MIT Sloan School of Management

MIT Sloan School Working Paper 5040-13

An Interpretable Stroke Prediction Model using Rules and Bayesian Analysis

Benjamin Letham, Cynthia Rudin, Tyler H. McCormick,
David Madigan

An Interpretable Stroke Prediction Model using Rules and Bayesian Analysis

Benjamin Letham Operations Research Center, MIT
Cynthia Rudin MIT Sloan School of Management
Tyler H. McCormick Department of Statistics, University of Washington
David Madigan Department of Statistics, Columbia University

Abstract

We aim to produce predictive models that are not only accurate, but are also interpretable to human experts. Our models are decision lists, which consist of a series of *if...then...* statements (for example, *if high blood pressure, then stroke*) that discretize a high-dimensional, multivariate feature space into a series of simple, readily interpretable decision statements. We introduce a generative model called the Bayesian List Machine which yields a posterior distribution over possible decision lists. It employs a novel prior structure to encourage sparsity. Our experiments show that the Bayesian List Machine has predictive accuracy on par with the current top algorithms for prediction in machine learning. Our method is motivated by recent developments in personalized medicine, and can be used to produce highly accurate and interpretable medical scoring systems. We demonstrate this by producing an alternative to the CHADS$_2$ score, actively used in clinical practice for estimating the risk of stroke in patients that have atrial fibrillation. Our model is as interpretable as CHADS$_2$, but more accurate.

1. INTRODUCTION

Our goal is to build predictive models that are highly accurate, yet are highly interpretable. These predictive models will be in the form of sparse *decision lists*, which consist of a series of *if... then...* statements where the *if* statements define a partition of a set of features and the *then* statements correspond to the predicted outcome of interest. Because of this form, a decision list model naturally provides a reason for each prediction that it makes. Figure 1 presents an example decision list that we created using the NHBLI Framingham Heart Study coronary heart disease (CHD) inventory (Wilson et al., 1998) for a 45 year-old male. The list provides an explanation of the risk factors that can be used by both healthcare providers and patients. A patient is at risk for CHD based on Figure 1, for example, *because* he has high blood pressure and smokes. The list in Fig. 1 is one accurate and interpretable decision list for predicting CHD, and there may be many other such lists. Our goal is to learn these lists from data.

Our model, called the Bayesian List Machine (BLM), produces a posterior distribution over permutations of *if... then...* rules, starting from a large set of possible pre-mined rules.

The decision lists with the highest posterior values tend to be both very accurate and very interpretable, where the interpretability comes from a novel sparsity-inducing prior structure over permutations of rules. The prior favors concise decision lists that have a small number of total rules, where the rules have few terms in the left-hand side.

The BLM provides a new type of balance between accuracy, interpretability and computation. Consider the challenge of constructing a predictive model that discretizes the input space in the same way as decision trees (Breiman et al., 1984; Quinlan, 1993) or decision lists (Rivest, 1987; Liu et al., 1998). Greedy construction methods like classification and regression trees (CART), C4.5, or typical Bayesian decision tree methods (Dension et al., 1998; Chipman et al., 1998, 2002) are not particularly computationally demanding, however in practice the greediness heavily affects the quality of the solution, both in terms of accuracy and interpretability. On the other hand, it is infeasible to fully optimize a decision tree due to the exponential size of the search space in the possibilities for the leaves of the tree. The BLM strikes a balance between these extremes, in that its solutions are not constructed in a greedy way, yet it can solve problems at the scale required to have an impact on modern healthcare.

BLM's practical feasibility is due to its two step procedure. The construction of the pre-mined rules in the first step massively reduces computation in the second step, where the rules are fully ordered: a computation over permutations of rules is substantially less demanding than a full optimization over the set of trees. As long as the pre-mined set of rules is sufficiently expressive, the quality of the decision list will be similar to that of a fully-optimized tree over the exponential search space. The rule ordering step for the BLM uses Bayesian analysis, where the prior structure encourages decision lists that are sparse. This serves not only the purpose of producing a more interpretable model, but also reduces computation, as most of the sampling iterations take place within a small set of permutations corresponding to the sparse decision lists. In practice, the BLM is able to compute predictive models that are as accurate as the state-of-the-art machine learning methods, on a scale that is much larger than that used for most modern medical scoring systems, yet maintain the same level of interpretability as medical scoring systems.

The motivation for our work lies in developing interpretable patient-level predictive models using massive observational medical data. To this end, we use the BLM to construct an alternative to the CHADS$_2$ score of Gage et al. (2001). CHADS$_2$ is widely-used in medical practice to predict stroke in patients with atrial fibrillation. Our model is built from

if total cholesterol $\geq$160 **and** smoke **then** *10 year CHD risk* $\geq$ *5%*
else if smoke **and** systolic blood pressure$\geq$140 **then** *10 year CHD risk* $\geq$ *5%*
else *10 year CHD risk* $<$ *5%*

Figure 1: Example decision list created using the NHBLI Framingham Heart Study coronary heart disease (CHD) inventory for a 45 year old male. The total number of rules is only 3, and the average number of terms on the left is 2, since both rules have 2 terms.

over a thousand times the amount of data used to build the CHADS$_2$ score, and is just as interpretable as CHADS$_2$ but much more accurate. In our experiments we compare the stroke prediction performance of BLM to CHADS$_2$, as well as to a collection of state-of-the-art machine learning algorithms: C4.5 (Quinlan, 1993), CART (Breiman et al., 1984), ℓ_1-regularized logistic regression, support vector machines (Vapnik, 1995), boosted decision trees (Freund and Schapire, 1995), and random forests (Breiman, 2001).

2. MOTIVATION AND RELATED WORK

Most widely used medical scoring systems are designed to be interpretable, but are not necessarily optimized for accuracy, and are generally derived from few factors. The Thrombolysis In Myocardial Infarction (TIMI) Score (Antman et al., 2000), Apache II score for infant mortality in the ICU (Knaus et al., 1985), the CURB-65 score for predicting mortality in community-acquired pneumonia (Lim et al., 2003), and the CHADS$_2$ score (Gage et al., 2001) are examples of interpretable predictive models that are very widely used. Each of these scoring systems involves very few calculations, and could be computed by hand during a doctor's visit. In the construction of each of these models, heuristics were used to design the features and coefficients for the model - none of these models were fully learned from data.

In contrast with these hand-designed interpretable medical scoring systems, recent advances in the collection and storing of medical data present unprecedented opportunities to develop powerful models that can predict a wide variety of outcomes (Shmueli, 2010). The front-end user interface of risk assessment tools are increasingly available online, however in large part the tools were developed using statistical models that are not interpretable. At the end of the assessment, a patient may be told he or she has a high risk for a particular outcome but have no understanding of why the risk is high or what steps can be taken to reduce risk.

In this work, we focus on the CHADS$_2$ score for predicting stroke in patients with atrial fibrillation. A patient's score is computed by assigning one "point" each for the presence of congestive heart failure (C), hypertension (H), age 75 years or older (A), and diabetes mellitus (D), and by assigning 2 points for history of stroke, transient ischemic attack, or thromoembolism (S$_2$). The CHADS$_2$ score considers only 5 factors, whereas the updated CHA$_2$DS$_2$-VASc score (Lip et al., 2010) includes three additional risk factors: vascular disease (V), age 65 to 74 years old (A), and female gender (Sc). Higher scores correspond to increased risk. In the study defining the CHADS$_2$ score (Gage et al., 2001), the scores was calibrated with stroke risks using a database of 1,733 Medicare beneficiaries followed for, on average, about a year. These calibration data demonstrate a key challenge in making predictions for (relatively) rare but important events. During the follow-up period, there were only 94 strokes across all risk categories (scores 0-6). Most patients were in lower risk categories, leaving very few patients to calibrate risk for patients with the highest scores. There were 65 patients with a score of 5 and only 5 patients with the maximum score of 6, for example. Thus, the CHADS$_2$ score is calibrated using the least data for patients most at risk. For the study in this paper, we use data from nearly ten times as many patients (n=12,586 with 1,786 strokes) and do not have the same problem with small samples for the highest risk categories.

2.1 Related work in machine learning and statistics

In general, humans can handle only a handful of cognitive entities at once (Miller, 1956; Jennings et al., 1982). It has long since been hypothesized that simple models predict well, both in the machine learning literature (Holte, 1993), and in the psychology literature (Dawes, 1979). The concept of explanation in statistical modeling (which is related to interpretability) has been explored in several past works (Madigan et al., 1997; Giraud-Carrier, 1998; Vellido et al., 2012; Rüping, 2006; Bratko, 1997, for example).

Decision lists have the same form as models used in the expert systems literature from the 1970's and 1980's (Leondes, 2002), which were among the first successful types of artificial intelligence. The knowledge base of an expert system is composed of natural language statements that are *if... then...* rules. Decision lists are a type of associative classifier, meaning that the list is formed from association rules. In general these rules indicate only correlated factors, not necessarily causative factors. In the past, associative classifiers have been constructed from heuristic sorting mechanisms (Rivest, 1987; Liu et al., 1998; Li et al., 2001; Yin and Han, 2003; Yi and Hüllermeier, 2005; Marchand and Sokolova, 2005; Rudin et al., 2011). Some of these sorting mechanisms work provably well in special cases, for instance when the decision problem is easy and the classes are easy to separate, but are not optimized to handle more general problems. Sometimes associative classifiers are formed by averaging several rules together, but the resulting classifier is not generally interpretable (Friedman and Popescu, 2008; Meinshausen, 2010). In the thesis of Chang (2012), rules are ordered using discrete optimization.

Decision trees are closely related to decision lists, and are in some sense equivalent. Decision trees are almost always constructed greedily from the top down, and then pruned heuristically upwards and cross-validated to ensure accuracy. Because the trees are not fully optimized, if the top of the decision tree happened to have been chosen badly at the start of the procedure, it could cause problems with both accuracy and interpretability. Bayesian decision trees (Dension et al., 1998; Chipman et al., 1998, 2002) use Markov chain Monte Carlo (MCMC) to sample from a posterior distribution over trees. Early sampling approaches in Bayesian decision trees converged slowly and required repeatedly restarting the sampling procedure. Wu et al. (2007) improve chain convergence using a "radical restructure" Metropolis-Hastings move. The space of decision lists using pre-mined rules is significantly smaller than the space of decision trees, making it easier to obtain MCMC convergence. Moreover, rule mining allows for the rules to be individually powerful.

This work is related to the Hierarchical Association Rule Model (HARM) presented recently by McCormick et al. (2012). HARM is a Bayesian model that uses rules, but for a different medical context and a different statistical problem. HARM estimates the conditional probabilities of each rule in a conservative way, and does not explicitly aim to learn the ordering of rules, as the BLM does.

3. THE BAYESIAN LIST MACHINE

We begin by presenting our method as a generative model. We are in the setting of multi-class classification, where the set of possible labels is $1, \dots, L$. In the case of predicting stroke risk, there are only two possible labels: stroke or no stroke. The training data are pairs $\{(x_i, y_i)\}_{i=1}^n$, where $x_i \in \mathbb{R}^d$ are the features of observation i, and y_i are the labels, within classes $1, ..., L$. Our predictions are based on a set of rules $r = 1, \ldots, R$ (left-hand sides) which are constructed from the training data and used to form lists. Informally, we can describe the generative model as follows, where each step is discussed in more detail below.

- Generate an exhaustive list of rules $r = 1, \ldots, R$ using a rule-mining algorithm. These rules come from the set of features and are used to make lists.
- Generate a random permutation over rules π from a prior $\text{PRIOR}(p, C)$.
- Using this ordering, for each observation x_i, select the first rule that applies, meaning it matches the observed features. Call the rule $\tilde{r}_i$.
- Generate the label y_i from a Dirichlet-Multinomial distribution $\theta^{(\tilde{r}_i)}$, with Dirichlet parameters $\alpha_1, \ldots, \alpha_L$ and counts $n_{\tilde{r}_i 1}, \ldots, n_{\tilde{r}_i L}$ for rule $\tilde{r}_i$ chosen in the previous step.

The posterior and the prior are distributions over rule lists. To obtain a single rule list, we could choose, for instance, the rule list having the highest posterior probability (the maximum *a posteriori* estimate).

The first step in the model is to generate the set of rules. For situations where the dimensionality of the features is fairly low, we may consider all possible candidate rules; however in most applications we select a smaller number of rules using an algorithm for frequent itemset mining. In our experiments we used the FP-Growth algorithm (Borgelt, 2005) which finds all itemsets that satisfy constraints on minimum support and maximum cardinality. As long as the set of rules is large enough, we should be able to find subsets of them, and permutations of the subsets, that form useful decision lists.

3.1 A prior over decision lists

After the set of rules is constructed, we draw an ordering over rules from a prior distribution over permutations of rules, $\pi \sim \text{PRIOR}(p, C)$. The prior favors shorter decision lists (small total number of rules, sparse in the vertical direction of the list), and prefers rules with a small number of conditional statements (small left-hand sides of rules, sparse in the horizontal direction). Let R_π be the number of rules in the list, A_π be the average size of the left-hand sides of the rules on the list, and M be the maximum allowed size of the rules. For example, the decision list in Figure 1 has $R_\pi = 3$ and $A_\pi = (2 + 2 + 0)/3$. Then,

$$\text{PRIOR}(\pi) \propto \frac{1}{\left(R_\pi + C\frac{A_\pi}{M}\right)^p},$$

where the user-specified parameter C in the prior trades off between horizontal and vertical sparseness, and parameter p controls the overall strength of the prior. A_π/M is a fraction between 0 and 1, allowing the parameter C to be calibrated in a more intuitive way. For instance, when $C = 1$, because $A_\pi/M \leq 1$, reducing the length of the decision list R_π would be favored over reducing A_π. Both prior hyperparameters p and C can be adjusted to the user's view of interpretability, or can be cross-validated. To promote sparsity, one can mine only rules with small left-hand-sides, in which case M would be relatively small. In our experiments we set $C = 1$ and used the single prior hyperparameter p to directly control the length of the decision list, and set it either using cross-validation or to a specific value to obtain a list of desired length. M was chosen to be 2 or 5 in our experiments.

3.2 The likelihood function

Define $\tilde{r} \in \mathbb{R}^n$ as a vector of rule labels such that element $\tilde{r}_i = r$ if r is the first rule in the decision list π that applies to observation x_i. The vector $\tilde{r}$ partitions the set of outcomes y_i so that the likelihood for each response is computed under exactly one rule. We then use these rule assignments to construct multinomial counts $n_{r\ell}$ for each rule $r = 1, \dots, R$ and for each class $\ell = 1, \dots, L$ by tallying the number of times rule r was associated with an outcome in class ℓ. That is, $n_{r\ell}$ is the number of observations x for which r was the first rule in the list that applied, and which have label $y = \ell$. Let $n_r = \sum_{\ell=1}^{L} n_{r\ell}$ be the total number of observations classified by rule r. The likelihood is then

$$\mathcal{L}(y_1, \dots, y_n | \theta^{(1)}, \dots, \theta^{(R)}, \tilde{r}) = \prod_{r=1}^{R} \text{Multinomial}(n_{r1}, \dots, n_{rL} | n_r, \theta^{(r)}),$$

where

$$\theta^{(r)} \sim \text{Dirichlet}(\alpha_1, \dots, \alpha_L).$$

Since the $\theta^{(r)}$ are not of primary interest, we can marginalize over $\theta^{(r)}$ in each Multinomial distribution in the above product, obtaining, through the standard derivation of the Dirichlet-Multinomial distribution,

$$\begin{aligned} p(y_1, \dots, y_n | \alpha_1, \dots, \alpha_L, \tilde{r}) &= \prod_{r=1}^{R} \frac{\Gamma(\sum_{\ell=1}^{L} \alpha_\ell)}{\Gamma(\sum_{\ell=1}^{L} n_{r\ell} + \alpha_\ell)} \times \prod_{\ell=1}^{L} \frac{\Gamma(n_{r\ell} + \alpha_\ell)}{\Gamma(\alpha_\ell)} \\ &\propto \prod_{r=1}^{R} \frac{\prod_{\ell=1}^{L} \Gamma(n_{r\ell} + \alpha_\ell)}{\Gamma(\sum_{\ell=1}^{L} n_{r\ell} + \alpha_\ell)}, \end{aligned} \tag{1}$$

The above equation depends on the rule indicators $\tilde{r}_i$ through the counts $n_{r\ell}$.

In practice, many datasets are extremely imbalanced. For example, many fewer medical patients have a stroke than do not have a stroke. In such circumstances, without an appropriate correction, the likelihood can be dominated by negative responses and as a result the method will simply predict "no stroke" for each patient using a single default rule. We may instead desire to trade off between sensitivity and specificity of the classifier. To do this, we introduce an altered likelihood for imbalanced data:

$$p(y_1, \dots, y_n | \alpha_1, \dots, \alpha_L, \tilde{r}) \propto \prod_{r=1}^{R} \frac{\prod_{\ell=1}^{L} \Gamma(\upsilon_\ell n_{r\ell} + \alpha_\ell)}{\Gamma(\sum_{\ell=1}^{L} \upsilon_\ell n_{r\ell} + \alpha_\ell)}, \tag{2}$$

where $\upsilon_\ell = L/P(y = \ell)$. The υ_ℓ terms re-weight the observations in each class to introduce additional weight in the likelihood for underrepresented cases. For imbalanced datasets, we apply the rule mining algorithm separately to each class to ensure that rules that are powerful for a particular underrepresented class are not rejected by the minimum support threshold.

3.3 Markov chain Monte Carlo sampling

The rule that ends the usable part of the list is called the "default" rule. The default rule has an empty left hand side, so that every observation applies to it. In the example of Fig. 1, the default rule is "*else 10 year CHD risk* $< 5\%$." The default rule supplies the prediction for every observation for which a prediction was not made at an earlier rule in the list.

We do Metropolis sampling, generating the proposed π^* from the current π_t using one of three options: 1) Swap the position of two rules on the decision list. 2) Add a rule to the decision list, by moving it above the default rule. 3) Remove a rule from the decision list, by moving it below the default rule. Which rules to adjust, and their new positions, are chosen randomly at each step. The option to swap, add, or remove is also chosen randomly. As long as the probabilities of adding and removing are the same, the proposal distribution is symmetric. In our experiments we used a uniform distribution over these choices. This sampling algorithm is related to those used for Bayesian Decision Tree models (Chipman et al., 2002, 1998; Wu et al., 2007), however, due to the reduction in the size of the search space from the rule mining step, we do not have similar problems with local maxima. We ensure that the sampler is not trapped in a local maximum using a convergence diagnostic which we now describe.

3.4 Convergence diagnostic

We assess chain convergence using the method of Brooks et al. (2003) with the novel addition of a randomization test on the chi-squared statistic. We begin J chains from randomly selected initial conditions and run them for N iterations. We discard the first half of the samples as burn-in, and thin the remaining samples at a rate of 100. Suppose that the J chains visited a total of c decision lists. We define N_ν^j as the number of times chain j visited decision list ν in the thinned samples, $j = 1, \dots, J$ and $\nu = 1, \dots, c$. We then implement a chi-square test of homogeneity across the chains. If the chains were homogeneous, the expected number of visits per chain to each decision list ν would be $E_\nu = \frac{1}{J}\sum_{j=1}^J N_\nu^j$ and the chi-squared statistic is

$$\chi^2 = \sum_{\nu=1}^{c}\sum_{j=1}^{J} \frac{(N_\nu^j - E_\nu)^2}{E_\nu}.$$

Brooks et al. (2003) use Pearson's chi-squared test to compute a p-value. If the p-value is sufficiently large (*e.g.*, greater than 0.05) then the null hypothesis of chain homogeneity cannot be rejected, and the chains can be considered converged. Pearson's chi-squared test tends to perform poorly when counts are less than around 5, which is often the case for chains over decision lists because the space of decision lists is very large. Thus rather than use the χ^2 distribution which is only asymptotically accurate, here we empirically estimate the actual distribution of the χ^2 statistic. This is done by randomly sampling a large number of contingency tables with the same marginals as

$$\begin{pmatrix} N_1^1 & \dots & N_c^1 \\ \vdots & & \vdots \\ N_1^J & \dots & N_c^J \end{pmatrix}$$

and computing their χ^2 statistic. Random contingency tables with fixed marginals can be efficiently sampled using Patefield's algorithm (Patefield, 1981), which is available as the R function "r2dtable." This provides an empirical distribution for the χ^2 statistic and the p-value can be estimated directly as the fraction of randomly generated tables with a χ^2 value larger than that of the Markov chain Monte Carlo sample chains. In our experiments, we used 3 chains and determined chains had converged if $p > 0.05$, for which in our experiments $N = 10^6$ was sufficient.

4. SIMULATION AND BENCHMARK DATASET STUDIES

We empirically analyzed the performance of the BLM using simulation studies to investigate consistency, and benchmark dataset studies to investigate generalization and prediction performance.

4.1 Simulation studies

We use simulation studies to demonstrate that when data are generated by a decision list model, the BLM method is able to recover the true decision list. We generated independent, random observations with $d = 1000$ binary features. Given observations with arbitrary features and a collection of rules on those features, the observations can be transformed to a new, binary feature space where each feature corresponds to a rule, and takes value 1 if the rule applies to that observation and 0 otherwise. Thus, we can consider observations to be binary vectors without loss of generality. We generated a random decision list of size 10 as a random ordering of 9 features, plus the default rule. Each rule in the decision list was assigned a distribution over labels using a random draw from the Beta$(1/2, 1/2)$ distribution, which ensures that the rules are informative about labels. Labels were then assigned to each observation using the decision list: For each observation, the label was taken as a draw from the label distribution corresponding to the first rule that applied to that observation. We applied BLM as described in Section 3, for a range of numbers of observations n and prior strengths p.

To appropriately visualize the posterior distribution, we binned the posterior rules according to their distance from the true decision list, using the number of incorrect positions on the list as the distance metric. A position on a list was considered incorrect if it did not contain exactly the rule that the true decision list had at that position, thus the number of incorrect positions ranged from 0 (exactly the true decision list) to 10 (an error at every position). This metric is conservative, because lists are equivalent when the positions of two rules are swapped if the sets of observations satisfying those rules are disjoint. The results of the simulations are given in Fig. 2.

Figure 2(a) shows that when the number of observations is small, the posterior mass is primarily concentrated on decision lists with a large number of incorrect positions, and as the number of observations is increased the posterior shifts to decision lists that are closer to the true list. Figure 2(b) illustrates the balance between the prior, which encourages small decision lists, and the data, which eventually overwhelm the prior. Together, Figs. 2(a) and (b) show that even with very few observations many of the highest rules on the list are in their correct positions, but the prior is encouraging lists that are shorter than the true,

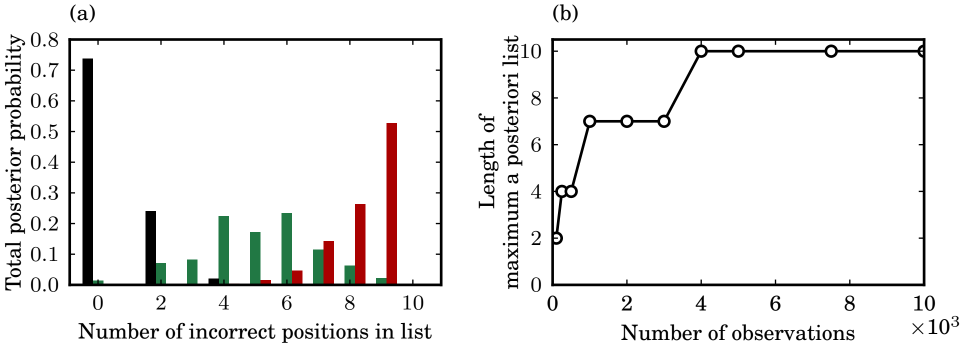

Figure 2: (a) Total posterior probability of all posterior decision lists with the specified number of incorrect positions, for 100 observations (red), 1,000 observations (green), and 10,000 observations (black). (b) The length of the maximum *a posteriori* decision list for for simulations with varying numbers of observations.

generating list. With 4,000 observations and higher, the maximum *a posteriori* list was the true decision list.

4.2 Benchmark dataset studies

The simulation studies primarily address the question of consistency, and show that when the data follow a decision list model, we are able to recover that model. We now use benchmark dataset studies to address the questions of generalization and prediction accuracy. We consider a collection of datasets from the UCI Machine Learning Repository (Bache and Lichman, 2013) that are frequently used in the benchmark dataset studies of machine learning papers: Tic-Tac-Toe Endgame, Mammographic Mass (Elter et al., 2007), Titanic, Breast Cancer Wisconsin (Original) (Mangasarian and Wolberg, 1990), and Adult.

For each dataset, categorical features were separated into binary features and real-valued features were split at their median into two binary features each. We used 5-fold cross validation and measured classification accuracy on each fold. None of these datasets suffer from extreme class imbalance, so we used the form of the likelihood given in (1). For all datasets except Adult, the parameters for rule mining were 5% minimum support and maximum cardinality of 5. For Adult the minimum support threshold was increased to 20% due to a large number of itemsets. We chose the strength of the prior p using 5-fold cross validation on each training set with $p = 0.5$, 2, and 5, and set p at the value that maximized area under the receiver operating characteristic curve (AUC) over the validation sets. We made predictions using the maximum *a posteriori* decision list.

In Table 1 we compare the prediction accuracy to C4.5, CART, ℓ_1-regularized logistic regression, support vector machines (SVM), boosted decision trees (BDT), and random forests. The implementation details for these comparison algorithms are in the Appendix. For all of these datasets, the decision lists created by BLM had prediction power on par with the other commonly used learning algorithms.

The BLM decision lists for these datasets were also all interpretable. For example, the Tic-Tac-Toe dataset contains all possible end board configurations for the game Tic-Tac-Toe, with the task of determining whether or not player "X" won. The dataset is deterministic,

Table 1: Mean classification accuracy across 5 folds of cross-validation, and in parentheses standard deviation, for various machine learning algorithms applied to UCI benchmark datasets

	Tic-Tac-Toe	Mammogram	Titanic	Wisconsin	Adult
BLM	1.00 (0.00)	0.81 (0.05)	0.79 (0.02)	0.95 (0.02)	0.82 (0.01)
C4.5	0.86 (0.04)	0.81 (0.05)	0.77 (0.02)	0.93 (0.03)	0.83 (0.01)
CART	0.88 (0.04)	0.79 (0.06)	0.79 (0.02)	0.93 (0.02)	0.82 (0.01)
Logistic Reg.	0.98 (0.01)	0.81 (0.06)	0.78 (0.02)	0.96 (0.01)	0.85 (0.01)
SVM	0.99 (0.01)	0.79 (0.06)	0.78 (0.02)	0.96 (0.02)	0.84 (0.02)
BDT	0.86 (0.04)	0.81 (0.06)	0.78 (0.02)	0.96 (0.02)	0.85 (0.01)
Rand. Forest	0.98 (0.01)	0.81 (0.06)	0.79 (0.02)	0.96 (0.02)	0.84 (0.02)

if male **and** adult **then** *died* (80%)
else if 3rd class **then** *died* (59%)
else *survived* (93%)

Figure 3: Decision list for Titanic. In parentheses, we give the proportion of observations satisfying that rule and no previous rule for which the classification was correct.

and there are exactly 8 ways that player “X” can win. The BLM decision list contained exactly the 8 ways of winning, and thus achieved perfect prediction accuracy, something that no other machine learning algorithm could do. As another example, the Titanic dataset contains the ticket class, gender, and age for the passengers of the Titanic, with the task of determining if the passenger survived or not. In Fig. 3 we give the fitted decision list for the Titanic dataset, which is consistent with historical accounts of space on lifeboats being limited to women and children, particularly those with higher-class tickets. Across all of the benchmark dataset experiments, the mean size of the BLM decision list ranged from 3.8 to 8.4, which could certainly be considered interpretable.

The benchmark dataset studies show that for these simple, frequently used example datasets, the decision lists produced by BLM have predictive power similar to popular machine learning algorithms. In the next section, we apply BLM to a large dataset of medical histories and learn decision lists for the real problem of stroke prediction in atrial fibrillation patients.

5. STROKE PREDICTION COMPARED TO CHADS$_2$

We use the Bayesian list machine to derive a competitor to CHADS$_2$ using the MarketScan Medicaid Multi-State Database (MDCD). MDCD contains administrative claims data for 11.1 million Medicaid enrollees from multiple states. This database forms part of the suite of databases that the Observational Medical Outcomes Partnership (OMOP, `http://omop.fnih.org`) has mapped to a common data model (Stang et al., 2010). We extracted every

```
if hemiplegia then stroke risk 58.0% (14.5%)
else if cerebrovascular disorder then stroke risk 46.6% (12.5%)
else if transient ischaemic attack and essential hypertension
    then stroke risk 23.2% (8.3%)
else if occlusion and stenosis of carotid artery without cerebral infarction
    then stroke risk 16.4% (7.8%)
else if $\text{age} \leq 60$ then stroke risk 3.7% (7.4%)
else stroke risk 8.5%
```

Figure 4: Decision list for determining 1-year stroke risk following diagnosis of atrial fibrillation from patient medical history. For each rule we give in parentheses the base risk for all patients that make it to that depth on the list.

patient in the MDCD database with a diagnosis of atrial fibrillation, one-year of atrial fibrillation-free observation time prior to the diagnosis, and one year of observation time following the diagnosis (n=12,586). Of these, 1,786 (14%) had a stroke within a year of the atrial fibrillation diagnosis. This is a much larger dataset than the one originally used to develop the CHADS$_2$ score (n=1,733, with 94 strokes).

As candidate predictors we considered all drugs and all conditions. Specifically, for every drug and condition, we created a binary predictor variable indicating the presence or absence of the drug or condition in the longitudinal record prior to the atrial fibrillation diagnosis. These totaled 4,146 unique medications and conditions. We included features for age and gender. Specifically, we used 50, 60, 70, and 80 years of age as split points, and for each split point introduced a pair of binary variables indicating whether the patient's age is less than or greater than the split point. We mined rules separately for each class (stroke or no stroke) using a minimum support threshold of 10% and a maximum cardinality M of 2. The total number of rules used in five folds of cross-validation ranged from 2162 to 2240. We used the likelihood model for imbalanced data, (2), and set the BLM prior hyperparameter at $p = 700$ to obtain a list of similar complexity to the CHADS$_2$ score. We followed the sampling procedure and then evaluated the performance of the maximum *a posteriori* decision list using 5-fold cross-validation, constructing a receiver operating characteristic (ROC) curve and measuring AUC for each fold.

In Fig. 4 we show the maximum *a posteriori* decision list recovered from one of the folds. For each rule we give the stroke risk estimated from the training data as the number of patients satisfying that rule (and no preceding rule) that had a stroke. We give in parentheses the stroke risk across all patients that did not satisfy any of the preceding rules in the list. For example, the second line in the list indicates that among patients without hemiplegia the stroke risk was 12.5%, which increased to 46.6% when patients had a cerebrovascular disorder.

The list indicates that past history of stroke reveals a lot about the vulnerability toward future stroke. In particular, the first half of the decision list focuses on a history of stroke, in order of severity. Hemiplegia, the paralysis of an entire side of the body, is a symptom of

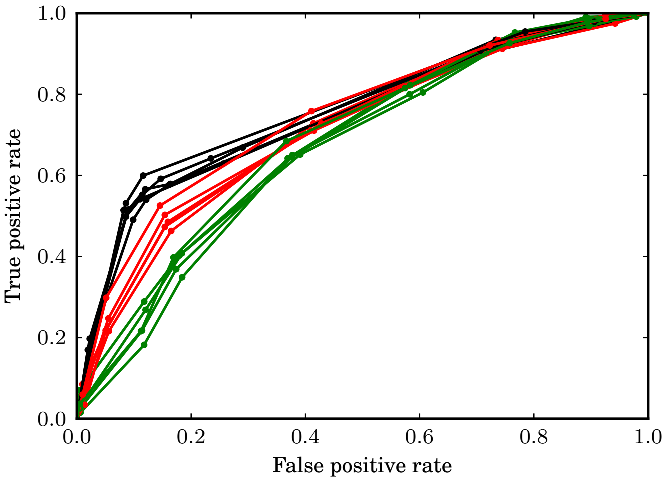

Figure 5: ROC curves for stroke prediction on the MDCD database for each of 5 folds of cross-validation, for BLM (black), $CHADS_2$ (red), and CHA_2DS_2-VASc (green).

a severe stroke. Cerebrovascular disorder indicates a prior stroke, and transient ischaemic attacks are generally referred to as "mini-strokes." The second half of the decision list includes age factors and vascular disease, which are known risk factors and are included in the CHA_2DS_2-VASc score. The lists that we obtained in the 5 folds of cross-validation were of similar complexity to the $CHADS_2$ score: the mean list length was 6.8 (standard deviation 0.8). For comparison, $CHADS_2$ uses 5 features and CHA_2DS_2-VASc uses 8 features.

In Fig. 5 we give ROC curves for all 5 folds for BLM, $CHADS_2$, and CHA_2DS_2-VASc, and in Table 2 we report mean AUC across the folds. These results show that with complexity and interpretability similar to $CHADS_2$, the BLM decision lists performed significantly better at stroke prediction than both $CHADS_2$ and CHA_2DS_2-VASc ($p < 0.01$, t-test). Interestingly, we also found that $CHADS_2$ outperformed CHA_2DS_2-VASc despite CHA_2DS_2-VASc being an extension to $CHADS_2$. This is likely because the model for the CHA_2DS_2-VASc score, in which risk factors are added linearly, is a poor model of actual stroke risk. For instance, the stroke risk percentages calibrated to the CHA_2DS_2-VASc scores are not a monotonic function of score: The stroke risk with a CHA_2DS_2-VASc score of 7 is 9.6%, whereas a score of 8 corresponds to a stroke risk of 6.7%. The fact that more stroke risk factors can correspond to a lower stroke risk suggests that the CHA_2DS_2-VASc model is misspecified, and highlights the difficulty in constructing these interpretable models manually.

Table 2 also gives performance results for the same collection of machine learning algorithms used in Section 4.2. The decision tree algorithms CART and C4.5, the only other interpretable classifiers, were outperformed even by $CHADS_2$. The BLM performance was comparable to that of the standard, generally uninterpretable, machine learning algorithms.

6. DISCUSSION AND CONCLUSION

We are working under the hypothesis that many real datasets permit predictive models that can be surprisingly small. This was hypothesized over a decade ago (Holte, 1993), however, we now are starting to have the computational tools to truly test this hypothesis. The BLM

Table 2: Mean, and in parentheses standard deviation, of AUC across 5 folds of cross-validation for stroke prediction

	AUC
BLM	0.750 (0.007)
CHADS$_2$	0.721 (0.014)
CHA$_2$DS$_2$-VASc	0.677 (0.007)
C4.5	0.553 (0.019)
CART	0.703 (0.010)
Logistic Reg.	0.767 (0.011)
SVM	0.763 (0.013)
BDT	0.780 (0.017)
Rand. Forest	0.776 (0.012)

method introduced in this work aims to hit the "sweet spot" between predictive accuracy, interpretability, and tractability.

For problems where interpretability requires extra constraints on the ordering or form of the rules, the framework introduced here can be adapted to handle that, and there are several ways to do this. First, the prior can be set to zero for lists that are "uninterpretable" according to a given definition. Second, post-processing on the lists can be performed in order to engineer the lists towards the desired level of interpretability. In that case, one should beware of changing the accuracy level when working manually with the lists. Third, one can explore the set of lists having high posterior values, and can choose among those lists for the one that is the most interpretable.

Interpretable models have the benefits of being both concise and convincing. A small set of trustworthy rules can be the key to communicating with domain experts and to allow machine learning algorithms to be more widely implemented and trusted. In practice, a preliminary interpretable model can help domain experts to troubleshoot the inner workings of a complex model, in order to make it more accurate and tailored to the domain. We demonstrated that interpretable models lend themselves to the domain of predictive medicine, but there are a wide variety of domains in science, engineering, and industry, where these models would be a natural choice.

APPENDIX

Comparison algorithm implementations

Support vector machines: LIBSVM (Chang and Lin, 2011) with a radial basis function kernel. We selected the slack parameter C_{SVM} and the kernel parameter γ using a grid-search over the ranges $C_{\text{SVM}} \in \{2^{-4}, 2^{-3}, \ldots, 2^4\}$ and $\gamma \in \{2^{-6}, 2^{-5}, \ldots, 2^0\}$ to find the parameters that maximized AUC over a 5-fold cross-validation over each training set. *C4.5*: C4.5 Release 8, distributed by Quinlan. *CART*: The R library "rpart" with default parameters and pruned using the complexity parameter that minimized cross-validation error. *Logistic regression*: The LIBLINEAR (Fan et al., 2008) implementation of logistic regression with ℓ_1 regularization. We selected the regularization parameter C_{LR} from $\{2^{-4}, 2^{-3}, \ldots, 2^4\}$ using

5-fold cross-validation over each training set to find the parameter that maximized AUC. *Boosted decision trees*: The R library "gbm" with shrinkage = 0.005, ntrees = 10000, and the number of iterations selected with 5-fold cross-validation. *Random forests*: The R library "randomForest" with $10,000$ trees.

ACKNOWLEDGEMENT

Cynthia Rudin's research was partly funded by grant IIS-1053407 from the National Science Foundation. Tyler McCormick's research was partially funded by a Google Faculty Award and NIAID grant R01 HD54511. David Madigan's research was partly funded by grant R01 GM87600-01 from the National Institutes for Health. The authors thank Zachary Shahn and the OMOP team for help with the data.

REFERENCES

Antman, E. M., Cohen, M., Bernink, P. M., and et al (2000). The TIMI risk score for unstable angina/nonST elevation MI: A method for prognostication and therapeutic decision making. *The Journal of the American Medical Association*, 284(7):835–842.

Bache, K. and Lichman, M. (2013). UCI machine learning repository.

Borgelt, C. (2005). An implementation of the fp-growth algorithm. In *Proceedings of the 1st international workshop on open source data mining: frequent pattern mining implementations*, OSDM '05, pages 1–5.

Bratko, I. (1997). Machine learning: Between accuracy and interpretability. *Courses and Lectures-International Centre for Mechanical Sciences*, pages 163–178.

Breiman, L. (2001). Random forests. *Machine Learning*, 45:5–32.

Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. (1984). *Classification and Regression Trees*. Wadsworth.

Brooks, S. P., Giudici, P., and Philippe, A. (2003). Nonparametric convergence assessment for MCMC model selection. *Journal of Computational and Graphical Statistics*, 12(1):1–22.

Chang, A. (2012). *Integer Optimization Methods for Machine Learning*. PhD thesis, Massachusetts Institute of Technology.

Chang, C.-C. and Lin, C.-J. (2011). LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Chipman, H. A., George, E. I., and McCulloch, R. E. (1998). Bayesian CART Model Search. *Journal of the American Statistical Association*, 93(443):935–948.

Chipman, H. A., George, E. I., and McCulloch, R. E. (2002). Bayesian Treed Models . *Machine Learning*, 48(1/3):299–320.

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

Dension, D., Mallick, B., and Smith, A. (1998). A Bayesian CART algorithm. *Biometrika*, 85(2):363–377.

Elter, M., Schulz-Wendtland, R., and Wittenberg, T. (2007). The prediction of breast cancer biopsy outcomes using two cad approaches that both emphasize an intelligible decision process. *Medical Physics*, 34:4164–4172.

Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., and Lin, C.-J. (2008). LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874.

Freund, Y. and Schapire, R. (1995). A decision-theoretic generalization of on-line learning and an application to boosting. *Computational Learning Theory*, 904:23–37.

Friedman, J. H. and Popescu, B. E. (2008). Predictive learning via rule ensembles. *The Annals of Applied Statistics*, 2(3):916–954.

Gage, B., Waterman, A., Shannon, W., Boechler, M., Rich, M., and Radford, M. (2001). Comparing hospitals on stroke care: The need to account for stroke severity. *Journal of the American Medical Association*, 285:2864–2870.

Giraud-Carrier, C. (1998). Beyond predictive accuracy: What? In *Proceedings of the ECML-98 Workshop on Upgrading Learning to Meta-Level: Model Selection and Data Transformation*, pages 78–85.

Holte, R. C. (1993). Very simple classification rules perform well on most commonly used datasets. *Machine Learning*, 11(1):63–91.

Jennings, D. L., Amabile, T. M., and Ross, L. (1982). Informal covariation assessments: Data-based versus theory-based judgements. In Kahneman, D., Slovic, P., and Tversky, A., editors, *Judgment Under Uncertainty: Heuristics and Biases,*, pages 211–230. Cambridge Press, Cambridge, MA.

Knaus, W. A., Draper, E. A., Wagner, D. P., and Zimmerman, J. E. (1985). APACHE II: A severity of disease classification system. *Critical Care Medicine*, 13:818–829.

Leondes, C. T. (2002). *Expert systems: the technology of knowledge management and decision making for the 21st century*. Academic Press.

Li, W., Han, J., and Pei, J. (2001). CMAR: Accurate and efficient classification based on multiple class-association rules. *IEEE International Conference on Data Mining*, pages 369–376.

Lim, W., van der Eerden, M., Laing, R., Boersma, W., Karalus, N., Town, G., Lewis, S., and Macfarlane, J. (2003). Defining community acquired pneumonia severity on presentation to hospital: an international derivation and validation study. *Thorax*, 58(5):377–382.

Lip, G., Nieuwlaat, R., Pisters, R., Lane, D., and Crijns, H. (2010). Refining clinical risk stratification for predicting stroke and thromboembolism in atrial fibrillation using a novel risk factor-based approach: the euro heart survey on atrial fibrillation. *Chest*, 137:263–272.

Liu, B., Hsu, W., and Ma, Y. (1998). Integrating classification and association rule mining. In *Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining*, pages 80–96.

Madigan, D., Mosurski, K., and Almond, R. (1997). Explanation in belief networks. *Journal of Computational and Graphical Statistics*, 6:160–181.

Mangasarian, O. L. and Wolberg, W. H. (1990). Cancer diagnosis via linear programming. 23(5):1–18.

Marchand, M. and Sokolova, M. (2005). Learning with decision lists of data-dependent features. *Journal of Machine Learning Research*.

McCormick, T. H., Rudin, C., and Madigan, D. (2012). Bayesian hierarchical rule modeling for predicting medical conditions. *The Annals of Applied Statistics*, 6:652–668.

Meinshausen, N. (2010). Node harvest. *The Annals of Applied Statistics*, 4(4):2049–2072.

Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits to our capacity for processing information. *The Psychological Review*, 63(2):81–97.

Patefield, W. M. (1981). Algorithm as159. an efficient method of generating r x c tables with given row and column totals. *Applied Statistics*, 30:91–97.

Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning.* Morgan Kaufmann.

Rivest, R. L. (1987). Learning decision lists. *Machine Learning*, 2(3):229–246.

Rudin, C., Letham, B., Salleb-Aouissi, A., Kogan, E., and Madigan, D. (2011). Sequential event prediction with association rules. In *Proceedings of the 24th Annual Conference on Learning Theory (COLT).*

Rüping, S. (2006). *Learning interpretable models.* PhD thesis, Universität Dortmund.

Shmueli, G. (2010). To Explain or to Predict? *Statistical Science*, 25(3):289–310.

Stang, P., Ryan, P., Racoosin, J., Overhage, J., Hartzema, A., Reich, C., Welebob, E., Scarnecchia, T., and Woodcock, J. (2010). Advancing the science for active surveillance: rationale and design for the observational medical outcomes partnership. *Annals of Internal Medicine*, 153:600–606.

Vapnik, V. N. (1995). *The Nature of Statistical Learning Theory.* Springer-Verlag, New York.

Vellido, A., Martín-Guerrero, J. D., and Lisboa, P. J. (2012). Making machine learning models interpretable. In *Proc. European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning.*

Wilson, P. W. F., D'Agostino, R. B., Levy, D., Belanger, A. M., Silbershatz, H., and Kannel, W. B. (1998). Prediction of coronary heart disease using risk factor categories. *Circulation*, 97(18):1837–1847.

Wu, Y., Tjelmeland, H., and West, M. (2007). Bayesian CART: Prior Specification and Posterior Simulation. *Journal of Computational and Graphical Statistics*, 16(1):44–66.

Yi, Y. and Hüllermeier, E. (2005). Learning complexity-bounded rule-based classifiers by combining association analysis and genetic algorithms. In *Proc. Joint 4th Int. Conf. in Fuzzy Logic and Technology and 11th French Days on Fuzzy Logic and Applications*, pages 47–52.

Yin, X. and Han, J. (2003). CPAR: Classification based on predictive association rules. In *Proceedings of the 2003 SIAM International Conference on Data Mining*, pages 331–335.