errors. There have been cases such as that of Glenn Rodríguez, a prisoner with a nearly perfect record, who was denied parole as a result of an incorrectly calculated COMPAS score (Wexler 2017b,a), with little recourse to argue, or even to determine how his score was computed. There have been cases where criminological risk scores (even simple ones) were miscalculated, allowing dangerous criminals to be released, who subsequently commit murders (Ho 2017) or other crimes. Issues like those discussed above have led to new regulations such as the European Union's "right to explanation" (Goodman & Flaxman 2016), which requires explanations from any algorithmic decision-making tool that significantly affects humans.

Because mistakes in healthcare and criminal justice can be serious, or even deadly, it can be beneficial for companies not to disclose their models. If the model is allowed to be hidden, the company never needs to fully justify why any particular prediction was made, and could avoid liability when the model makes mistakes. This leads to misaligned incentives, where the users of the tools would strongly benefit from transparent predictive models, but this would equally undermine profits for selling predictive models. Since these industries have a strong disincentive from building transparent models, there has been little work done on determining the answers to the following questions:

1. *Are there interpretable predictive models that are as accurate as black box models?* When we trust companies to build black box models, we are implicitly assuming that their models are more accurate than transparent models. Is it possible that for many given black box models, an alternative model exists that is just as accurate, but that is so simple that it can fit on an index card? We claim the answer is yes. A compelling argument of Breiman (2001), called the *Rashomon effect*, indicates that for many applications, there may exist a large class of models that predict almost equally well. Among this large class of models are those from the various black box machine learning methods (e.g., support vector machines, random forests, boosted decision trees, neural networks). There is no inherent reason that this class would exclude interpretable models. This observation also helps to explain the 40 years of literature on the surprising performance of simple linear models (Dawes 1979, Holte 1993).
2. *What are the desired characteristics of an interpretable model, if one exists?* The answer to this question changes for each audience and application (Kodratoff 1994, Pazzani 2000, Freitas 2014). We might desire accuracy in predictions, risks that are calibrated, and we might want the model to be calculated by a judge or a doctor without a calculator, which makes it easier to explain to a defendant or medical patient. Predictions from simpler models are much easier to verify, leading to fewer calculation errors and more robust decisions. A model with all of the characteristics listed above may not exist for any given problem, but if it does, it would be better to use than a black box.
3. *If an interpretable model does exist, is it possible to find it?* Interpretability, transparency, usability, and other desirable characteristics in predictive models lead to computationally hard optimization problems, such as mixed-integer non-linear programs. It is much easier to find an accurate unintelligible model than an interpretable one.

The renaissance from proprietary predictive models back to interpretable predictive models can only be partially determined by regulations such as "right to explanation." Instead, the restoration to interpretable models should fundamentally be driven by technology. It must be demonstrated that interpretable models can achieve performance comparable with black box models. That is what this work focuses on.

We will present two machine learning algorithms, called *Supersparse Linear Integer Models* (SLIM) and *Risk-Calibrated Supersparse Linear Integer Models* (RiskSLIM), which solve mixed-integer linear and nonlinear programs. They produce sparse linear models directly from data that are faithful to the century-old scoring-system model form, similar to the predictive models that have been used over the last century. SLIM produces scoring systems optimized for desired true positive / false positive tradeoffs, whereas RiskSLIM produces risk scores. Both methods leverage modern optimization tools and avoid well-known pitfalls of rounding methods. The models come with optimality guarantees, meaning that they allow one to test for the existence of interpretable models that are as accurate as black box models. RiskSLIM's models are risk-calibrated across the spectrum of true positives and false positives (or sensitivity and specificity), and both methods honor constraints imposed by the domain. Software for both methods is public, and could be used to challenge the use of black box models for high-stakes decisions.

SLIM and RiskSLIM are already challenging decision-making processes for applications in medicine and criminal justice. We will focus on three of them in this work. (i) *Sleep Apnea Screening*: In joint work with Massachusetts General Hospital (Ustun *et al.* 2016), we determined that a scoring system built using a patient's medical history can be as accurate as one that relies on reported symptoms. This yields savings in the efficiency and effectiveness of medical care for sleep apnea patients. (ii) *ICU Seizure Prediction*: In joint work with Massachusetts General Hospital (Struck *et al.* 2017), we created the first scoring system that uses continuous EEG measurements to predict seizures, called 2HELPS2B. The model provides concise reasons why a patient may be at risk. (iii) *Recidivism Prediction*: The recent public debate regarding recidivism prediction, and whether COMPAS' proprietary predictions are racially biased (Angwin *et al.* 2016) leads to the question of whether interpretable models exist for recidivism prediction. In our studies of recidivism (Zeng *et al.* 2017, Ustun & Rudin 2016a, 2017), we used the largest publicly available dataset on recidivism, and showed that SLIM and RiskSLIM could produce small scoring systems that are as accurate as state-of-the-art machine learning models. This calls into question the necessity of tools like COMPAS, and the reasons for government expenditures for predictions from proprietary models.

Scoring Systems: Applications and Prior Art

The use of predictive models is not new to society, only the use of black box models is relatively new. Scoring systems, which are a widely used form of interpretable predictive model, have dated back at least to work on parole violation by Burgess (1928). An example of a scoring system is the CHADS$_2$ score (Gage *et al.* 2001), shown in Figure 1, which predicts stroke in patients with atrial fibrillation, and is arguably the most widely used predictive model in medicine. Scoring systems are sparse linear models with small integer coefficients. The coefficients are the "point scores:" for CHADS$_2$, the coefficients are 1, 1, 1, 1 and 2.

The vast majority of predictive models in the healthcare system and justice system are scoring systems. Other examples from healthcare include: SAPS I, II and III (Le Gall *et al.* 1993, Moreno *et al.* 2005); APACHE I, II and III to assess ICU mortality risk (Knaus *et al.* 1981, 1985, 1991); TIMI to assess the risk of death and ischemic events (Antman

1.	*Congestive Heart Failure*	1 point		$\cdots$
2.	*Hypertension*	1 point	+	$\cdots$
3.	*Age* $\geq$ *75*	1 point	+	$\cdots$
4.	*Diabetes Mellitus*	1 point	+	$\cdots$
5.	*Prior Stroke or Transient Ischemic Attack*	2 points	+	$\cdots$
	ADD POINTS FROM ROWS 1–5	**SCORE**	=	$\cdots$

SCORE	0	1	2	3	4	5	6
STROKE RISK	1.9%	2.8%	4.0%	5.9%	8.5%	12.5%	18.2%

Figure 1 **CHADS$_2$ score to assess stroke risk (Gage et al. 2001). For each patient, the score is computed as the sum of the patients' points. The score is translated into the 1-year stroke risk using the lower table.**

et al. 2000), HEART (Six *et al.* 2008) and EDACS (Than *et al.* 2014) for cardiac events; PCL to screen for PTSD (Weathers *et al.* 2013), and SIRS to detect system inflammatory response syndrome (Bone *et al.* 1992). Examples from criminal justice include the Ohio Risk Assessment System (Latessa *et al.* 2009), the Kentucky Pretrial Risk Assessment Instrument (Austin *et al.* 2010), the Salient Factor Score (Hoffman & Adelberg 1980, Hoffman 1994), and the Criminal History Category (CHC) (U.S. Sentencing Commission 1987).

None of the scoring systems listed in the previous paragraphs were optimized for predictive performance on data. Each scoring system was created using a different method. Some of them were built using domain expertise alone (no data), and some were created using logistic regression followed by rounding of coefficients to obtain integer-valued point scores.

Serious problems with rounding heuristics are well documented in the optimization literature. When we solve a relaxed problem and round values to integers afterward, we know that (unless the problem has specific properties) either the solutions become infeasible or suboptimal. It is easy to find problems in discrete optimization textbooks where rounding leads to flawed solutions. In the case of linear regression or linear classification models, coefficients that are small are all rounded to zero, and thus an important part of the signal can easily be lost. We should not be using rounding heuristics if we want a reliable high quality solution, despite the government's recommendation (Gottfredson & Snyder 2005) to round logistic regression coefficients.

An additional set of challenges arises when models need to satisfy *operational constraints*, which are user-defined requirements for the model (e.g., false positive rate below 20%). It is extremely difficult to design rounding heuristics that produce accurate models that also obey operational constraints. Heuristics for model design lead to suboptimal models, which in turn could lead to poor decision-making for high-stakes applications.

Since its inception, the field of discrete optimization has been advancing, while all of the scoring systems have been built without using discrete optimization technology. Let us describe the optimization problems that we actually desire to solve when building scoring systems.

Optimization Problems and Methods

We will discuss two kinds of scoring systems:

1. Decision rules, which are scoring systems for decision-making, produced by SLIM. Here, predictions are based on whether the total score exceeds a threshold value (i.e., predict "yes" if total score > 1). The choice of variables and points in the score function is optimized for accuracy at a specific decision point (a specific true positive rate or false positive rate). The desired choice of true positive rate (TPR) or false positive rate (FPR) depends on the application. For medical screening, one might desire a larger false positive rate so that the test is more likely to falsely identify someone as positive for a disease than to dismiss someone who has the disease by giving them a negative test result. The user could specify the maximum false positive rate they are willing to tolerate, and SLIM will optimize the true positive rate subject to that constraint.

2. Risk scores, which are scoring systems for risk assessment, produced by RiskSLIM. These models use the score to output a risk estimate. The choice of variables and points in the score function is optimized for risk calibration. A scoring system is risk calibrated when the predicted risk of the outcome (from the model) matches the risk of outcome in the data. These models do not optimize a specific TPR/FPR tradeoff, rather they aim to achieve the highest true positive rate for each false positive rate.

We illustrate the difference between these two types of scoring systems in Figure 2, where we show SLIM and RiskSLIM models for predicting whether a prisoner will be arrested within three years of being released from prison. Both models were built using the largest publicly available dataset on recidivism and perform similarly to state-of-the-art machine learning models (as discussed in the applications section). The SLIM scoring system outputs a decision rule (predict "yes" if the total score exceed a threshold score), whereas the RiskSLIM scoring system outputs a table of risk estimates for each distinct score. In both cases, the choice of variables and the number of points are chosen to optimize the relevant performance metric by solving a discrete optimization problem.

SLIM solves one constrained optimization problem to produce decision rules, and RiskSLIM solves a different problem to produce risk scores. Solving these optimization problems directly is principled, obviates the need for rounding and other manipulation, and directly encodes what we desire in a scoring system. The optimization problems are described mathematically in the appendix. In particular:

- In both optimization problems (the decision rule optimization and risk score model optimization), hard constraints are used to force the coefficients to integer values.
- In both optimization problems, the objective we minimize includes a term that encourages the number of questions asked in the scoring system to be small (model sparsity).

SLIM scoring system

1.	Age at Release between 18 to 24	2 points	$\cdots$
2.	Prior Arrests $\geq$ 5	2 points	$+ \cdots$
3.	Prior Arrest for Misdemeanor	1 point	$+ \cdots$
4.	No Prior Arrests	-1 point	$+ \cdots$
5.	Age at Release $\geq$ 40	-1 point	$+ \cdots$
		SCORE	$= \cdots$

PREDICT ARREST FOR ANY OFFENSE IF SCORE $>$ 1

RiskSLIM risk score

1.	Prior Arrests $\geq$ 2	1 point	$\cdots$
2.	Prior Arrests $\geq$ 5	1 point	$+ \cdots$
3.	Prior Arrests for Local Ordinance	1 point	$+ \cdots$
4.	Age at Release between 18 to 24	1 point	$+ \cdots$
5.	Age at Release $\geq$ 40	-1 points	$+ \cdots$
		SCORE	$= \cdots$

SCORE	-1	0	1	2	3	4
RISK	11.9%	26.9%	50.0%	73.1%	88.1%	95.3%

Figure 2 Optimized scoring systems for recidivism prediction built using SLIM (top) and RiskSLIM (bottom). The outcome variable for both models is whether a prisoner is arrested within 3 years of release from prison. The SLIM scoring system outputs a predicted outcome. It has a test TPR/FPR of 76.6%/44.5%, and a mean 5-fold cross validation TPR/FPR of 78.3%/46.5%. The RiskSLIM scoring system outputs a risk estimate. It has a 5-fold cross validation mean test CAL/AUC of 1.7%/0.697 and training CAL/AUC of 2.6%/0.701. We provide a definition of these performance metrics in the Evaluation section. See Zeng et al. (2017), Ustun & Rudin (2016a) for more details.

- In the objective for SLIM, there is a term that encourages the point values to be small (e.g., it prefers value '1 point' rather than value '7 points'). This also encourages the point values to be co-prime, meaning they share no common prime factors. Thus, this formulation would never choose point scores '10, 10, 20, 10, 40', rather it would choose '1, 1, 2, 1, 4' to solve the same problem.
- In the formulation for RiskSLIM, the objective includes a term used in logistic regression (the *logistic loss*) that encourages the scores to be small and risk calibrated. As we define later, a model is risk calibrated when its predicted risks agree with risks calculated directly from the data.

Both optimization problems can accommodate constraints on the solution that are specific to the domain (operational constraints). Some types of constraints are in Table 1.

Constraint Type	**Example**
Feature Selection	Choose up to 10 features
Group Sparsity	Include either *Male* or *Female*, not both
Optimal Thresholding	Use at most 3 thresholds for *Age*, e.g., (Age$\leq$30, Age$\leq$50, Age$\leq$75).
Logical Structure	If *Male* is in model, then also include *Hypertension*
Probability	Predict $\Pr(y=+1\|\boldsymbol{x}) \geq 0.90$ when *Male* = TRUE
Fairness	Ensure that the predicted outcome $\hat{y}$ is +1 an equal number of times for *Male* and *Female*

Table 1 Examples of operational constraints that can be addressed. Both SLIM and RiskSLIM can handle constraints on model form. SLIM handles constraints related to error metrics (e.g., fairness constraints). RiskSLIM handles constraints on risk estimates (e.g., probability constraints, as in the second last row).

Both optimization problems are computationally hard, but theoretical results allow practical improvements in speed. As a result, both the decision rule optimization problem and the risk score optimization problem can be solved for reasonably large datasets in minutes.

The risk score problem is a mixed-integer non-linear program, because the logistic loss is nonlinear. However, since the logistic loss is convex, cutting planes would be a natural type of technique for this problem. Cutting plane techniques produce piecewise linear approximations to the objective (cuts), which produce a surrogate lower bound, labeled “cutting plane approximation” in the illustration in Figure 3. However, traditional cutting plane methods fail badly for the risk score problem. Since the feasible region is the integer lattice, a traditional cutting plane method would need to solve a mixed-integer program (MIP) to optimality to develop each new cut. If this surrogate MIP is not solved to optimality, we have no way of knowing when we have reached the solution to the risk score problem. After several iterations, enough cuts would accumulate that the mixed integer program

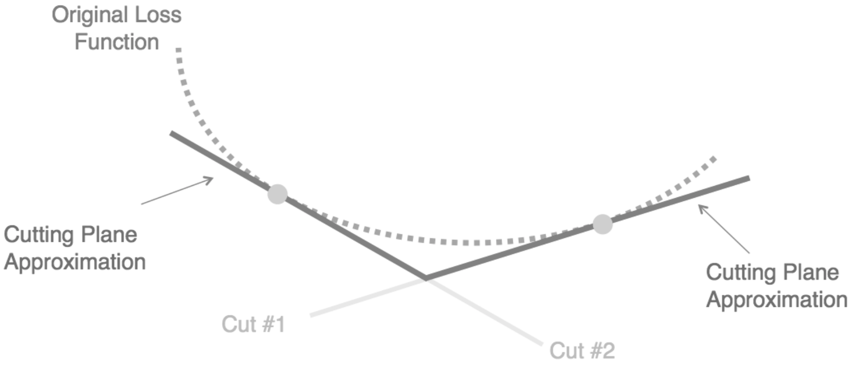


Figure 3 **A convex loss function (smooth curve) and its surrogate lower bound (lines).**

could not be solved to optimality in a reasonable amount of time, and the program would stall and fail to provide optimal scoring systems. This necessitates a new approach.

We developed a new branch-and-bound cutting plane method used in RiskSLIM for solving the risk score problem. This method does not stall, involves solving linear programs rather than mixed-integer programs, and can be implemented using standard callback functions in CPLEX (ILOG 2007). The method gracefully handles arbitrarily large datasets (even millions of observations), since computation scales linearly with the number of observations. The RiskSLIM model in Figure 2 was fit on a dataset with $N = 22,530$ observations in 20 minutes.

SLIM's decision rule problem (unlike the risk-score problem we just described for RiskSLIM) is a mixed-integer linear program. It can be solved with optimization software like CPLEX, but the solver is made more efficient with a specialized bound that we constructed, which reduces the amount of data we use without changing the solution to the optimization problem (discussed in Ustun & Rudin 2016b).

In the appendix, we discuss the optimization problems solved by SLIM and RiskSLIM. Before we discuss applications, let us discuss means of evaluation.

Evaluation Methodology for Machine Learning Models

The fields of machine learning and data mining use rigorous empirical evaluation techniques. *Cross validation* is commonly used to provide a measure of uncertainty of prediction quality. To perform 5 fold cross-validation, the data are divided into five equal size folds. Four of the folds are used to train the algorithm, and predictions are made out-of-sample on the fifth "test" fold. The test fold rotates, and we report a mean and standard deviation (or range) across folds.

In this work, we are interested in the following evaluation measures for classification problems: The *true positive rate* (TPR) is the fraction of positive test observations predicted to be positive. *Sensitivity* is also the true positive rate. *Specificity* is the true negative rate, the fraction of negative test observations predicted to be negative. The *false positive rate* (FPR) is the fraction of negative test observations predicted to be positive, and FPR is equal to one minus the specificity. The *Receiver Operator Characteristic (ROC) curve* is a plot of true positive rate for each possible value of the false positive rate. The *area under the ROC curve* (AUC) is important, since if the true positive rate is high for each value of the false positive rate, the algorithm has a high AUC and is performing well. An AUC value of .5 would be obtained for random guessing, an AUC of 1 is perfect, and for most of the problems we consider here, an AUC value of .8 would be considered excellent. AUC is a useful evaluation measure particularly when the positive and negative classes are imbalanced, meaning that only a small fraction of the data are positive (or negative). For instance, for the seizure prediction problem we discuss below, only 13.5% of observations in the seizure prediction data correspond to true seizures, while the rest were non-seizures.

For risk score prediction, we are also interested in *calibration* (CAL), which is a measure of how closely the predicted positive rate from the model matches the empirical positive rate in the data. We will discuss CAL later.

In general we find that when the form or size of the model is not constrained, then for the majority of applications, AUC values for all machine learning algorithms tend to be similar. AUC's start to differ when operational constraints are imposed. We will see this in more depth for the sleep apnea and seizure examples below.

Applications and Insights

Both SLIM and RiskSLIM have had an impact on several applications in healthcare and criminal justice. In what follows, we discuss three applications, and provide insight gained by producing interpretable models.

Sleep Apnea Screening

Obstructive Sleep Apnea (OSA) is a serious medical condition that can lead to morbidity and mortality, and can severely affect quality of life. A major goal of every sleep clinic is to screen patients for this disease correctly. Testing for OSA is problematic. Preliminary screening is mainly based on patient-reported symptoms and scoring systems. However, surprisingly, patient-reported symptoms are not particularly reliably reported, nor are they very useful for determining whether a patient has OSA. In particular, doctors often use the Epworth Sleepiness scale (Johns *et al.* 1991) or other scoring systems to screen for OSA, which are based on typical reported OSA symptoms like snoring, nocturnal gasping, witnessed apneas, sleepiness and other daytime complaints. Each of these predictive factors alone is weak; the comorbidities provided in medical records are much stronger. Hypertension, for instance, is a good predictor of OSA. Thus, it is reasonable that the staff of the Massachusetts General Hospital hypothesized that an accurate scoring system could be created that uses information from only routinely available medical records – without reported symptoms – that could be just as accurate as the widely used scoring systems.

The data provided for this study were records from all patients at the Massachusetts General Hospital Sleep Lab above 18 years old that underwent an definitive test for OSA called *polysomnography* (1,922 patients) between 2009 and 2013. Polysomnography is an expensive test for obstructive sleep apnea in which patients stay at the hospital overnight in order to collect information about brain activity, blood oxygen levels, heart rate, breathing patterns, eye movements and leg movements. Our goal was to predict OSA using only information that was available before the polysomnography. Such information included standard medical information (e.g. gender, age, BMI, past heart problems, hypertension, diabetes, smoking), as well as self-reported information on sleep patterns (e.g. caffeine consumption, insomnia, snoring, gasping, dry mouth in morning, leg jerks, falls back to sleep slowly). A full list of the features is provided in Table 1 of Ustun *et al.* (2016).

The domain experts also required several operational constraints on the form of the model, such as constraints on the size of the model, and the signs of the coefficients. The domain experts considered these constraints vital to their trust in the model.

If a scoring system could be developed that accurately screens patients for sleep apnea, using only the patient's medical records, without using the patient-reported symptoms, it would create an actionable tool that could allow automatic screening (as opposed to manual screening where a doctor would be involved). This type of automated scoring would allow wise usage of limited resources available for direct patient encounters.

To summarize, our domain experts (Brandon Westover and Matt Bianchi at Massachusetts General Hospital) had two important goals: (i) create an accurate transparent model for obstructive sleep apnea that obeyed operational constraints; (ii) determine the value of the patient-reported symptoms (e.g. gasping, insomnia, caffeine consumption) as compared with information that is already within the patient's medical record.

Prior to our work, the best previous scoring system for sleep apnea screening was arguably the STOP-BANG score (Chung *et al.* 2008). STOP-BANG relies on 8 features including self-reported snoring, tiredness, and breathing problems in addition to medical record information. Its sensitivity is 83.6% and specificity is 56.4%, which precludes it from being used as a screening tool. The specificity is the percentage of negatives identified correctly, meaning that the false positive rate is $100\% - 56.4\% = 43.6\%$, much higher than the goal on FPR that our domain experts were looking for, which was 20%.

SLIM **Model for Sleep Apnea Screening**

One of the models that our collaboration produced has sensitivity 61.4% and specificity 79.1%, so that the FPR was 20.9%. The scoring system was produced by SLIM, and is in Figure 4.

1.	Age $\geq$ 60	4 points	$\cdots$
2.	Hypertension	4 points	$+$ $\cdots$
3.	BMI $\geq$ 30	2 points	$+$ $\cdots$
4.	BMI $\geq$ 40	2 points	$+$ $\cdots$
5.	Female	-6 points	$+$ $\cdots$
		SCORE	$=$ $\cdots$

PREDICT OBSTRUCTIVE SLEEP APNEA IF SCORE $>$ 1

Figure 4 **SLIM scoring system for sleep apnea screening. This model achieves a 10-fold cross validation mean test TPR/FPR of 61.4/20.9%, and obeys all operational constraints. The model predicts OSA if the score exceeds 1. There are no common prime factors, since the threshold 1 is included in the set of factors; the coefficients are 1,4,4,2,2,-6, which are co-prime. See Ustun et al. (2016) for more details.**

Note that the model in Figure 4 does not contain patient-reported symptoms. After finding models like this, we wondered whether patient-reported symptoms were needed at all to achieve good prediction performance.

Patient-Reported Symptoms vs. Medical Record Information

Using any machine learning algorithm, it was easy to answer the second question of domain experts – that of measuring the importance of patient-reported symptoms. Patient-related symptoms are not nearly as important as medical history information. Across every machine learning method we tried, the models that used only patient-reported symptoms performed poorly, whereas models that used only medical record information performed almost as well (often as well) as the models that used both sets of information (see Table S2 in Ustun *et al.* 2016, for the AUC values of all machine learning methods we tried). To illustrate this, Figure 5 shows the ROC curves for models built using all features (dashed curve), patient-reported symptoms only (lower solid curve), and features that were extracted from an electronic health record (gray curve, overlapping the dashed curve). This figure shows that performance does not degrade when omitting the patient-reported variables all together.

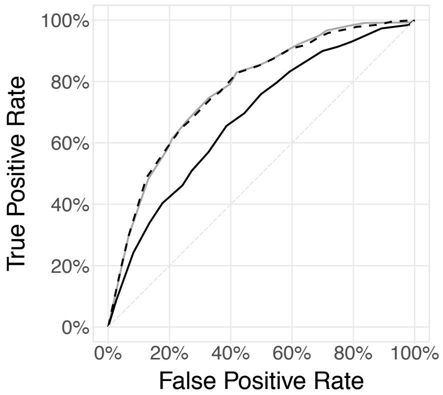


Figure 5 **Decision points of** SLIM **models over the full ROC curve for: (i) all features (gray, overlapping with dashed curve); (ii) features that can be extracted from an electronic health record (dashed); (iii) features related to patient-reported symptoms (black). See Ustun et al. (2016) for more details.**

To summarize: SLIM was able to find a model using medical-record information only – without the patient-reported symptoms – with prediction quality that is essentially identical to the models that use both types of information.

An Insight from the Apnea Study: Operational Constraints Are Challenging for Non-Mathematical-Programming-Based Machine Learning Algorithms

The experiment for the sleep apnea project revealed severe shortcomings for non-mathematical-programming-based machine learning methods, in that they are almost incapable of handling operational constraints.

Without considering operational constraints, our experiments indicated that SLIM's models have similar performance to other machine learning methods, such as support vector machines with radial basis function kernels (Ustun *et al.* 2016). The differences between methods arise when operational constraints are considered.

Our collaborators at Massachusetts General Hospital wanted a model fulfilling three simple operational constraints:

- *Max FPR*: Less than 20% false positive rate. Our goal was to correctly detect as many cases of OSA as possible, limiting the falsely detected cases to 20%.
- *Model Size*: Less than 5 terms in the model. Also small integer coefficients.
- *Sign Constraints*: Some point values needed to be constrained to be either positive or negative. For instance, it would not make sense to subtract points (predict lower risk of OSA) for patients that have hypertension, than for those who do not. This is because hypertension alone provides a significant risk for sleep apnea.

How would one obtain a model obeying these constraints with a standard machine learning algorithm that does not use mathematical programming? As it turns out, this is not trivial. For standard methods, the only degrees of freedom given to the experimenter are parameters that govern the shape of the model. These parameters can be tuned until the constraints are obeyed, but this proved to be challenging in practice. In particular, our results showed that for the standard machine learning methods, even if we searched extensively through parameter values, we can rarely find feasible models (model that satisfy all constraints). Table 2 shows the number of parameter values we chose using a grid search, which is recorded in the "Total Instances Trained" column, and the parameter values we chose are in the "Values for Free Parameters" column. For instance, we ran 975,000 instances of the standard machine learning algorithm called "Elastic Net." Despite the large number of instances we trained, Table 2 indicates that the grid search rarely produced models that satisfied the constraints. The decision tree methods we tried (CART, C5.0 rules, C5.0 trees) had the worst problems: they were unable to produce any models with FPR<20% despite tuning. This can be seen in the column under "Percent of total instances satisfying" labeled "MaxFPR." SVMs with linear kernels were unable to produce models with simultaneously less than 5 terms and FPR<20%, while ridge regression had the same problem. SVM with RBF kernels is nonparametric (meaning it adapts dynamically to the data), and is highly nonlinear and thus not interpretable. The only algorithms that could be tuned to accommodate the constraints were Elastic Net, Lasso, and SLIM. For SLIM, the constraints are directly incorporated into the solver, and every solution it produces is feasible.

Of the feasible models found from the standard machine learning methods, almost none are accurate predictive models. Figure 6 shows how Elastic Net, Lasso and SLIM perform as we vary the model size. Here, both Lasso and Elastic Net would need 8 variables to attain the accuracy of the 5-variable SLIM model.

What we have illustrated is a serious concern regarding the use of machine learning methods for practical problems: in almost all machine learning algorithms, user-defined constraints are not accommodated. Mathematical programming tools solve this issue.

Our work on sleep apnea was published in the Journal of Clinical Sleep Medicine (Ustun *et al.* 2016), which is the official journal of the American Academy of Sleep Medicine. More details can be found in SLIM's paper in the journal Machine Learning (Ustun & Rudin 2016b).

			Percent of Total Instances Satisfying		
Algorithm	**Values for Free Parameters**	**Total Instances Trained**	**Max FPR**	**Max FPR & Model Size**	**Max FPR, Model Size & Signs**
CART	39 values of $w^+ \in \{0.025, 0.05, \ldots, 0.975\}$	39	0.0%	0.0%	0.0%
C5.0R	39 values of $w^+ \in \{0.025, 0.05, \ldots, 0.975\}$	39	0.0%	0.0%	0.0%
C5.0T	39 values of $w^+ \in \{0.025, 0.05, \ldots, 0.975\}$	39	0.0%	0.0%	0.0%
Lasso	39 values of $w^+ \in \{0.025, 0.05, \ldots, 0.975\}$ × 1000 values of λ chosen by **glmnet**	39000	19.6%	4.8%	4.8%
Ridge	39 values of $w^+ \in \{0.025, 0.05, \ldots, 0.975\}$ × 1000 values of α chosen by **glmnet**	39000	20.9%	0.0%	0.0%
Elastic Net	39 values of $w^+ \in \{0.025, 0.05, \ldots, 0.975\}$ × 1000 values of λ chosen by **glmnet** × 19 values of $\alpha \in \{0.05, 0.10, \ldots, 0.95\}$	975000	18.3%	1.0%	1.0%
SVM Linear	39 values of $w^+ \in \{0.025, 0.05, \ldots, 0.975\}$ × 25 values of $C \in \{10^{-3}, 10^{-2.75}, \ldots, 10^3\}$	975	18.7%	0.0%	0.0%
SVM RBF	39 values of $w^+ \in \{0.025, 0.05, \ldots, 0.975\}$ × 25 values of $C \in \{10^{-3}, 10^{-2.75}, \ldots, 10^3\}$	975	15.8%	0.0%	0.0%
SLIM	$w^+ = n^-/(1+n^-)$, $C_0 = 0.9w^-/nd$, $\lambda_0 \in \{-100, \ldots, 100\}$, $\lambda_j \in \{-10, \ldots, 10\}$	1	100.0%	100.0%	100.0%

Table 2 Classification methods used for sleep apnea screening. We show the parameter settings, total number of instances trained, and the percentage of instances that fulfilled various combinations of operational constraints. Each instance is a unique combination of free parameters for a given method. The w^+ parameter is a unit misclassification cost for positive points. See Ustun et al. (2016), Ustun & Rudin (2016b) for more details.

Seizure Prediction in the ICU

Patients in the intensive care unit of a hospital who may be at risk for dangerous seizures are monitored using continuous electroencephalography cEEG, where electrodes monitor electrical signals in the brain. A clinician monitors the patient and identifies features in

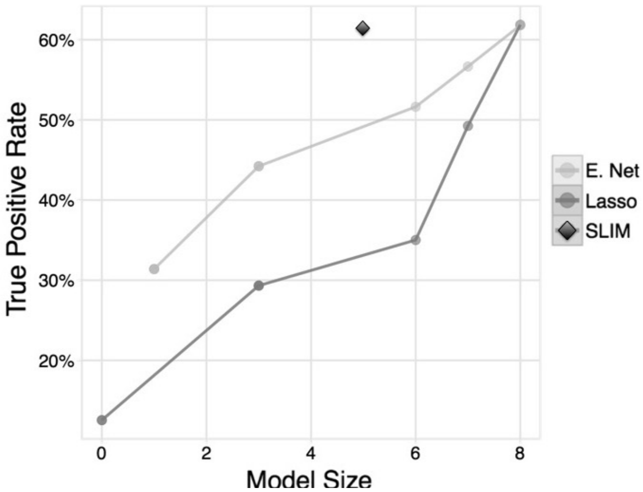


Figure 6 **Sensitivity and model size of Lasso and Elastic Net models that satisfy the sign and FPR constraints. For each method, we plot the instance that attains the highest 10-fold cross validation mean test TPR at model sizes between 0 and 8. Lasso and Elastic Net need at least 8 coefficients to produce a model with the same sensitivity as SLIM. See Ustun & Rudin (2016b) for details.**

the cEEG signal that may be predictive of seizure. The clinician may determine that the patient requires an intervention to prevent seizures, which could be dangerous, or (expensive) continued monitoring. Rather than have clinicians estimate seizure risk manually from cEEG signals, Massachusetts General Hospital staff aimed to assist clinicians by estimating this risk in a transparent way. We worked with a dataset from the Critical Care EEG Monitoring Research Consortium, collected at several hospitals (Emory University Hospital, Brigham and Womens Hospital, and Yale University Hospital) over the course of 3 years. The database contains 5,427 cEEG recordings with 87 variables, and each patient had at least 6 hours of uninterrupted cEEG monitoring. The variables from cEEG included important pattern types: lateralized periodic discharges (LPD); lateralized rhythmic delta (LRDA); generalized periodic discharges (GPD); generalized rhythmic delta (GRDA); bilateral periodic discharges (BiPD). Additionally, we had medical history and secondary symptoms for each patient. The outcome we aimed to predict was whether the patient would have a seizure within 24 hours. A transparent automated tool to help with seizure risk prediction would be particularly helpful in preventing false negatives: situations where clinicians mistakenly label the patient as being not-at-risk.

RiskSLIM **Model for Seizure Prediction**

In Figure 7 we show a model that we built using RiskSLIM. This model has a mean AUC over 5 cross-validation folds of 0.819 (with a range of 0.776-0.849 over the 5 folds). It is similar to other medical scoring systems in that it can be memorized by an acronym: the "**2H**" stands for: "GRDAs, LRDAs, BiPDs, LPDs, or GPDs with a frequency > 2 **H**z" (1 point), "**E**" stands for **E**pileptiform discharges (1 point), "**L**" stands for **L**PD or LRDA or BiPD (1 point), "**P**" stands for GRDAs, LRDAs, BiPDs, LPDs, or GPDs with plus features (superimposed rhythmic, fast, or sharp activity) (1 point); "**S**" is any history of **s**eizures (1 point), and "**2B**" is Brief Potentially Ictal Rhythmic Discharges (2 points).

1.	Any cEEG Pattern with Frequency **2 H**z	1 point	· · ·
2.	**E**pileptiform Discharges	1 point	+ · · ·
3.	Patterns include [**L**PD, LRDA, BIPD]	1 point	+ · · ·
4.	**P**atterns Superimposed with Fast or Sharp Activity	1 point	+ · · ·
5.	Prior **S**eizure	1 point	+ · · ·
6.	**B**rief Rhythmic Discharges	**2** points	+ · · ·
		SCORE	= · · ·

SCORE	0	1	2	3	4	5	6+
RISK	<5%	11.9%	26.9%	50.0%	73.1%	88.1%	95.3%

Figure 7 2HELPS2B scoring system, constructed by RiskSLIM **(reproduced from Struck et al. 2017).**

The 2HELP2B score has no predecessors; it is the first scoring system to be developed for cEEG monitoring for seizure prediction. It can be directly integrated into clinical workflow.

Our work on seizure prediction was published in JAMA Neurology (Struck *et al.* 2017). More details are in the RiskSLIM methodology paper (Ustun & Rudin 2017, 2016a).

Calibration was an important concern for our collaborators – models were deemed unacceptable if they were poorly calibrated. While constructing the 2HELP2B score, it became

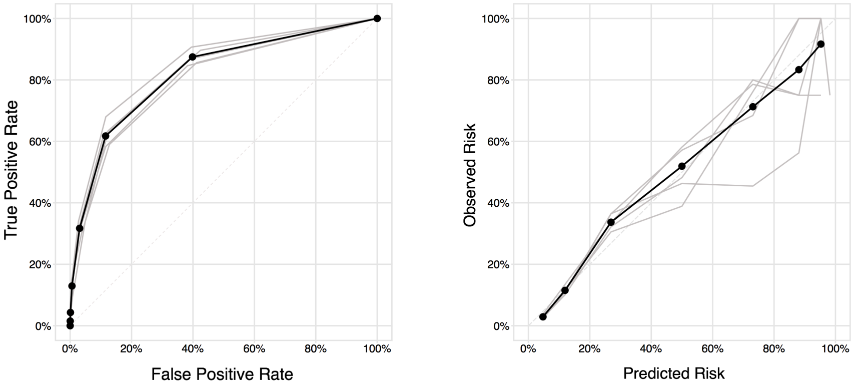


Figure 8 **ROC curves and calibration curves for the 2HELPS2B score produced by** RiskSLIM**.**

apparent that the typical methods one might use to construct scoring systems had systematic problems with calibration. This is our second insight, which we now discuss.

An Insight from the Seizure Study: Risk Calibration Suffers when we use Rounding to Compute Risk Scores

Risk calibration (CAL) measures how closely the estimated risks from the model match risks in the data. Risk calibration is essential for practical use in risk-scoring applications.

Let us define CAL precisely. The estimated risks for each individual i are calculated using the scoring system (e.g., from 2HELPS2B), and the risk for patient i from the model is denoted by p_i. Separately, for each possible value of the score s, we estimate the probability of the outcome $y=1$ given s from the data, that is, $p(s)=P(y=1|s)$. Then we compute the Euclidean distance between p_i and $p(s_i)$ across all patients i, and this is precisely CAL. A calibration plot is a plot of $p(s_i)$ vs p_i. If the plot is a diagonal line, the model is nicely calibrated.

RiskSLIM minimizes the logistic loss that is used for logistic regression. Logistic regression produces risk-calibrated models (Caruana & Niculescu-Mizil 2004, Zadrozny & Elkan 2002) but when rounding or other post-processing steps are done to a logistic regression model, it can drastically alter calibration. As discussed earlier, rounding sends all small coefficients to zero (which eliminates part of the signal), and rounding coefficients upwards makes variables more important than they should be in a calibrated model. An extensive set of experiments in the work of Ustun & Rudin (2017, 2016a) considered several types of rounding techniques. In particular, it considered naïve rounding (denoted Rd), which simply rounds coefficients to the nearest integer within the range $\{-5, -4, .., 0, ...4, 5\}$, and rescaled rounding (denoted RsRd), which scales all coefficients so that the largest one is ± 5, and then rounds to the nearest integer. Rescaled rounding tends to mitigate the problem of too many coefficients being rounded to zero.

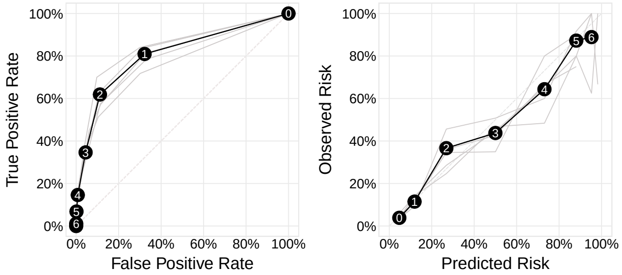
Calibration curves should always go upwards: as the score increases, the risk should always increase. However, this does not hold for either Rd or RsRd. Our collaborators determined that this was problematic since it is unreasonable that (for instance) a patient with a score of 3 has a higher risk of seizure than a patient with a score of 4. Figure 9 shows results from a controlled cross-validation experiment, including ROC curves and calibration curves for RiskSLIM and also for the Rd and RsRd methods. The black curves in the figures are from a model computed across the 5 cross-validation folds, and models in gray are from each of the 5 folds. The problems with calibration are apparent: the curves simply do not always increase. Here, RiskSLIM's 5-fold mean CAL was 2.5% (the best is 0%), whereas Rd's was 3.7% and RsRd's was 11.5%. 2HELPS2B was determined separately from the controlled experiment, and its ROC and calibration curves are in Figure 8. It has mean CAL over the 5 folds of 2.7%.

These experiments with rounding are not surprising – when we move in an arbitrary direction in a high dimensional space, we know from integer programming textbooks (Wolsey 1998) that there are problems with solution quality. Further, by using rounding, all guarantees of optimality are lost. This becomes problematic for applications like recidivism prediction, discussed next.

Optimized Risk Score (RiskSLIM)

1.	Brief Rhythmic Discharges	2 points	···
2.	Patterns Include LPD	2 points	+ ···
3.	Prior Seizure	1 point	+ ···
4.	Epileptiform Discharge	1 point	+ ···
		SCORE	= ···

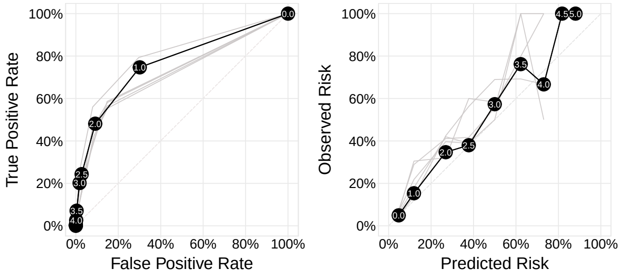
SCORE	0	1	2	3	4	5	6
RISK	4.7%	11.9%	26.9%	50.0%	73.1%	88.1%	95.3%



$\ell_1 + \ell_2$ Penalized Logistic Regression + Rounding

1.	Any Prior Seizure	1 point	···
2.	Patterns Include BiPD, LRDA, LPD	1 point	+ ···
3.	MaxFrequency LPD	× 1 point per Hz	+ ···
		SCORE	= ···

SCORE	0.0	1.0	2.0	2.5	3.0	3.5	4.0	4.5	5.0
RISK	4.7%	11.9%	26.9%	37.8%	50.0%	62.2%	73.1%	81.8%	88.1%



$\ell_1 + \ell_2$ Penalized Logistic Regression + Scaling + Rounding

1.	AnyPriorSeizure	5 points	···
2.	Patterns Include BiPD, LRDA, LPD	1 point	+ ···
3.	MaxFrequency LPD	× 5 points per Hz	+ ···
		SCORE	= ···

SCORE	0 to 10	12.5	15.0	20.0	20 to 25
RISK	$< 5.0\%$	7.6%	50.0%	92.4%	$> 95.0\%$

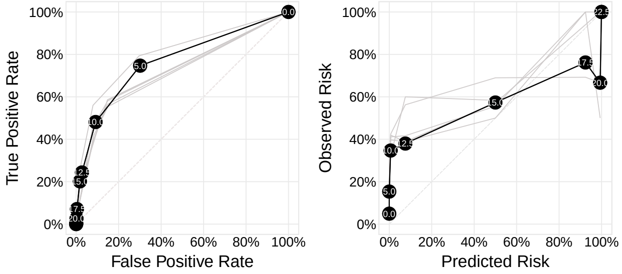


Figure 9 **Risk scores, ROC curves, and reliability diagrams for RiskSLIM and heuristic rounding techniques. We show the final model on training data in black, and fold-based models on test data in gray. This figure was reproduced from Ustun & Rudin (2017).**

Recidivism Prediction

In the U.S., criminal sentencing is done according to a mandated federal guideline (e.g., the Criminal History Category, U.S. Sentencing Commission 2004). One of the latest public guidelines for recidivism risk prediction in the U.S. is the Pennsylvania Commission on Sentencing (Pennsylvania Commission on Sentencing 2012), and other methods are used in Canada (Hanson & Thornton 2003), the Netherlands (Tollenaar & van der Heijden 2013), and the U.K. (Howard *et al.* 2009). There are a very large number of different risk scores for various applications, including sentencing, parole, and prison administration (see Zeng *et al.* 2017, for a longer list). These scores can be helpful: it is possible for a data-driven calculation to mitigate irregularities in decisions made by people. No human can keep a database in their head and accurately calculate recidivism risks. In fact, the decision-making process of judges can have high variance and rely on arbitrary factors. For instance, there is (debated) evidence that judges are much less likely to make a favorable ruling just before a lunch break (Kahneman 2011, Danziger *et al.* 2011). Worse than this, judges are not generally provided with feedback on the quality of their recidivism predictions, meaning they cannot learn from past mistakes.

Over the last few years, there has been an ongoing debate in the statistical community of criminologists. Some of them have claimed that traditional statistical methods are as accurate for predicting recidivism as modern machine learning tools, when the proper preprocessing has been done to create features (see e.g. Tollenaar & van der Heijden 2013, Berk & Bleich 2013, Bushway 2013). As we showed above, however, traditional statistical tools have serious flaws when paired with rounding methods, in terms of risk calibration and inability to incorporate operational constraints.

At the same time as this debate is happening, companies like Northpointe (now called Equivant) are selling predictions to the government, which are used widely. These risk scores have the potential to be racially biased, as argued by ProPublica (Angwin *et al.* 2016), though it is not easy to determine whether it is actually biased (see Fisher *et al.* 2018, who debate this using variable importance arguments). In 2016, in the case State v. Loomis, the Wisconsin Supreme Court ruled that black box risk scores like Northpointe's COMPAS can be used by judges, but minimized the role that such scoring systems could play as evidence. An appeal was filed at the U.S. Supreme Court, who declined to hear the case in June 2017.

The goal of our project was to determine whether such black box scoring systems were needed at all for recidivism prediction. If we find a transparent model with the same accuracy as the best black box model, we no longer require the black box model.

We used the largest publicly available dataset on recidivism, which is the "Recidivism of Prisoners Released in 1994" dataset collected by the U.S. Department of Justice, Bureau of Justice Statistics (U.S. Department of Justice, Bureau of Justice Statistics 2014). This dataset contains information that we used from 33,796 prisoners, including criminal history from record-of-arrest-and-prosecution (RAP) sheets, along with demographic factors such as gender and age. We omitted socioeconomic factors such as race for the main study, but conducted experiments using race afterwards (see Zeng *et al.* 2017). The outcomes we aimed to predict within three years of release were: (1) arrest for any crime, (2) arrest for drug-related crime, (3) arrest for violent crime (general_violence), (4) arrest for a domestic violence crime, (5) arrest for a sexual violence crime, and (6) arrest for a crime involving fatal violence.

Results for Recidivism Prediction

Our results were consistent with those from other applications, in that most machine learning algorithms performed almost identically across the full ROC curve, for all of the six prediction problems, as shown in Figure 10. The decision tree methods (CART, C5.0T, C5.0R – in green in Figure 10) sometimes performed poorly, particularly for imbalanced problems. This could potentially illustrate the reason why people often believe that an interpretable modeling algorithm does not perform as accurately as a black box method – methods that produce interpretable models like CART are indeed not as accurate as other methods. CART (Breiman *et al.* 1984) is not based on optimization, and was designed to operate within the limits of computers from 1984. CART's poor performance is not a convincing reason as to why all interpretable modeling methods might perform poorly.

Our work on this problem was published in the Journal of the Royal Statistical Society (Zeng *et al.* 2017). Figure 2 shows two of the models we produced using SLIM and RiskSLIM.

The basic findings (that interpretable models are as accurate as black box models) was confirmed by Angelino *et al.* (2017) using another publicly available dataset, namely the Propublica Broward County data; in this case, small logical models were shown to be as accurate as the COMPAS score for predicting 2 year recidivism.

Insight for Recidivism Prediction: Importance of Certifiable Optimality

Methods like SLIM and RiskSLIM produce certificates of optimality, or they provide distance to optimality (optimality gaps) in the case where the problems are not fully solved to optimality. These types of guarantees are useful for answering questions such as: "*Does there exist an interpretable model (of a given form) that achieves a particular value for predictive performance on the dataset?*"

While it is true that optimizing performance on the training set does not correspond exactly to performance on the test set, training and test performances are guaranteed to

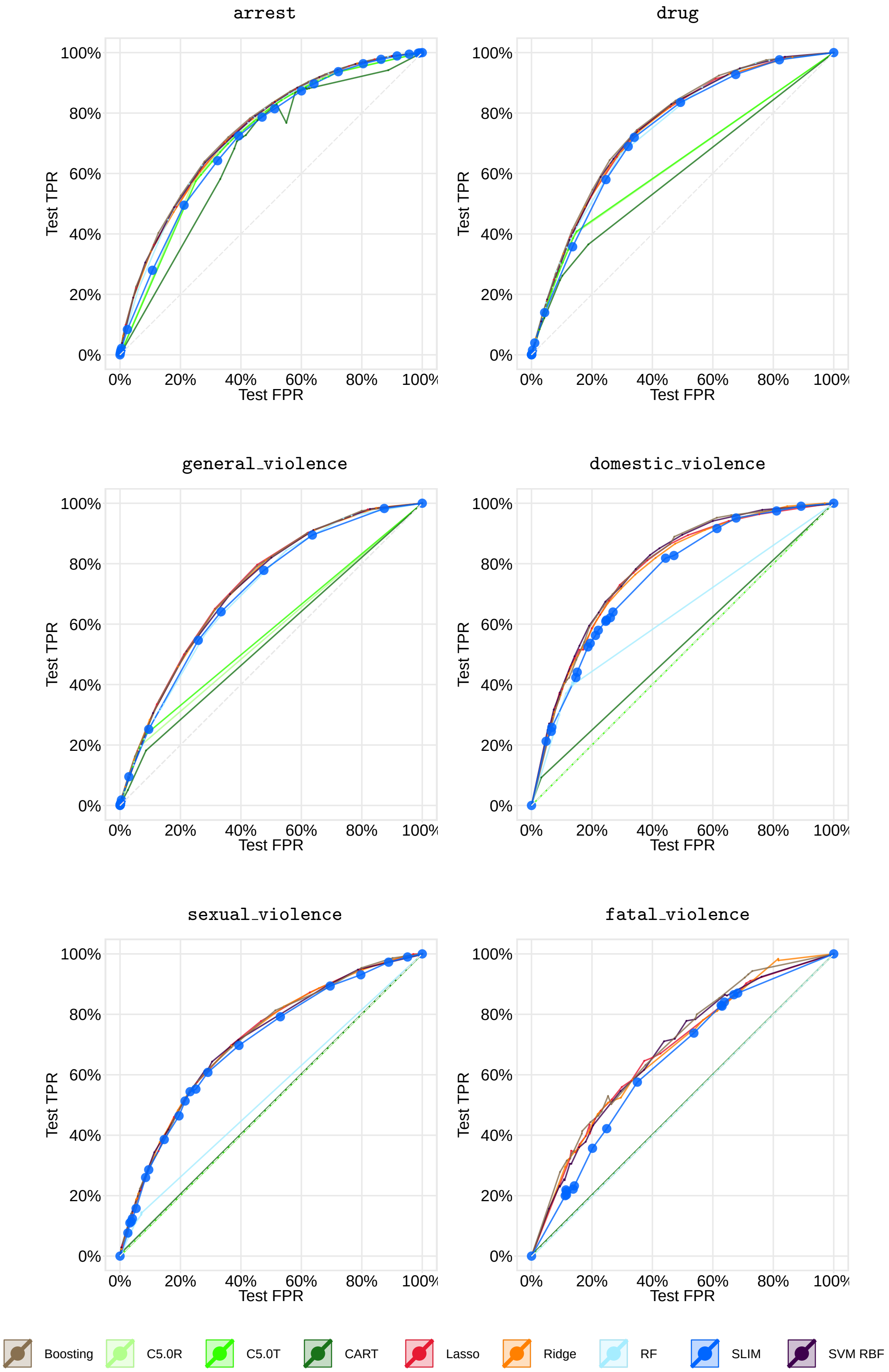


Figure 10 ROC curves for recidivism prediction problems. TPR/FPR for SLIM models are plotted using large blue dots. All models perform similarly except those from C5.0R, C5.0T, and CART. This figure was reproduced from Zeng et al. (2017)

be similar by statistical learning theory. In fact, if a method cannot achieve high quality in-sample performance, it is difficult for it to achieve high quality out-of-sample performance.

This work provides tools that can determine whether an interpretable model exists that performs well on a given dataset. If an accurate, interpretable model does exist (which it does in many cases), we should use it rather than resorting to a black box, particularly for high-stakes decisions such as bail, parole, and sentencing.

Other Applications

SLIM and RiskSLIM have been used for purposes besides those discussed above. SLIM has been used to detect cognitive impairment, such as Alzheimer's disease, dementia and Parkinson's disease. In particular, the Clock Drawing Test, which is a pen-and-paper test that has been used for a century to diagnose these disorders, has been updated to be digitized. Patients draw clocks with a digital pen, and this digitized test is automatically scored with a SLIM-based system (Souillard-Mandar *et al.* 2016). The new scoring system far surpasses the accuracy of all previously published scoring systems for the Clock Drawing Test, and is a promising non-invasive technique for early identification of cognitive impairment. Our work on this project, in conjunction with several collaborators, was published in the Machine Learning journal, and won the 2016 INFORMS Innovative Applications in Analytics Award.

In a separate project using RiskSLIM, we created a screening scale for adult ADHD (attention deficit hyperactivity disorder) in collaboration with a team of psychiatrists (Ustun *et al.* 2017). The test allows for a quick, risk-calibrated diagnosis based on the answers to 6 questions on a self-reported questionnaire. The questions include: "How often do you have difficulty concentrating on what people say to you, even when they are speaking to you directly?" and "How often do you leave your seat in meetings and other situations in which you are expected to remain seated?" The prediction performance was optimized based on clinical diagnoses using DSM-5 criteria, which is the new standard for adult ADHD diagnosis. The work was published in JAMA Psychiatry in May 2017, and has 12,502 views as of January 9, 2018.

SLIM and RiskSLIM are optimization-based approaches. A Bayesian approach to forming scoring systems is that of Ertekin & Rudin (2015).

It is important to note that scoring systems are not the only forms of interpretable models. Logical models, such as decision trees and decision lists, have existed since the beginning of artificial intelligence. Recent work on those models have been useful for recidivism prediction (Angelino *et al.* 2017, Yang *et al.* 2017), credit scoring (Chen & Rudin 2018), hospital readmission (Wang & Rudin 2015), and stroke prediction in atrial fibrillation patients (Letham *et al.* 2015). Logical models (in particular "and's of or's") are useful for modeling consideration sets used in marketing (Wang *et al.* 2017, 2016, Goh & Rudin 2014). Logical models with operational constraints can also be constructed with specialized optimization techniques (e.g., varieties of monotonicity constraints, see Wang & Rudin 2015, Chen & Rudin 2018).

Looking Forward

Within the foreseeable future, there will be a business need to keep the details of machine learning models as a trade secret. In some domains this may not be problematic, particularly when decisions have a minor effect on people's lives. In other domains, such as healthcare and criminal justice, decisions are serious and actions need to be defensible. The machine learning algorithms presented here represent a fundamental change to the way transparent models are constructed, leveraging modern discrete optimization techniques (cutting planes, data reduction bounds, mixed-integer programming) and capabilities (callback functions, modern solvers). Code for SLIM and RiskSLIM is publicly available, at http://github.com/ustunb/slim-python and http://github.com/ustunb/risk-slim.

References

Angelino, Elaine, Larus-Stone, Nicholas, Alabi, Daniel, Seltzer, Margo, & Rudin, Cynthia. 2017. Learning certifiably optimal rule lists for categorical data. *In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.*

Angwin, Julia, Larson, Jeff, Mattu, Surya, & Kirchner, Lauren. 2016. *Machine Bias.* https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing.

Antman, Elliott M, Cohen, Marc, Bernink, Peter JLM, McCabe, Carolyn H, Horacek, Thomas, Papuchis, Gary, Mautner, Branco, Corbalan, Ramon, Radley, David, & Braunwald, Eugene. 2000. The TIMI risk score for unstable angina/non–ST elevation MI. *The Journal of the American Medical Association*, **284**(7), 835–842.

Austin, James, Ocker, Roger, & Bhati, Avi. 2010. Kentucky pretrial risk assessment instrument validation. *Bureau of Justice Statistics. Grant.*

Berk, Richard A, & Bleich, Justin. 2013. Statistical Procedures for Forecasting Criminal Behavior. *Criminology & Public Policy*, **12**(3), 513–544.

Bone, RC, Balk, RA, Cerra, FB, Dellinger, RP, Fein, AM, Knaus, WA, Schein, RM, Sibbald, WJ, Abrams, JH, Bernard, GR, *et al.* . 1992. American College of Chest Physicians/Society of Critical Care Medicine Consensus Conference: Definitions for sepsis and organ failure and guidelines for the use of innovative therapies in sepsis. *Critical Care Medicine*, **20**(6), 864–874.

Breiman, Leo. 2001. Statistical modeling: The two cultures. *Statistical Science*, **16**(3), 199–231.

Breiman, Leo, Friedman, Jerome, Stone, Charles J, & Olshen, Richard A. 1984. *Classification and regression trees.* CRC press, Boca Raton, Florida.

Burgess, Ernest W. 1928. Factors determining success or failure on parole. *The workings of the indeterminate sentence law and the parole system in Illinois*, 221–234.

Bushway, Shawn D. 2013. Is There Any Logic to Using Logit. *Criminology & Public Policy*, **12**(3), 563–567.

Caruana, Rich, & Niculescu-Mizil, Alexandru. 2004. Data mining in metric space: an empirical analysis of supervised learning performance criteria. *Pages 69–78 of: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining.* ACM.

Chen, Chaofan, & Rudin, Cynthia. 2018. An optimization approach to learning falling rule lists. *In: Artificial Intelligence and Statistics (AISTATS).*

Chung, Frances, Yegneswaran, Balaji, Liao, Pu, Chung, Sharon A., Vairavanathan, Santhira, Islam, Sazzadul, Khajehdehi, Ali, & Shapiro, Colin M. 2008. STOP Questionnaire: a tool to screen patients for obstructive sleep apnea. *Anesthesiology*, **108**, 812–821.

Citron, Danielle. 2016. (Un)Fairness of Risk Scores in Criminal Sentencing. *Forbes, Tech section*, July.

Danziger, Shai, Levav, Jonathan, & Avnaim-Pesso, Liora. 2011. Extraneous factors in judicial decisions. *Proceedings of the National Academy of Sciences*, **108**(17), 6889–6892.

Dawes, Robyn M. 1979. The robust beauty of improper linear models in decision making. *American psychologist*, **34**(7), 571–582.

Ertekin, Şeyda, & Rudin, Cynthia. 2015. A Bayesian Approach to Learning Scoring Systems. *Big Data*, **3**(4), 267–276.

Fisher, Aaron, Rudin, Cynthia, & Dominici, Francesca. 2018. Model Class Reliance: Variable Importance Measures for any Machine Learning Model Class, from the "Rashomon" Perspective. *arXiv:1801.01489 [stat.ME]*.

Freitas, Alex A. 2014. Comprehensible classification models: a position paper. *ACM SIGKDD Explorations Newsletter*, **15**(1), 1–10.

Gage, Brian F, Waterman, Amy D, Shannon, William, Boechler, Michael, Rich, Michael W, & Radford, Martha J. 2001. Validation of clinical classification schemes for predicting stroke. *The Journal of the American Medical Association*, **285**(22), 2864–2870.

Goh, Siong Thye, & Rudin, Cynthia. 2014. Box Drawings for Learning with Imbalanced Data. *In: Proceedings of the 20th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*.

Goodman, Bryce, & Flaxman, Seth. 2016. EU regulations on algorithmic decision-making and a "right to explanation". *arXiv:1606.08813 [stat.ML]*. Presented at 2016 ICML Workshop on Human Interpretability in Machine Learning (WHI 2016).

Gottfredson, Don M., & Snyder, Howard N. 2005. *The Mathematics of Risk Classification: Changing Data into Valid Instruments for Juvenile Courts*. Tech. rept. NCJ 209158. Department of Justice, Washington, D.C. Office of Juvenile Justice and Delinquency Prevention.

Hanson, RK, & Thornton, D. 2003. Notes on the development of Static-2002. *Ottawa, Ontario: Department of the Solicitor General of Canada*.

Ho, Vivian. 2017. Miscalculated score said to be behind release of alleged Twin Peaks killer. *SFGate (San Francisco Chronicle)*, August.

Hoffman, Peter B. 1994. Twenty years of operational use of a risk prediction instrument: The United States Parole Commission's Salient Factor Score. *Journal of Criminal Justice*, **22**(6), 477–494.

Hoffman, Peter B, & Adelberg, Sheldon. 1980. The Salient Factor Score: A Nontechnical Overview. *Fed. Probation*, **44**, 44–52.

Holte, Robert C. 1993. Very simple classification rules perform well on most commonly used datasets. *Machine learning*, **11**(1), 63–90.

Howard, Philip, Francis, Brian, Soothill, Keith, & Humphreys, Leslie. 2009. *OGRS 3: The revised offender group reconviction scale.* Tech. rept. Ministry of Justice London, UK.

ILOG. 2007. *CPLEX 11.0 User's Manual.* ILOG, Inc.

Johns, Murray W, *et al.* . 1991. A new method for measuring daytime sleepiness: the Epworth sleepiness scale. *Sleep*, **14**(6), 540–545.

Kahneman, Daniel. 2011. *Thinking, fast and slow.* Macmillan.

Knaus, William A, Zimmerman, Jack E, Wagner, Douglas P, Draper, Elizabeth A, & Lawrence, Diane E. 1981. APACHE-acute physiology and chronic health evaluation: a physiologically based classification system. *Critical Care Medicine*, **9**(8), 591–597.

Knaus, William A, Draper, Elizabeth A, Wagner, Douglas P, & Zimmerman, Jack E. 1985. APACHE II: a severity of disease classification system. *Critical Care Medicine*, **13**(10), 818–829.

Knaus, William A, Wagner, DP, Draper, EA, Zimmerman, JE, Bergner, Marilyn, Bastos, PG, Sirio, CA, Murphy, DJ, Lotring, T, & Damiano, A. 1991. The APACHE III prognostic system. Risk prediction of hospital mortality for critically ill hospitalized adults. *Chest Journal*, **100**(6), 1619–1636.

Kodratoff, Y. 1994. The comprehensibility manifesto. *KDD Nugget Newsletter*, **94**(9).

Latessa, Edward, Smith, Paula, Lemke, Richard, Makarios, Matthew, & Lowenkamp, Christopher. 2009. Creation and validation of the Ohio risk assessment system: Final report. *Center for Criminal Justice Research, School of Criminal Justice, University of Cincinnati, Cincinnati, OH. Retrieved from http://www. ocjs. ohio. gov/ORAS_FinalReport. pdf.*

Le Gall, Jean-Roger, Lemeshow, Stanley, & Saulnier, Fabienne. 1993. A new simplified acute physiology score (SAPS II) based on a European/North American multicenter study. *The Journal of the American Medical Association*, **270**(24), 2957–2963.

Letham, Benjamin, Rudin, Cynthia, McCormick, Tyler H., & Madigan, David. 2015. Interpretable classifiers using rules and Bayesian analysis: Building a better stroke prediction model. *Annals of Applied Statistics*, **9**(3), 1350–1371.

Moreno, Rui P, Metnitz, Philipp GH, Almeida, Eduardo, Jordan, Barbara, Bauer, Peter, Campos, Ricardo Abizanda, Iapichino, Gaetano, Edbrooke, David, Capuzzo, Maurizia, & Le Gall, Jean-Roger. 2005. SAPS 3 - From evaluation of the patient to evaluation of the intensive care unit. Part 2: Development of a prognostic model for hospital mortality at ICU admission. *Intensive Care Medicine*, **31**(10), 1345–1355.

Northpointe. 2015. *Correctional Offender Management Profiling for Alternative Sanctions (COMPAS).* http://www.northpointeinc.com/downloads/compas/Practitioners-Guide-COMPAS-Core-_031915.pdf.

Pazzani, Michael J. 2000. Knowledge discovery from data? *Intelligent systems and their applications, IEEE*, **15**(2), 10–12.

Pekkala, Timoa, Hall, Anettea, Lötjönen, Jyrkib, Mattila, Jussic, Soininen, Hilkkaa, Ngandu, Tiiae, Laatikainen, Tiinae, Kivipelto, Miiaa, & Solomon, Alina. 2017. Development of a Late-Life Dementia Prediction Index with Supervised Machine Learning in the Population-Based CAIDE Study. *Journal of Alzheimer's Disease*, **55**, 1055–1067.

Pennsylvania Commission on Sentencing. 2012 (June). *Interim Report 4: Development of Risk Assessment Scale*. Tech. rept.

Six, AJ, Backus, BE, & Kelder, JC. 2008. Chest pain in the emergency room: value of the HEART score. *Netherlands Heart Journal*, **16**(6), 191–196.

Souillard-Mandar, William, Davis, Randall, Rudin, Cynthia, Au, Rhoda, Libon, David J., Swenson, Rodney, Price, Catherine C., Lamar, Melissa, & Penney, Dana L. 2016. Learning Classification Models of Cognitive Conditions from Subtle Behaviors in the Digital Clock Drawing Test. *Machine Learning*, **102**(3), 393–441.

Struck, Aaron F., Ustun, Berk, Rodriguez Ruiz, Andres, Lee, Jong Woo, LaRoche, Suzette, Hirsch, Lawrence J., Gilmore, Emily J., Rudin, Cynthia, & Westover, Brandon M. 2017. A Practical Risk Score for EEG Seizures in Hospitalized Patients. *JAMA Neurology*, **74**(12).

Than, Martin, Flaws, Dylan, Sanders, Sharon, Doust, Jenny, Glasziou, Paul, Kline, Jeffery, Aldous, Sally, Troughton, Richard, Reid, Christopher, Parsonage, William A, Frampton, Christopher, Greenslade, Jaimi H, Deely, Joanne M, Hess, Erik, Sadiq, Amr Bin, Singleton, Rose, Shopland, Rosie, Vercoe, Laura, Woolhouse-Williams, Morgana, Ardagh, Michael, Bossuyt, Patrick, Bannister, Laura, & Cullen, Louise. 2014. Development and validation of the Emergency Department Assessment of Chest pain Score and 2 h accelerated diagnostic protocol. *Emergency Medicine Australasia*, **26**(1), 34–44.

Tollenaar, Nikolaj, & van der Heijden, P.G.M. 2013. Which method predicts recidivism best?: a comparison of statistical, machine learning and data mining predictive models. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, **176**(2), 565–584.

U.S. Department of Justice, Bureau of Justice Statistics. 2014. *Recidivism of Prisoners Released in 1994*. http://doi.org/10.3886/ICPSR03355.v8.

U.S. Sentencing Commission. 1987 (November). *2012 Guidelines Manual: Chapter Four - Criminal History and Criminal Livelihood*.

U.S. Sentencing Commission. 2004. Measuring Recidivism: The Criminal History Computation of the Federal Sentencing Guidelines.

Ustun, Berk, & Rudin, Cynthia. 2016a. Learning Optimized Risk Scores for Large-Scale Datasets. *arXiv:1610.00168.*

Ustun, Berk, & Rudin, Cynthia. 2016b. Supersparse Linear Integer Models for Optimized Medical Scoring Systems. *Machine Learning*, **102**(3), 349–391.

Ustun, Berk, & Rudin, Cynthia. 2017. Optimized Risk Scores. *In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.*

Ustun, Berk, Westover, M. Brandon, Rudin, Cynthia, & Bianchi, Matt T. 2016. Clinical Prediction Models for Sleep Apnea: The Importance of Medical History over Symptoms. *Journal of Clinical Sleep Medicine*, **12**(2), 161–168.

Ustun, Berk, Adler, Lenard A, Rudin, Cynthia, Faraone, Stephen V, Spencer, Thomas J, Berglund, Patricia, Gruber, Michael J, & Kessler, Ronald C. 2017. The World Health Organization Adult Attention-Deficit/Hyperactivity Disorder Self-Report Screening Scale for DSM-5. *JAMA Psychiatry*, **74**(5), 520–526.

Wang, Fulton, & Rudin, Cynthia. 2015. Falling Rule Lists. *In: Proceedings of Artificial Intelligence and Statistics (AISTATS).*

Wang, Tong, Rudin, Cynthia, Doshi, Finale, Liu, Yimin, Klampfl, Erica, & MacNeille, Perry. 2016. Bayesian Or's of And's for Interpretable Classification with Application to Context Aware Recommender Systems. *In: International Conference on Data Mining (ICDM).*

Wang, Tong, Rudin, Cynthia, Doshi-Velez, Finale, Liu, Yimin, Klampfl, Erica, & MacNeille, Perry. 2017. A Bayesian Framework for Learning Rule Sets for Interpretable Classification. *Journal of Machine Learning Research*, **18**(70), 1–37.

Weathers, Frank W, Litz, Brett T, Keane, Terence M, Palmieri, Patrick A, Marx, Brian P, & Schnurr, Paula P. 2013. The PTSD Checklist for DSM-5 (PCL-5). *Scale available from the National Center for PTSD at www.ptsd.va.gov.*

Weng, Stephen F., Reps, Jenna, Kai, Joe, Garibaldi, Jonathan M., & Qureshi, Nadeem. 2017. Can machine-learning improve cardiovascular risk prediction using routine clinical data? *PLoS ONE*, April.

Wexler, Rebecca. 2017a. Code of Silence: How private companies hide flaws in the software that governments use to decide who goes to prison and who gets out. *Washington Monthly*, June/July/August.

Wexler, Rebecca. 2017b. When a Computer Program Keeps You in Jail: How Computers are Harming Criminal Justice. *New York Times*, June.

Wolsey, Laurence A. 1998. *Integer programming.* Vol. 42. Wiley New York.

Yang, Hongyu, Rudin, Cynthia, & Seltzer, Margo. 2017. Scalable Bayesian Rule Lists. *In: Proceedings of the 34th International Conference on Machine Learning (ICML).* Preprint at arXiv:1602.08610.

Zadrozny, Bianca, & Elkan, Charles. 2002. Transforming classifier scores into accurate multiclass probability estimates. *Pages 694–699 of: Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM.

Zeng, Jiaming, Ustun, Berk, & Rudin, Cynthia. 2017. Interpretable classification models for recidivism prediction. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, **180**(3), 689–722.

Appendix: Optimization Problems

We start with a dataset of N i.i.d. training examples $\mathcal{D} = \{(\boldsymbol{x}_i, y_i)_{i=1}^N\}$ where $\boldsymbol{x}_i \in \mathcal{X} \subseteq \mathbb{R}^{d+1}$ denotes a vector of features $[1, x_{i,1}, \ldots, x_{i,d}]^\top$ and $y_i \in \mathcal{Y} = \{-1, 1\}$ denotes a class label. We consider linear classification models of the form $\hat{y} = \text{sign}(\langle \boldsymbol{\lambda}, \boldsymbol{x} \rangle)$, where $\boldsymbol{\lambda} = [\lambda_0, \lambda_1, \ldots, \lambda_d]^\top$ represents a vector of coefficients and λ_0 represents an intercept.

In this setup, the coefficient vector $\boldsymbol{\lambda}$ determines all parameters of a scoring system. In particular, the coefficient λ_j represents the *points* for feature j for $j = 1, \ldots, d$. Given an example with features $\boldsymbol{x}_i$, users first tally the points for all features such that $\lambda_j \neq 0$ to obtain a total *score* $\sum_{j=1}^d \lambda_j x_{i,j}$ then use the total score to obtain a predicted label (i.e. for decision-making) or a estimate of predicted risk (i.e. for risk assessment).

SLIM's **Optimization Framework for Decision-Making**

In decision-making applications, we use the score to output a predicted label $\hat{y} \in \{-1, 1\}$ through a decision rule of the form:

$$\hat{y}_i = \begin{cases} +1 & \text{if } \displaystyle\sum_{j=1}^d \lambda_j x_{i,j} + \lambda_0 > 0, \\ -1 & \text{if } \displaystyle\sum_{j=1}^d \lambda_j x_{i,j} + \lambda_0 \leq 0. \end{cases} \tag{1}$$

In this setting, we learn the values of coefficients by solving a discrete optimization problem that we refer to as the *decision rule problem.* The optimal solution to the decision rule problem is a *Supersparse Linear Integer Model.* The decision rule problem is a discrete optimization problem of the form:

$$\begin{aligned} \min_{\boldsymbol{\lambda}} \quad & l_{01}(\boldsymbol{\lambda}) + C_0 \left\|\boldsymbol{\lambda}\right\|_0 \\ \text{s.t.} \quad & \boldsymbol{\lambda} \in \mathcal{L}, \\ & \text{gcd}(\boldsymbol{\lambda}) = 1, \end{aligned} \tag{2}$$

where:

- $l_{01}(\boldsymbol{\lambda}) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}\left[\hat{y}_i \neq y_i\right]$ is the fraction of misclassified observations;
- $\|\boldsymbol{\lambda}\|_0 = \sum_{j=1}^{d} \mathbb{1}\left[\lambda_j \neq 0\right]$ is the count of non-zero coefficients, ℓ_0-seminorm;
- $\mathcal{L} \subset \mathbb{Z}^{d+1}$ is a finite user-provided set of feasible coefficient vectors, usually chosen to be small integers, $\mathcal{L} = \{-10, \ldots, 10\}^{d+1}$;
- $C_0 > 0$ is a user-chosen trade-off parameter to balance accuracy and sparsity;
- $\gcd(\boldsymbol{\lambda}) = 1$ is a symmetry-breaking constraint to ensure coefficients are co-prime. Here "gcd" stands for greatest common divisor.

Here, the objective minimizes the empirical probability of misclassification, and penalizes the number of non-zero terms to encourage the model to be sparse. The feasible region can be customized to include additional operational constraints (see Table 1).

To implement the decision rule problem as a mathematical program, there is a simple trick for encoding the constraint that the gcd of the coefficients is 1. In particular, if we add a term to the objective that is the sum of the absolute coefficients, multiplied by a very small number (ϵ in the formulation below), it forces the gcd to be 1 without influencing either accuracy or sparsity. The reason this trick works is because the loss and sparsity terms take on only discrete values. Among all models that are equally accurate and equally sparse, the formulation will choose the one with the smallest absolute sum of terms, $\sum_j |\lambda_j|$, also written $\|\boldsymbol{\lambda}\|_1$. Since the values of the λ_j are also integers, they must be co-prime.

In practice, the fraction of misclassifications in the objective is replaced with a weighted sum of false positives and false negatives, for applications where the user has determined that one of these is more important to reduce than the other.

Incorporating the separate weights for false positives and false negatives (w^- and w^+), and using the additional term in the objective to force the gcd to 1, the optimization problem is as follows.

$$\begin{aligned} \min_{\boldsymbol{\lambda}} \quad & \frac{w^+}{N_+}\sum_{i:y_i=1}\mathbb{1}\left[\hat{y}_i \neq -1\right] + \frac{w^-}{N_-}\sum_{i:y_i=-1}\mathbb{1}\left[\hat{y} \neq -1\right] + C_0\left\|\boldsymbol{\lambda}\right\|_0 + \epsilon\|\boldsymbol{\lambda}\|_1 \\ \text{s.t.} \quad & \boldsymbol{\lambda} \in \mathcal{L}, \end{aligned} \tag{3}$$

where $\hat{y}$ depends on $\boldsymbol{\lambda}$ through Equation 1, and N_+ and N_- are the number of positive observations and negative observations respectively. The value ϵ needs to be sufficiently small that the gcd term only makes the coefficients in $\boldsymbol{\lambda}$ coprime and does not effect the solution in any other way.

The relative importance of false positives and false negatives, w^+ and w^-, should generally be chosen by the user, depending on how much a false positive is worth relative to a false negative in the application. Often, we try many possible values of w^+ and w^- to create several models that are optimized for specific points on the ROC curve.

The optimization problem above is amenable to mixed integer linear programming, discussed in depth in Ustun & Rudin (2016b).

We have finished discussing the optimization problem solved by SLIM, now we move on to RiskSLIM.

RiskSLIM's **Optimization Framework for Risk Assessment**

In risk assessment applications, we use the score to estimate of predicted risk. Specifically, we estimate the *predicted risk* that example i belongs to the positive class using the logistic link function as:

$$\Pr\left(y_i = +1 \mid \boldsymbol{x}_i\right) = \frac{1}{1+\exp(-\boldsymbol{\lambda}^T\boldsymbol{x}_i)}.$$

We learn the values of the coefficients from data by solving the following mixed integer nonlinear program (MINLP), which we refer to as the *risk score problem* or RiskSlim-MINLP:

$$\begin{aligned} \min_{\boldsymbol{\lambda}} \quad & l(\boldsymbol{\lambda}) + C_0 \left\|\boldsymbol{\lambda}\right\|_0 \\ \text{s.t.} \quad & \boldsymbol{\lambda} \in \mathcal{L}, \end{aligned} \tag{4}$$

where:

- $l(\boldsymbol{\lambda}) = \frac{1}{N}\sum_{i=1}^{N} \log(1+\exp(-\boldsymbol{\lambda}^T y_i \boldsymbol{x}_i))$ is the logistic loss function;
- $\left\|\boldsymbol{\lambda}\right\|_0 = \sum_{j=1}^{d} \mathbb{1}\left[\lambda_j \neq 0\right]$ is the ℓ_0-seminorm;
- $\mathcal{L} \subset \mathbb{Z}^{d+1}$ is a set of feasible coefficient vectors (user-provided);
- $C_0 > 0$ is a trade-off parameter to balance fit and sparsity (user-provided);

The optimal solution to the risk score problem is a scoring system that we refer to as a *Risk-calibrated Supersparse Linear Integer Model.*

Here, the objective minimizes the *logistic loss* from logistic regression in order to achieve high values of the area under the ROC curve (AUC) and to achieve risk calibration. The objective penalizes the ℓ_0-seminorm for sparsity. The trade-off parameter C_0 controls the balance between these competing objectives, and represents the maximum log-likelihood that is sacrificed to remove a feature from the optimal model. The feasible region restricts coefficients to a small set of bounded integers such as $\mathcal{L} = \{-10, \ldots, 10\}^{d+1}$, and may be further customized to include operational constraints, such as those in Table 1.

In order to fit a RiskSLIM scoring system, we need to solve the MINLP above. This MINLP is difficult to solve using any commercial solver. Cutting plane algorithms are a natural choice for this problem because the objective is continuous and convex, but we were not able to use a traditional cutting plane algorithm because of the discrete domain of the optimization problem. Instead, we designed a specialized cutting plane technique that creates a series of branches, where we compute cutting planes on each branch. This allows us to solve very large problems and parallelize easily. This algorithm is called the "Lattice Cutting Plane Method," and more details can be found in the work of Ustun & Rudin (2016a).