A Computationally Efficient Algorithm for Producing Risk Score Models with Applications to Tuberculosis Diagnosis and Treatment Adherence

Hannah Eglinton

Brown University, May 2024

Master's Thesis

Submitted in fulfillment of the requirements for ScM in Biostatistics in the

Brown University School of Public Health.

Acknowledgements

I would like to thank my thesis advisor, Dr. Alice Paul, for her invaluable guidance, support, and expertise. Her mentorship has been pivotal in shaping this thesis and my academic growth.

Abstract

Risk score models are simple scoring systems that map patient characteristics to the probability of an outcome occurring. These models are popular with clinicians because they are easy to memorize and can be quickly calculated by hand. Risk score models can be created by rounding the estimated coefficients from a logistic regression model, though rounding can reduce the performance of the models. We introduce a new cyclical coordinate descent algorithm to estimate integer risk score models, expanding on recent work that has aimed to directly solve for the maximum likelihood with integer constraints. By offering an associated R package, we aim to foster wider accessibility and utilization in the medical research community. In a simulation study, our algorithm demonstrates comparable performance to the current state-of-the-art methods while being substantially more efficient. Further, we highlight our method with two applications in tuberculosis (TB) research. First, we develop a risk score model for TB diagnosis in sub-Saharan Africa that shows higher validation AUC than previous rounding methods. Second, we develop a novel model for TB treatment non-adherence of adolescents in Peru. Our risk score model identifies key characteristics influencing non-adherence, aligning with previous qualitative research findings. This study showcases the effectiveness and efficiency of our algorithm in constructing integer risk score models.

Contents

1 Introduction

Machine learning is increasingly implemented in healthcare to make predictions about patient outcomes. However, black box machine learning models do not explain how variables are being used to make a prediction. In a domain such as medicine where accurate prediction can directly influence treatment plans and outcomes, the ability to understand and trust the reasoning behind a model's predictions is critical. Transparent models allow clinicians to validate, refine, and make informed choices that are aligned with their expertise. Risk score models are one way to generate predictions that contain transparent and interpretable reasoning [1].

Risk scores are sparse linear models that map an integer linear combination of covariates to a probability of an outcome occurring. They are a popular predictive model in healthcare because they are easy to use and interpret, allowing clinicians to calculate a patient's risk for a given outcome by hand. Risk score models are typically sparse and involve small integer coefficients and dichotomous covariates. These characteristics are important because they allow risk score predictions to be easily computed by adding or subtracting a few small numbers. Additionally, these characteristics result in a transparent and interpretable model that clearly defines the individual effect of each variable on the risk score prediction.

Many common risk score models in healthcare were developed by first identifying predictors through regression analysis and then manually assigning point scores to each predictor using expert knowledge and the relative coefficient values (e.g. TIMI score for the risk of heart-related mortality [2]; HAS-BLED score for the risk for increased bleeding [3]; qSOFA score for the risk of sepsis [4]; and the CHA2DS2-VASc score for stroke risk for patients with atrial fibrillation [5]). As an example, the HAS-BLED risk score model is presented in Table 1. While these simple models allow clinicians to quickly calculate risk at the bedside, they were not developed using a standardized or optimized method,

making them difficult to reproduce in different contexts or update with new information. Rudin et al. [1] consider the optimization of integer risk score models as one of the top ten challenges in interpretable machine learning.

Table 1: The HAS-BLED risk score model estimating the 1-year risk of major bleeding in patients with atrial fibrillation.

	Score
H: Hypertension	+1
A: Abnormal liver or renal function	+1 each
S: Stroke history	+1
B: Bleeding tendency	+1
L: Labile INRs	+1
E: Elderly (> 65 yrs)	+1
D: Drugs or alcohol	+1 each
	Max 9 pts

Scaling and rounding coefficients is a common method used to convert logistic regression models into simple integer risk score models [6, 7]. However, scaling and rounding regression coefficients results in notable decreases in model performance [8]. In contrast to rounding logistic regression models, optimization approaches can directly include an integer constraint to find the maximum likelihood integer risk model among all possible integer solutions. Optimization approaches can also include penalties that encourage sparsity, another important characteristic of risk score models.

Ustun and Rudin [9] introduce an optimization approach to learn risk scores by solving a mixed-integer nonlinear program (MINLP). This model, called RiskSLIM, ensures rank accuracy by minimizing the logistic loss function. At the same time, the model promotes sparsity by incorporating a penalty for the L_0-norm, effectively reducing the number of features in the final model. Additionally, the model constrains coefficient values to small integers by including these constraints in the MINLP formulation, which is solved using a cutting plane algorithm. They demonstrate that the RiskSLIM method outperforms penalized logistic regression, naive rounding (rounding each penalized logistic regression

coefficient to the nearest integer), and rescaled rounding (applying rescaling to coefficients before rounding) [9].

Liu et al. [10] expand upon the work of Ustun and Rudin [9] by introducing a scaling parameter to the optimization problem to expand the search space of possible solutions and introducing a new algorithm called FasterRisk to solve this optimization problem. In their experiments, FasterRisk outperforms RiskSLIM in both accuracy and speed. However, the authors note that the algorithm scales poorly with the number of covariates. As the current state-of-the-art method, FasterRisk uses a beam-search algorithm to identify a pool of continuous solutions with low logistic loss and then identifies a multiplier that maintains low logistic loss after rounding. Rather than using an L_0-norm penalty, FasterRisk ensures sparsity by constraining the number of nonzero coefficients to a value set by the user [10].

The development of risk score models is related to another body of research on sparse integer classifiers which seek to directly classify points based on an integer combination of features [11–18]. In another approach, Xie et al. [19] and Li et al. [20] use machine learning variable importance measures to do variable selection before using a rounded logistic regression model including the selected covariates to create an integer risk score model. A key limitation of these methods is that they do not predict a corresponding risk probability and don't quantify the uncertainty in the estimates.

Although both RiskSLIM and FasterRisk models have improved performance over rounding methods, both algorithms are still relatively slow, especially with a larger number of candidate variables. Further, both methods require Python and RiskSLIM requires the use of the commercial optimization software CPLEX. We introduce a new method, called RiskCD, that uses cyclical coordinate descent to minimize logistic loss under regularization and integer constraints. Coordinate descent has been shown to be an effective and efficient optimization method for other sparse or regularized regression applications [21, 22]. Our method is available in an accessible R package to broaden the clinical audience. Further, in

our simulation study, our algorithm demonstrates comparable performance to FasterRisk with improved computational efficiency.

Importantly, we also apply our method to two applications related to tuberculosis (TB) diagnosis and treatment to highlight how directly solving for an integer risk score model yields better calibration and discrimination compared to rounding. This adds to past evidence that rounding methods limit model performance [8, 9].

In Section 2, we introduce the risk score optimization problem and our coordinate descent algorithm RiskCD. In Section 3, we demonstrate how to use the associated **riskscores** package to run the RiskCD algorithm. In Section 4, we present our simulation study and results to demonstrate the efficacy and efficiency of our method. In Section 5, we compare RiskCD to existing methods using publicly available datasets. In Section 6 we develop a risk score model for TB diagnosis in sub-Saharan Africa that shows higher validation AUC than previous rounding methods [23]. In Section 7 we develop a novel model for TB treatment non-adherence among adolescents in Peru. Last, in Section 8 we discuss limitations to the current work and possible extensions.

2 Methods

2.1 Risk Score Objective Function

We consider a regression setting with a binary outcome and p covariates. Suppose that we have data $(\boldsymbol{x}^i, y_i)$ for observations $i = 1, 2, \ldots, n$ where $\boldsymbol{x}^i = (1, x_1^i, x_2^i, \ldots, x_p^i)$ is a vector consisting of an intercept term plus p covariates and $y_i \in \{0, 1\}$ is a binary indicator for whether or not each observation experienced the outcome. We consider setting coefficients $\boldsymbol{\beta} = [\beta_0, \beta_1, \ldots, \beta_p] \in \mathbb{R}^{p+1}$ and scalar $\gamma \in \mathbb{R}$ such that the estimated probability for the outcome is given by

$$\Pr(y = 1|\boldsymbol{x}) = \frac{\exp\big(\gamma \boldsymbol{\beta}^T \boldsymbol{x}\big)}{1 + \exp(\gamma \boldsymbol{\beta}^T \boldsymbol{x})}. \tag{1}$$

For $\gamma = 1$, this corresponds to standard logistic regression. However, we add further restrictions to the $\boldsymbol{\beta}$ coefficients. In particular, we restrict β_j to be integer-valued and within the range $[l_j, u_j]$. The limits $\boldsymbol{l}$ and $\boldsymbol{u}$ allow the user to ensure that the scores remain easy to calculate by hand (reasonable limits for the integer coefficients might be $[-10, 10]$ or $[-5, 5]$). Our goal is to set $\boldsymbol{\beta}$ and γ to maximize the likelihood of the observed data, equivalent to minimizing the negative log-likelihood. This optimization problem is given in Equation 2.

$$\begin{aligned} \min_{\gamma,\beta} \quad & -\frac{1}{n}\sum_{i=1}^{n}\left(y_i \log\left(\frac{\exp\left(\gamma\boldsymbol{\beta}^T\boldsymbol{x}^i\right)}{1+\exp(\gamma\boldsymbol{\beta}^T\boldsymbol{x}^i)}\right) + (1-y_i)\log\left(\frac{1}{1+\exp(\gamma\boldsymbol{\beta}^T\boldsymbol{x}^i)}\right)\right) \\ \text{s.t.} \quad & l_j \leq \beta_j \leq u_j \quad \forall j = 1, 2, \ldots, p \\ & \beta_j \in \mathbb{Z} \quad \forall j = 1, 2, \ldots, p \\ & \beta_0, \gamma \in \mathbb{R} \end{aligned} \tag{2}$$

The scale parameter $\gamma \in \mathbb{R}$ in the optimization problem rescales the linear term which is restricted by the constraints on $\boldsymbol{\beta}$. That is, γ maps the parameter space of the linear term from integers between $\boldsymbol{l}$ and $\boldsymbol{u}$ to all real numbers. We can rewrite the optimization problem to highlight the underlying risk scores. We let $z_i = \sum_{j=1}^{p} \beta_j x_j^i$ be the risk score for observation i. Since $\beta_j \in \mathbb{Z}$, each score is an integer combination of the covariates. Given these scores, we can rewrite this optimization problem as in Equation 3. This shows that the risk score optimization problem can be written as a simple logistic regression problem on the estimated risk scores with corresponding coefficients $\gamma\beta_0$ and γ. While the risk score itself must be an integer combination of covariates, we allow for a non-integer mapping of these coefficients to the estimated probabilities using a logit link function.

$$
\begin{aligned}
\min_{\gamma,\beta} \quad & -\frac{1}{n}\sum_{i=1}^{n}\left(y_i \log\left(\frac{\exp(\gamma\beta_0+\gamma z_i)}{1+\exp(\gamma\beta_0+\gamma z_i)}\right)+(1-y_i)\log\left(\frac{1}{1+\exp(\gamma\beta_0+\gamma z_i)}\right)\right) \\
\text{s.t.} \quad & z_i = \sum_{j=1}^{p}\beta_j x_j^i \quad \forall i = 1,2,\ldots,n \\
& l_j \le \beta_j \le u_j \quad \forall j = 1,2,\ldots,p \\
& \beta_j \in \mathbb{Z} \quad \forall j = 1,2,\ldots,p \\
& \beta_0, \gamma \in \mathbb{R}
\end{aligned}
\tag{3}
$$

Last, we expand upon the optimization problem above by adding an optional regularization penalty to further improve interpretability and reduce potential overfitting. In the optimization in Equation 4, we include an L_0 penalty term on the number of included covariates but the algorithm presented in Section 2.2 easily extends to include L_1 or L_2 norm penalties. The penalty coefficient λ_0 controls the amount of regularization – a larger value of λ_0 will result in fewer non-zero coefficients. We call the optimization problem in Equation 4 the Risk Score Optimization problem (RISK-OPT).

$$
\begin{aligned}
\min_{\gamma,\beta} \quad & -\frac{1}{n}\sum_{i=1}^{n}\left(y_i \log\left(\frac{\exp\left(\gamma\boldsymbol{\beta}^T\boldsymbol{x}^i\right)}{1+\exp(\gamma\boldsymbol{\beta}^T\boldsymbol{x}^i)}\right)+(1-y_i)\log\left(\frac{1}{1+\exp(\gamma\boldsymbol{\beta}^T\boldsymbol{x}^i)}\right)\right)+\lambda_0\sum_{j=1}^{p}\beta_j \\
\text{s.t.} \quad & l_j \le \beta_j \le u_j \quad \forall j = 1,2,\ldots,p \\
& \beta_j \in \mathbb{Z} \quad \forall j = 1,2,\ldots,p \\
& \beta_0, \gamma \in \mathbb{R}
\end{aligned}
\tag{4}
$$

2.2 Cyclical Coordinate Descent Algorithm

We now introduce a heuristic algorithm to solve RISK-OPT. The optimization problem in Equation 4 is a mixed integer nonlinear optimization problem. The integer constraints on β_j make the risk score optimization problem difficult to solve directly. Coordinate descent algorithms are a popular derivative-free optimization method for NP-hard optimization problems [24]. This type of algorithm has proved efficient in practice and allows for penalty and integer constraints, outperforming more complex methods in other sparse or regularized regression applications [21, 22].

In each iteration of coordinate descent, we have a current solution γ and $\boldsymbol{\beta}$ and consider fixing all values except for a single β_j where $j \in \{1, 2, \ldots, p\}$. We then solve the reduced optimization problem of finding the optimal value of β_j to maximize the objective function. If we ignore the penalty term for the number of non-zero coefficients, minimizing the negative log-likelihood is a convex optimization problem with a single constrained integer variable. To exploit this structure, we use bisection search to find the optimal value, $\hat{\beta}_j$. We then use the full objective function with the penalty term to compare setting $\beta_j = \hat{\beta}_j$ with setting $\beta_j = 0$. Given the binary nature of the penalty, the optimal of the two is the optimal value for β_j. These steps are outlined in Algorithm 1.

We repeat this process over all β_j until convergence, updating γ and β_0 between each step by running a simple logistic regression model on the current estimated risk scores. The RiskCD algorithm is summarized in Algorithm 2.

Note that Algorithm 2 requires an initial starting solution. To find this starting solution, we relax our constraints and find an initial solution using logistic regression. Let $\boldsymbol{\beta}^{LR}$ be the optimal coefficients found when we set $\gamma = 1$ and remove all constraints on $\boldsymbol{\beta}$. We then set the scalar

$$\gamma = \min_{j=1,2,\ldots,p} \frac{|\beta_j^{LR}|}{1(\beta_j^{LR} < 0) \cdot |l_j| + 1(\beta_j^{LR} \geq 0) \cdot |u_j| + 0.5}$$

Algorithm 1 Bisection Search for Optimization of β_j

Require: Numeric data $(\boldsymbol{x}^i, y_i), i = 1, 2, \ldots, n$ where $y_i \in \{0, 1\}$.
Require: Penalty parameter $\lambda_0 \geq 0$.
Require: Current solution γ, $\boldsymbol{\beta}$.
Require: Integer bounds l, u.
Require: Index value j, where $j \in \{1, 2, \ldots p\}$.

Define $f(\boldsymbol{\beta}) = \sum_{i=1}^{n} \left(y_i \log \left(\frac{\exp(\gamma \boldsymbol{\beta}^T \boldsymbol{x}^i)}{1+\exp(\gamma \boldsymbol{\beta}^T \boldsymbol{x}^i)} \right) + (1 - y_i) \log \left(\frac{1}{1+\exp(\gamma \boldsymbol{\beta}^T \boldsymbol{x}^i)} \right) \right)$
Define $g(\boldsymbol{\beta}) = f(\boldsymbol{\beta}) + \lambda_0 \sum_{j=1}^{p} \beta_j$
while $(u - l) > 1$ **do**
  Set $m \leftarrow \frac{l+u}{2}$.
  Set $\boldsymbol{\beta}^l \leftarrow \boldsymbol{\beta}$ where $\beta_j = l$.
  Set $\boldsymbol{\beta}^u \leftarrow \boldsymbol{\beta}$ where $\beta_j = u$.
  Set $\boldsymbol{\beta}^m \leftarrow \boldsymbol{\beta}$ where $\beta_j = m$.
  if $\frac{\partial}{\partial \beta_j} f(\boldsymbol{\beta}^m) = 0$ **then**
    Set $l \leftarrow m$
    Set $u \leftarrow m$
  else if $\frac{\partial}{\partial \beta_j} f(\boldsymbol{\beta}^m)$ has the same sign as $\frac{\partial}{\partial \beta_j} f(\boldsymbol{\beta}^l)$ **then**
    Set $l \leftarrow m$
  else if $\frac{\partial}{\partial \beta_j} f(\boldsymbol{\beta}^m)$ has the same sign as $\frac{\partial}{\partial \beta_j} f(\boldsymbol{\beta}^u)$ **then**
    Set $u \leftarrow m$
  end if
end while
Set $\boldsymbol{\beta}^l \leftarrow \boldsymbol{\beta}$ where $\beta_j = l$.
Set $\boldsymbol{\beta}^u \leftarrow \boldsymbol{\beta}$ where $\beta_j = u$.
Set $\boldsymbol{\beta}^0 \leftarrow \boldsymbol{\beta}$ where $\beta_j = 0$.
if $g(\boldsymbol{\beta}^0) \leq g(\boldsymbol{\beta}^l)$ and $g(\boldsymbol{\beta}^0) \leq g(\boldsymbol{\beta}^u)$ **then**
  Return $\hat{\beta}_j = 0$.
else if $g(\boldsymbol{\beta}^l) \leq g(\boldsymbol{\beta}^u)$ **then**
  Return $\hat{\beta}_j = l$.
else if $g(\boldsymbol{\beta}^u) \leq g(\boldsymbol{\beta}^l)$ **then**
  Return $\hat{\beta}_j = u$.
end if

Algorithm 2 Cyclical Coordinate Descent for Risk Score Optimization (RiskCD)

Require: Numeric data $(\boldsymbol{x}^i, y_i), i = 1, 2, \ldots, n$ where $y_i \in \{0, 1\}$.
Require: Penalty parameter $\lambda_0 \geq 0$.
Require: Initial solution γ, $\boldsymbol{\beta}$.
Require: Maximum iterations maxiter $\in \mathbb{Z}$.
  Shuffle the indices $\{1, 2, \ldots, p\}$ to obtain a random permutation P.
  for Iteration it $= 1, 2, \ldots,$ maxiter **do**
    Set $\boldsymbol{\beta}_{\text{old}} \leftarrow \boldsymbol{\beta}$.
    for $j \in P$ **do**
      Find optimal β_j fixing all other variables using bisection search in the range $[l_j, u_j]$.
      Calculate current risk scores $\boldsymbol{z}$.
      Update γ and β_0 using logistic regression of $\boldsymbol{y}$ on current risk scores $\boldsymbol{z}$
    end for
    if $\boldsymbol{\beta}_{\text{old}} = \boldsymbol{\beta}$ **then**
      Break.
    end if
  end for

to ensure the the coefficients will be between $\boldsymbol{l}$ and $\boldsymbol{u}$. Then, we convert $\boldsymbol{\beta}^{LR}$ to a solution satisfying the bounded integer constraints by multiplying the logistic regression coefficients by γ and rounding to the nearest integer. The resulting $\boldsymbol{\beta}$ satisfies the bound and integer constraints and can be used as the initial solution for the RiskCD algorithm.

$$\beta_j = \begin{cases} \beta_j^{LR}/\gamma & j = 0 \\ \text{round}(\beta_j^{LR}/\gamma) & \text{otherwise} \end{cases} \tag{5}$$

In our corresponding R package, we also include a function to implement cross-validation to tune λ_0. Additionally, our implementation offers the ability to run the cross-validation folds in parallel given a parallel environment. To expand the solution space explored, we also offer an option to initialize RiskCD with a random solution rather than the rounded logistic regression solution. Using this option, the user specifies a number of random starts *nstart* and the algorithm is run *nstart* times, each starting from a random $\boldsymbol{\beta}$ vector generated by sampling values in $\{-1, 0, 1\}$. We then return the solution that minimizes

RISK-OPT.

3 Package Demonstration

In this section, we demonstrate how the RiskCD algorithm is run using the **riskscores** package in R. In this example, we develop a risk score model that predicts whether a breast tissue sample is malignant using features recorded during a biopsy. We use the "breastcancer" dataset, originally accessed from the UCI Repository [25], which can be loaded directly from the **riskscores** package.

```
library(riskscores)
data("breastcancer")
```

This dataset contains 683 observations and 9 covariates. Our goal is to develop a risk score model that predicts whether a breast tissue sample is benign using nine (or fewer) features recorded during a biopsy: clump thickness, uniformity of cell size, uniformity of cell shape, marginal adhesion, single epithelial cell size, bare nuclei, bland chromatin, normal nucleoli, and mitoses. Each covariate is integer-valued and ranges from 1 to 10.

The dataset needs to be split into a design matrix with the covariates (X) and a vector with the outcome data (y). The first column in this dataset contains the outcome variable.

```
X <- as.matrix(breastcancer[,-1])
y <- breastcancer[,1]
```

3.1 Cross-Validation

We use cross-validation to find a λ_0 value that minimizes the model deviance. Ideally, each cross-validation fold should contain an approximately equal proportion of cases. The **riskscores** package contains the function `stratify_folds()` that creates fold IDs with an equal proportion of cases in each fold.

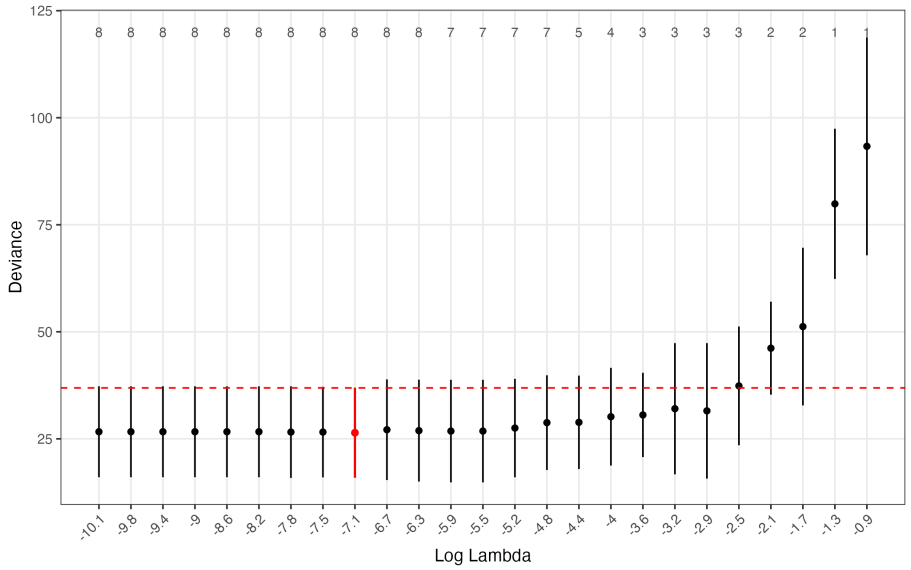
```
foldids <- stratify_folds(y, nfolds = 5, seed = 1)
```

The `cv_risk_mod()` function runs cross validation for a grid of possible λ_0 values. If the user does not specify the vector of λ_0 values to test, the program constructs this λ_0 sequence. The maximum λ_0 in this sequence is the smallest value such that all coefficients in the logistic regression model are zero. The minimum λ_0 in the sequence is calculated using the user-defined `lambda_ratio` argument. If $n > p$, the default `lambda_ratio` value is 0.0001, close to zero. If $n < p$, the default is 0.01. The λ_0 grid is created by generating `nlambda` values linear on the log scale from the minimum λ_0 to the maximum λ_0. Here, we set `nlambda` to 25, so the program constructs an appropriate sequence of 25 λ_0 values to test using cross validation. The fold IDs that we generated above can be entered into the `cv_risk_mod()` function under the `foldids` parameter. Otherwise, `cv_risk_mod()` will set random fold IDs.

```
cv_results <- cv_risk_mod(X, y, foldids = foldids, nlambda = 25)
```

Running `plot()` on a `cv_risk_mod` object creates a plot with the mean deviance for each λ_0 value in the grid. The number of nonzero coefficients that are produced by each λ_0 value when fit on the full data are listed at the top of the plot. The λ_0 value with the lowest mean deviance ("lambda_min") is indicated in red, and its standard deviation is marked with a red dashed line. Its precise value can be accessed by calling `cv_results$lambda_min`. If we want a sparser model, we could increase λ_0 to "lambda_1se", the largest value whose mean deviance is within one standard error of "lambda_min". This value can be accessed by calling `cv_results$lambda_1se`. In our example, "lambda_min" creates a model with 8 non-zero coefficients and "lambda_1se" creates a model with 3 non-zero coefficients. To view a dataframe with the full cross-validation results (including both deviance and accuracy metrics), run `cv_results$results`.

```
plot(cv_results)
```




```
cv_results$lambda_min
```

```
[1] 0.0008453613
```

```
cv_results$lambda_1se
```

```
[1] 0.0575938
```

3.2 Fitting a RiskCD Model

After running cross-validation, we fit a risk score model on the full data using the function `risk_mod()`. We use the "lambda_1se" value determined by cross-validation as the λ_0 parameter.

```
mod <- risk_mod(X, y, lambda0 = cv_results$lambda_1se)
```

The integer risk score model can be viewed by calling `mod$model_card`. An individual's risk score can be calculated by multiplying each covariate response by its respective number

of points and then adding all points together. In our example below, a patient with a ClumpThickness value of 1, a BareNuclei value of 5, and a BlandChromatin value of 10 would receive a score of $10(1) + 7(5) + 8(10) = 125$.

```
mod$model_card
               Points
ClumpThickness     10
BareNuclei          7
BlandChromatin      8
```

Each score can then be mapped to a risk probability. The `mod$score_map` dataframe maps an integer range of scores to their associated risk. For this example dataset, `mod$score_map` includes a range of integer scores from 25 to 200, which are the minimum and maximum scores predicted from the training data. We can see that a patient who received a score of 125 would have a 77.9% risk of their tissue sample being malignant.

```
mod$score_map
  Score   Risk
1    25 0.0006
2    50 0.0054
3    75 0.0446
4   100 0.2886
5   125 0.7788
6   150 0.9683
7   175 0.9962
8   200 0.9996
```

The function `get_risk()` can be used to calculate the risk from a given score (or a vector of scores). Likewise, the function `get_score()` calculates the score associated with a given risk (or vector of risk probabilities).

```
get_risk(mod, score = 125)
```

```
[1] 0.7787763
```

```
get_score(mod, risk = 0.7788)
```

```
[1] 124.9976
```

The function `get_metrics()` returns the accuracy, sensitivity, and specificity of the risk score model under different thresholds. The user can input either probability thresholds or score thresholds. Here, we evaluate score thresholds between 100 and 120 above which samples would be predicted malignant. This dataset has a strong relationship between the predictors and the outcome and the resulting model makes predictions close to 0 and 1. Therefore, we don't observe large changes in the sensitivity and specificity as the threshold changes.

```
get_metrics(mod, threshold = seq(100, 120, 5), threshold_type = "score")
```

```
  threshold_risk threshold_score  accuracy sensitivity specificity
1          0.289             100 0.9648609   0.9665272   0.9639640
2          0.385             105 0.9648609   0.9539749   0.9707207
3          0.491             110 0.9648609   0.9497908   0.9729730
4          0.597             115 0.9619327   0.9372385   0.9752252
5          0.696             120 0.9604685   0.9205021   0.9819820
```

3.3 Applying Generic Functions to RiskCD

Many generic functions that are used on `glm` objects can also be used on `risk_mod` objects, such as `summary()`, `coef()`, `predict()`, and `plot()`.

```
summary(mod)
```

```
Intercept: -110.4395
```

```
Non-zero coefficients:                .
ClumpThickness 10
BareNuclei      7
BlandChromatin  8

Gamma (multiplier):  0.08643598
Lambda (regularizer):  0.0575938

Deviance:  143.9915
AIC:  163.9915
```

```
coef(mod)
             (Intercept)           ClumpThickness
               -110.4395                  10.0000
    UniformityOfCellSize    UniformityOfCellShape
                  0.0000                   0.0000
        MarginalAdhesion SingleEpithelialCellSize
                  0.0000                   0.0000
              BareNuclei           BlandChromatin
                  7.0000                   8.0000
          NormalNucleoli                  Mitoses
                  0.0000                   0.0000
```

We can map our integer score model to an equivalent logistic regression model by multiplying the integer and coefficients by γ (saved as `$gamma` in the `risk_mod` object).

```
mod$beta * mod$gamma
             (Intercept)           ClumpThickness
              -9.5459487                0.8643598
```

```
UniformityOfCellSize    UniformityOfCellShape
           0.0000000                0.0000000
MarginalAdhesion SingleEpithelialCellSize
           0.0000000                0.0000000
          BareNuclei           BlandChromatin
           0.6050519                0.6914879
       NormalNucleoli                  Mitoses
           0.0000000                0.0000000
```

Running `predict()` on a `risk_mod` object allows for three types of prediction, as the `type` parameter can be set to either "link", "response", or "score". These first two options are the same as when `predict()` is run on a logistic `glm` object. The added "score" option returns each subject's score, as calculated from the integer coefficients in the risk score model.

The table below compares the three possible prediction types for five example subjects. The first three columns contain data for clump thickness, bare nuclei, and bland chromatin, respectively.

Covariates			Prediction		
CT	BN	BC	'score'	'link'	'response'
5	1	3	81	-2.54	0.073
5	10	3	144	2.90	0.948
3	2	3	68	-3.67	0.025
6	4	3	112	0.13	0.534
4	1	3	71	-3.41	0.032

The "score" is a linear combination of the covariates and their integer coefficients:

$$\text{score} = 10(\text{CT}) + 7(\text{BN}) + 8(\text{BC})$$

The "link" is a linear combination of the covariates using the full logistic regression equation:

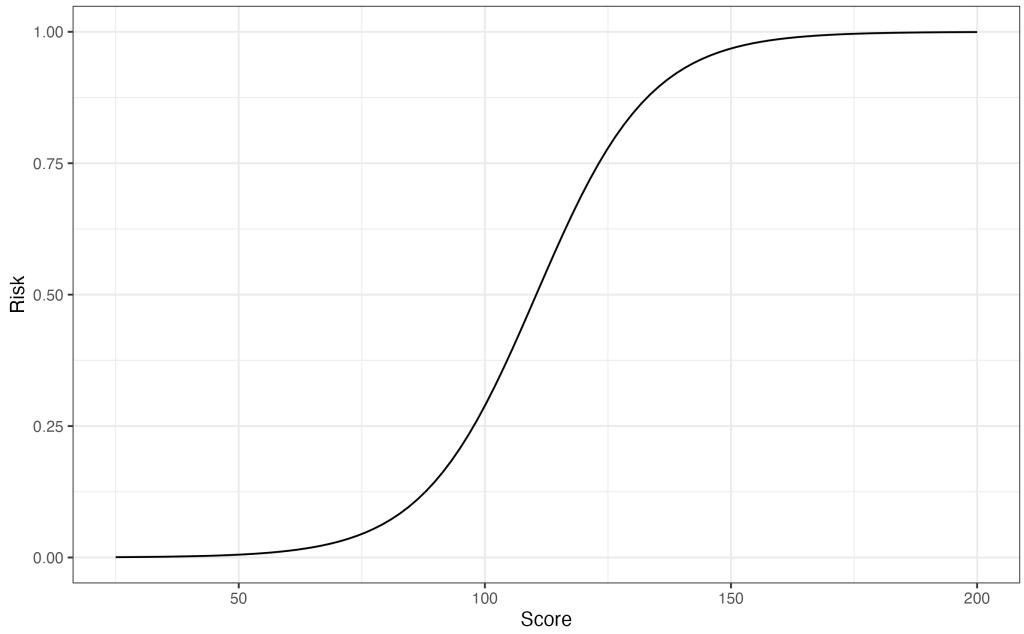
$$\text{link} = -9.54 + 0.86(\text{CT}) + 0.61(\text{BN}) + 0.69(\text{BC})$$

The "response" converts these link values to probabilities:

$$\text{response} = e^{\text{link}}/(1 + e^{\text{link}})$$

Finally, the relationship between scores and risk can be visualized by calling `plot()` on a `risk_mod` object.

```
plot(mod, score_min = 25, score_max = 200)
```



4 Simulation Study

4.1 Set Up

In this section, we evaluate the performance of our algorithm compared to rounding methods and the current state-of-the-art method FasterRisk [10]. We simulate data with different dimensions, proportion of covariates associated with the outcome, and signal-to-noise ratios. We also consider building models with and without regularization.

Suppose we have a fixed number of observations n, number of covariates p, proportion of noise predictors (PNP), and signal to noise ratio (SNR). We first generate n observations with p covariates by generating $\boldsymbol{x}^i$ where $x_j^i \sim \text{Bernoulli}(p_j)$ where $p_j \sim \text{Unif}(0.1, 0.9)$. We also generate test data with $10 \cdot p$ observations using the same data generating mechanism. Next, to generate the outcome, we select a subset S of $p' = \lceil (1 - \text{PNP}) \cdot p \rceil$ random columns to be associated with the outcome. Further, we randomly partition S into S_1 and S_2 such

that $|S_1| = \lfloor 0.75 \cdot p' \rfloor$. We then set coefficients

$$\beta_j = \begin{cases} \sim \text{Unif}(0.2, 1.5) & j \in S_1 \\ \sim \text{Unif}(2, 5) & j \in S_2 \\ = 0 & \text{otherwise} \end{cases} . \tag{6}$$

This ensures that the coefficients include mostly small to moderate effects with a few large effects, a scenario that impacts rounding methods.

Last, we generate y for both our training and validation data using β. Let $\upsilon = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{\beta}^T \boldsymbol{x}^i$. We generate y from a Bernoulli distribution

$$y \sim \text{Bernoulli}\left(p = \frac{\exp\big(-\upsilon + \boldsymbol{\beta}^T \boldsymbol{x}^i + \epsilon\big)}{1 + \exp(-\upsilon + \boldsymbol{\beta}^T \boldsymbol{x}^i + \epsilon)}\right), \tag{7}$$

where $\epsilon \sim N(0, \sigma^2)$ where $\sigma = \sqrt{\text{Var}(\boldsymbol{\beta}^T \boldsymbol{x})/\text{SNR})}$. This term controls the signal-to-noise ratio of the linear predictor. A lower value for SNR means that there is less signal to learn from. The offset term υ ensures that the signal is centered.

We simulate data with the number of observations n in the training set ranging from 100 to 5000, the number of candidate predictors p ranging from 10 to 50, the proportion of noise predictors PNP ranging from 0% to 50%, and the signal-to-noise ratio SNR ranging from 1 to 3 (48 unique scenarios, Table 2). Each scenario is simulated 10 times, resulting in 480 total datasets.

Table 2: Values for each simulation parameter. For all possible combinations of parameters, we generate 10 datasets.

Number of training observations (n)	100, 500, 1000, 5000
Number of candidate predictors (p)	10, 25, 50
Proportion of noise predictors (PNP)	0.0, 0.5
Signal-to-noise ratio (SNR)	1, 3

For each simulated dataset, we compare several methods.

- **Logistic Regression** We fit a standard logistic regression model for y containing all p covariates. The resulting model does not generate an integer risk score model since the estimated coefficients are not guaranteed to be integer valued. However, since we generated the data using a logistic form, we include this model as an oracle.
- **Rounded Logistic Regression** Using the estimated coefficients from the logistic regression model $\hat{\beta}^{LR}$, we scale the coefficients by the maximum absolute value of $\hat{\beta}^{LR}$ divided by 10. We divide by 10 within the scalar to restrict the resulting coefficients to values between -10 and 10. We then create a risk score model by rounding the estimated coefficients

$$\text{round}\left(\frac{\hat{\beta}^{LR}}{\frac{1}{10}\max|\hat{\beta}^{LR}|}\right).$$

- **Rounded Lasso** We use the **glmnet** package [21] to fit a lasso regression model, tuning the L1 penalty term λ_1 using 5-fold cross-validation. We then round the estimated coefficients $\hat{\beta}^{L1}$ to obtain a risk score model

$$\text{round}\left(\frac{\hat{\beta}^{L1}}{\frac{1}{10}\max|\hat{\beta}^{L1}|}\right).$$

- **FasterRisk** [10] We run the FasterRisk algorithm with parameters with range $[-10, 10]$ and no sparsity constraint ($k = p$).
- **FasterRisk-CV** [10] We run the FasterRisk algorithm with range $[-10, 10]$, tuning the sparsity constraint k using 5-fold cross-validation. Note that cross-validation is not built into the FasterRisk package. We do not report results when $p = 50$ and $n \geq 500$ because cross-validation timed out at this data size ($\geq$ 30 minutes per dataset).

- **RiskCD** We run our cyclical coordinate descent algorithm with $\lambda_0 = 0$ and range $[-10, 10]$ for all β's using our rounded starting solution.
- **RiskCD-CV** We run our cyclical coordinate descent algorithm, tuning the L0 penalty term λ_0 using 5-fold cross-validation to estimate the test deviance.

Each method was evaluated using the AUC on the test dataset, computation time (seconds), and the number of non-zero coefficients. Out of the 40 datasets in each unique scenario of n and p, we calculate the percentage that each method achieved the highest test AUC ("% Best"). The experiments were run using R on a Apple 2020 13.3" Macbook Air, M1 chip: 8GB unified memory; 8-core CPU (4 performance, 4 efficiency); 7-core GPU; 16-core Neural Engine.

4.2 Results

Table 3 reports the average performance metrics for all values of n and p for risk score models that do not use cross-validation to tune regularization parameters (rounded logistic regression, FasterRisk with $k = p$, and RiskCD with $\lambda_0 = 0$). We observe no patterns in which method performed best across different values for PNP and SNR. All methods have similar test AUCs across all simulation scenarios and no method consistently outperforms the others. For $n = 100$, we observe the best performance for FasterRisk. However, we also observe higher average test AUC compared to logistic regression, indicating a possible need for regularization.

FasterRisk is less efficient than rounded logistic regression and RiskCD. Figure 1 plots the average computation time of FasterRisk and RiskCD in seconds. RiskCD is considerably faster than FasterRisk and the computation time did not substantially increase as n and p increased as it did for FasterRisk. This indiciates that RiskCD scales better as the data size increases.

Table 4 compares the performance of the methods that use cross-validation to tune

Table 3: Performance of each risk score model without regularization across different data dimensions n and p. Each row corresponds to the average across 40 datasets, 10 datasets for each combination of the proportion of noise predictors (PNP) and signal-to-noise ratios (SNR). We report the average AUC, the average number of non-zero coefficients, and the percentage of instances on which each method achieved the highest validation AUC.

		Logistic Regression	Rounded Logistic Regression			RiskCD			FasterRisk		
n	p	AUC	AUC	# Nonzero	% Best	AUC	# Nonzero	% Best	AUC	# Nonzero	% Best
100	10	0.783	0.780	8.8	47.5	0.780	8.8	35.0	0.782	8.7	40.0
100	25	0.762	0.768	22.2	27.5	0.769	22.1	32.5	0.773	22.0	40.0
100	50	0.654	0.654	45.0	0.0	0.668	44.5	2.5	0.704	44.2	97.5
500	10	0.804	0.803	8.3	45.0	0.802	8.2	40.0	0.803	8.1	52.5
500	25	0.827	0.825	21.1	37.5	0.824	21.1	27.5	0.826	20.9	45.0
500	50	0.828	0.826	41.9	37.5	0.827	41.4	37.5	0.826	41.6	30.0
1000	10	0.809	0.809	8.3	57.5	0.807	8.3	50.0	0.808	8.2	37.5
1000	25	0.845	0.844	20.6	35.0	0.844	20.3	25.0	0.845	20.3	57.5
1000	50	0.846	0.845	41.2	42.5	0.844	40.9	37.5	0.845	41.0	27.5
5000	10	0.816	0.814	7.3	57.5	0.813	7.2	45.0	0.814	7.2	52.5
5000	25	0.848	0.847	19.1	45.0	0.846	19.0	32.5	0.847	19.0	40.0
5000	50	0.859	0.858	38.4	47.5	0.858	38.0	32.5	0.858	38.0	32.5

regularization parameters (rounded lasso, RiskCD-CV, and FasterRisk-CV). In this setting, RiskCD-CV typically has a slightly higher average test AUC than rounded lasso. FasterRisk typically has a higher average test AUC than RiskCD and is the better method for a higher percentage of simulations. However, FasterRisk-CV is notably less efficient than RiskCD-CV, even when the time spent in the cross-validation step is not considered. With large datasets ($p = 50$ and $n \geq 500$), running cross-validation with FasterRisk is not feasible, as it takes over 30 minutes to complete cross-validation on a single dataset. Average computation times across all integer methods are reported in Table 5. Overall, FasterRisk and FasterRisk-CV are considerably slower than rounding and RiskCD methods.

Last, Figure 2 visualizes how well each model is able to detect noise predictors (i.e. variables with true coefficients of zero). For simulated data with 50% noise predictors, all three regularized integer methods have similar accuracy. However, when the simulated data were generated with 0% noise predictors, the rounded lasso models are not able to detect

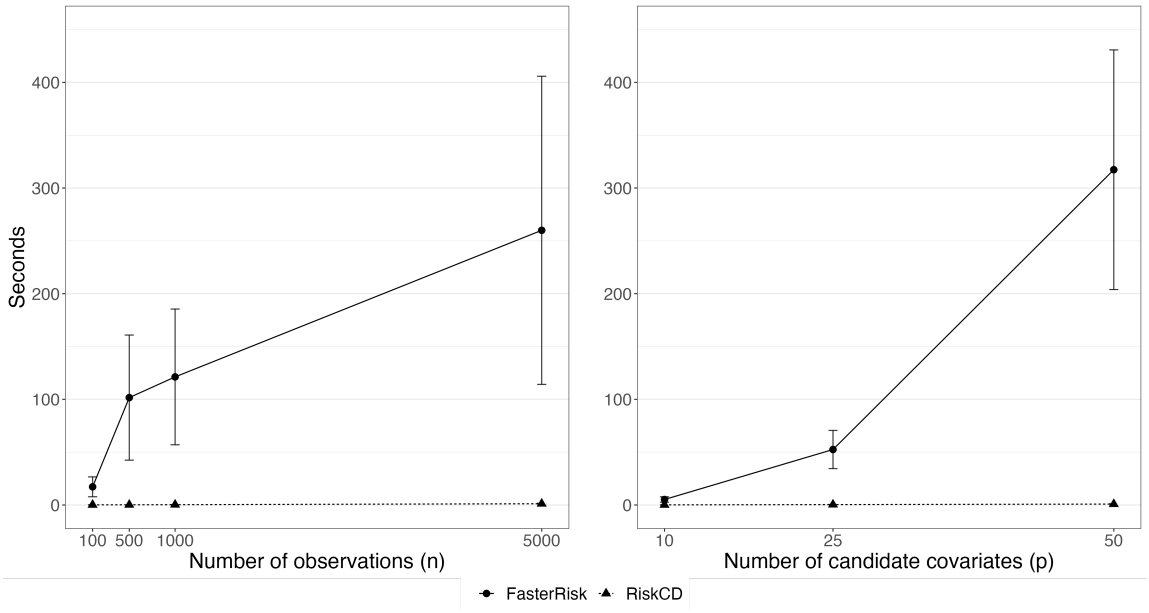


Figure 1: Comparison of FasterRisk and RiskCD computation time (seconds) across data dimensions n and p.

these true nonzero coefficients and underestimate the number of predictors. RiskCD-CV and FasterRisk-CV are better able to identify these nonzero coefficients, resulting in a higher accuracy.

Table 4: Performance of each risk score model with regularization across different data dimensions n and p. Each row corresponds to the average across 40 datasets, 10 datasets for each combination of the proportion of noise predictors (PNP) and signal-to-noise ratios (SNR). We report the average AUC, the average number of non-zero coefficients, and the percentage of instances on which each method achieved the highest validation AUC. For FasterRisk-CV, cross-validation for the sparsity parameter timed out for $p = 50$ and $n \geq 500$.

		Lasso	Rounded Lasso			RiskCD-CV			FasterRisk-CV		
n	p	AUC	AUC	# Nonzero	% Best	AUC	# Nonzero	% Best	AUC	# Nonzero	% Best
100	10	0.784	0.759	3.2	45.0	0.751	4.2	32.5	0.772	5.2	45.0
100	25	0.784	0.747	6.0	47.5	0.645	3.0	7.5	0.764	10.5	50.0
100	50	0.713	0.663	5.7	35.0	0.666	39.1	12.5	0.711	18.0	57.5
500	10	0.803	0.789	4.8	17.5	0.799	6.5	42.5	0.802	6.7	52.5
500	25	0.829	0.823	12.1	40.0	0.823	14.7	27.5	0.824	15.2	32.5
500	50	0.833	0.825	19.7	62.5	0.820	17.6	37.5	–	–	–
1000	10	0.808	0.803	5.4	27.5	0.807	7.2	35.0	0.809	7.2	47.5
1000	25	0.845	0.840	12.9	35.0	0.843	17.4	35.0	0.843	16.3	30.0
1000	50	0.847	0.842	24.0	45.0	0.844	29.4	55.0	–	–	–
5000	10	0.816	0.811	6.0	30.0	0.814	7.2	40.0	0.813	7.0	37.5
5000	25	0.848	0.845	15.9	25.0	0.846	18.5	30.0	0.847	17.9	47.5
5000	50	0.860	0.857	31.0	50.0	0.858	35.8	52.5	–	–	–

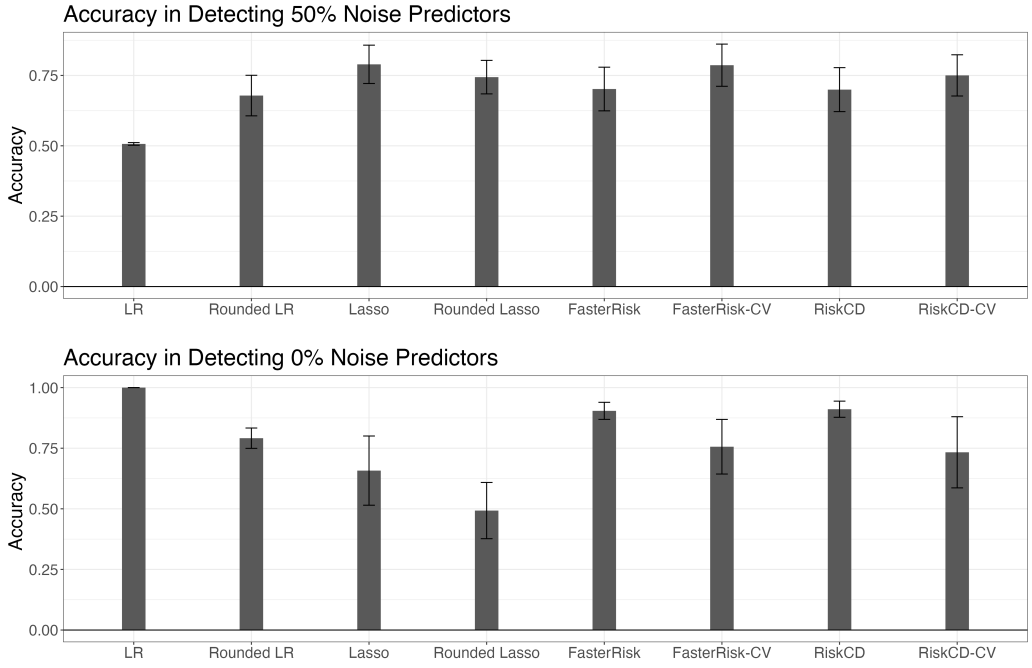


Figure 2: Model accuracy in detecting noise predictors when 50% of true coefficients were zero (top) and when 0% of true coefficients were zero (bottom). Accuracy was defined as the proportion of model coefficients that were correctly assigned a nonzero value plus the proportion of coefficients that were correctly assigned a zero value.

Table 5: Average computation time (in seconds) of the three integer non-regularized method and the three integer regularized methods. Each row corresponds to the average across 40 datasets, 10 datasets for each combinations of the proportion of noise predictors (PNP) and signal-to-noise ratios (SNR). For regularized methods, the times do not include cross-validation for parameter tuning. For FasterRisk-CV, cross-validation for the sparsity parameter timed out for $p = 50$ and $n \geq 500$.

		Non-Regularized			Regularized		
n	p	Rounded Logistic Regression	RiskCD	FasterRisk	Rounded Lasso	RiskCD-CV	FasterRisk-CV
100	10	0.00	0.03	0.76	0.04	0.04	0.18
100	25	0.00	0.07	8.06	0.11	0.13	2.52
100	50	0.01	0.28	42.76	0.09	0.65	12.15
500	10	0.00	0.05	1.20	0.04	0.08	0.37
500	25	0.01	0.17	37.97	0.07	0.19	18.41
500	50	0.01	0.32	265.77	0.11	0.52	–
1000	10	0.01	0.06	6.01	0.08	0.10	1.22
1000	25	0.01	0.19	58.32	0.14	0.29	65.46
1000	50	0.02	0.58	299.38	0.19	0.81	–
5000	10	0.02	0.34	12.83	0.33	0.48	17.15
5000	25	0.04	1.02	105.66	0.59	1.98	118.82
5000	50	0.14	2.23	661.44	1.05	5.30	–

5 Experiments with Test Bed Datasets

We evaluate the performance of the RiskCD method using publicly available datasets that were also used by the authors of RiskSLIM [9] and FasterRisk [10] (Table 6).

For each dataset, we develop risk score models using the following non-regularized integer risk score methods: (1) rounded logistic regression; (2) RiskCD ($\lambda_0 = 0$, $l = -10$, $u = 10$); (3) FasterRisk ($k = p$, $l = -10$, $u = 10$). Each dataset is split into a training set (70% of observations) and a test set (30% of observations). The training and test sets are split to include equal proportions of each outcome class.

Performance of each model is evaluated using the test AUC, the number of nonzero coefficients in the final model, and the computation time in seconds. Model performance results are reported in Table 7. Although FasterRisk has the highest test AUC (or tied

Table 6: Names, dimensions, and sources for the public datasets used for comparison. Available on the UCI Repository, accessed from `https://github.com/ustunb/risk-slim/tree/master/examples/data`.

Dataset	n	p	Event Fraction	Source
adult	32,561	36	24.1%	[26]
bank	41,188	57	11.3%	[27]
breastcancer	683	9	35.0%	[25]
compas	7,214	6	45.1%	[28]
mammo	961	14	46.3%	[29]
mushroom	8,124	113	48.2%	[30]
spambase	4,601	57	11.3%	[31]

for highest) for six of the seven datasets, it has a considerably slower computation time. Across the seven datasets, FasterRisk is 17 to 231 times slower than RiskCD.

Table 7: Non-regularized model performance on test bed datasets. Maximum point value refers the maximum integer coefficient in the model (absolute value). Maximum point values above 10 are marked in red. Computation times above 100 seconds are marked in red.

	Logistic Regression	Rounded Logistic Regression			RiskCD			FasterRisk		
Dataset	AUC	AUC	# Nonzero	Seconds	AUC	# Nonzero	Seconds	AUC	# Nonzero	Seconds
adult	0.889	0.877	23	0.80	0.874	21	22.25	0.890	25	1513.34
bank	0.781	0.675	9	1.91	0.769	30	65.16	0.773	28	3751.82
breastcancer	0.997	0.997	9	0.01	0.995	9	0.16	0.995	8	13.27
compas	0.680	0.681	6	0.03	0.682	6	0.12	0.682	6	2.05
mammo	0.874	0.873	10	0.01	0.881	13	0.59	0.883	10	15.96
mushroom	1.000	1.000	27	2.58	1.000	64	44.67	1.000	45	2569.79
spambase	0.764	0.935	44	0.42	0.968	43	3.53	0.968	35	817.18

6 Application to Tuberculosis Diagnosis in Sub-Saharan Africa

Now we reproduce an existing model that estimates risk of tuberculosis (TB) among symptomatic patients. Baik et al. [23] developed a risk score model using symptom, demographic, and identified risk factor data to estimate a patient's risk of having TB while awaiting microbiological results. Early diagnosis allows for earlier treatment and helps

prevent pretreatment loss to follow-up.

The given model was derived on data from rural South Africa and validated using data from urban Uganda. The data include only patients that presented with a classic TB symptom at the health clinic (cough, fever, night sweats, or weight loss). The derivation data include 1,407 participants, 702 of which tested positive for TB (49.9%). The validation data include 387 participants, 106 of which tested positive for TB (27.4%). Each dataset contains information typically collected at health clinics, including the patient's age group, sex, self-reported HIV status, diabetes status, past TB diagnoses, smoking status, education, number of TB symptoms, and length of time experiencing TB symptoms.

Baik et al. [23] develop their risk score model using rounded lasso regression coefficients. To round the lasso coefficients, they identify clusters of coefficients that have similar associations with the outcome, and then round each coefficient by the median value in the coefficient cluster. They also alter coefficients manually to increase usability (i.e. the number of TB symptoms is equivalent to the point score for that variable). The final coefficients are reported in Table 8 under Baik2020.

Table 8: Estimated risk score model coefficients predicting TB diagnosis including those reported in Baik et al. [23], our replication using a rounded lasso model, and the results of RiskCD-CV.

	Lasso	**Rounded Lasso**	**Baik2020**	**RiskCD-CV**
Age 15-24	0.000	0	0	0
Age 25-34	0.270	0	1	2
Age 35-44	0.045	0	1	1
Age 45-54	0.000	0	0	0
HIV-positive	0.766	1	2	2
Diabetes mellitus	0.000	0	1	0
Ever smoked	0.000	0	0	0
Previous TB diagnosis	0.000	0	0	0
Male	0.495	1	1	2
High school education or less	0.000	0	0	1
Duration of TB symptoms >2 weeks	0.668	1	1	2
Reports 1 TB symptom	0.000	0	1	0
Reports 2 TB symptoms	0.168	0	2	1
Reports 3 TB symptoms	1.214	2	3	4
Reports 4 TB symptoms	1.618	3	4	5

We reproduce the Baik2020 model and compare the performance to a rounded lasso regression model where each coefficient is divided by the median coefficient and rounded to the nearest integer, and a RiskCD-CV model using our algorithm with a logistic regression start and coefficient bounds between -5 and 5.

Coefficients of the resulting models are reported in Table 8. Although we followed the data pre-processing steps outlined by Baik et al. [23], our lasso regression coefficients differed than those reported by Baik et al. [23]. Thus, our rounded lasso model differs from the Baik2020 model. The RiskCD-CV model identifies many of the same covariates as the lasso model. However, while rounding the lasso model converts the smallest coefficients to zero, these variables are retained in the RiskCD-CV model. The RiskCD-CV model has higher AUC values than the other models in both the derivation and validation datasets.

RiskCD-CV achieves higher derivation and validation AUC values than lasso regression and the rounded methods (Table 9). The RiskCD-CV model has similar calibration as the Baik2020 model (Figure 3). In practice, Baik et al. [23] recommends adjusting for the sampling fraction of TB in a given population, which would counteract the underestimation observed for the validation set.

Table 9: Risk score model performance (AUC) on derivation and validation datasets for TB diagnosis.

	Derivation Data AUC (95% CI)	**Validation Data AUC (95% CI)**
Lasso	0.804 (0.782, 0.827)	0.743 (0.688, 0.798)
Rounded Lasso	0.789 (0.767, 0.812)	0.741 (0.686, 0.795)
Baik2020	0.799 (0.776, 0.822)	0.728 (0.673, 0.783)
RiskCD-CV	0.806 (0.784, 0.829)	0.756 (0.702, 0.810)

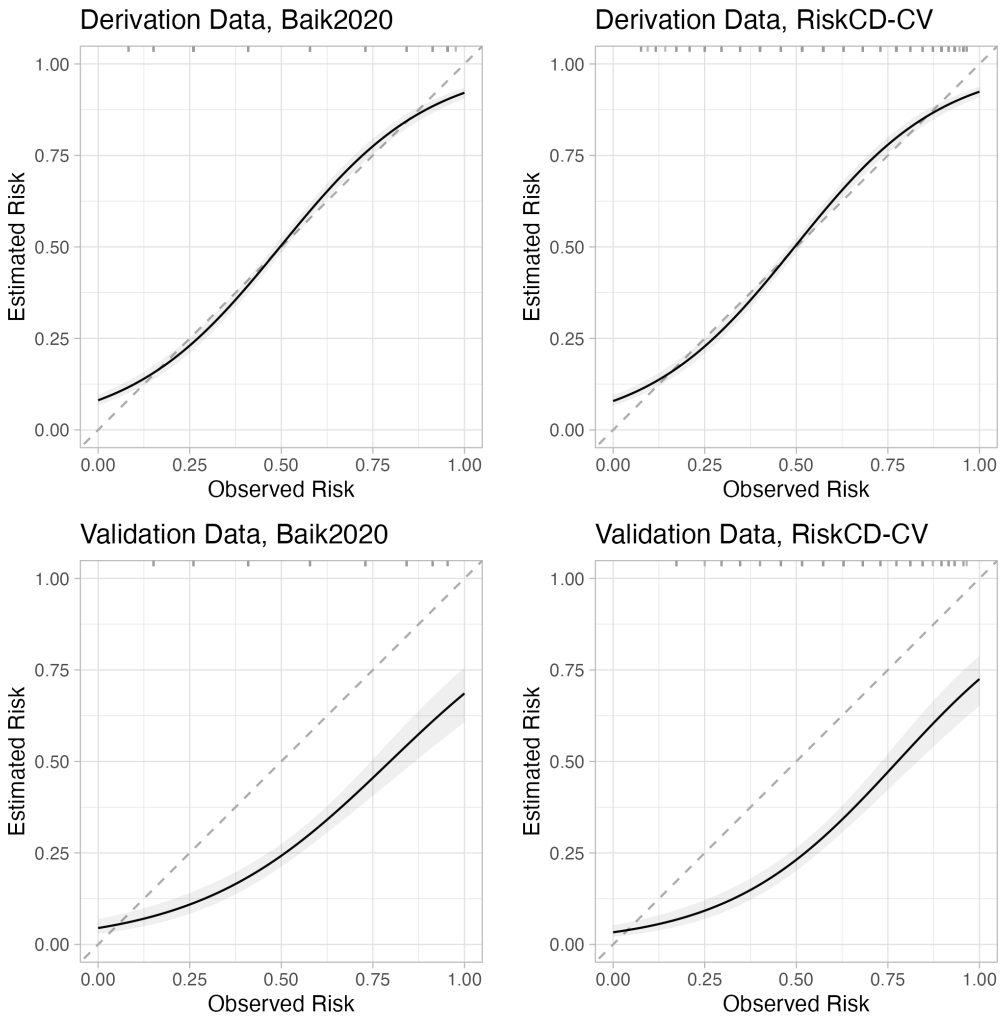


Figure 3: Baik2020 and RiskCD-CV model calibration on the derivation and validation datasets. Calibration is visualized by fitting a logistic regression model of the observed outcomes against the estimated risk. The 90% confidence intervals are also shown. If a model is well calibrated, the fitted curve should align with the diagonal line.

Through this example, we demonstrate that RiskCD-CV is a viable alternative to existing risk score methods that is standardized, simple to implement, and doesn't require manual manipulation of coefficients. This gives clinicians a more principled approach to producing risk score models.

7 Application to Tuberculosis Treatment Adherence in Peru

Next, we apply our method in a novel setting to estimate risk of non-adherence to treatment regimens among adolescents with TB in Peru.

7.1 Background

Despite advancements in TB diagnosis and treatment, medication adherence remains a critical challenge in TB management, especially among adolescents [32, 33]. Suboptimal treatment adherence leads to worse outcomes for those with TB, as well as increased transmission within their community. Understanding the determinants of adolescent medication adherence is critical to identifying patients requiring additional support. Prior quantitative research on TB treatment adherence has used surveillance data rather than psychosocial and clinical data [34]. Qualitative research suggests that family relationships, mental health, and type of treatment administration contribute to an individual's adherence [35, 36], suggesting that these data should be considered when assessing risk.

Chiang et al. [34] collected demographic and socioemotional data from 249 TB-positive adolescents aged 10-19 years in Lima, Peru. They applied k-means cluster analysis, grouping the participants into three clusters. They analyzed the characteristics of each cluster to identify factors that were associated with sub-optimal adherence. However, it may be difficult to directly apply these results to assess risk in new patients. An integer risk score model offers an alternative that is highly interpretable and usable. Clinic workers with a validated risk score model could quickly estimate a new patient's risk of sub-optimal adherence and respond appropriately.

7.2 Data Processing

We use the data collected by Chiang et al. [34] to develop risk score models to predict medication non-adherence. We exclude subjects with missing covariate data, leaving 210

participants. In total, 31 covariates are included as candidates, with continuous variables converted to categorical variables according to cutoffs based on knowledge of the measurement (e.g. using a cutoff of 10 for the depression score, the value that separates none/mild depression from moderate/severe depression) or at the midpoint of possible values (e.g. a cutoff of 12 for the self-efficacy score, which has a possible range of 4 to 20). The full list of candidate variables with descriptions is available in Appendix A.1. The outcome, medication adherence, was measured as the percentage of doses taken on time. Although using a cutoff of below 90% adherence to classify sub-optimal adherence would align with established TB medication research [37], only 17 participants in this study had medication adherence below 90%. Given the limited data size at this cutoff, we define a patient as "non-adherent" when their medication adherence is less than 95%. Using this cutoff, we observe 39 "non-adherent" participants (19%). Population summaries for candidate variables by adherence status are reported in Appendix A.2.

7.3 Rounded Lasso Model

To compare RiskCD to a popular integer risk score method, we first fit a rounded lasso model. Lasso regression, run according to the description in Section 4.1, results in seven nonzero coefficients among the 31 candidate covariates (Table 10). Living with a single dad, needing to take more pills, having a family that dislikes their friends, and anticipating stigma at receiving TB care are identified as features that increase risk of non-adherence. Past studies identify pill burden and TB stigma as barriers to TB medication adherence in adolescents [35, 36].

Family support, in-person directly observed therapy (DOT) with family supervision, and a higher health services score are identified as factors that decrease risk of non-adherence. Five types of treatment administration were measured: in-person DOT, family supervision only, in-person DOT with family supervision, no supervision, and virtual DOT with family supervision. Only in-person DOT with family supervision is selected in the

lasso regression model. Past studies identify family support and adherence support services as facilitators to TB medication adherence in children, adolescents, and young adults [35, 36]. We create the "Rounded Lasso" model in Table 10 according to the method described in Section 4.1.

Table 10: Non-zero coefficients from Lasso and Rounded Lasso models predicting TB medication non-adherence.

	Lasso	**Rounded Lasso**
Lives with single dad (0/1)	0.170	10
Number of pills (scale from 1 to 5)	0.009	1
Family dislikes friends (scale from 1 to 5)	0.023	1
Family support (scale from 1 to 5)	-0.023	-1
In-person DOT + family supervision (0/1)	-0.058	-3
Health services (scale from 1 to 5)	-0.001	0
Total stigma score > 30 (0/1)	0.014	1

7.4 RiskCD Model

The Rounded Lasso model does not optimize over all possible integer solutions. We present RiskCD as a method to convert the lasso coefficients to integer scores by optimizing over an objective function that includes integer constraints. We use the rounded lasso model as a "warm start", which is then tuned using cyclical coordinate descent. We parameterize RiskCD with bounds of [-10, 10] to keep the integer coefficients within a range that can be easily multiplied and memorized. The resulting RiskCD-CV integer coefficients are reported in Table 11. In this case, the RiskCD algorithm does not add any variables that aren't also selected by lasso regression, though it does shrink the health services and stigma coefficients to zero.

RiskCD's associated R package **riskscores** includes a function that maps all possible scores to the probability of the outcome occurring, which is reported for this model in Table 12. The combination of Table 11 and Table 12 is the information a clinic worker

would need to predict a patient's risk of non-adherence. Table 12 can also be expressed visually as in Figure 4.

Table 11: The RiskCD-CV risk score model estimating the risk of TB medication non-adherence among adolescents in Peru.

	Possible Responses	Score
Lives with single dad	0, 1	+4
Number of pills	1, 2, 3, 4, 5	+1
Family dislikes friends	1, 2, 3, 4, 5	+1
Family support	1, 2, 3, 4, 5	−1
In-person DOT + family supervision	0, 1	−10
		Min −13 pts
		Max 13 pts

Table 12: Predicted TB medication non-adherence risk per RiskCD score.

SCORE	≤ -1	0	1	2	3	4	5	6	7	8	9	10	11	≥ 12
RISK	≤ 0.071	0.105	0.154	0.218	0.300	0.398	0.504	0.610	0.706	0.787	0.851	0.898	0.931	≥ 0.954

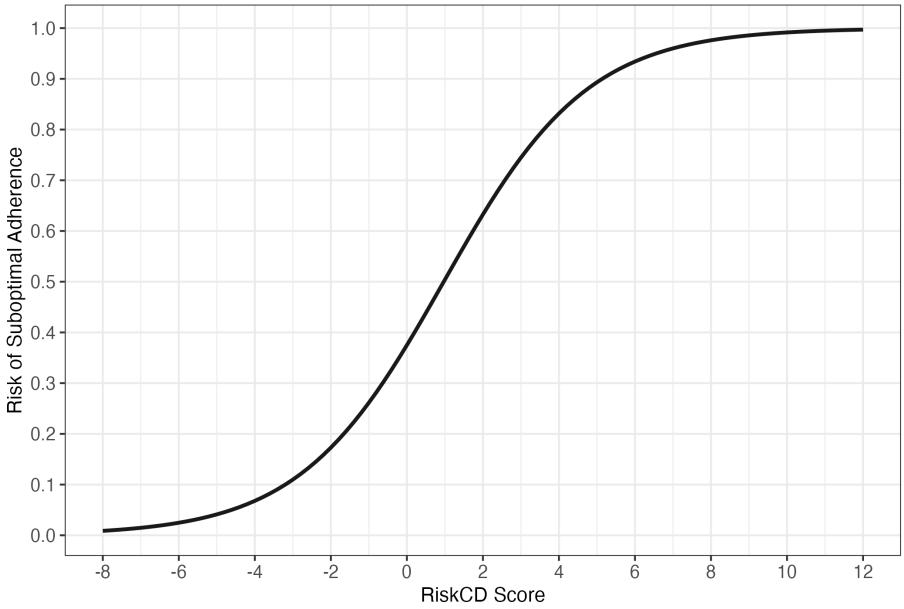


Figure 4: Relationship between RiskCD score and risk of TB treatment non-adherence. For each possible risk score, we also plot the number of observations in the derivation data with that estimated risk score and the proportion that experienced the outcome.

7.5 Model Comparison

The derivation AUC for each model is reported in Table 13. Along with having a slightly higher AUC, the RiskCD model is noticeably better calibrated with the derivation dataset than the rounded lasso model (Figure 5).

Table 13: Risk score model performance (AUC) on derivation dataset for TB treatment non-adherence.

	AUC	**95% CI**
Lasso	0.7613	0.6707, 0.8518
Rounded Lasso	0.7716	0.6849, 0.8583
RiskCD-CV	0.7782	0.6976, 0.8589

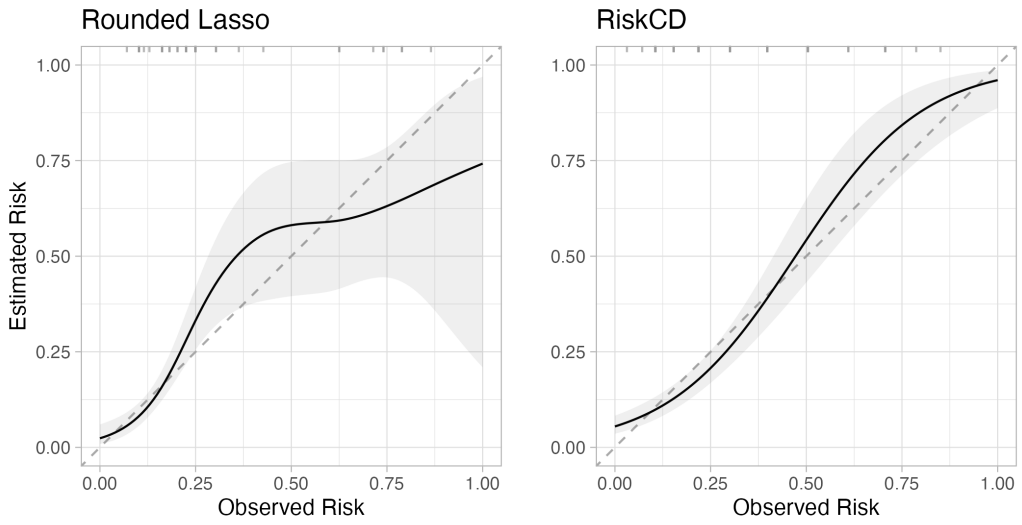


Figure 5: Rounded lasso and RiskCD calibration on derivation dataset for TB treatment non-adherence. Calibration is visualized by fitting a logistic regression model of the observed outcomes against the estimated risk. The 90% confidence intervals are also shown. If a model is well calibrated, the fitted curve should align with the diagonal line.

Overall, the resulting risk score model highlights important factors associated with a higher risk of non-adherence that are corroborated by past research. Given the limited sample size, further work is needed to validate the usefulness of the proposed risk score models.

8 Discussion

In this paper, we introduce a novel algorithm for estimating integer risk score models for binary outcomes. A simulation study, experiments on test bed datasets, and two applications demonstrate the efficiency and efficacy of our algorithm. We compare our algorithm

to popular rounding methods, showing that these methods often sacrifice performance and can produce coefficients that are too large to be easily applied in practice. RiskCD offers an alternative to rounding that heuristically optimizes over an integer constraint.

Further, we compare our algorithm to the current state-of-the-art optimization method, FasterRisk. While FasterRisk could produce higher validation AUC values than RiskCD, it's relative inefficiency and inability to scale to larger datasets is a substantial limitation. Additionally, FasterRisk requires the user to choose the sparsity parameter, while RiskCD-CV uses cross-validation to optimally regularize the model.

RiskCD is the first integer risk score model to be implemented in an R package. This allows the RiskCD algorithm to be accessible to researchers in clinical fields, who are more likely to use R than Python. While rounding methods can be implemented in R, they need to be coded by individual researchers since they are not currently available in an R package, making these methods less standardized.

Future work could consider other more flexible functions to map between the estimated risk scores and the estimated probabilities for the outcome. Further, one limitation of risk score models in general is that in order for the risk scores to be easily interpreted the data is often converted to all categorical variables. Identifying optimal cut-points for continuous variables is another open question.

The R package associated with this paper is available at `https://cran.r-project.org/web/packages/riskscores/index.html`. Code for reproducing the simulations and applications is available at `https://github.com/hjeglinton/RiskCD`.

References

[1] Cynthia Rudin, Chaofan Chen, Zhi Chen, Haiyang Huang, Lesia Semenova, and Chudi Zhong. Interpretable machine learning: Fundamental principles and 10 grand challenges. *Statistics Surveys*, 16(none), 2022. ISSN 1935-7516. doi: 10.1214/21-ss133. URL `http://dx.doi.org/10.1214/21-ss133`.

[2] D A Morrow, E M Antman, A Charlesworth, R Cairns, S A Murphy, J A de Lemos, R P Giugliano, C H McCabe, and E Braunwald. TIMI risk score for ST-elevation myocardial infarction: A convenient, bedside, clinical score for risk assessment at presentation: An intravenous nPA for treatment of infarcting myocardium early II trial substudy. *Circulation*, 102(17):2031–2037, October 2000.

[3] Ron Pisters, Deirdre A Lane, Robby Nieuwlaat, Cees B de Vos, Harry J G M Crijns, and Gregory Y H Lip. A novel user-friendly score (HAS-BLED) to assess 1-year risk of major bleeding in patients with atrial fibrillation: the euro heart survey. *Chest*, 138 (5):1093–1100, November 2010.

[4] Mervyn Singer, Clifford S Deutschman, Christopher Warren Seymour, Manu Shankar-Hari, Djillali Annane, Michael Bauer, Rinaldo Bellomo, Gordon R Bernard, Jean-Daniel Chiche, Craig M Coopersmith, Richard S Hotchkiss, Mitchell M Levy, John C Marshall, Greg S Martin, Steven M Opal, Gordon D Rubenfeld, Tom van der Poll, Jean-Louis Vincent, and Derek C Angus. The third international consensus definitions for sepsis and septic shock (sepsis-3). *JAMA*, 315(8):801–810, February 2016.

[5] Gregory Y H Lip, Robby Nieuwlaat, Ron Pisters, Deirdre A Lane, and Harry J G M Crijns. 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(2):263–272, February 2010.

[6] Yann Chevaleyre, Frédéerick Koriche, and Jean-Daniel Zucker. Rounding methods for

discrete linear classification. In *International Conference on Machine Learning*, pages 651–659. PMLR, 2013.

[7] TJ Cole. Algorithm as 281: scaling and rounding regression coefficients to integers. *Applied statistics*, pages 261–268, 1993.

[8] Vigneshwar Subramanian, Edward J Mascha, and Michael W Kattan. Developing a clinical prediction score: Comparing prediction accuracy of integer scores to statistical regression models. *Anesthesia & Analgesia*, 132(6):1603–1613, June 2021.

[9] Berk Ustun and Cynthia Rudin. Learning optimized risk scores. *Journal of Machine Learning Reseach*, 20:1–75, 2019.

[10] Jiachang Liu, Chudi Zhong, Boxuan Li, Margo Seltzer, and Cynthia Rudin. Fasterrisk: Fast and accurate interpretable risk scores. 2022.

[11] Lieven Billiet, Sabine Van Huffel, and Vanya Van Belle. Interval coded scoring extensions for larger problems. In *2017 IEEE Symposium on Computers and Communications (ISCC)*, pages 198–203. IEEE, 2017.

[12] Lieven Billiet, Sabine Van Huffel, and Vanya Van Belle. Interval coded scoring: a toolbox for interpretable scoring systems. *PeerJ Computer Science*, 4:e150, 2018.

[13] Emilio Carrizosa, Amaya Nogales-Gómez, and Dolores Romero Morales. Strongly agree or strongly disagree?: Rating features in support vector machines. *Information Sciences*, 329:256–273, 2016.

[14] Antoine Dedieu, Hussein Hazimeh, and Rahul Mazumder. Learning sparse classifiers: Continuous and mixed integer optimization perspectives. *The Journal of Machine Learning Research*, 22(1):6008–6054, 2021.

[15] Şeyda Ertekin and Cynthia Rudin. A bayesian approach to learning scoring systems. *Big Data*, 3(4):267–276, 2015.

[16] Nataliya Sokolovska, Yann Chevaleyre, Karine Clément, and Jean-Daniel Zucker. The fused lasso penalty for learning interpretable medical scoring systems. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 4504–4511. IEEE, 2017.

[17] Nataliya Sokolovska, Yann Chevaleyre, and Jean-Daniel Zucker. A provable algorithm for learning interpretable scoring systems. In *International Conference on Artificial Intelligence and Statistics*, pages 566–574. PMLR, 2018.

[18] Berk Ustun and Cynthia Rudin. Supersparse linear integer models for optimized medical scoring systems. *Machine Learning*, 102:349–391, 2016.

[19] Feng Xie, Bibhas Chakraborty, Marcus Eng Hock Ong, Benjamin Alan Goldstein, and Nan Liu. AutoScore: A Machine Learning-Based Automatic Clinical Score Generator and Its Application to Mortality Prediction Using Electronic Health Records. *JMIR Med Inform*, 8(10):e21798, Oct 2020.

[20] Anthony Li, Ming Lun Ong, Chien Wei Oei, Weixiang Lian, Hwee Pin Phua, Lin Htun Htet, and Wei Yen Lim. Unified auto clinical scoring (uni-acs) with interpretable ml models. In Zachary Lipton, Rajesh Ranganath, Mark Sendak, Michael Sjoding, and Serena Yeung, editors, *Proceedings of the 7th Machine Learning for Healthcare Conference*, volume 182 of *Proceedings of Machine Learning Research*, pages 26–53. PMLR, 05–06 Aug 2022. URL `https://proceedings.mlr.press/v182/li22a.html`.

[21] Jerome Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(5): 1–22, 2010. doi: 10.18637/jss.v033.i01. URL `https://www.jstatsoft.org/article/view/v033i01`.

[22] Hussein Hazimeh, Rahul Mazumder, and Tim Nonet. L0learn: A scalable package

for sparse learning using l0 regularization. *Journal of Machine Learning Research*, 24 (205):1–8, 2023.

[23] Yeonsoo Baik, Hannah M Rickman, Colleen F Hanrahan, Lesego Mmolawa, Peter J Kitonsa, Tsundzukana Sewelana, Annet Nalutaaya, Emily A Kendall, Limakatso Lebina, Neil Martinson, Achilles Katamba, and David W Dowdy. A clinical score for identifying active tuberculosis while awaiting microbiological results: Development and validation of a multivariable prediction model in sub-saharan africa. *PLoS medicine*, 17(11):e1003420, 2020. doi: 10.1371/journal.pmed.1003420. URL https://doi.org/10.1371/journal.pmed.1003420.

[24] Stephen J Wright. Coordinate descent algorithms. *Mathematical programming*, 151 (1):3–34, 2015.

[25] Olvi L Mangasarian, W Nick Street, and William H Wolberg. Breast cancer diagnosis and prognosis via linear programming. *Operations Research*, 43(4):570–577, 1995.

[26] Ron Kohavi. Scaling up the accuracy of naive-bayes classifiers: a decision-tree hybrid. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, page 202–207. AAAI Press, 1996.

[27] Sérgio Moro, Paulo Cortez, and Paulo Rita. A data-driven approach to predict the success of bank telemarketing. *Decision Support Systems*, 62:22–31, 2014. ISSN 0167-9236. doi: https://doi.org/10.1016/j.dss.2014.03.001. URL https://www.sciencedirect.com/science/article/pii/S016792361400061X.

[28] Jeff Larson, Julia Angwin, Lauren Kirchner, and Surya Mattu. How we analyzed the compas recidivism algorithm, May 2016. URL https://www.propublica.org/article/how-we-analyzed-the-compas-recidivism-algorithm.

[29] M Elter, R Schulz-Wendtland, and T Wittenberg. The prediction of breast cancer

biopsy outcomes using two CAD approaches that both emphasize an intelligible decision process. *Med. Phys.*, 34(11):4164–4172, November 2007.

[30] Jeffrey C Schlimmer. Concept acquisition through representational adjustment. *PhD thesis, University of California, Irvine*, page AAI8724747, 1987.

[31] Lorrie Faith Cranor and Brian A LaMacchia. Spam! *Communications of the ACM*, 41:74–83, 1998. doi: http://doi.acm.org/10.1145/280324.280336.

[32] L A Enane, E D Lowenthal, T Arscott-Mills, M Matlhare, L S Smallcomb, B Kgwaadira, S E Coffin, and A P Steenhoff. Loss to follow-up among adolescents with tuberculosis in gaborone, botswana. *International Journal of Tuberculosis Lung Disease*, 20(10):1320–1325, October 2016.

[33] Eva-Maria Guix-Comellas, Librada Rozas, Eneritz Velasco-Arnaiz, Victoria Morín-Fraile, Enriqueta Force-Sanmartín, and Antoni Noguera-Julian. Adherence to antituberculosis drugs in children and adolescents in a low-endemic setting: A retrospective series. *Pediatric Infectious Disease Journal*, 36(6):616–618, June 2017.

[34] Silvia S Chiang, Joshua Ray Tanzer, Jeffrey R Starke, Jennifer F Friedman, Betsabe Roman Sinche, Katya León Ostos, Rosa Espinoza Meza, Elmer Altamirano, Catherine B Beckhorn, Victoria E Oliva Rapoport, Marco A Tovar, and Leonid Lecca Lecca. Identifying adolescents at risk for suboptimal adherence to tuberculosis treatment: A prospective cohort study. *PLOS global public health*, 4(2):e0002918, 2024. doi: https://doi.org/10.1371/journal.pgph.0002918.

[35] Silvia S Chiang, Liz Senador, Elmer Altamirano, Milagros Wong, Catherine B Beckhorn, Stephanie Roche, Julia Coit, Victoria Elena Oliva Rapoport, Leonid Lecca, and Jerome T Galea. Adolescent, caregiver and provider perspectives on tuberculosis treatment adherence: a qualitative study from lima, peru. *BMJ Open*, 13(5):e069938, May 2023.

[36] Anna M Leddy, Devan Jaganath, Rina Triasih, Eric Wobudeya, Marcia C Bellotti de Oliveira, Yana Sheremeta, Mercedes C Becerra, and Silvia S Chiang. Social determinants of adherence to treatment for tuberculosis infection and disease among children, adolescents, and young adults: A narrative review. *Journal of the Pediatric Infectious Disease Society*, 11(Supplement_3):S79–S84, October 2022.

[37] Marjorie Z Imperial, Payam Nahid, Patrick PJ Phillips, Geraint R Davies, Katherine Fielding, Debra Hanna, David Hermann, Robert S Wallis, John L Johnson, Christian Lienhardt, et al. A patient-level pooled analysis of treatment-shortening regimens for drug-susceptible pulmonary tuberculosis. *Nature medicine*, 24(11):1708–1715, 2018.

A Appendix

A.1 TB Medication Adherence Data: Variable Descriptions

Table 14: Full list of variables in TB medication adherence dataset

Variable	**Type**	**Description**
TB medication non-adherence	binary (0/no, 1/yes)	Whether participant took $< 95\%$ of doses on time.
Gender	binary (male, female)	Self-reported identity. Multiple options were provided, but all participants identified as either male or female.
Age	categorical (< 16, $16-17$, $18+$)	Age of participant.
Concomitant TB	binary (no, yes)	Whether participant had concomitant TB.
Lives with mom	binary (no, yes)	Whether participant lives with their mother.
Lives with parents	categorical (no parents, single mom, single dad, 2 parents)	Which parent(s) participant lives with.
No current symptoms	binary (0/no, 1/yes)	Whether participant had no current symptoms at time of survey (0 = symptoms, 1 = no symptoms).
Pills	categorical (1, 2, 3, 4, 5)	1 = 0-3 pills, 2 = 4-6 pills, 3 = 7-9 pills, 4 = 10-11 pills, 5 = 12+ pills
Fixed doses	binary (no, yes)	Whether participant was prescribed fixed-dose combination therapy.
Isoniazid monoresistance	binary (no, yes)	Whether participant had isoniazid monoresistance detected.

Table 14: Full list of variables in TB medication adherence dataset

Variable	Type	Description
Frequency of adverse reactions	categorical (0, 1, 2, 3, 4)	Number of days per week participant has side effects from TB treatment (0 = 0 days, 1 = 1-2 days, 2 = 3-4 days, 3 = 5-6 days, 4 = 7 days).
Accompanied by family	categorical (1, 2, 3, 4, 5)	Response to statement: "Someone from home accompanies me when I go to the health center for my medications" (from 1 for never to 5 for always).
Family dislikes friends	categorical (1, 2, 3, 4, 5)	Response to statement: "My mother/father/guardian does not like my friends" (from 1 for never to 5 for always).
Obedient autonomy	categorical (1, 2, 3, 4, 5)	From 1 for never to 5 for always.
Self-efficacy	binary (≤ 12, > 12)	Score on validated PROMIS treatment self-efficacy scale with 4 items; higher score is more self-efficacious
Depression	binary (≤ 10, > 10)	Score on validated (including in adolescents) PHQ-9 depression scale.
Alcohol use	binary (0, > 0)	Alcohol Use Disorders Identification Test (AUDIT), validated in adults.
Frequency of tobacco use	categorical (0, 1, 2, 3)	0 = never, 1 = 1-2 times, 2 = monthly, 3 = weekly
Drug use	binary (0/no, 1/yes)	Whether patient uses drugs, other than tobacco and alcohol, on a regular basis.
Adverse childhood experiences	categorical (0, 1, > 1)	Validated score to measure adverse childhood experiences.
Health center stigma	categorical (1, 2, 3, 4, 5)	Response to statement: "I feel ashamed to be seen at the health center" (1 for never to 5 for always).
Total stigma	binary (≤ 30, > 30)	Score on validated stigma scale.
Prior TB treatment	binary (0/no, 1/yes)	Whether participant previously received treatment for TB disease.
Prior COVID	categorical (no, suspected, confirmed)	Whether participant previously had COVID-19.
COVID concerns	categorical (0, 1, 2, 3, 4)	Response to question: "How worried are you about getting COVID at the health center?" (1 = not worried, 2 = a little worried, 3 = somewhat worried, 4 = very worried).
Treatment administration	categorical (in person DOT, family supervision only, in-person DOT + family supervision, no supervision, VDOT + family supervision)	How participant's treatment was monitored (DOT = directly observed therapy).
Psychological intervention	categorical (no intervention needed, MINSA referral, SAME, not evaluated)	Whether participant received psychological intervention.

Table 14: Full list of variables in TB medication adherence dataset

Variable	Type	Description
Family support	categorical (1, 2, 3, 4, 5)	Median response to the following statements (from 1 for never to 5 for always): "My mother/father/guardian treats me with kindness"; "I get along well with my mother/father/guardian"; "I get along well with other family members"; "I confide in my family"; "My mother/father/guardian supports me emotionally with my TB treatment"; "Other family members support me emotionally with my TB treatment"; "Generally my mother/father/guardian and I always have had emotional support from other family members"; "Generally, I am happy with my relationship with my mother/father/-guardian"
Health services	categorical (1, 2, 3, 4, 5)	Median response to the following statements (from 1 for strongly disagree to 5 for strongly agree): "The health worker at the TB program always treats me with respect"; "The providers at the health center have clearly explained to me what TB is and what the treatment is like"; "The physical space at the TB program at the health center is comfortable for adolescents"; "The providers at the TB program care about my recovery"; " I am happy with the schedule and quality of care at the TB program"
Motivation	categorical (1, 2, 3, 4, 5)	Median response to the following statements (from 1 to strongly disagree to 5 for strongly agree): "I want to finish my treatment as soon as possible so as not to infect my family members"; "I want to finish my treatment as soon as possible so my family no longer has to worry about me"; "I want to finish my treatment as soon as possible so I can continue my studies or work"; "I want to finish my treatment as soon as possible so I can return to my normal, personal activities (go out with friends, play soccer, dance, go skating, etc.)"
TB knowledge	categorical (1, 2, 3, 4, 5)	Median response to the following statements (from 1 to strongly disagree to 5 for strongly agree): "TB can be completely cured"; "If I miss some days of my TB treatment, my TB could come back stronger"; "If I completely stop taking my TB treatment, my TB could come back "stronger"

A.2 TB Medication Adherence Data: Population Characteristics

Table 15: Population summary by adherence status among 210 TB-positive participants in Lima, Peru.

Characteristic	**Adherent**, N = 171[1]	**Non-Adherent**, N = 39[1]
Gender		
female	65 (38%)	12 (31%)
male	106 (62%)	27 (69%)
Age		
< 16	51 (30%)	10 (26%)
16-17	47 (27%)	12 (31%)
18+	73 (43%)	17 (44%)
Concomitant TB		
no	155 (91%)	33 (85%)
yes	16 (9.4%)	6 (15%)
Lives with mom		
no	21 (12%)	10 (26%)
yes	150 (88%)	29 (74%)
Lives with parents		
2 parents	99 (58%)	17 (44%)
no parents	13 (7.6%)	1 (2.6%)
single dad	7 (4.1%)	8 (21%)
single mom	52 (30%)	13 (33%)
No current symptoms		
0	131 (77%)	32 (82%)
1	40 (23%)	7 (18%)
Pills		
1	33 (19%)	2 (5.1%)
2	35 (20%)	6 (15%)
3	15 (8.8%)	7 (18%)
4	88 (51%)	24 (62%)
Fixed doses		
no	111 (65%)	29 (74%)
yes	60 (35%)	10 (26%)

Table 15: Population summary by adherence status among 210 TB-positive participants in Lima, Peru.

Characteristic	**Adherent**, N = 171[1]	**Non-Adherent**, N = 39[1]
Isoniazid monoresistance		
no	160 (94%)	39 (100%)
yes	11 (6.4%)	0 (0%)
Frequency of adverse reactions		
0	73 (43%)	15 (38%)
1	54 (32%)	14 (36%)
2	32 (19%)	5 (13%)
3	7 (4.1%)	3 (7.7%)
4	5 (2.9%)	2 (5.1%)
Accompanied by family		
1	25 (15%)	8 (21%)
2	25 (15%)	7 (18%)
3	32 (19%)	8 (21%)
4	11 (6.4%)	5 (13%)
5	78 (46%)	11 (28%)
Family dislikes friends		
1	61 (36%)	9 (23%)
2	58 (34%)	11 (28%)
3	36 (21%)	9 (23%)
4	9 (5.3%)	4 (10%)
5	7 (4.1%)	6 (15%)
Obedient autonomy		
1	0 (0%)	1 (2.6%)
2	1 (0.6%)	4 (10%)
3	35 (20%)	6 (15%)
4	58 (34%)	10 (26%)
5	77 (45%)	18 (46%)
Self-efficacy		
≤ 12	146 (85%)	37 (95%)
> 12	25 (15%)	2 (5.1%)

Table 15: Population summary by adherence status among 210 TB-positive participants in Lima, Peru.

Characteristic	**Adherent**, N = 171[1]	**Non-Adherent**, N = 39[1]
Depression		
≤ 10	115 (67%)	29 (74%)
> 10	56 (33%)	10 (26%)
Alcohol use		
> 0	26 (15%)	6 (15%)
0	145 (85%)	33 (85%)
Tobacco use		
0	148 (87%)	32 (82%)
1	18 (11%)	6 (15%)
2	5 (2.9%)	0 (0%)
3	0 (0%)	1 (2.6%)
Drug use frequency		
no	151 (88%)	34 (87%)
yes	20 (12%)	5 (13%)
Adverse childhood experiences		
> 1	79 (46%)	21 (54%)
0	46 (27%)	9 (23%)
1	46 (27%)	9 (23%)
Health center stigma		
1	109 (64%)	27 (69%)
2	29 (17%)	5 (13%)
3	20 (12%)	3 (7.7%)
4	7 (4.1%)	1 (2.6%)
5	6 (3.5%)	3 (7.7%)
Total stigma		
≤ 30	154 (90%)	30 (77%)
> 30	17 (9.9%)	9 (23%)
Prior TB treatment		
0	169 (99%)	38 (97%)
1	2 (1.2%)	1 (2.6%)

Table 15: Population summary by adherence status among 210 TB-positive participants in Lima, Peru.

Characteristic	**Adherent**, N = 171[1]	**Non-Adherent**, N = 39[1]
Prior COVID		
confirmed	19 (11%)	6 (15%)
no	125 (73%)	23 (59%)
suspected (unconfirmed)	27 (16%)	10 (26%)
COVID concerns		
0	51 (30%)	17 (44%)
1	96 (56%)	20 (51%)
2	23 (13%)	2 (5.1%)
3	1 (0.6%)	0 (0%)
Treatment administration		
family supervision only	38 (22%)	5 (13%)
in-person DOT + family supervision	25 (15%)	0 (0%)
in-person DOT only	98 (57%)	30 (77%)
no supervision	1 (0.6%)	0 (0%)
VDOT + family supervision	9 (5.3%)	4 (10%)
Psychological intervention		
MINSA referral	20 (12%)	4 (10%)
no intervention needed	38 (22%)	9 (23%)
not evaluated	73 (43%)	16 (41%)
SAME	40 (23%)	10 (26%)
Family support		
1	1 (0.6%)	2 (5.1%)
2	3 (1.8%)	2 (5.1%)
3	28 (16%)	13 (33%)
4	51 (30%)	7 (18%)
5	88 (51%)	15 (38%)
Health services		
3	6 (3.5%)	6 (15%)
4	76 (44%)	16 (41%)
5	89 (52%)	17 (44%)

Table 15: Population summary by adherence status among 210 TB-positive participants in Lima, Peru.

Characteristic	**Adherent**, N = 171[1]	**Non-Adherent**, N = 39[1]
Motivation		
2	1 (0.6%)	1 (2.6%)
3	1 (0.6%)	0 (0%)
4	45 (26%)	10 (26%)
5	124 (73%)	28 (72%)
TB knowledge		
2	1 (0.6%)	1 (2.6%)
3	47 (27%)	9 (23%)
4	123 (72%)	29 (74%)