An Integrated Machine Learning Approach to Stroke Prediction

Aditya Khosla
Dept. of Computer Science
Stanford University
Stanford, CA 94305
aditya86@stanford.edu

Yu Cao
Dept. of Electrical Engineering
Stanford University
Stanford, CA 94305
yufcao@stanford.edu

Cliff Chiung-Yu Lin
Dept. of Electrical Engineering
Stanford University
Stanford, CA 94305
chiungyu@stanford.edu

Hsu-Kuang Chiu
Dept. of Electrical Engineering
Stanford University
Stanford, CA 94305
hkchiu@stanford.edu

Junling Hu*
eBay, Inc
2145 Hamilton Avenue
San Jose, CA 95125
juhu@ebay.com

Honglak Lee†
Dept. of Computer Science
Stanford University
Stanford, CA 94305
hllee@cs.stanford.edu

ABSTRACT

Stroke is the third leading cause of death and the principal cause of serious long-term disability in the United States. Accurate prediction of stroke is highly valuable for early intervention and treatment. In this study, we compare the Cox proportional hazards model with a machine learning approach for stroke prediction on the Cardiovascular Health Study (CHS) dataset. Specifically, we consider the common problems of data imputation, feature selection, and prediction in medical datasets. We propose a novel automatic feature selection algorithm that selects robust features based on our proposed heuristic: *conservative mean*. Combined with Support Vector Machines (SVMs), our proposed feature selection algorithm achieves a greater area under the ROC curve (AUC) as compared to the Cox proportional hazards model and L_1 regularized Cox model. Furthermore, we present a *margin-based censored regression* algorithm that combines the concept of margin-based classifiers with censored regression to achieve a better concordance index than the Cox model. Overall, our approach outperforms the current state-of-the-art in both metrics of AUC and concordance index. In addition, our work has also identified potential risk factors that have not been discovered by traditional approaches. Our method can be applied to clinical prediction of other diseases, where missing data are common and risk factors are not well understood.

Categories and Subject Descriptors

J.3 [**Computer Application**]: Life and medical sciences; I.2.6 [**Artificial Intelligence**]: Learning; I.5.2 [**Pattern recognition**]: Design methodology

General Terms

Experimentation, Algorithms, Performance

*This work was done while the author was at Robert Bosch LLC, Research and Technology Center

†Corresponding author.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

KDD'10, July 25–28, 2010, Washington, DC, USA.

Copyright 2010 ACM 978-1-4503-0055-1/10/07 ...$10.00.

1. INTRODUCTION

Stroke is the third leading cause of death and the principal cause of serious long-term disability in the United States [2]. Stroke risk prediction can contribute significantly to its prevention and early treatment. Numerous medical studies and data analyses have been conducted to identify effective predictors of stroke. The Framingham Study [6, 34] reported a list of stroke risk factors including age, systolic blood pressure, the use of anti-hypertensive therapy, diabetes mellitus, cigarette smoking, prior cardiovascular disease, atrial fibrillation, and left ventricular hypertrophy by electrocardiogram. Furthermore, in the past decade, a number of other studies [25, 23, 24, 26] have led to the discovery of more risk factors such as creatinine level, time to walk 15 feet, and others.

Most previous prediction models have adopted features (risk factors) that are verified by clinical trials or selected manually by medical experts. For example, Lumley et al. [24] built a 5-year stroke prediction model based on the Cardiovascular Health Study [8] dataset using a set of 16 manually selected features (given in [25]) from a total of roughly one thousand features. With a large number of features in current medical datasets, it is a cumbersome task to identify and verify each risk factor manually. On the other hand, machine learning algorithms are capable of identifying features highly related to stroke occurrence efficiently from the huge set of features; therefore, we believe machine learning can be used to: (i) improve the prediction accuracy of stroke risk and (ii) discover new risk factors.

Lumley et al.'s [24] 5-year stroke prediction model adopted the Cox proportional hazards model, one of the most commonly used statistical methods in medical research [3]. It has been extensively studied [1, 3] and applied to the prediction of various diseases including stroke [15, 24, 21]. However, the performance of the original Cox model depends heavily on the quality of the pre-selected features. To address this problem, several approaches have been proposed recently [9, 28].

Thus far, there have been very few studies on comparing the Cox regression with machine learning methods in making predictions on censored data. Kattan [18] compared Cox proportional hazards regression with several machine learning methods (neural networks and tree-based methods) based on three urological datasets. However, Kattan's

study focused on datasets with only five features, while machine learning algorithms are expected to effectively handle many more features. In addition, the paper considered only some relatively simple machine learning algorithms and high-performance machine learning algorithms such as SVM and logistic regression were not explored.

This paper presents an integrated machine learning approach for stroke risk prediction. We investigated machine learning algorithms to improve the prediction accuracy and conducted extensive comparisons between our results and those with the Cox proportional hazards model. Using the CHS dataset as a benchmark, we first duplicated the results of Lumley et al. [24] as a baseline. We then compared our machine learning approach with the baseline results and an extended version of the Cox model with feature selection. According to our experiments, our approach consistently outperformed the Cox model.

Our approach considers the problems of data imputation, feature selection, and prediction in medical datasets. We propose a novel automatic feature selection algorithm that selects robust features based on our proposed heuristic: *conservative mean.* We combine this feature selection algorithm with the popular SVM algorithm.

Furthermore, we present a *margin-based censored regression* algorithm that combines the concept of margin-based classifiers with censored regression to achieve a better concordance index than the Cox model. In addition, our work has also identified potential risk factors that have not been discovered by traditional approaches. Last, we note that this method can be applied to clinical prediction of other diseases, where missing data are common and risk factors are not well understood.

In summary, our main contributions are:

1. An extensive evaluation of the problems of data imputation, feature selection and prediction in medical data, with comparisons against the Cox proportional hazards model.

2. A novel feature selection algorithm, Conservative Mean feature selection, that outperforms both L_1 regularized Cox model and L_1 regularized logistic regression on the CHS dataset.

3. A novel risk prediction algorithm, Margin-based Censored Regression, that outperforms the Cox model given the same set of features.

4. Discovery of new (previously unknown) potential risk factors of stroke.

5. An integrated machine learning approach that significantly outperforms the current state-of-the-art algorithm in stroke prediction.

This paper is organized as follows. Section 2 describes Cox proportional hazards regression and the L_1 regularized Cox models that we use as the baselines in this study. Section 3 describes the various machine learning-based methods we compare against the Cox models, and Section 4 provides the experimental results of our approach. Finally, Section 5 presents our conclusions.

2. RELATED WORK

Cox proportional hazards model is widely adopted in clinical studies and used heavily in stroke prediction. We briefly compare the Cox model to some of our other approaches.

2.1 Cox proportional hazards model

The Cox proportional hazards model is given by

$$h(t|\mathbf{x}) = h_0(t)\exp(\beta^T\mathbf{x}), \tag{1}$$

where $h(t|\mathbf{x})$ is the hazard value at time t given the feature vector $\mathbf{x} \in \mathbb{R}^d$ for an individual, $h_0(t)$ is an arbitrary baseline hazard function, $\beta \in \mathbb{R}^d$ are the parameters that we are trying to estimate for the model, and d is the number of features for each individual.

This model is known as a semi-parametric model because the baseline hazard function is treated non-parametrically. Thus, we can see that the parameters have a multiplicative effect on the hazard value which makes it different from the linear regression models [20].

The Cox model is part of the Generalized Linear Model (GLM) family. Another member of this family is the logistic regression model, where the output takes the following form:

$$h(\mathbf{x}) = (1 + \exp(-\beta^T\mathbf{x}))^{-1} \tag{2}$$

In this study, we investigated both the Cox model and logistic regression model for stroke prediction. In addition, we broadened our approach to other non-regression models such as SVM, taking an agnostic view on what the best model is for stroke prediction. We found that while the Cox model performs reasonably well for stroke prediction, it is inferior to more general machine learning models, such as SVM or margin-based censored regression (proposed in this paper).

2.2 Feature selection and L_1 regularization

Finding the best estimate for β in equation (1) and (2) is typically computationally difficult, particularly given a large number of features. By introducing a complexity-based penalty term, we can identify irrelevant features and remove them from our model. The L_1 regularized sparse learning problem has the following general form:

$$\min_{\beta} g(\beta) + \lambda||\beta||_1, \tag{3}$$

where $g(\cdot)$ is a convex function, β is a vector of length d, and $\lambda > 0$ is a regularization parameter.

In this study, we evaluated both L_1 regularized Cox model and L_1 regularized logistic regression. We found that L_1 regularized feature selection gives better performance over the baselines (i.e., selecting features manually) by reducing the feature set to the most relevant ones.

3. OUR APPROACH

We present an integrated machine learning approach to stroke prediction. Our approach takes the following steps:

1. Apply a systematic method for imputing the missing entries in the dataset.

2. Select the relevant feature subset based on an automatic procedure.

3. Apply learning algorithms to evaluate the prediction performance.

3.1 Performance Metrics

For evaluating the performance of our methods, we used the following metrics: area under the ROC curve and concordance index. To define these precisely, we first outline the notation.

3.1.1 Notation

Consider a dataset $\{(\mathbf{x}^{(1)}, y^{(1)}, t^{(1)}), ..., (\mathbf{x}^{(m)}, y^{(m)}, t^{(m)})\}$, where $\mathbf{x}^{(i)} \in \mathbb{R}^d$ is the feature vector[1] for individual i (i.e., d is the number of features), m is the number of individuals in the dataset, $y^{(i)}$ is the occurrence of stroke within a pre-defined time frame ($y = 1$ if stroke occurs and 0 otherwise), and $t^{(i)}$ is the time of stroke. If the stroke does not occur within the pre-defined time frame for individual i, we set $t^{(i)} = t_{max}$, where t_{max} is the duration of the time frame (e.g., 5 years). Now, we define the set of indexes of all positive and negative examples as $\mathcal{M}_p = \{i | y^{(i)} = 1\}$ and $\mathcal{M}_n = \{i | y^{(i)} = 0\}$ respectively. Given a prediction function $f : \mathbb{R}^d \rightarrow \mathbb{R}$, we can compute the prediction estimate for individual i as $f(\mathbf{x}^{(i)})$.[2]

3.1.2 Area under the ROC curve

The area under the ROC curve (or AUC) is one of the most important metrics for evaluating the performance of classifiers in the medical diagnosis domain (where positive samples are usually small in number) as it considers both sensitivity and specificity, providing a balanced measure for classifier performance. Specifically, the AUC (associated with the function f) is defined as follows [5, 14]:

$$AUC = \frac{1}{|\mathcal{M}_p| \cdot |\mathcal{M}_n|} \sum_{i \in \mathcal{M}_p} \sum_{j \in \mathcal{M}_n} \mathbf{1}_{f(\mathbf{x}^{(i)}) > f(\mathbf{x}^{(j)})}, \quad (4)$$

where $\mathbf{1}_{(\cdot)}$ is an indicator function. The AUC is used to evaluate the performance of the binary stroke classification task. Essentially, this metric gives an estimate of how accurately the model can answer the question, "is individual A likely to have a stroke within the next 5 years?".

3.1.3 Concordance Index

We would also like to measure how accurately the predictions reflect relative risk of stroke of two randomly selected individuals. A commonly used metric in survival models for this evaluation is the concordance index [17, 29]. The concordance index is a generalization of the concept of AUC designed to handle (i) continuous values for prediction and (ii) censored data. Similar to the AUC, it takes values from 0.5 (completely random) to 1.0 (perfect prediction). The concordance index gives an estimate of how well the output of a prediction model matches the relative time of the event for all pairs of individuals that can be ordered. In essence, it allows us to measure the ability of the model to answer the question, "is individual A or individual B more likely to have a stroke?"

Formally, the concordance index is defined as:

$$\text{Concordance Index} = \frac{1}{|\varepsilon|} \sum_{i \in \mathcal{M}_p} \sum_{t^{(j)} > t^{(i)}} \mathbf{1}_{f(\mathbf{x}^{(i)}) > f(\mathbf{x}^{(j)})}, \quad (5)$$

where $\mathbf{1}_{(\cdot)}$ is the indicator function as before, and $|\varepsilon|$ denotes the number of edges in the ordered graph of $\mathbf{t}$.[3] We assume that a larger value of f corresponds to a higher risk of stroke.

In the following sections, we describe the details of data imputation, feature selection, and prediction models.

3.2 Missing Data Imputation

Clinical data often has significant omissions due to individuals dropping out of the survey, errors in data collection and so on. Missing data often leads to an inaccurate predictive model. Data imputation can be used to remedy missing data. We filled-in missing entries using the following methods:

- Column mean: replace each missing value with the mean of the feature's observed values
- Column median: replace each missing value with the median of the feature's observed values
- Imputation through linear regression [19]
- Regularized Expectation Maximization (EM) [31]

As a post-processing step to impute discrete-valued features, we rounded the imputed values to the nearest discrete value. The imputation algorithms were evaluated using the following metrics:

1. Imputation accuracy (adopted from [7]):
   (a) Root-Mean-Square Deviation (RMSD)
   (b) Mean Absolute Deviation (MAD)
   (c) Bias (mean of imputed values - mean of ground-truth data)
2. Overall stroke prediction performance (measured by the area under the ROC curve).

3.3 Feature Selection

Selecting relevant features [13] is crucial for building an accurate model of clinical data. For example, the CHS dataset has a large number of attributes, ranging from demographic information and clinical history to biomedical and physical measurements. However, only a small subset of attributes is highly relevant to stroke prediction. The traditional approach to stroke prediction has been to use manually selected features based on risk factors analyzed by medical and clinical studies [4, 24, 33, 36]. Instead of manually selecting features, we evaluate three machine learning-based algorithms for selecting features automatically: forward feature selection, L_1 regularized logistic regression, and "conservative mean" feature selection.

3.3.1 Forward feature selection

Forward feature selection [12] greedily adds one feature at a time. The best subset of features was selected based on cross-validation. Note that adding more features does not necessarily improve the test performance since overfitting may occur.

[1] We extend the feature vector with a constant 1 as the intercept term.

[2] Throughout this paper, we interpret the binary classification output $\hat{y} \in \{0, 1\}$ of the prediction function value $f(\mathbf{x})$ as follows: $\hat{y} = 1 \iff f(\mathbf{x}) > 0$.

[3] An ordered graph $G(N, E)$ of $\mathbf{t} = \{t^{(1)}, ..., t^{(m)}\}$ is defined on a set of m nodes, $N = \{n_1, ..., n_m\}$, and a set of edges, E, where $(n_i, n_j) \in E \iff t^{(i)} < t^{(j)}$.

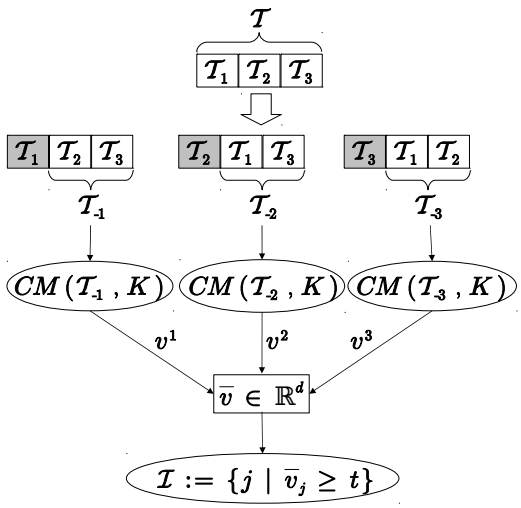

Figure 1: An illustration of Algorithm 2 for $K = 3$. We use the sets $\mathcal{T}_1, \mathcal{T}_2$ and $\mathcal{T}_3$ shown in the shaded boxes as validation sets, and the corresponding sets $\mathcal{T}_{-1}, \mathcal{T}_{-2}$ and $\mathcal{T}_{-3}$ for training to find cross-validation estimates to optimize the value of the threshold, t. $CM(\mathcal{T}, K)$ refers to the $ConservativeMean(\mathcal{D}, K)$ function defined in Algorithm 1.

3.3.2 L_1 regularized logistic regression

L_1 regularized logistic regression [30] is a popular algorithm for feature selection. L_1 regularization has the beneficial effect of regularizing model coefficients (as in L_2 regularization), but yields sparse models that are more easily interpretable [27, 32, 35]. This model has a regularization parameter that controls the "sparseness" of the weights. Consequently, the features with nonzero weights are selected for prediction.

3.3.3 Conservative mean feature selection

Here we present a novel and efficient feature selection algorithm, Conservative Mean (CM) feature selection. Consider a setting where positive examples are small in number and non-homogeneous. Then, the prediction performance may vary significantly depending on how the training and testing examples are sampled. We want to select features that are relevant, yet robust to variations due to sampling over a small number of non-homogeneous examples.

In order to incorporate the above intuition, we introduce the heuristic of conservative mean ($\mu - \sigma$), where μ and σ refer to the mean and standard deviation of the AUC of a feature[4] respectively. The setting is similar to K-fold cross-validation, but we also want to consider the variance across different folds along with the average of the prediction performance. In addition, we want to evaluate the performance of each feature individually. Therefore, subtracting the standard deviation from the mean provides a more 'conservative' estimate of the performance of each feature as compared to using the mean alone, which is the typical approach.[5]

[4] μ (or σ) refers to mean (or standard deviation) of the AUC of a given feature over K-folds of the dataset. See Algorithm 1 for precise definition.

[5] The same heuristic can be used to optimize any other metric such as classification accuracy. Furthermore, this heuristic could be applied to other feature selection algorithms that use cross-validation for selecting features (e.g., forward feature selection).

Table 1: Notation for a dataset $\mathcal{D}$ with d features and m examples, i.e., $\mathcal{D} = \{(\mathbf{x}^{(1)}, y^{(1)}), ..., (\mathbf{x}^{(m)}, y^{(m)})\}$ ($\mathbf{x}^{(i)} \in \mathbb{R}^d, y^{(i)} \in \{0, 1\}, \forall i$)

Symbol	Description
$\mathcal{D}.\mathbf{x}^{(i)} \triangleq \mathbf{x}^{(i)}$	i-th example in the dataset
$\mathcal{D}.y^{(i)} \triangleq y^{(i)}$	i-th label in the dataset
$\mathcal{D}.\mathbf{x}_j \triangleq (x_j^{(1)}, ..., x_j^{(m)})$	j-th features in the dataset
$\mathcal{D}.\mathbf{y} \triangleq (y^{(1)}, ..., y^{(m)})$	set of all labels in the dataset

To describe the conservative mean heuristic formally, we first introduce the notation in Table 1. The key observation here is that when we consider monotonic prediction functions over a single feature, then we only need to compute the AUC over the feature values and the labels (without considering the prediction functions). This is because AUC is invariant under mapping from monotonic functions. For example, this eliminates the need to compute the weight vector or intercept term for a linear SVM when using a single feature as input. The AUC is only affected by the sign of the weight vector which can be easily determined to be the one that ensures the AUC is greater than or equal to 0.5. Specifically, the following two Lemmas provide the formal basis for the efficient computation.

Lemma 1. *Given any monotonically increasing function $f : \mathbb{R} \to \mathbb{R}$ and a dataset $\mathcal{D} = \{(\mathbf{x}^{(1)}, y^{(1)}), ..., (\mathbf{x}^{(m)}, y^{(m)})\}$, where $\mathbf{x}^{(i)} \in \mathbb{R}^d, y^{(i)} \in \{0, 1\}, \forall i$, the AUC for predicted function values of j-th feature, $f(\mathcal{D}.\boldsymbol{x}_j)$, and the labels, $\mathcal{D}.\boldsymbol{y}$, is equal to $AUC(\mathcal{D}.\boldsymbol{x}_j, \mathcal{D}.\boldsymbol{y})$.*

Proof. For notational convenience, we define

$$f(\mathcal{D}.\mathbf{x}_j) \triangleq (f(x_j^{(1)}), ..., f(x_j^{(m)})),$$

i.e., function values for j-th features. Then, the following equalities hold (for all j's):

$$AUC(f(D.\mathbf{x}_j), \mathcal{D}.\mathbf{y}) \tag{6}$$
$$\triangleq AUC((f(x_j^{(1)}), ..., f(x_j^{(m)})), (y^{(1)}, ..., y^{(m)})) \tag{7}$$
$$= AUC((x_j^{(1)}, ..., x_j^{(m)}), (y^{(1)}, ..., y^{(m)})) \tag{8}$$
$$\triangleq AUC(\mathcal{D}.\mathbf{x}_j, \mathcal{D}.\mathbf{y})) \tag{9}$$

The second step holds because a monotonically increasing function does not affect the relative ordering of $D.\mathbf{x}_j$, causing the AUC to remain unchanged. □

Lemma 2. *Given a hypothesis space $\mathcal{H}$ of monotonic (either strictly increasing or strictly decreasing) prediction functions $f : \mathbb{R} \to \mathbb{R}$ and a training set, $\mathcal{D}_{tr}$, and a validation set, $\mathcal{D}_{val}$, we define f^* as follows:*

$$f^* \triangleq \arg\max_{f \in \mathcal{H}} AUC(f(\mathcal{D}_{tr}.\mathbf{x}_j), \mathcal{D}_{tr}.\mathbf{y}) \tag{10}$$

Then, the following holds:

$$AUC(f^*(\mathcal{D}_{val}.\mathbf{x}_j), \mathcal{D}_{val}.\mathbf{y}) \tag{11}$$
$$= \begin{cases} AUC(\mathcal{D}_{val}.\mathbf{x}_j, \mathcal{D}_{val}.\mathbf{y}) & if\ AUC(\mathcal{D}_{tr}.\mathbf{x}_j, \mathcal{D}_{tr}.\mathbf{y}) \geq 0.5 \\ AUC(-\mathcal{D}_{val}.\mathbf{x}_j, \mathcal{D}_{val}.\mathbf{y}) & otherwise \end{cases}$$

Algorithm 1 Computing the conservative mean vector

function ConservativeMean($\mathcal{D}, K$):
Input:
$\mathcal{D}$: dataset with d features
K: number of folds
Output:
v: Conservative mean vector (of length d)
begin
Divide $\mathcal{D}$ evenly into K disjoint sets $\mathcal{D}_1, ..., \mathcal{D}_K$ such that $\mathcal{D} = \mathcal{D}_1 \cup ... \cup \mathcal{D}_K$ and $\mathcal{D}_k \cap \mathcal{D}_l = \emptyset, \forall k, l$.
Set $\mathcal{D}_{-k} \triangleq \mathcal{D} - \mathcal{D}_k, \forall k$.
for $j := 1$ to d **do**
  $s := \vec{0} \in \mathbb{R}^K$
  for $k := 1$ to K **do**
    if AUC($\mathcal{D}_{-k}.\mathbf{x}_j, \mathcal{D}_{-k}.\mathbf{y}$) ≥ 0.5 **then**
      $s_k :=$AUC($\mathcal{D}_k.\mathbf{x}_j, \mathcal{D}_k.\mathbf{y}$)
    else
      $s_k :=$AUC($-\mathcal{D}_k.\mathbf{x}_j, \mathcal{D}_k.\mathbf{y}$)
    end if
  end for
  $v_j := \mu(s) - \sigma(s)$
end for

where
AUC(predictions, labels) returns the area under the ROC curve given the predictions and labels.
$\mu(s) \triangleq \frac{1}{d}\sum_{j=1}^{d} s_j$
$\sigma(s) \triangleq \sqrt{\frac{1}{d}\sum_{j=1}^{d}(s_j - \mu(s))^2}$
end

Proof. By definition of f^*, $AUC(f^*(\mathcal{D}_{tr}.\mathbf{x}_j), \mathcal{D}_{tr}.\mathbf{y}) \geq 0.5$. Since we consider only monotonic functions, f^* is either monotonically increasing or monotonically decreasing. If f^* is monotonically increasing, we have (from the Lemma 1):

$$AUC(f^*(D.\mathbf{x}_j), \mathcal{D}.\mathbf{y}) = AUC(\mathcal{D}.\mathbf{x}_j, \mathcal{D}.\mathbf{y})), \forall \mathcal{D} \quad (12)$$
$$\Rightarrow AUC(f^*(D_{val}.\mathbf{x}_j), \mathcal{D}_{val}.\mathbf{y}) = AUC(\mathcal{D}_{val}.\mathbf{x}_j, \mathcal{D}_{val}.\mathbf{y}))$$

Similarly, if f^* is monotonically decreasing, we have

$$AUC(f^*(D.\mathbf{x}_j), \mathcal{D}.\mathbf{y}) = AUC(-\mathcal{D}.\mathbf{x}_j, \mathcal{D}.\mathbf{y})), \forall \mathcal{D} \quad (13)$$
$$\Rightarrow 0.5 \leq AUC(f^*(D_{tr}.\mathbf{x}_j), \mathcal{D}_{tr}.\mathbf{y}) = AUC(-\mathcal{D}_{tr}.\mathbf{x}_j, \mathcal{D}_{tr}.\mathbf{y}))$$
$$\Rightarrow AUC(\mathcal{D}_{tr}.\mathbf{x}_j, \mathcal{D}_{tr}.\mathbf{y})) \leq 0.5, and$$
$$AUC(f^*(D_{val}.\mathbf{x}_j), \mathcal{D}_{val}.\mathbf{y}) = AUC(-\mathcal{D}_{val}.\mathbf{x}_j, \mathcal{D}_{val}.\mathbf{y})) \quad \square$$

Therefore, we can efficiently compute a robust estimate of prediction performance for each feature (summarized as conservative mean vector v) as described in Algorithm 1.

Based on the conservative mean heuristic for ranking the features, we can now describe an algorithm to select the appropriate number of features. The overall procedure is described in Algorithm 2 (see Figure 1 for illustration). The training data is initially split into K folds, and we compute the conservative mean vector by holding out the k-th fold each time, resulting in a total of K vectors. We then compute the vector $\bar{v}$ by taking the average of the conservative mean vectors to select a robust set of features that generalize well across all folds. Finally, given a threshold value $t \in [0, 1]$, we select all features $\mathcal{I} \triangleq \{j|\bar{v}_j \geq t\}$.[6] This threshold value can be determined using cross-validation, as described in the caption of Figure 1.

[6]Note that selecting a threshold is equivalent to selecting a number of features ranked according to their value of the $\bar{v}$ vector.

Algorithm 2 Conservative mean feature selection

Input:
$\mathcal{T}$: dataset with d features
K: number of folds
t: threshold $\in [0, 1]$
Output:
$\mathcal{I}$: Set of selected feature indexes $\subset \{1, ..., d\}$.
begin
Divide $\mathcal{T}$ evenly into K disjoint sets $\mathcal{T}_1, ..., \mathcal{T}_K$ such that $\mathcal{T} = \mathcal{T}_1 \cup ... \cup \mathcal{T}_K$ and $\mathcal{T}_k \cap \mathcal{T}_l = \emptyset, \forall k, l$.
Set $\mathcal{T}_{-k} \triangleq \mathcal{T} - \mathcal{T}_k, \forall k$.
$\bar{v} := \vec{0} \in \mathbb{R}^d$
for $k := 1$ to K **do**
  $\mathcal{D} := \mathcal{T}_{-k}$
  $\bar{v} := \bar{v} + \frac{1}{K}$ ConservativeMean($\mathcal{D}, K$)
end for
$\mathcal{I} := \{j|\bar{v}_j \geq t\}$
end

3.4 Learning Algorithms for Prediction

In this section, we describe the learning algorithms that we used for stroke prediction: Support Vector Machines and Margin-based Censored Regression (MCR).

3.4.1 Support Vector Machines

SVM is a popular machine learning algorithm that is widely used for classification. Conceptually, SVM optimizes the "margin" between positive and negative examples. We can formulate the stroke prediction problem as predicting the occurrence of stroke over a pre-defined time frame, which makes it a binary classification problem that fits into the framework of SVM. Furthermore, SVM solvers can be used to optimize the area under the ROC curve directly, so they are well suited for the task of stroke prediction. We used linear SVMs implemented using SVM-perf [16] in this study.

3.4.2 Margin-based Censored Regression

Since the SVM is in principle developed for classification, we use it to predict whether or not a stroke would occur within a given time frame while ignoring the information about when the stroke occurred. However, the time of stroke is indicative of the relative risk level of an individual. Incorporating this information would enable us to answer questions such as "is individual A or individual B more likely to have a stroke?" and "when is a stroke likely to occur?," whereas SVM predictions are generally unable to address these concerns.

To address these concerns, we propose the Margin-based Censored Regression algorithm that unifies linear regression with an SVM-like classifier on censored data. More specifically, we propose a convex optimization problem as described below.

Consider a dataset $\{(\mathbf{x}^{(1)}, y^{(1)}, t^{(1)}), ..., (\mathbf{x}^{(m)}, y^{(m)}, t^{(m)})\}$, where $\mathbf{x}^{(i)} \in \mathbb{R}^d$, $y^{(i)} \in \{0, 1\}$ and $t^{(i)} \in \mathbb{R}$, as described in Section 3.1. The time of stroke is then normalized as $\tilde{t}^{(i)} = t^{(i)}/t_{max}$. We then use a monotonically decreasing function to transform the normalized time of stroke to a "hazard value," as in the Cox model. We use the function $z(\tilde{t}^{(i)}) = -\log(\tilde{t}^{(i)})$ in our experiments. With this trans-

formation, we have $z(\tilde{t}^{(i)}) = 0$ for $i \in \{i|y^{(i)} = 0\}$ and $z(\tilde{t}^{(i)}) > 0$ for $i \in \{i|y^{(i)} = 1\}$.[7]

Given the above transformation, our goal is to find a weight vector $\mathbf{w}$ such that $\mathbf{w}^T\mathbf{x}^{(i)}$ is "close to" $z(\tilde{t}^{(i)})$. In addition, we want to be able to distinguish between positive and negative examples; in other words, we want to find $\mathbf{w}$ such that the individuals who experienced a stroke are well separated from the individuals who did not. This is achieved by imposing $\mathbf{w}^T\mathbf{x}^{(i)} \leq -\epsilon$ for the individuals who did not have a stroke, where ϵ is the desired margin between positive and negative examples (set to 1 in our experiments). Finally, similar to SVM, we introduce a penalty term $\sum_i \xi^{(i)}$ to allow for non-separable datasets, and to reduce the sensitivity to outliers.

To sum up, we formulate the problem as

$$\begin{aligned} \underset{w,\xi}{\text{minimize}} \quad & \sum_{i:y^{(i)}=1} \phi(\mathbf{w}^T\mathbf{x}^{(i)} - z(\tilde{t}^{(i)})) + C \sum_{i:y^{(i)}=0} \xi^{(i)} + \gamma ||\mathbf{w}||_2^2 \\ \text{subj. to} \quad & \mathbf{w}^T\mathbf{x}^{(i)} \leq -\epsilon + \xi^{(i)}, \ \forall i \in \{i|y^{(i)} = 0\}, \\ & \xi^{(i)} \geq 0, \ \forall i, \end{aligned} \tag{14}$$

where C and γ are the hyperparameters for the misclassification loss penalty and for regularization respectively, and $\phi : \mathbb{R} \to \mathbb{R}$ is the regression loss penalty, which we fixed as the Huber function [11][8] in our study. To solve problem (14), we used `CVX`, a package for specifying and solving convex programs [11, 10].

Note that we can easily apply the kernel trick to this model, as in SVM. Furthermore, the objective function can be modified to optimize the AUC directly in the same way as SVM-perf [16].

4. EXPERIMENTAL RESULTS

4.1 Dataset and Preprocessing

The Cardiovascular Heart Study [8] is a study of risk factors for cardiovascular diseases in people over the age of 65. According to the Centers for Disease Control and Prevention, nearly three-quarters of all strokes occur in people over the age of 65.[9] This makes CHS an invaluable resource for the investigation of risk factors and the prediction of stroke. In the original cohort recruited in the first phase of the CHS, 5,201 individuals were examined yearly from 1989 to 1999, with a total of about one thousand attributes collected annually through medical examinations, questionnaires, and phone contacts. Events such as stroke and hospitalization were verified by specialists and recorded for each individual.

The comprehensiveness of the CHS dataset makes it one of the most widely used benchmarks for studying risk factors for cardiovascular diseases, including stroke. However, it is also very challenging to use the CHS dataset effectively due to a significant fraction of missing values and a large number of features in the dataset. For example, about 25% of the baseline measurements in the CHS dataset are missing, and some entries are recorded as "unknown" or "refuse to answer."

In our experiments, we considered the 5-year stroke prediction[10] problem on the original cohort. To begin with, we removed the individuals with pre-baseline stroke (as done in [24]) and the features with more than 60% missing entries.[11] This criterion was chosen because features with too many missing entries often turn out to be irrelevant. After preprocessing, the final dataset consisted of 796 features and $4,988$ examples with 299 occurrences of stroke. Then the data was divided randomly with a ratio of $9:1$ for training and testing respectively, while keeping the ratio of the positive and negative examples constant. This process was repeated to obtain a fixed set of 5 randomly sampled train and test sets.[12] In the remaining sections, "average test AUC" refers to the average of AUC obtained by evaluating the prediction algorithm on the test set over these 5 random trials. We also define "average test concordance index" in a similar way.

4.2 Data Imputation

The data imputation quality was evaluated using 10-fold cross-validation. For each feature j, we first removed all the examples that contained missing values for the particular feature. Then, we divided the examples in the resulting data into training and validation sets with the ratio of 9:1 respectively. Treating feature j of the validation set as being unobserved, we used the training data to impute the values of the particular feature of the validation data.[13] The imputed values were then compared with the actual values in the validation dataset to obtain the performance metrics described in Section 3.2. This process was repeated for every feature and the results were averaged. The summary results are shown in Table 2.[14] For the computation of the area under the ROC curve, we used conservative mean feature selection and SVM for stroke prediction.[15]

Among the imputation methods, linear regression gave the smallest RMSD and MAD values, which suggested that it achieved the highest imputation accuracy. However, the overall stroke prediction quality with column median was the best with an area of 0.774 under the ROC curve. In the following sections, we report the results using column median as the default imputation method.

4.3 Feature Selection

4.3.1 *Forward feature selection*

As this method is computationally very expensive, the number of features was initially reduced from 796 to about 200 using L_1 regularized logistic regression. Then we ran for-

[7] For the CHS dataset, $z(\tilde{t}^{(i)})$ roughly ranges from 0 to 5.

[8] The Huber function $\phi(x)$ is defined as $2|x| - 1$ for $|x| \geq 1$, and $|x|^2$ for $|x| \leq 1$.

[9] http://www.cdc.gov/Stroke/facts.htm

[10] Only the cases of stroke that occurred within 5 years after the baseline measurements were considered positive examples.

[11] Some features represented "refuse to answer" values as 9 or 99. We replaced these entries with "missing" before data imputation.

[12] These random trials were used for all the remaining experiments to ensure all the results are directly comparable.

[13] For the linear regression imputation, any missing values in the other features were filled in using the column mean.

[14] Imputation methods without rounding have been left out as the results were very similar.

[15] Imputation with regularized EM was computationally expensive, so we used L_1 logistic regression on data imputed with column median to reduce the number of features to about 200 before applying EM.

Table 2: Data Imputation Results

Imputation Method	RMSD	MAD	Bias	Avg. Test AUC
Column Median	0.0755	0.0125	0.0039	**0.774**
Linear Regression (with rounding)	0.0526	0.0114	$< 10^{-4}$	0.768
Regularized EM	0.9563	0.5537	0.0002	0.765
Column Mean (with rounding)	0.0747	0.0129	0.0032	0.765

ward feature selection on this reduced set to obtain the final set of features. Using SVM for prediction, we selected the optimal number of features through 10-fold cross-validation. The final prediction performance was an average test AUC of 0.751, which is slightly worse than that of using SVM and the 16 features used in [24]. In our experiments, this method selected a much larger number of features than other feature selection algorithms, which indicates that it may be susceptible to overfitting.

4.3.2 L_1 regularized logistic regression

The L_1 regularized logistic regression (L1LR) was used for feature selection, followed by SVM for prediction. The implementation of L1LR was done using the SLEP package [22]. The optimal regularization parameter λ^* was assigned to be the value that maximized the area under the ROC curve for 10-fold cross-validation. The value of λ^* was then used to run L1LR on the entire training set to select the final set of features for testing. The average test AUC was 0.764, which is better than that of the L_1 regularized Cox feature selection algorithm.

4.3.3 Conservative mean selection

Conservative mean selection was run using 10-fold cross-validation for both the generation of the conservative mean vectors as described in Algorithm 1, and to obtain the subset of optimal features as described in Algorithm 2. As observed for the forward search feature selection, using the maximum cross-validation AUC may result in overfitting, and thus we propose a simple method to reduce this effect.

Throughout our experiments, we observed that overfitting may occur when the performance on the training set increases with increasing number of features while the cross-validation performance decreases or remains fairly constant. An example of this can be seen in Figure 2. Hence, we can say that the 'extent of overfitting' increases as the gap between the cross-validation AUC (CV AUC) and train AUC increases. Therefore, we estimated this extent of overfitting by subtracting the cross-validation AUC from the training AUC. Then, we computed a more conservative estimate of the CV AUC as 'CV AUC' − ('train AUC' − 'CV AUC').[16] In Figure 2, we can see that the CV AUC remains fairly constant after about 30 features. If the CV AUC was maximized without accounting for the overfitting, we would select 120 features instead of 30.

Furthermore, we compare the effect of using conservative mean ("$v_j = \mu(s) - \sigma(s)$") against using mean ("$v_j = \mu(s)$") in Algorithm 1 while keeping the remaining algorithms unchanged. The average test AUC decreases from 0.774 (conservative mean) to 0.759 (mean) when using SVM for prediction. The difference in performance clearly shows that conservative mean selects more robust features as compared

[16]This method improved average test AUC by about 0.5% for all the algorithms.

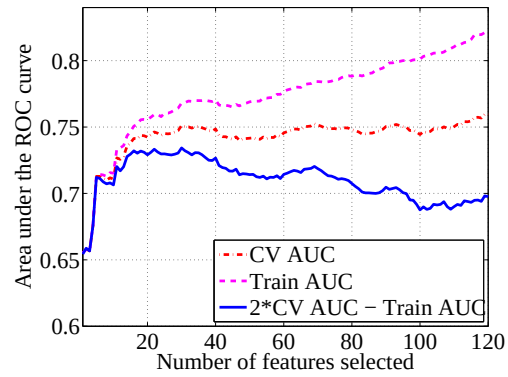

Figure 2: Plot showing the cross-validation AUC and train AUC as features are added. We used CM feature selection with SVM for prediction on a random trial.

Table 3: Average test AUC combining various feature selection algorithms with our prediction algorithms

Feature selection algorithm	SVM	MCR
Conservative Mean	0.774	0.777
L_1 logistic regression	0.764	0.771
16 features (used in [24])	0.753	0.765

to mean alone. Furthermore, we note that the best performance for both SVM and MCR are obtained using the conservative mean feature selection algorithm.

Also note that the conservative mean selection algorithm is significantly more computationally efficient than the forward feature selection algorithm. On the same machine, forward feature selection took about 60 hours to select features from a set of about 200 features, while conservative mean took less than 10 minutes to select from a set of 796 features.

4.4 Stroke Prediction

First, we evaluated the performance of our prediction algorithms based on the area under the ROC curve. The prediction performance when using the feature selection algorithms (described in the previous sections) is compared against the set of 16 manually selected features used by Lumley et al., as shown in Table 3. When using SVM or MCR[17] for prediction, we found that all the feature selection algorithms except forward feature selection performed better than using manually selected features. Overall, CM feature selection performed the best for both prediction methods. We also found that MCR performed better than SVM for all the feature selection methods.

In Table 4, we compare the performance of our algorithms against the current state-of-the-art Cox proportional haz-

[17]In our experiments, we added a small L_1 regularization penalty to the MCR objective function.

Table 4: Average test AUC using different algorithms with comparison to the Cox models

Algorithm	Avg. Test AUC
MCR + CM feature selection	**0.777**
SVM + CM feature selection	0.774
SVM + 16 features (used in [24])	0.753
SVM + forward selection	0.751
Cox + L_1 feature selection	0.747
Cox + 16 features (used in [24])	0.734

Table 5: Average test concordance index of Cox model and MCR using different sets of features

Method	Concordance Index
MCR + CM feature selection	**0.770**
MCR + 16 features (used in [24])	0.757
SVM + CM feature selection	0.760
SVM + 16 features (used in [24])	0.747
Cox + CM feature selection (for MCR)	0.737
Cox + 16 features (used in [24])	0.730

ards model and L_1 regularized Cox model. All our methods outperformed these baseline models. The best method was combining CM feature selection with MCR for prediction, which achieved a 16% error reduction in the average test AUC as compared to the Cox model (as used in [24]).

Second, we evaluated the concordance index to compare the ability of MCR, SVM and Cox model to predict the relative risk of stroke. From Table 5, we observe that the MCR algorithm outperforms the other models when using the same set of features. Also, the features selected using CM significantly increased the concordance index for all the models. It is interesting to note that the SVM performs better than the Cox model even though it does not use information about the relative risk of stroke. The combination of CM feature selection and MCR for prediction gave the best performance with a concordance index of 0.770.

4.5 Identifying risk factors

In addition to achieving better results, our method can automatically identify potential risk factors without carrying out extensive medical studies to understand each one in detail. This would allow for a quick method of characterizing a new disease and identifying its predictors before other studies confirm them. Furthermore, this procedure could also be used to suggest risk factors that might have been previously unexplored.

In our experiments, we found the top features by ranking the average of the conservative mean vectors over multiple random trials in descending order. Table 6 shows a representative set of features found among the top 40 features. Note that there is a large overlap between the top features selected by our feature selection algorithm and those identified by medical studies. This verifies that our algorithm is reliable in identifying risk factors. Thus, the features that are highly ranked but have not been clinically tested might be probable risk factors.

For example, "any ECG abnormality" is a highly ranked factor, whereas "atrial fibrilliation by ECG" is a commonly accepted risk factor of stroke. It may be possible that all ECG abnormalities are more indicative of stroke than just atrial fibrilliation. Also, "minimental score" could be an important risk factor because it gives an indication of the cerebrovascular activity of an individual, which could be correlated to stroke risk. In addition, our results suggest a few other potential risk factors for stroke, such as total number of medications and FVC percent predicted. Further investigation of these features could lead to improved stroke prediction.

Table 6: A representative set of features obtained from the top 40 features selected by CM feature selection

Feature description	Average ($\mu - \sigma$)
Age†	0.6064
Number of symbols correctly coded*	0.5828
Maximal inflation level*	0.5820
Systolic blood pressure†	0.5738
Calculated 100 point score*	0.5681
Total medications*	0.5634
Isolated systolic hypertension†	0.5588
General health*	0.5519
Calculated hypertension status†	0.5500
Time (in sec) to walk 15 feet†	0.5485
Any ECG abnormality*	0.5461
Right/left % Stenosis†	0.5444
Cardiac Injury Score†	0.5426
Min. ankle arm ratio*	0.5374
Diabetic status defined by ADA†	0.5342
Minimental score 35 point*	0.5337
Left ventricular mass†	0.5337
Creatinine†	0.5273
FVC percent predicted*	0.5239

* Potential risk factors that are not found in previous work (to the best of our knowledge)

† Clinically established risk factors of stroke

5. DISCUSSIONS AND CONCLUSION

As we have shown in this study, the conservative mean feature selection performs very well for the CHS dataset. However, we realize that this feature selection algorithm may not work well in other datasets with highly correlated features as it evaluates the performance of each feature individually. To address this problem, we could use an L_1 regularized feature selection algorithm (e.g., L_1 regularized logistic regression) to prune the features before applying conservative mean feature selection for fine-tuning.

In this paper, we have presented an integrated machine learning approach combining the elements of data imputation, feature selection and prediction. We provide an extensive comparison of machine learning methods with the Cox proportional hazards model and show that the machine learning methods significantly outperform the Cox model in terms of both binary stroke prediction and stroke risk estimation. Specifically, we propose the conservative mean heuristic for feature selection, which gives us the best performance as compared to other methods. In addition, we present a novel prediction algorithm, Margin-based Censored Regression, that achieves a better concordance index than the Cox model. Further, our method can be used for identifying potential risk factors for diseases without performing clinical trials. We hope that this paper will motivate the application of machine learning methods in healthcare data analysis.

Acknowledgments

We thank Andrew Ng and Chaitanya Rastogi for helpful advice and the National Heart, Lung, and Blood Institute (NHLBI) for providing the CHS dataset. Support for this work from the Research and Technology Center of Robert Bosch LLC is gratefully acknowledged.

6. REFERENCES

[1] K. Akazawa and T. Nakamura. Simulation program for estimating statistical power of Cox's proportional hazards model assuming no specific distribution for the survival time. *Elseview Ireland*, 1991.

[2] American Heart Association. *Heart Disease and Stroke Statistics 2009 Update*. American Heart Association, Dallas, Texas, 2009.

[3] R. Bender, T. Augustin, and M. Blettner. Generating survival times to simulate Cox proportional hazards models. *Statistics in Medicine*, 24:1713–1723, 2005.

[4] L. E. Chambless, G. Heiss, E. Shahar, M. J. Earp, and J. Toole. Prediction of ischemic stroke risk in the atherosclerosis risk in communities study. *Am. J. Epidemiol.*, 160(3):259–269, 2004.

[5] C. Cortes, , C. Cortes, and M. Mohri. Auc optimization vs. error rate minimization. In *in Advances in Neural Information Processing Systems*. MIT Press, 2003.

[6] T. R. Dawber, G. F. Meadors, and F. E. Moore. Epidemiological approaches to heart disease: The framingham study. *Am J Public Health Nations Health*, 41:279–286, March 1951.

[7] J. M. Engels and P. Diehr. Imputation of missing longitudinal data: a comparison of methods. *J. Clin. Epidemiol.*, 56(10):968–976, 2003.

[8] L. P. Fried, N. O. Borhani, P. Enright, C. D. Furberg, J. M. Gardin, R. A. Kronmal, L. H. Kuller, T. A. Manolio, M. B. Mittelmark, A. Newman, D. H. O'Leary, B. Psaty, P. Rautaharju, R. P. Tracy, and P. G. Weiler. The Cardiovascular Health Study: design and rationale. *Ann Epidemiol.*, 1(3):263–276, February 1991.

[9] J. Goeman. l_1 penalized estimation in the Cox proportional hazards model. *Biom J.*, November 2009.

[10] M. Grant and S. Boyd. Graph implementations for nonsmooth convex programs. In V. Blondel, S. Boyd, and H. Kimura, editors, *Recent Advances in Learning and Control*, Lecture Notes in Control and Information Sciences, pages 95–110. Springer-Verlag Limited, 2008. `http://stanford.edu/~boyd/graph_dcp.html`.

[11] M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming, version 1.21. `http://cvxr.com/cvx`, May 2010.

[12] E. I. Guyon, S. Gunn, M. Nikravesh, and L. A. Zadeh. *Feature Extraction: Foundations and Applications*. Springer, 2006.

[13] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *J. Mach. Learn. Res.*, 3:1157–1182, 2003.

[14] J. A. Hanley and B. J. McNeil. The meaning and use of the area under a receiver operating characteristic (roc) curve. 1982.

[15] K. Ikeda, H. Kumada, and S. Saitoh. Effect of repeated transcatheter arterial embolization on the survival time in patients with hepatocellular carcinoma. *Cancer*, 2006.

[16] T. Joachims. A support vector method for multivariate performance measures. In *ICML '05: Proceedings of the 22nd international conference on Machine learning*, pages 377–384. ACM, 2005.

[17] F. E. H. Jr. *Regression Modeling Strategies, With Applications to Linear Models, Logistic Regression, and Survival Analysis*. Springer, 2001.

[18] M. W. Kattan. Comparison of cox regression with other methods for determining prediction models and nomograms. *The Journal of Urology*, 170:S6–S10, December 2003.

[19] H. Kim, G. H. Golub, and H. Park. Imputation of missing values in DNA microarray gene expression data. In *IEEE Computational Systems Bioinformatics Conference (CSB'04)*, pages 572–573, 2004.

[20] J. Klein and M. Moeschberger. *Survival Analysis: Techniques for Censored and Truncated Data*. Springer, 2003.

[21] K.-Y. Liang, S. G. Self, and X. Liu. The Cox proportional hazards model with change point: An epidemiologic application. *Biometrics*, 46:783–793, 1990.

[22] J. Liu, S. Ji, and J. Ye. *SLEP: Sparse Learning with Efficient Projections*. Arizona State University, 2009.

[23] W. T. Longstreth, Jr., C. Bernick, A. Fitzpatrick, M. Cushman, L. Knepper, J. Lima, and C. Furberg. Frequency and predictors of stroke death in 5,888 participants in the Cardiovascular Health Study. *Neurology*, 56:368–375, February 2001.

[24] T. Lumley, R. A. Kronmal, M. Cushman, T. A. Manolio, and S. Goldstein. A stroke prediction score in the elderly: Validation and web-based application. *J. Clin. Epidemiol.*, 55(2):129–136, February 2002.

[25] T. A. Manolio, R. A. Kronmal, G. L. Burke, D. H. O'Leary, and T. R. Price. Short-term predictors of incident stroke in older adults: The Cardiovascular Health Study. *Stroke*, 27:1479–1486, September 1996.

[26] A. P. McGinn, R. C. Kaplan, J. Verghese, D. M. Rosenbaum, B. M. Psaty, A. E. Baird, J. K. Lynch, P. A. Wolf, C. Kooperberg, J. C. Larson, and S. Wassertheil-Smoller. Walking speed and risk of incident ischemic stroke among postmenopausal women. *Stroke*, 39:1233–1239, April 2008.

[27] A. Y. Ng. Feature selection, l_1 vs. l_2 regularization, and rotational invariance. In *Proc. International Conf. on Machine Learning*, 2004.

[28] M.-Y. Park and T. Hastie. An l_1 regularization-path algorithm for generalized linear models. *JRSSB*, 69(4):659–677, 2007.

[29] V. Raykar, H. Steck, B. Krishnapuram, C. Dehing-Oberije, and P. Lambin. On ranking in survival analysis: Bounds on the concordance index. In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 1209–1216. MIT Press, Cambridge, MA, 2008.

[30] M. Schmidt, G. Fung, and R. Rosales. Fast optimization methods for l_1 regularization: A comparative study and two new approaches. In *European Conference on Machine Learning*, 2007.

[31] T. Schneider. Analysis of incomplete climate data: Estimation of mean values and covariance matrices and imputation of missing values. *Journal of Climate*, 14:853–871, 2001.

[32] R. Tibshirani. Regression shrinkage and selection via the lasso. *J. R. Statist. Soc. B*, 58(1):267–288, 1996.

[33] Z. Vokó, M. Hollander, P. J. Koudstaal, A. Hofman, and M. M. Breteler. How do American stroke risk functions perform in a western European population? *Neuroepidemiology*, 23(5):247–253, September-October 2004.

[34] P. A. Wolf, R. B. D'Agostino, A. J. Belanger, and W. B. Kannel. Probability of stroke: a risk profile from the framingham study. *Stroke*, 22:312–318, March 1991.

[35] E. P. Xing, M. I. Jordan, and R. M. Karp. Feature selection for high-dimensional genomic microarray data. In *Proc. 18th International Conf. on Machine Learning*, pages 601–608. Morgan Kaufmann, San Francisco, CA, 2001.

[36] X.-F. Zhang, J. Attia, C. D'este, X.-H. Yu, and X.-G. Wu. A risk score predicted coronary heart disease and stroke in a Chinese cohort. *Journal of clinical epidemiology*, 58(9):951–958, 2005.