Patient Risk Prediction Model via Top-k Stability Selection

Jiayu Zhou[*] Jimeng Sun[†] Yashu Liu[*] Jianying Hu[†] Jieping Ye[*]

Abstract

The patient risk prediction model aims at assessing the risk of a patient in developing a target disease based on his/her health profile. As electronic health records (EHRs) become more prevalent, a large number of features can be constructed in order to characterize patient profiles. This wealth of data provides unprecedented opportunities for data mining researchers to address important biomedical questions. Practical data mining challenges include: How to correctly select and rank those features based on their prediction power? What predictive model performs the best in predicting a target disease using those features?

In this paper, we propose top-k stability selection, which generalizes a powerful sparse learning method for feature selection by overcoming its limitation on parameter selection. In particular, our proposed top-k stability selection includes the original stability selection method as a special case given $k = 1$. Moreover, we show that the top-k stability selection is more robust by utilizing more information from selection probabilities than the original stability selection, and provides stronger theoretical properties. In a large set of real clinical prediction datasets, the top-k stability selection methods outperform many existing feature selection methods including the original stability selection. We also compare three competitive classification methods (SVM, logistic regression and random forest) to demonstrate the effectiveness of selected features by our proposed method in the context of clinical prediction applications. Finally, through several clinical applications on predicting heart failure related symptoms, we show that top-k stability selection can successfully identify important features that are clinically meaningful.

1 Introduction

The electronic health records (EHRs) capture longitudinal patient information in digital forms from diverse sources: demographics, diagnosis, medication, lab results and physician notes. The adoption of EHRs has increased dramatically recently in response to the government's EHR incentive program. Many healthcare institutions and practitioners are looking for secondary use of EHR data; among which, clinical decision support and care management systems are the most prominent applications. As more EHR data becomes available, a large number of features can be constructed that may be potentially useful for risk prediction, which has important applications in both clinical decision support and care management systems. The challenge of getting high quality features is how to rank all available features based on the predictive power to a specific condition. For that, appropriate feature selection is required.

The need for feature selection is further amplified by the challenge of obtaining high quality training data. The form of training data depends on the specific tasks and the source data quality. Because of the highly noisy nature of EHR data, clinical experts often have to be involved in the annotation process in order to obtain reliably labeled training data. As a result, in many cases only limited labeled data can be obtained. Because of the small sample size, over-fitting becomes one of the biggest concerns for predictive modeling in the healthcare domain. The key to avoiding over-fitting is to construct robust and parsimonious models, where recent development of sparse learning [9, 11, 17, 20, 25] can offer many insights.

Another reason why feature selection is particularly important in building risk prediction models for healthcare is the need for interpretability and actionability. For both clinical decision support and care management support, it is often not sufficient to simply have a model that produces a risk score, even if it is highly accurate. The decision makers need to understand what are the leading risk factors so that proper interventions can be taken to address the risk. Parsimonious models that achieve high accuracy with a relatively small number of succinct features clearly have an advantage of offering better interpretability. This again points to feature selection as the center piece of the discussion.

We focus on addressing this central issue of feature selection for risk prediction in this paper. In many embedded feature selection algorithms, there is a tuning parameter that determines the amount of regularization and correspondingly how many features are included in the model. Examples are regularization factors of spar-

[*] Computer Science and Engineering, Arizona State University
[†] IBM T.J. Watson Research Lab

sity induced penalties (e.g. Lasso regression [25], sparse logistic regression [11, 14], group Lasso [9, 17]), and the iteration number of orthogonal matching pursuit [20]. The choice of such tuning parameters is data-dependent and therefore there is no general guidance on how the parameters should be chosen. Typically cross-validation process is used to estimate the tuning parameters from the training data. Due to overfitting, the estimated parameters via cross validation usually include too many (irrelevant) features in the model [18], especially in high dimensional problems, and thus lead to sub-optimal performance. Moreover, chances are that not all relevant features are selected given a particular tuning parameter, and therefore the information of relevant features may be contained in the models given by more than one tuning parameter.

To address the aforementioned problems, we propose to introduce and extend a recently developed sparse learning method, stability selection [19], which is currently not widely used in the data mining community, despite its success in bioinformatics especially in genome-related biomarker selection problems where sample size is much smaller than feature dimension ($n \ll p$) [6, 22, 23, 26]. Stability selection, based on subsampling/bootstrapping, provides a general method to perform model selection using information from a set of regularization parameters. The stability ranking score gives a probability which makes it naturally interpretable. However, the existing stability selection uses a rather arbitrary and limited maximum selection criterion, which may lead to suboptimal results.

To tackle the limitation of stability selection, we propose a general top-k stability selection algorithm and establish stronger theoretical properties. The original selection method can be shown to be a special case of our proposed top-k stability selection given $k = 1$. We also show theoretically that the proposed top-k stability selection is more robust by utilizing more information from selection probabilities when $k > 1$ than the original stability selection, and is less sensitive to the input parameters. Overall, the top-k stability selection method has the following advantages over the original stability selection:

- **More stable ranking:** The feature ranking is more robust against data perturbation and less sensitive to the input parameters.
- **Improved theoretical guarantee:** Improved theoretical guarantees can be established that provides a stronger bound of the false positive error rate.

We design experiments to compare top-k stability selection given different k and systematically compare our proposed method against six different feature selection methods in the context of clinical prediction. Our top-k stability selection method outperforms the traditional feature selection methods in all datasets. Moreover, the proposed top-k stability selection achieves better performance than the original stability selection on both synthetic and real datasets, which empirically confirms the superiority of this generalization. Finally, we compare three competitive classification methods using selected features from top-k stability selection: SVM, Random Forest and Logistic Regression, and discuss several practical considerations in how to choose the classifiers in the context of clinical prediction application. Logistic regression gives the best prediction performance in terms of AUC, while random forest is shown to be less sensitive to the input parameter value.

The rest of paper is organized as follows: Section 2 describes the motivating application and shows how to convert EHR data into features for risk prediction. Section 3 presents the idea of stability selection, and discusses the top-k stability selection extension and its theoretical properties. Section 4 presents systematic evaluation of different methods for risk prediction. We conclude in Section 5.

2 Motivating Application

Risk prediction has important applications in both clinical decision support and care management systems. For risk prediction to provide actionable insight, it often requires building a predictive model (e.g., classifier) for a specific disease condition. For example, based on the EHR data, one may want to assess the risk scores of a patient developing different disease conditions, such as diabetes complication [24] and heart failure [7, 27], so that proper intervention and care plan can be designed accordingly. The common pipeline for building any predictive model involves the following steps:

- **Feature construction** prepares the input features and the target variable for a specific task. In this step, the labeled training data sets are constructed from EHR data.
- **Feature selection** ranks the input features and keeps only the most relevant ones for subsequent model building.
- **Model building** uses the selected features and the target variable to construct a predictive model. Depending on the tasks, the model can be either classification or regression. In this paper, we focus on classification tasks with binary target variables.
- **Model validation** evaluates the models against a diverse set of performance metrics. In the presence

of an unbalanced class distribution, the Area Under the Curve (AUC) is typically employed.

Feature selection, model building and validation are familiar concepts in the data mining field. Feature construction is often application dependent. Next we illustrate the common strategies for constructing features in clinical applications.

EHR data documents detailed patient encounters in time, which often includes diagnosis, medication, lab result and clinical notes. We systematically construct features from different data sources, recognizing that longitudinal data on even a single variable (e.g., blood pressure) can be represented in a variety of ways. The objective of our feature construction effort is to capture sufficient clinical nuances for subsequent risk prediction task. We model features as longitudinal sequences of observable measures such as demographic, diagnoses, medication, lab, vital, and symptom. As clinical events are recorded over time, for a given time window (observation window), we summarize those events into feature variables. As shown in Figure 1, at the index date, we want to construct features for a specific patient. For that, we summarize all clinical events in observation window right before the index date into features about this patient.

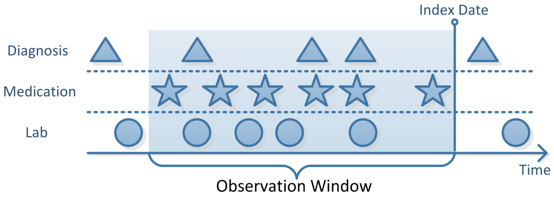

Figure 1: Illustration of longitudinal EHR data.

The sparseness of the data will vary among patients and we expect the sequencing and clustering of what is observed to also vary. Different types of clinical events arise in different frequencies and in different orders. Therefore, we construct summary statistics for different types of event sequences based on the feature characteristics: For static features such as gender and ethnicity, we will use a single static value to encode the feature. For temporal numeric features such as lab measures, we will use summary statistics such as point estimate - mean, variance, and trend statistics to represent the features. For temporal discrete features such as diagnoses and medications, we use the event frequency (e.g., number of occurrences of a specific International Classification of Diseases code). For complex variables, like medication prescribing, we model medication use as a time dependent variable and also express medication usage (i.e., percent of days pills may have been used) at different time intervals before the index date, taking account of relevant changes (i.e., intensification, de-intensification, switching, multiple therapies) for a given condition.

3 Top-k Stability Selection

Stability selection is a feature selection method recently proposed to address the issues of selecting proper amount of regularization in existing embedded feature selection algorithms [19]. In this section we introduce a more general top-k stability selection, which 1) is more robust against data perturbation and less sensitive to parameters compared to original stability selection; 2) possesses improved theoretical guarantee on the false positive of features selected.

3.1 Algorithm In this section we describe the detailed process of the stability selection using sparse logistic regression, and introduce the top-k stability selection. Sparse logistic regression serves as an embedded feature selection algorithm that simultaneously performs feature selection and classification. Given binary class label $y \in \{+1, -1\}$ and input vector $\mathbf{x} \in \mathbb{R}^p$, where p is the dimension of feature space, logistic regression assumes that the posterior probability of a class y is a logistic sigmoid function acting on the linear combination of features $\mathbf{x}$ [1], which is given by:

$$P(y|\mathbf{x}) = \frac{1}{1 + \exp(-y(\mathbf{w}^T\mathbf{x} + b))},$$

where $\mathbf{w} \in \mathbb{R}^p$ and b are model parameters. Given training data $T = \{\mathbf{x}_i, y_i\}_{i=1}^n$, the training of logistic regression is to minimize the empirical loss function:

$$\begin{aligned} f(\mathbf{w}, b) &= -\frac{1}{n} \log \prod_{i=1}^{n} P(y_i|\mathbf{x}_i) \\ &= \frac{1}{n} \sum_{i=1}^{n} \log(1 + \exp(-y_i(\mathbf{w}^T\mathbf{x}_i + b))). \end{aligned}$$

Sparse logistic regression adds a sparsity-inducing ℓ_1-norm regularization on the model parameter $\mathbf{w}$ [14], which solves the following optimization problem:

$$\min_{\mathbf{w},b} f(\mathbf{w}, b) + \lambda\|\mathbf{w}\|_1,$$

where λ is the regularization parameter that controls complexity (sparsity) of the model. Let F be the index set of features, and let $f \in F$ denote a feature. For a feature f, if $\mathbf{w}_f \neq 0$ then the feature is included in the model. The sparse logistic regression is sensitive to parameter and therefore it is hard to control the exact number of features included in the model.

Given a set of regularization parameter values Λ and an iteration number β, the stability selection iteratively performs the following procedure. Let $D_{(j)}$

be a random subsample from T of size $\lfloor n/2 \rfloor$ drawn without replacement. For a given $\lambda \in \Lambda$, let $\hat{\mathbf{w}}^{(j)}$ be the optimal solution of sparse logistic regression on $D_{(j)}$. Denote $\hat{S}^{\lambda}(D_{(j)}) = \{f : \hat{\mathbf{w}}_f^{(j)} \neq 0\}$ as the features selected by $\hat{\mathbf{w}}^{(j)}$. The procedure is repeated for β times and selection probability $\hat{\Pi}_f^{\lambda}$ is computed as $\frac{1}{\beta}\sum_{j=1}^{\beta} \mathbf{1}\left\{f \in \hat{S}^{\lambda}(D_{(j)})\right\}$, where $\mathbf{1}\{\cdot\}$ is the indicator function. The stability score of f is given by:

$$\mathcal{S}_{\mathrm{SS(1)}}(f) = \max_{\lambda \in \Lambda}(\hat{\Pi}_f^{\lambda}). \tag{3.1}$$

There are two methods of choosing features from the stability score: the first one is to give a threshold π_{thr} and select features that have higher stability score than this threshold. Second, given the number of features we desired to include in the model, we can choose top features ranked by stability score as in filters-based feature selection methods; this is the method used in our experiments.

Stability selection resembles a mixture of multiple models of different regularization parameters by using the maximum selection probability as in Eq. (3.1). The choice of using the maximum selection probability as the stability score is questionable in some discussions of the paper [19]. In this paper, we address this issue by computing the stability score in an alternative way; specifically, we choose the average of top-k selection probabilities:

$$\mathcal{S}_{\mathrm{SS(k)}}(f) = \sum_{\lambda:\hat{\Pi}_f^{\lambda} \text{ranks top-}k} \hat{\Pi}_f^{\lambda}/k. \tag{3.2}$$

It follows that the original method of computing stability score in Eq. (3.1) is a special case of this more general form in Eq. (3.2) given $k = 1$. Intuitively, as k increases the selection score uses information from more parameters. In this sense, given a large k the stability selection boosts features that are more stable compared to others in the given parameter range Λ. Features will have high scores if they perform well in many parameter settings. On the other hand, stability selection with a small k boosts features that are sensitive to the parameters. A feature may have a high score even if it performs well only in a few parameter settings. To achieve the best performance, the selection of k is data-dependent and can be determined by using the standard cross-validation technique on the training data. Since the score of different k can be easily calculated given the selection probabilities, the cross-validation can be used at very low cost. The algorithm of computing top-k stability selection is given in Algorithm 1. Though the paper focuses on top-k stability selection for classification problems, it can also be applied to the feature selection in the regression settings, for example, using Lasso in top-k stability selection.

Algorithm 1 Top-k Stability Selection

```
Input: dataset $T = \{\mathbf{x}_i, y_i\}_{i=1}^n$, iteration number $\beta$,
    parameter set $\Lambda$
Output: top-$k$ stability scores $\mathcal{S}_{\mathrm{SS(k)}}(f)$ for all $f \in F$
for $j = 1$ to $\beta$ do
    subsample $D_{(j)}$ from $T$ without replacement
    for $\lambda \in \Lambda$ do
        compute the sparse logistic regression on $D_{(j)}$
        using parameter $\lambda$ and obtain the result $\hat{\mathbf{w}}^{(j)}$
        store indices of selected features:
            $\hat{S}^{\lambda}(D_{(j)}) = \{f : \hat{\mathbf{w}}_f^{(j)} \neq 0\}$
    end for
end for
for $f \in F$ do
    compute selection probability for all $\lambda \in \Lambda$:
        $\hat{\Pi}_f^{\lambda} = \frac{1}{\beta}\sum_{i=1}^{\beta} \mathbf{1}\left\{f \in \hat{S}^{\lambda}(D_{(i)})\right\}$
    compute top-$k$ stability score:
        $\mathcal{S}_{\mathrm{SS(k)}}(f) = \sum_{\lambda:\hat{\Pi}_f^{\lambda}\text{ranks top}-k} \hat{\Pi}_f^{\lambda}/k$
end for
```

Another open question in stability selection is how to obtain the set of parameters Λ. The stability selection is said to be not sensitive to the selected regularization parameters [19]. Practically, however, the values of the parameter should not be too large, or very few features are selected in each iteration; nor should they be very small, in which case the selection probability will be very high for all the features. The selection of the parameters also depends on the data set used. Some datasets may be very sensitive to the parameter chosen, in terms of sparsity in the model of sparse logistic regression. In this paper we use a heuristic algorithm to compute a proper set of parameters Λ. The algorithm firstly uniformly samples a set of parameters from the entire range and finds minimum $\lambda_{\min}$ and maximum $\lambda_{\max}$ that give us $p/3$ and 1 feature(s) respectively, and we uniformly choose 8 parameters in this range. In the case that the data is sensitive to the parameter, whose $\lambda_{\min}$ and $\lambda_{\max}$ reduce to the same value λ_{s}, the algorithm iterates the above search algorithm on $[\lambda_{\mathrm{s}} - \varepsilon, \lambda_{\mathrm{s}} + \varepsilon]$, where ε is a small value that reduces in each iteration.

One of the key technical contributions of this paper is to establish the upper bound of the average number of falsely selected features for the proposed top-k stability selection in the next section.

3.2 Theoretical Analysis In this section, we show the theoretical upper bound of the average number of falsely selected features in top-k stability selection. The error bound of original stability selection in [19] is a special case of our established bound when $k = 1$. We show that the error bound in our paper is guaranteed to be at least as tight as the bound when $k = 1$, and we show that the proposed top-k stability selection is more robust by utilizing more information from selection probabilities.

Let $\mathbf{w} \in \mathbb{R}^p$ be the ground truth of the sparse model, S be the set of non-zero components of $\mathbf{w}$, i.e., $S = \{f : \mathbf{w}_f \neq 0\}$. Let T be the total n samples $\{1, 2, \dots, n\}$ and D be a random subsample of T with size $\lfloor n/2 \rfloor$ drawn without replacement. For simplicity, we denote the selected features $\hat{S}^\lambda(T)$ by $\hat{S}^\lambda$ except when the dependence on data is necessary to be clarified. The set of zero components of $\mathbf{w}$ is denoted by $N = \{f : \mathbf{w}_f = 0\}$ and the estimated set of zeros under the model with regularization parameter λ is $\hat{N}^\lambda$. It follows that $\hat{S}^\Lambda = \bigcup_{\lambda\in\Lambda} \hat{S}^\lambda$, $\hat{N}^\Lambda = \bigcup_{\lambda\in\Lambda} \hat{N}^\lambda$. The average number of selected features from models built on Λ is $u_\Lambda = \mathbb{E}\left(\left|\hat{S}^\Lambda\right|\right)$. The selection probability of a subset of features $F \subseteq \{1, 2, \dots, p\}$ under model with λ is $\hat{\Pi}_F^\lambda = P^*\left(F \subseteq \hat{S}^\lambda(D)\right)$, where the probability P^* is with respect to the randomness of subsampling. By averaging the top-k selection probabilities of features, we define the top-k stable features as follows:

Definition 1. (Top-k Stable Features) *For a cut-off threshold π_{thr} with $0 < \pi_{thr} < 1$ and a set of regularization parameters Λ, the set of top-k stable features is defined as*

$$\hat{S}_k^{stable} = \left\{ f : \mathcal{M}_k^{\lambda\in\Lambda}(\hat{\Pi}_f^\lambda) \geq \pi_{thr} \right\},$$

where $\mathcal{M}_k^{\lambda\in\Lambda}(\hat{\Pi}_f^\lambda) = \frac{1}{k}\sum_{i=1}^k \max_{\lambda\in\Lambda_f^{-i}}\left(\hat{\Pi}_f^\lambda\right)$, Λ_f^{-i} *is defined as* $\Lambda \backslash \left\{\lambda_f^1, \lambda_f^2, \dots, \lambda_f^{i-1}\right\}$, $i = 1, \dots, k$, $\lambda_f^i = \operatorname*{argmax}_{\lambda\in\Lambda_f^{-i}}\left(\hat{\Pi}_f^\lambda\right)$, $\Lambda_f^{-1} = \Lambda$.

Note that the top-k stable feature is defined with respect to a single feature f. The definition and error control can be easily extended to the case of a subset of features $F \subseteq \{1, 2 \dots, p\}$. Let V be the set of falsely selected features of top-k stability selection, $V_k = |N \bigcap \hat{S}_k^{\text{stable}}|$. The average number of falsely selected features by top-k stability selection, $\mathbb{E}(V_k)$ is guaranteed to be controlled by the following theorem:

Theorem 3.1. (Top-k Error Control) *Assume that the original feature selection method is not worse than random guessing, i.e.*

$$\frac{\mathbb{E}\left(\left|S \bigcap \hat{S}^\Lambda\right|\right)}{|S|} \geq \frac{\mathbb{E}\left(\left|N \bigcap \hat{S}^\Lambda\right|\right)}{|N|},$$

and for $\forall\lambda \in \Lambda$ *the distribution of* $\left\{\mathbf{1}\left\{f \in \hat{S}^\lambda\right\}, f \in N\right\}$ *is exchangeable, then,*

$$\mathbb{E}(V_k) \leq \frac{\sum_{i=1}^k u_{\Lambda,i}^2}{k \cdot p \cdot (2\pi_{thr} - 1)}, \tag{3.3}$$

where $\frac{1}{2} < \pi_{thr} < 1$, $u_{\Lambda,i}^2 = \mathbb{E}_f[\Lambda_f^{-i}]^2$, $i = 1, 2, \dots, k$, Λ_f^{-i} *is defined the same as in the definition of top-k stable features.*

The proof of the Theorem 3.1 is given in the supplemental materials [29]. Obviously, the error bound $\mathbb{E}(V_k) \leq \frac{1}{2\pi_{\text{thr}}-1}\frac{u_\Lambda^2}{p}$ in [19] is a special case of our bound in Eq. (3.3) given $k = 1$. For each $u_{\Lambda,i}, i = 1, 2, \dots, k$, it measures the average of $u_{\Lambda_f^{-i}}$ with respect to all features f. Since $u_{\Lambda,i}^2 \leq u_\Lambda^2$ for $i = 1, 2, \dots, k$, we have $\sum_{i=1}^k u_{\Lambda,i}^2/k \leq u_\Lambda^2$, which says that the upper bound of error in top-k stability selection is guaranteed to be at least as tight as the original bound. $u_{\Lambda,i}^2$ expresses the average number of selected features from Λ excluding top $i-1$ λ's maximizing the selection probability of the features. By utilizing the information from k $u_{\Lambda,i}^2$'s, the top-k stability selection is more robust in terms of average number of falsely selected features, compared with the $k = 1$ stability selection.

4 Experiments

In the experiment we study the empirical performance of feature selection algorithms and classification methods in both synthetic and real clinical prediction datasets. In the experiments the implementation of top-k stability selection uses Lasso and the sparse logistic regression from the SLEP package [15].

4.1 Simulation We use synthetic datasets to study the proposed top-k selection probability in the setting of regression. We start with generating the true model $\mathbf{w} \in \mathbb{R}^{1000}$. There are 50 non-zero elements at random locations generated with distribution $\mathcal{N}(0, 1)$. We then generate a data matrix of 100 samples $\mathbf{X} \in \mathbb{R}^{100\times1000}$ with elements generated i.i.d. by $\mathcal{N}(0, 1)$. The response is generated using $\mathbf{y} = \mathbf{X}\mathbf{w} + \epsilon$, where $\epsilon \sim \mathcal{N}(0, 0.5)$. We apply the top-$k$ stability selection on the data $(\mathbf{X}, \mathbf{y})$, and calculate the number of falsely selected features $|V_k| = |N \cap S_k^{\text{stable}}|$ and the ratio $|V_k|/|S_k^{\text{stable}}|$ using $k = \{1, 2, 4, 8, 10\}$. We repeat the experiments for 10

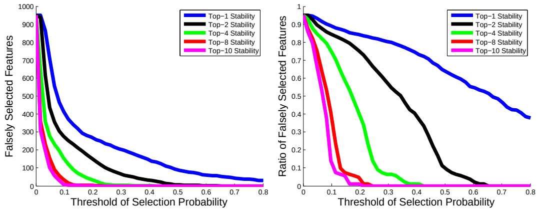

Figure 2: Comparisons of top-k stability selection with $k = 1, 2, 4, 8, 10$. When k increases, we observe improved empirical performance using top-k stability selection in terms of the number of false positive features included in the model (left) and the ratio (right), which is consistent with the analysis in Theorem 3.1.

iterations and report the averaged results in Figure 2. We observe that for a given threshold on the stability score, the top-k stability selection performs better when $k > 1$. When k increases, we observe improved empirical performance using the top-k stability selection in terms of both the number and ratio of the false positive features included in the model, which is consistent with the discussion in Theorem 3.1.

4.2 Clinical Application Setting The individual and societal impact of heart failure (HF) is staggering. One in five US citizens over age 40 is expected to develop HF during their lifetimes. It is currently the leading cause of hospitalization among Medicare beneficiaries. Framingham risk criteria [16], being the most commonly used diagnostic criteria for HF, are signs and symptoms that are often buried in clinical notes, which may not be in the structured fields in the EHR. Framingham criteria include Acute pulmonary edema (APEdema), Ankle Edema, Dyspnea on ordinary exertion, and etc. Despite their importance, Framingham criteria are not systematically documented in structured EHR. Processing clinical notes requires manual annotations or building customized natural language processing (NLP) applications, which are both expensive and time-consuming. Plenty of structured EHRs, on the other hand, are available, which include diagnosis, medication, and lab results.

Our goal here is to construct features from structured EHR data and to use only a small number of labeled annotations of Framingham criteria to build predictive models, so that we can identify individuals that are likely to have Framingham criteria (though they are not documented). The reasons for identifying potentially missing Framingham criteria are 1) to enable timely diagnosis of HF, and 2) to use the Framingham criteria as additional features for predicting future onset of HF. In this paper we construct 9 datasets targeting different Framingham criteria from the EHR of approximately 4000 patients over 5 years from a hospital. The samples size of the data is given in Table 1, where each sample corresponds to exactly one patient in the study. The number of features is 132, which include 79 diagnosis features, 36 medication features, 17 lab test features.

Table 1: The sample size of the EHR data sets.

Dataset#	Task Name	Training		Testing	
		+	-	+	-
1	APEdema	424	1576	424	1576
2	AnkleEdema	1205	794	1206	795
3	ChestPain	809	1190	810	1191
4	DOExertion	1065	935	1065	935
5	Neg-Hepatomegaly	1537	463	1537	463
6	Neg-JVDistention	1232	767	1233	768
7	Neg-NightCough	1070	929	1071	930
8	Neg-PNDyspnea	869	1130	870	1131
9	Neg-S3Gallop	1552	447	1553	448

4.3 Feature Selection Comparison In this experiment we evaluate the performance of top-k stability selection given different k and existing feature selection methods including Fisher Score [5], ReliefF [10, 12], Gini Index [8], Info Gain [3], $\chi^{\mathbf{2}}$ [13], Minimum-Redundancy Maximum-Relevance (mRMR) [21]. Note that for the feature selection methods that only take categorical inputs, we can still use continuous features by applying discretization techniques [4]. For feature selection algorithms other than stability selection (SS), we use the implementations from an existing package [28]. For each feature selection algorithm, we compute the feature ranking on the training data. We then use top t features to build classification model using logistic regression and evaluate the model on the testing data, where we vary t from 2 to 20. To study the effects of top-k stability selection when k changes, we treat $k = 1, 2, 4, 8$ as different feature selection algorithms in the evaluation.

Since the class distribution of the data sets is not balanced, we focus on the Area Under Curve (AUC) metric. Given a specific feature number, for all feature selection methods we select the number of top features ranked respectively and build classification models using logistic regression on the training data, and the models are then evaluated using the testing data. We repeat this procedure for 10 times, and the average AUC over the 10 times is used to rank the feature selection algorithms. We report the mean rank and standard deviation of 9 datasets in Table 2. We highlight the top 3 feature selection algorithms according to the mean rank. One important observation is that stability selection algorithms perform better than all other feature selection algorithms. The performance results also demonstrate the potential of top-k stability selection with $k \geq 2$. These results are consistent with our theoretical analysis in Section 3.2.

Sparse Learning Comparison. In our experiments, the stability selection uses sparse logistic regression,

Table 2: Average AUC ranks of feature selection algorithms over the 9 datasets.

Feature #	4	6	8	10	12	14	16	18	20
ReliefF	10.0 ± 0.0	10.0 ± 0.0	10.0 ± 0.0	10.0 ± 0.0	10.0 ± 0.0	10.0 ± 0.0	10.0 ± 0.0	10.0 ± 0.0	10.0 ± 0.0
Fisher	7.0 ± 2.1	7.2 ± 1.5	6.2 ± 2.4	6.0 ± 2.6	5.8 ± 2.6	5.6 ± 2.2	4.9 ± 2.8	5.0 ± 2.1	5.6 ± 1.5
Gini	7.0 ± 1.5	7.1 ± 2.0	6.0 ± 1.8	6.8 ± 1.8	6.9 ± 2.1	6.4 ± 2.8	5.9 ± 2.0	6.6 ± 2.0	6.1 ± 2.1
InfoGain	4.7 ± 2.3	6.2 ± 2.0	5.7 ± 1.7	5.7 ± 1.7	5.8 ± 1.1	5.9 ± 1.3	6.0 ± 2.4	7.0 ± 1.6	6.8 ± 1.7
ChiSquare	5.3 ± 2.2	6.8 ± 1.8	6.4 ± 2.5	6.6 ± 1.6	6.2 ± 1.4	5.8 ± 2.0	6.7 ± 1.4	7.2 ± 1.5	7.2 ± 1.6
mRMR	5.8 ± 2.6	5.7 ± 1.7	7.1 ± 1.5	7.3 ± 1.7	7.8 ± 1.5	7.6 ± 1.6	7.2 ± 1.7	6.9 ± 2.4	6.4 ± 3.4
SS($k = 1$)	5.1 ± 3.0	4.1 ± 2.2	4.8 ± 2.5	4.4 ± 2.5	4.6 ± 2.7	4.2 ± 3.6	4.7 ± 3.2	3.8 ± 2.4	4.8 ± 2.3
SS($k = 2$)	$\mathbf{3.8 \pm 3.0}$	$\mathbf{3.0 \pm 2.4}$	$\mathbf{2.8 \pm 2.0}$	$\mathbf{2.4 \pm 1.9}$	$\mathbf{3.2 \pm 2.0}$	$\mathbf{3.7 \pm 2.1}$	$\mathbf{3.2 \pm 2.7}$	$\mathbf{3.3 \pm 1.9}$	$\mathbf{3.0 \pm 1.7}$
SS($k = 4$)	$\mathbf{3.0 \pm 1.8}$	$\mathbf{2.3 \pm 1.6}$	$\mathbf{2.8 \pm 2.3}$	$\mathbf{2.2 \pm 1.6}$	$\mathbf{2.6 \pm 1.6}$	$\mathbf{2.8 \pm 1.6}$	$\mathbf{2.9 \pm 1.4}$	$\mathbf{2.3 \pm 1.3}$	$\mathbf{2.3 \pm 1.0}$
SS($k = 8$)	$\mathbf{3.3 \pm 1.9}$	$\mathbf{2.6 \pm 1.2}$	$\mathbf{3.2 \pm 2.4}$	$\mathbf{3.6 \pm 1.9}$	$\mathbf{2.2 \pm 1.5}$	$\mathbf{3.1 \pm 1.8}$	$\mathbf{3.6 \pm 1.9}$	$\mathbf{2.9 \pm 1.7}$	$\mathbf{2.8 \pm 1.8}$

Table 3: Classification performance on EHR datasets in terms of AUC. We compare three classifiers: support vector machine (SVM), random forest (RanF) and logistic regression (LReg).

Dataset	1	2	3	4	5	6	7	8	9
SVM	62.2 ± 3.0	71.9 ± 0.8	68.8 ± 1.2	75.2 ± 0.7	59.0 ± 1.4	64.3 ± 1.3	67.1 ± 0.5	64.5 ± 0.6	56.1 ± 2.0
RanF	69.6 ± 1.1	73.7 ± 0.9	69.7 ± 0.7	75.7 ± 0.6	59.3 ± 2.1	63.9 ± 0.6	66.0 ± 1.1	65.3 ± 1.3	52.3 ± 5.6
LReg	$\mathbf{70.0 \pm 1.1}$	$\mathbf{75.0 \pm 0.9}$	$\mathbf{69.7 \pm 0.6}$	$\mathbf{76.6 \pm 0.9}$	$\mathbf{65.2 \pm 1.3}$	$\mathbf{67.1 \pm 0.9}$	$\mathbf{67.6 \pm 0.8}$	$\mathbf{66.7 \pm 0.7}$	$\mathbf{63.9 \pm 0.9}$

which is the most popular sparse learning method for classification and also serves as an embedded feature selection method itself via the ℓ_1-norm penalty. This motivates us to investigate the performance of directly applying sparse logistic regression to the data sets. We vary the regularization parameter of sparse logistic regression and evaluate the performance in terms of AUC. In order to compare the performance, we perform the top-4 stability selection on the same training data and use the same number of features as selected in the model of sparse logistic regression to build models using logistic regression. We show the results of AnkleEdema and DOExertion datasets in Figure 3. Since different regularization parameters may give the same number of features, there may be more than one value at each feature number. We observe that when the same number of features are selected, the approach of using stability selection outperforms sparse logistic regression. Also note that sparse logistic regression is sensitive to the parameter and the estimation of which requires cross-validation. In the approach of logistic regression after stability selection, however, the performance is not very sensitive to the number of features selected. We observe similar patterns given different k values.

4.4 Classification Methods Comparison In this experiment we study the performance of popular classifiers on the EHR data sets. We include SVM, random forest (RanF), and logistic regression (LReg) in the study. For SVM, 5-fold cross validation is used to estimate the best parameter c. In random forest we fix the tree number to be 500. When building the classification models, we include top 20 features ranked by stability selection ($k = 4$). The performance of the classifiers on the 9 datasets are given in Table 3. We observe that in all data sets logistic regression performs better than other classifiers in terms of AUC.

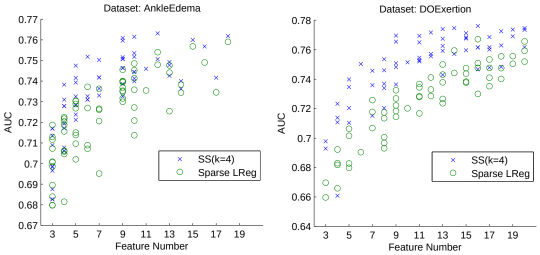

Figure 3: The AUC of sparse logistic regression (Sparse LReg), logistic regression after top-4 stability selection (SS $k = 4$) for 10 randomly splitting training/testing datasets. In each iteration we vary the regularization parameter of the Spare LReg and choose the same number of features in SS. Note that different regularization parameters may give the same number of features.

Sensitivity analysis. We perform experiments to study the sensitivity of methods with respect to their parameters. We randomly split the AnkleEdema dataset into training and testing sets (we observe similar patterns in other datasets). We perform stability selection ($k = 4$) on the training data, and choose top 20 features to build the classification models using different parameters. We repeat the procedure for 10 times and report AUC. In SVM we vary the parameter c, such that $\log(c)$ is from -4 to $+6$ in the study. For random forest we vary the tree number from 50 to 1000. In logistic regression, we vary the ℓ_2-norm regularization term λ in the range $\log(\lambda) \in [-4, 6]$. The results are given in Figure 4. We find that the performance of SVM and logistic regression on the dataset is very sensitive to the parameter. For logistic regression, we obtain better performance for a smaller parameter value, which leads to less regularization. This is because when a small number of features are included in the model, it is not necessary to add ℓ_2-norm regularization. For SVM, however, a cross-validation is necessary in order to select the best parameter. Unlike SVM and LReg, random forest is not sensitive to the tree number parameter and the performance remains almost the same when the tree number is large enough.

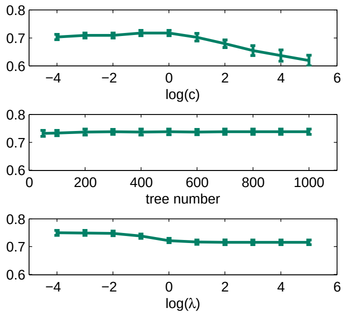

Figure 4: The sensitivity of classification methods with respect to varying parameters. SVM (top) is sensitive to parameter c; random forest (middle) is not sensitive to the tree number parameter; logistic regression (bottom) is sensitive to the regularization parameter (λ).

4.5 Top-k Stability Score Case Study In this experiment we study the top-k stability scores given different k. We show the distribution of stability scores for DOExertion dataset in Figure 5. We present the top features to the clinical experts and confirm their clinical validity in many cases. For example, G1-58 is actually shortness of breath (a symptom diagnose), which includes DOExertion as a special case. G3-33 are sympathomimetic agents, used as nasal decongestants to help breathing.

Moreover, we observe that the absolute value of stability score shrinks when k increases. The shrinkage is expected because we average the selection probabilities over more regularization parameters. In the stability score distribution given $k = 1$, we find that the average stability score decreases continuously. As k increases, we find that some steep drops of stability score emerge. In Figure 5 ($k = 2$), there is a drop between feature G1-44 and G3-4. When we increase k to $k = 8$, we find a more significant drop after the first three features. Such drop information can potentially be used as a guidance for the number of features to be included, or equivalently the choice of threshold in stability selection.

We are also aware that the rank of features can be different given different k. The feature G3-16, cardio selective beta blocker, has a subtle connection to DOExertion, since this is often prescribed among HF patients, and DOExertion is a common symptom among HF patients. In particular, G3-16 ranks 11 given $k = 1$. Its rank improves to 10, 6 and 4, respectively, when k increases to 2, 4 and 8. The improvement in rank implies that the feature's selection probability may not be highest with any regularization parameter λ, but across all parameters Λ the feature's stability probabilities are consistently high. Such features like G3-16 in this case are considered to be stable with respect to parameters, which can only be identified with top-k stability selection.

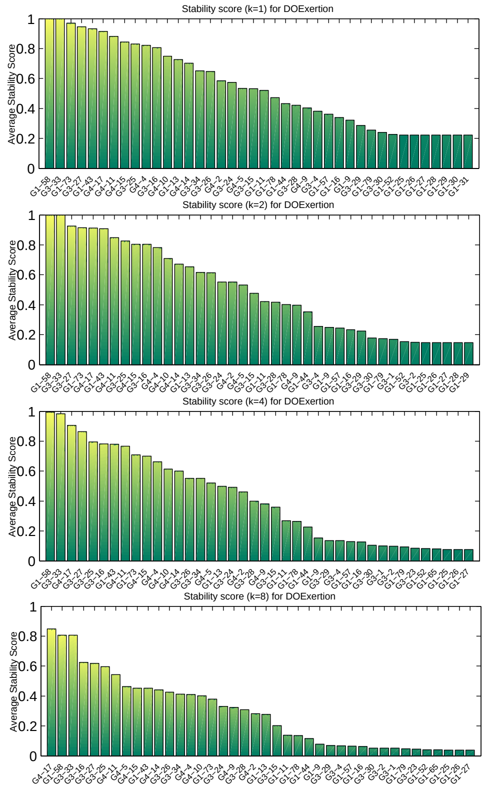

Figure 5: Average stability scores of DOExertion.

5 Conclusion

The goal of disease-specific risk prediction is to assess the risk of a patient in developing a target disease based on his/her health profile. As electronic health records (EHRs) become more prevalent, a large number of features can be constructed in order to characterize patient profiles. In this paper, we propose top-k stability selection, which generalizes a powerful sparse learning feature selection method by overcoming its limitation on parameter selection and providing stronger theoretical properties. In a large set of real clinical prediction datasets, the top-k stability selection methods outperform the original stability selection and many existing feature selection methods. We compare three competitive classification methods to demonstrate the effectiveness of selected features by our proposed method. We also show that in several clinical applications on predicting heart failure related symptoms, top-k stability selection can successfully identify important features that are clinically meaningful.

Recently the data mining and machine learning community has integrated its research efforts for structure sparsity. In our future works, we plan to combine stability selection and structure sparsity and apply to research on EHRs. Top-k stability selection is based on Lasso and therefore is vulnerable to strongly correlated features. We plan to develop methods that are more robust against such ill-conditioned design matrices.

Acknowledgments

We would like to thank Dr. Steven Steinhubl and Dr. Walter F. Stewart from Geisinger for facilitating the testing of our algorithm. We also want to thank Roy Byrd from IBM for providing symptom annotations on the data. This work was supported in part by NSF (IIS-0953662, CCF-1025177) and NIH (R01LM010730).

References

[1] C. Bishop. *Pattern recognition and machine learning*, volume 4. Springer New York, 2006.

[2] S. Boucheron and O. Bousquet. Concentration inequalities. In *Advanced Lectures in Machine Learning*, pages 208–240. Springer, 2004.

[3] T. Cover, J. Thomas, J. Wiley, et al. *Elements of information theory*, volume 6. Wiley Online Lib., 1991.

[4] J. Dougherty, R. Kohavi, and M. Sahami. Supervised and unsupervised discretization of continuous features. In *ICML*, pages 194–202, 1995.

[5] R. Duda, P. Hart, and D. Stork. Pattern classification and scene analysis 2nd ed. 1995.

[6] H. Eleftherohorinou, C. Hoggart, V. Wright, M. Levin, and L. Coin. Pathway-driven gene stability selection of two rheumatoid arthritis gwas identifies and validates new susceptibility genes in receptor mediated signalling pathways. *Hum. Mol. Genet.*, 20(17):3494–3506, 2011.

[7] G. Fonarow, K. Adams, W. Abraham, C. Yancy, W. Boscardin, et al. Risk stratification for in-hospital mortality in acutely decompensated heart failure. *J. of the American Medical Association*, 293(5):572, 2005.

[8] C. Gini. Variabilità e mutabilità. *Reprinted in Memorie di metodologica statistica (Ed. Pizetti E, Salvemini, T). Rome: Libreria Eredi Virgilio Veschi*, 1, 1912.

[9] L. Jacob, G. Obozinski, and J. Vert. Group lasso with overlap and graph lasso. In *ICML*, pages 433–440. ACM, 2009.

[10] K. Kira and L. Rendell. A practical approach to feature selection. In *Intl. Workshop on Machine Learning*, pages 249–256, 1992.

[11] B. Krishnapuram, L. Carin, M. Figueiredo, and A. Hartemink. Sparse multinomial logistic regression: Fast algorithms and generalization bounds. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(6):957–968, 2005.

[12] H. Liu and H. Motoda. *Computational methods of feature selection*. Chapman & Hall, 2008.

[13] H. Liu and R. Setiono. Chi2: Feature selection and discretization of numeric attributes. In *Intl. Conf. on Tools with Artificial Intelligence*, pages 388–391, 1995.

[14] J. Liu, J. Chen, and J. Ye. Large-scale sparse logistic regression. In *KDD*, pages 547–556. ACM, 2009.

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

[16] P. McKee, W. Castelli, P. McNamara, and W. Kannel. The natural history of congestive heart failure: the framingham study. *New England Journal of Medicine*, 285(26):1441–1446, 1971.

[17] L. Meier, S. Van De Geer, and P. Bühlmann. The group lasso for logistic regression. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 70(1):53–71, 2008.

[18] N. Meinshausen and P. Bühlmann. High-dimensional graphs and variable selection with the lasso. *The Annals of Statistics*, 34(3):1436–1462, 2006.

[19] N. Meinshausen and P. Bühlmann. Stability selection. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 72(4):417–473, 2010.

[20] Y. Pati, R. Rezaiifar, and P. Krishnaprasad. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In *Asilomar Conference on Signals, Systems and Computers*, pages 40–44. IEEE, 1993.

[21] H. Peng, F. Long, and C. Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(8):1226–1238, 2005.

[22] Z. Ryali, T. Chen, K. Supekar, and V. Menon. Estimation of functional connectivity in fmri data using stability selection-based sparse partial correlation with elastic net penalty. *Neuroimage*, 59(1):3852–3861, 2012.

[23] D. Stekhoven, L. Hennig, G. Sveinbjörnsson, I. Moraes, M. Maathuis, and P. Bühlmann. Causal stability ranking. 2011.

[24] M. Stern, K. Williams, D. Eddy, and R. Kahn. Validation of prediction of diabetes by the archimedes model and comparison with other predicting models. *Diabetes Care*, 31(8):1670, 2008.

[25] R. Tibshirani. Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, pages 267–288, 1996.

[26] M. Vounou, E. Janousova, R. Wolz, J. Stein, P. Thompson, D. Rueckert, and G. Montana. Sparse reduced-rank regression detects genetic associations with voxel-wise longitudinal phenotypes in alzheimer's disease. *NeuroImage*, 2011.

[27] J. Wu, J. Roy, and W. Stewart. Prediction modeling using ehr data: challenges, strategies, and a comparison of machine learning approaches. *Medical care*, 48(6):S106, 2010.

[28] Z. Zhao, F. Morstatter, S. Sharma, S. Alelyani, A. Anand, and H. Liu. Asu feature selection repository, featureselection.asu.edu, 2010.

[29] J. Zhou. http://www.public.asu.edu/∼jzhou29/papers/jzhousdm13_appendix.pdf.