CRISP: Consensus Regularized Selection based Prediction

Ping Wang
Dept. of Computer Science
Virginia Tech
Arlington, VA - 22203.
ping@vt.edu

Karthik K. Padthe
Dept. of Computer Science
Wayne State University
Detroit, MI - 48202.
karthikp@wayne.edu

Bhanukiran Vinzamuri
Dept. of Computer Science
Wayne State University
Detroit, MI - 48202.
bhanukiranv@wayne.edu

Chandan K. Reddy
Dept. of Computer Science
Virginia Tech
Arlington, VA - 22203.
reddy@cs.vt.edu

ABSTRACT

Integrating regularization methods with standard loss functions such as the least squares, hinge loss, etc., within a regression framework has become a popular choice for researchers to learn predictive models with lower variance and better generalization ability. Regularizers also aid in building interpretable models with high-dimensional data which makes them very appealing. It is observed that each regularizer is uniquely formulated in order to capture data-specific properties such as correlation, structured sparsity and temporal smoothness. The problem of obtaining a consensus among such diverse regularizers while learning a predictive model is extremely important in order to determine the optimal regularizer for the problem. The advantage of such an approach is that it preserves the simplicity of the final model learned by selecting a single candidate model which is not the case with ensemble methods as they use multiple candidate models for prediction. This is called the *consensus regularization* problem which has not received much attention in the literature due to the inherent difficulty associated with learning and selecting a model from an integrated regularization framework. To solve this problem, in this paper, we propose a method to generate a committee of non-convex regularized linear regression models, and use a consensus criterion to determine the optimal model for prediction. Each corresponding non-convex optimization problem in the committee is solved efficiently using the cyclic-coordinate descent algorithm with the generalized thresholding operator. Our Consensus RegularIzation Selection based Prediction (CRISP) model is evaluated on electronic health records (EHRs) obtained from a large hospital for the congestive heart failure readmission prediction problem. We also evaluate our model on high-dimensional synthetic datasets to assess its performance. The results indicate that CRISP outperforms several state-of-the-art methods such as additive, interactions-based and other competing non-convex regularized linear regression methods.

CIKM'16 , October 24-28, 2016, Indianapolis, IN, USA
© 2016 ACM. ISBN 978-1-4503-4073-1/16/10…$15.00
DOI: http://dx.doi.org/10.1145/2983323.2983779

Keywords

Consensus prediction; regularization; regression.

1. INTRODUCTION

Consensus modeling is an important topic which deals with assembling a committee of experts for a given problem and then obtaining a consensus among their votes to arrive at the final prediction. This has been applied to predictive analytics problems such as classification, ensemble modeling and active learning where a committee of models are created to cast their individual votes on a test case. Multiple classifier fusion is an application of consensus modeling where multiple classifiers are integrated within a single framework [1]. Query by Committee is also a well studied topic in the context of active learning where consensus modeling is used to determine the instance whose label must be queried [2, 3]. The effectiveness of consensus modeling in such scenarios like classification and active learning relies on the mechanism used to build the committee of models. Consensus modeling can be extended to the field of regularization in the context of regression which is described to be the *consensus regularization* problem in this paper.

Consensus regularization is the problem of identifying an optimal regularizer for a given regression problem among a set of regularized models by obtaining a consensus among all these models. The consensus is obtained using a predefined criterion which assesses each of the candidate regularizers separately and decides the best candidate regularizer for prediction. Solving such a problem is non-trivial since it is not easy to integrate multiple regularizers within a single framework. This is because the regularizers differ in their degree of complexity and how they interpret the inherent data structure which makes this problem of integration highly cumbersome. Optimization methods such as proximal algorithms also cannot be universally applied to solve multiple regularization problems because the cost of obtaining the proximal operator associated with each regularizer may significantly differ [4, 5]. Finally, ensuring diversity of regularizers within a multiple regularizer framework is not always guaranteed. This is the reason why the problem of unifying multiple regularizers has not received much attention in the data mining community.

To solve this problem, in this paper, we propose a two-step algorithm. The first step generates a committee of regularization models. Each model in this committee differs from the others, but the solution for each one of them can be expressed using a unique generalized thresholding operator [6–8]. The advantage of our approach is that this generalized thresholding operator can be computed efficiently for each individual model. In addition, to promote robustness in the model to capture sparsity more effectively, we use non-convex regularizers within our approach. Non-convex regularizers have certain unique advantages of unbiased feature selection and consistent results which make them a better choice compared to the prominent sparsity promoting convex regularizers such as the Lasso. We choose a non-convex regularizer called the minimax concave plus (MC+) penalty for the model proposed in this paper which is explained in Section 4.

The second stage of our approach involves using a *consensus criterion* among all these candidate regularizers to obtain the final model for prediction. A major advantage of our approach is that an expert can design an arbitrary consensus criterion and integrate it with this approach to obtain an optimal model for prediction. This is particularly important while building prediction models on real-world data where an expert aims at optimizing the model performance for domain-specific metrics.

We conduct extensive sets of experiments for this Consensus RegularIzed Selection based Prediction framework (CRISP) algorithm on electronic health records (EHRs) collected from a large hospital consisting of 8,000 patient records. We now summarize the major contributions of this paper.

- Propose a **C**onsensus **R**egular**I**zed **S**election based **P**rediction framework (CRISP) which builds a committee of non-convex regularized linear regression candidate models and integrates them using a consensus criterion to obtain the optimal model for prediction.

- Develop an efficient cyclic coordinate descent based solution for the optimization problem being solved for learning each candidate model in CRISP. We also provide proof of convergence for the proposed algorithm.

- Evaluate CRISP using state-of-the-art additive, interaction-based, and non-convex regularized linear regression models using metrics such as AUC, MSE and R^2. We also conduct experiments to assess the performance of CRISP on high-dimensional synthetic datasets.

In our evaluation, CRISP obtained very competitive AUC values for the 30-day and 365-day readmission problems compared to state-of-the-art regression methods. This paper is organized as follows: In Section 2, we provide a brief review of the related work on additive, hierarchical and non-convex regularized regression models. In Section 3, we provide the notations that are necessary for understanding the proposed CRISP model along with a brief overview of regularization theory. In Section 4, we present the details of the CRISP model including the MC+ penalty, the generalized thresholding operator and the corresponding cyclic coordinate descent algorithm employed in CRISP. In Section 5, we evaluate the performance of CRISP using various additive, interaction-based and non-convex regularized linear regression methods. Finally, we conclude our discussion and provide directions for future work in Section 6.

2. RELATED WORK

In this section, we review the existing works related to the topics of non-convex regularized linear regression, additive and interaction-based methods. We briefly mention how the contributions in this paper are distinctly different from these algorithms that are available in the literature.

- *Non-convex regularized linear regression models* [9,10]: Convex regularizers such as the ℓ_1 norm, ℓ_2 norm and the elastic net penalty are used popularly in the sparse learning literature [11–13]. However, based on some empirical studies, it has been observed that they are not perfect in capturing sparsity [14, 15]. In contrast, methods with non-convex penalties can recover sparsity more efficiently and are being actively pursued by researchers recently. Optimization methods such as Difference of Convex Functions (DC) programming, Alternating Direction Method of Multipliers (ADMM) and proximal algorithms are popular choices for solving such non-convex optimization problems efficiently.

- *Additive models* [16–18]: Generalized Additive Models (GAM) capture non-linear relationship between individual features and the response. However, the standard GAM does not perform well since it does not model any interactions between the features. To overcome this issue, generalized additive models plus interactions (GA2M) is proposed by adding selected terms of interacting pairs of features to GAM. In other words, GA2M consists of both univariate terms and a small number of pairwise interaction terms. The interactions can be determined by a greedy forward selection strategy for low-dimensional data and FAST interaction detection can be used for large high-dimensional datasets.

- *Interactions based models* [19, 20]: Additive models which only consider the main effects of the features are ineffective in many situations when predicting an outcome of interest. Regression models with interactions, which consider the effect of different features on the response variable except for the main effects, are more effective than additive models. In these models, the additive part corresponds to the main effect term and the quadratic part corresponds to the interaction term. In general, not all of the main effects and interactions are of interest, thus it is critical to select the variables of high significance. In statistics, a hierarchical structure between the main effects and interaction effects has been shown to be very effective in constraining the search space and identifying important individual features and interactions. Specifically, the hierarchical constraint requires that an interaction term is selected in the model only if its corresponding main effects are included. Strong theoretical properties have been established for such hierarchical models. We refer to these algorithms which model strong and weak interactions, in this paper, as hiernet-strong and hiernet-weak, respectively.

In contrast to these methods, our CRISP approach uses a non-convex penalty generating multiple candidate models in the process, and selects an optimal model using a consensus criterion among these candidate models for the final prediction.

3. PRELIMINARIES

This section introduces the preliminaries required to comprehend the proposed approach. First, the notations used in our work are presented in Table 1. We then review the concepts associated with regularized linear regression models followed by introducing the thresholding operators used in our CRISP algorithm. Eq. (1) describes the basic linear regression model which aims at estimating the relationship between the features $X = (x_1, x_2, ..., x_n)^T$ and the corresponding response variable $Y = (y_1, y_2, ..., y_n)^T$, where $x_i \in \mathbb{R}^m$ and $y_i \in \mathbb{R}$ for $i = 1, \ldots, n$.

Table 1: Notations used in this paper.

Name	Description
n	number of instances.
m	number of features.
X	$\mathbb{R}^{n \times m}$ feature vector matrix.
Y	$\mathbb{R}^n$ response variable.
β	$\mathbb{R}^m$ regression coefficient vector.
λ	scalar regularization parameter.
Λ	a vector of regularization parameters.
γ	scalar non-convexity parameter.
Γ	a vector of non-convexity parameters.
L	length of regularization sequence.
K	length of non-convexity sequence.
η	consensus matrix $\in \mathbb{R}^{L \times K}$ entries.
$P(\|\beta\|, \lambda, \gamma)$	a family of penalty functions.
$S(\tilde{\beta}, \lambda)$	soft-thresholding operator.
$H(\tilde{\beta}, \lambda)$	hard-thresholding operator.

$$Y = f(X) = X\beta + \epsilon. \tag{1}$$

In high-dimensional data, m is much greater than n. This motivates the use of a relatively small number of predictors to accurately predict the outcome.

$$\hat{\beta} = \arg \min_{\beta \in \mathbb{R}^m} \frac{1}{2} \sum_{i=1}^{n} (y_i - f(x_i))^2 + \lambda P(\beta) \tag{2}$$

Eq. (2) provides the standard regularized linear regression setting where λ is the penalty coefficient which controls the degree of regularization and $P(\beta)$ is a penalty function. A number of variable selection methods with convex penalty functions and the corresponding optimization methods have been proposed in the literature [11–13].

The ℓ_p norm for $p > 1$ does not provide a sparse solution. When $p \leq 1$, the solution is sparse. Lasso [11,21] with the ℓ_1 penalty function is convex and non-smooth which produces models with good prediction accuracy when the underlying model is reasonably sparse. The lasso penalty is often considered as the convex surrogate for the best-subset selection with the ℓ_0 penalty, $\| \beta \|_0 = \sum_{i=1}^{m} \mathbf{I}(|\beta_i| > 0)$, which penalizes the number of non-zero coefficients in the model, where $\mathbf{I}$ represents the indicator function.

However, there are two disadvantages for the lasso method. Empirical results show that the ℓ_1 penalty tends to generate biased estimates for large coefficients, which may prevent its consistent variable selection. In addition, lasso is effective at giving sparse solutions, but when variables are correlated, it excludes many correlated variables once a strong variable is included and fully fitted in the model. Also, when the regularity conditions are violated, the lasso can be sub-optimal in variable selection, which means it can fail as a variable selector. In order to include the full effect of a variable in the model, we have to relax the penalty to allow other redundant but possibly correlated features. Fan and Li [22] suggested some desirable properties of the penalization function, such as sparsity and unbiasedness of the estimated parameters.

To address these disadvantages associated with the lasso method, some non-convex penalty functions, which bridge the gap between ℓ_1 and ℓ_0 penalty, have also been considered. Non-convex penalties are known to be more efficient at recovering sparsity compared to convex penalties. We consider a generic dual parameter non-convex formulation from the literature as given in Eq. (3), where $P(\beta; \gamma)$ defines a family of penalty functions concave in $|\beta|$.

$$\hat{\beta} = \arg \min_{\beta \in \mathbb{R}^m} \frac{1}{2} \sum_{i=1}^{n} (y_i - f(x_i))^2 + \lambda P(\beta; \gamma) \tag{3}$$

In this optimization problem, both λ and γ are user provided parameters and they control the degree of the regularization and non-convexity, respectively. In other words, for a fixed λ, there will be a family of penalty functions, each of which corresponds to an optimization problem. This means that the penalty function $P(\beta; \gamma)$ can be updated to be $P(\beta; \lambda, \gamma)$ if we also consider λ as a parameter of the model. In addition, due to the fact that the penalty function is separable for the parameters $\beta = (\beta_1, \ldots, \beta_m)^T$, the optimization problem in Eq. (3) can be updated as follows after incorporating λ within the penalty function.

$$\hat{\beta} = \arg \min_{\beta \in \mathbb{R}^m} \frac{1}{2} \sum_{i=1}^{n} (y_i - f(x_i))^2 + \sum_{i=1}^{m} P(|\beta_i|; \lambda; \gamma) \tag{4}$$

In Eq. (4), for a fixed λ, the value of the parameter γ varies in the range of $[1+, \infty)$ where $1+$ represents values greater than 1. Each variation of γ corresponds to a separate problem. A family of threshold operators called the generalized thresholding operator [6–8], with soft-thresholding (ST) and hard-thresholding (HT) as its two extremes, will be obtained by solving all the optimization problems using the cyclic-coordinate descent method [23].

Also, the regularization parameter λ can vary, which generates different families of threshold operators. Each threshold operator corresponds to a solution of an optimization model with specific λ and γ values. This means a consensus matrix η will be obtained based on the family of threshold operators obtained by varying λ and γ. This matrix captures the information across all the different regularization models in the committee. Subsequently, we use a consensus criterion to select the optimal model parameters.

4. THE PROPOSED METHOD

In this section, we discuss the formulation of the minimax concave plus (MC+) penalty [9] function used in CRISP. We use this penalty and propose a novel consensus regularized selection based prediction method which generates a committee of regularized models and selects the best model among them. The selection among these different models is done using a consensus-based decision rule which differentiates our method compared to other ensemble techniques in machine learning.

Majority voting is a binary decision rule and it selects the candidate which obtains the highest number of votes. In other words, majority voting takes all the different choices into consideration by counting the occurrence when making decisions. However, in our method, we conduct an explicit search for the optimal model parameters (λ^*, γ^*) among all the entries in the consensus matrix η which effectively captures the information across all the different models.

4.1 Consensus Regularized Selection based Prediction Method

In this framework, we use the MC+ penalty which is a fast, continuous, nearly unbiased and accurate method for penalized variable selection in linear regression. The motivation for using this penalty arises from (i) it is an unbiased feature selection property which is one of the key disdvantages associated with the lasso, and (ii) it can be computed efficiently which makes it easier to employ it within ensemble-based models.

$$\begin{aligned} P(\beta;\lambda;\gamma) &= \lambda \int_0^{|\beta|} (1 - \frac{x}{\gamma\lambda})_+ dx \qquad (5) \\ &= \lambda(|\beta| - \frac{\beta^2}{2\lambda\gamma})\mathbf{I}(|\beta| < \lambda\gamma) + \frac{\lambda^2\gamma}{2}\mathbf{I}(|\beta| \geq \lambda\gamma) \end{aligned}$$

The MC+ penalty is defined in Eq. (5). For each value of $\lambda > 0$, there will be a continuum of penalties and threshold operators when γ varies from ∞ to 1. $(\cdot)_+$ represents the positive component. The threshold operators for the MC+ penalty will form a continuum between the soft- and hard-thresholding functions, which generates a natural and smooth transition across the set of solutions. In addition, we can also vary the value of λ, which will determine a specific model along with the non-convexity parameter γ. Thus, using the MC+ penalty we will develop a committee of prediction models to be used in CRISP.

By using the MC+ penalty, we can consider different combinations of the regularization parameter (λ) and the non-convexity parameter (γ), which will be helpful to avoid obtaining sub-optimal solutions. In other words, the MC+ penalty ensures a family of models for a fixed λ by interpolating between the ℓ_0 norm and ℓ_1 norm, which provides more candidates for the approximation of the ℓ_0 norm. In addition, it also generates a series of thresholding operators with the soft-thresholding operator and hard-thresholding operator as its two extremes. Thus, we can conclude that the MC+ penalty has the necessary and meaningful properties for capturing sparsity more efficiently.

$$Q^{(1)}(\beta) = \frac{1}{2}(\beta - \tilde{\beta})^2 + \lambda \int_0^{|\beta|} (1 - \frac{x}{\gamma\lambda})_+ dx \qquad (6)$$

Non-convex penalties such as the MC+ penalty can also perform better feature selection. When we use the MC+ penalty in the objective function in Eq. (6), the univariate penalized least squares objective function will be strictly convex, which ensures the descent property with coordinate descent method and the solution converges to a stationary point [9, 24]. The objective function used within the MC+ penalty is separable, which enables us to optimize the univariate case which is one-dimensional with the form using the standard coordinate-decent approach. If $\beta > 0$, the derivative of $Q^{(1)}(\beta)$ with respect to the β can be calculated as $\frac{dQ^{(1)}(\beta)}{d\beta} = \beta - \tilde{\beta} + \lambda(1 - \frac{\beta}{\gamma\lambda})_+$.

$$\begin{aligned} S_\gamma(\tilde{\beta}, \lambda) &\rightarrow S(\tilde{\beta}, \lambda) \qquad (7) \\ &= \arg\min_\beta \{\frac{1}{2}(\beta - \tilde{\beta})^2 + \lambda|\beta|\} \\ &= sgn(\tilde{\beta})(|\tilde{\beta}| - \lambda)_+ \end{aligned}$$

For a fixed λ, as γ varies, this generates a family of threshold operators $S_\gamma(\cdot, \lambda) : \mathbb{R} \rightarrow \mathbb{R}$, with the soft and hard thresholding operators as its two extremes. The soft-thresholding operator when $\gamma \rightarrow \infty$ is given as in Eq. (7).

$$\begin{aligned} S_\gamma(\tilde{\beta}, \lambda) &\rightarrow H(\tilde{\beta}, \lambda) \qquad (8) \\ &= \arg\min_\beta \{\frac{1}{2}(\beta - \tilde{\beta})^2 + \lambda\mathbf{I}(|\beta| > 0)\} \\ &= \tilde{\beta}\mathbf{I}(|\tilde{\beta}| > \lambda) \end{aligned}$$

The hard-thresholding operator when $\gamma \rightarrow 1+$ is given in Eq. (8). Since soft and hard thresholding functions are often used in the optimization problems with ℓ_1 and ℓ_0 norms, respectively, we assume $\gamma_{\ell_1} = \infty$ and $\gamma_{\ell_0} = 1+$ for the ℓ_1 and ℓ_0 norms.

$$\begin{aligned} S_\gamma(\tilde{\beta}, \lambda) &= \arg\min_\beta Q^{(1)}(\beta) \qquad (9) \\ &= \begin{cases} 0 & |\tilde{\beta}| \leq \lambda \\ sgn(\tilde{\beta})(\frac{|\tilde{\beta}| - \lambda}{1 - \frac{1}{\gamma}}) & \lambda < |\tilde{\beta}| \leq \lambda\gamma \\ \tilde{\beta} & |\tilde{\beta}| > \lambda\gamma \end{cases} \end{aligned}$$

Each coefficient in our optimization problem can be estimated by the generalized thresholding operator as given in Eq. (9) for the univariate problem. In each iteration, all of the m coefficients are repeatedly updated until convergence. In this case, all the solutions when varying λ and γ will form a two-dimensional solution surface whose coordinates can be represented as the matrix η. The goal of our work is to find optimal parameters (λ^*, γ^*) corresponding to the best solution. In order to find the best solution, our method will evaluate each solution. We now present the consensus criterion used in CRISP.

Consensus criterion: Squared error (se) of an estimator measures the square of the errors (or deviations) and assesses the quality of an estimator. It is used for assessing the performance of an estimator or a predictor. Generally, for the i^{th} training instance (x_i, y_i) and a linear fit $\hat{f}_{\lambda_\ell,\gamma_k}(x_i) = x_i\hat{\beta}_{\lambda_\ell,\gamma_k}$, when the values of λ_ℓ and γ_k are fixed, the squared error of the predictor will be of the form given below.

$$\eta_{\lambda_\ell,\gamma_k} = se(\hat{f}_{\lambda_\ell,\gamma_k}) = \sum_{i=1}^{n} (\hat{f}_{\lambda_\ell,\gamma_k}(x_i) - y_i)^2 \qquad (10)$$

Using this formulation of the consensus criterion, we evaluate the performance of each model for different pairs of (λ_ℓ, γ_k). We now present the CRISP algorithm which generates a family of solutions $\hat{\beta}_{\lambda_l,\gamma_k}$ based on Eq. (4) and selects the best one using this criterion based on the squared error of deviances. We assume that the matrix X is standardized with each column having zero mean and unit ℓ_2 norm. When γ=∞, the exact solution path for $Q(\beta)$ using coordinate-descent method will be used as a warm start for the minimization of $Q(\beta)$ with a non-convex penalty function.

The value of γ is decreasing until we have the solution path across a grid of values for γ [24]. The details of our approach are given in Algorithm 1. The univariate sub-problem in Eq. (6) will be optimized using coordinate descent method [23] which is a widely used non-derivative optimization algorithm. In each iteration of the coordinate descent method for the objective function $\text{argmin}_{\beta} Q(\beta_1, \beta_2, \ldots, \beta_m)$, it performs search along one coordinate direction at the current point and cyclically iterates through the other directions. In other words, in each iteration, the algorithm solves the optimization problem as shown in Eq.(11) for each variable $\beta_i (i = 1, 2, ..., m)$ of the problem.

$$\beta_i^{k+1} = \arg\min_{u \in \mathbb{R}} Q(\beta_1^{k+1}, ..., \beta_{i-1}^{k+1}, u, \beta_{i+1}^{k}, ..., \beta_m^{k}) \qquad (11)$$

That is, in each iteration of the optimization problem, each variable $\beta_i (i = 1, 2, \ldots, m)$ will be updated until convergence. Coordinate descent method minimizes a multivariable objective function by solving a series of univariate optimization problems in a loop.

4.2 Optimization

In this section, we discuss the optimization involved in the CRISP algorithm and also provide a detailed algorithmic description. We begin by providing the proof of convergence. The convergence of CRISP algorithm cannot directly follow the convergence property of coordinate-descent for functions with the form of the sum of a smooth loss function and a separable non-smooth convex penalty function due to its non-convex formulation [24]. This makes it important to discuss the convergence properties of our algorithm. The coordinate descent method used within CRISP updates the variables using Eq. (11). We will now show that the CRISP algorithm always converges to a global minimum of the objective function under certain assumptions which will be discussed below.

Consider the criterion in Eq. (4), where the data (X, Y) lies on a compact set and no column of the features in X is a multiple of the unit vector. Also, suppose that the penalty function $P(\beta; \lambda; \gamma)$ is symmetric around 0, which means that it satisfies $P(\beta; \lambda; \gamma) = P(-\beta; \lambda; \gamma)$; the first derivative of $P(\beta)$ with respect to β, $P^{'}(|\beta|)$, is non-negative, uniformly bounded and the second derivative $P^{''}(|\beta|)$ satisfies $inf_{\beta} P^{''}(|\beta|) > -1$; the sequence generated $\{\beta^k\}_k$ is bounded; for all the subsequences $\{\beta^{n_k}\}_k$ of $\{\beta^k\}_k$, the successive differences, i.e., $(\beta^{n_k} - \beta^{n_k - 1})$ converges to 0.

Theorem 1 *The univariate problem in Eq. (6) is strictly convex and the sequence of coordinate-updates $\{\beta^k\}_k$ converge to a minimum solution of Eq. (4).*

Proof. It should be noted that the MC+ penalty used in our work can meet all the required properties mentioned above. In addition, the assumption on data (X, Y) is used to ensure that the variables can be standardized and the non-degeneracy assumption on X means that all the columns are identically non-zero.

For a fixed i and $(\beta_1, \cdots, \beta_{i-1}, u, \beta_{i+1}, \cdots, \beta_m)$, we denote $Q(u)$ as

$$\begin{aligned} Q(u) &= Q^{i}_{(\beta_1, \cdots, \beta_{i-1}, u, \beta_{i+1}, \cdots, \beta_m)} \\ &= l(\beta_1, \cdots, \beta_{i-1}, u, \beta_{i+1}, \cdots, \beta_m) + P(|u|) \end{aligned} \qquad (12)$$

where $l(\cdot)$ is the loss function. Then, based on the Taylor's series expansions on f and penalty function $P(|u|)$, the sub-gradient at u will be

$$\begin{aligned} \partial Q(u) &= Q(u + \delta) - Q(u) \qquad (13) \\ &= \nabla_i l(\beta_1, \cdots, \beta_{i-1}, u, \beta_{i+1}, \cdots, \beta_m) + P^{'}(|u|) sgn(u) \\ &= l(\beta_1, \cdots, \beta_{i-1}, u + \delta, \beta_{i+1}, \cdots, \beta_m) \\ &- l(\beta_1, \cdots, \beta_{i-1}, u, \beta_{i+1}, \cdots, \beta_m) \\ &+ P(|u + \delta|) - P(|u|) \\ &= \nabla_i l(\beta_1, \cdots, \beta_{i-1}, u, \beta_{i+1}, \cdots, \beta_m)\delta + \frac{1}{2}\delta^2 \nabla_i^2 l \\ &+ P^{'}(|u|)(|u + \delta| - (|u|)) + \frac{1}{2} P^{''}(|u^*|)(|u + \delta| - |u|)^2 \end{aligned}$$

where $\delta \in \mathbb{R}$ and $\nabla_i^2 l = 1$ since it is the second derivative of the function f with respect to the i^{th} coordinate. $|u^*|$ is some number between $|u + \delta|$ and $|u|$. Assume that u_0 is the optimal value for $F(u)$, based on Eq. (13), we can have

$$\begin{aligned} & Q(u_0 + \delta) - Q(u_0) \qquad (14) \\ & \geq \frac{1}{2}\delta^2 \nabla_i^2 l + \frac{1}{2} P^{''}(|u^*|)(|u_0 + \delta| - |u_0|)^2 \\ & \geq \begin{cases} \frac{1}{2}\delta^2 \nabla_i^2 l + \frac{1}{2} P^{''}(|u^*|)\delta^2 & if\ P^{''}(|u^*|) \leq 0 \\ \frac{1}{2}\delta^2 \nabla_i^2 l + 0 & if\ P^{''}(|u^*|) \geq 0 \end{cases} \\ & \geq \frac{1}{2}\delta^2 (\nabla_i^2 l + min\{P^{''}(|u^*|), 0\}) \end{aligned}$$

Since for the MC+ penalty, $inf_{\beta} P^{''}(|\beta|) = -\frac{1}{\gamma}$ with $\gamma > 1$, $\nabla_i^2 l + inf_x P^{''}(|x|) > 0$. Then there exists a positive value $\theta = \frac{1}{2}\delta^2 (\nabla_i^2 l + min\{inf_x P^{''}(|x|), 0\})$ such that

$$Q(u_0 + \delta) - Q(u_0) \geq \theta \delta^2 \qquad (15)$$

Based on the analysis above, the boundedness of the sequence β^t for $t > 1$ will be

$$\begin{aligned} Q(\beta_i^{t-1}) - Q(\beta_{i+1}^{t-1}) &\geq \theta(\beta_{i+1}^{t-1} - \beta_{i+1}^{t})^2 \\ &= \theta \parallel \beta_i^{t-1} - \beta_{i+1}^{t-1} \parallel_2^2 \end{aligned} \qquad (16)$$

where $\beta_i^{t-1} = (\beta_1^t, \cdots, \beta_i^t, \beta_{i+1}^{t-1}, \cdots, \beta_m^{t-1})$. Using this boundedness for each coordinate, for every t, we will have

$$Q(\beta^{t+1}) - Q(\beta^t) \geq \theta \parallel (\beta^{t+1} - \beta^t) \parallel_2^2 \qquad (17)$$

From Eq. (17), we can see that the decreasing sequence $Q(\beta^t)$ converges. The sequence β^k cannot cycle without convergence and it must have a unique limit point. This completes the proof of convergence for β^k. □

We now provide a stepwise description of the CRISP algorithm. Algorithm 1 outlines the CRISP algorithm for selecting the best estimates among a family of solutions $\hat{\beta}_{\lambda_\ell, \gamma_k}$ to Eq. (4). A grid of increasing $\Lambda = \{\lambda_1, \lambda_2, \cdots, \lambda_L, \lambda_{L+1}\}$ and $\Gamma = \{\gamma_1, \gamma_2, \cdots, \gamma_K\}$ values are used for traversing different combinations of λ and γ, and generating different candidate models in the ensemble. Here, the additional λ_{L+1} values is used for the warm start of CRISP algorithm by Lasso.

In line 2, we initialize the estimator using the solution from Lasso for the minimization of $Q(\beta)$ at a smaller value of γ corresponding to a more non-convex penalty. In lines 4-8, each element of the coefficient vector is updated using the coordinate-wise update as shown in Eq. (9) until the solutions converge to the solution for Eq. (4) when $\lambda = \lambda_\ell$ and $\gamma = \gamma_k$. In line 10, we evaluate each model by obtaining

```
Algorithm 1: CRISP Algorithm
    Input: Predictor matrix ($X$); response variable ($Y$);
           regularization parameter sequence ($\Lambda$);
           non-convexity parameter sequence ($\Gamma$);
           length of $\Lambda$ ($L$); length of $\Gamma$ ($K$).
    Output: Optimal model parameters ($\lambda^*, \gamma^*$) and
            regression coefficient vector ($\beta^*$).
 1  for $\ell = L, \cdots, 2, 1$ do
 2      Use Lasso solution $\hat{\beta}_{\lambda_{\ell+1}, \gamma_K}$ as warm start;
 3      Initialize $\tilde{\beta} \leftarrow \hat{\beta}_{\lambda_{\ell+1}, \gamma_K}$;
 4      for $k = K, \cdots, 2, 1$ do
 5          repeat
 6              for $i = 1, 2, \cdots, p$ do
 7                  $\tilde{\beta}_i \leftarrow S_{\gamma_k}(\tilde{\beta}, \lambda_\ell)$ using Eq. (11);
 8              end
 9          until $\tilde{\beta}$ converges to $\hat{\beta}_{\lambda_\ell, \gamma_k}$;
10          Estimate $\eta_{\lambda_\ell, \gamma_k}$ using Eq. (10) for $\hat{\beta}_{\lambda_\ell, \gamma_k}$;
11      end
12  end
13  $(\lambda^*, \gamma^*) \leftarrow \arg\min_{\lambda_\ell, \gamma_k \in \mathbb{R}} Q(\tilde{\beta})$;
14  Select final model $\beta^*$ corresponding to $(\lambda^*, \gamma^*)$;
```

the value of the squared error (se) and populate a $L \times K$ consensus matrix η, in which $\eta_{\lambda_\ell, \gamma_k} = se(\hat{f}_{\lambda_\ell, \gamma_k})$ for $\ell = 1, 2, ..., L$ and $k = 1, 2, ..., K$. Here L and K represent the number of elements in Λ and Γ, respectively. In Line 13, according to the value of the se, the best model parameters which has the minimum se value among the LK entries in η will be selected as the final model parameters. Subsequently, the model β^* corresponding to these parameters (λ^*, γ^*) will be used for prediction.

4.3 Complexity Analysis

CRISP uses a cyclic coordinate descent based method to generate a committee of regularized models. The selection procedure using the squared error criterion for different (λ_ℓ, γ_k) values takes linear time in general, as we have to find the minimum entry among a set of LK entries in the consensus matrix η. Filling up each entry of the matrix η constitutes $O(m)$ time. When (λ^*, γ^*) are selected, these model parameters are used for the final prediction. Hence, the overall time complexity of the CRISP algorithm is $O(nm)$.

5. EXPERIMENTAL RESULTS

In this section, we conduct different experiments to evaluate the performance of the CRISP algorithm. We evaluate the goodness of prediction and scalability of CRISP by comparing it with various state-of-the-art algorithms.

5.1 Experimental Setup

We evaluate the performance of our CRISP algorithm using real-world EHRs and synthetic datasets which are summarized in Table 2.

Table 2: Description of the EHRs and synthetic datasets used in our experiments.

EHRs	# Features	#Instances
HF-cohort	77	8132
EHR-0	73	4416
EHR-1	72	3409
EHR-2	72	2748
EHR-3	72	2208
EHR-4	71	1800
Syn-1	1000	500
Syn-2	5000	500
Syn-3	10000	500

5.1.1 Electronic Health Records (EHRs)

The EHRs used in this paper were obtained from Henry Ford Health System in Detroit, Michigan in the United States for patients admitted with congestive heart failure (CHF) condition over a period of 10 years. In Figure 1, we show the class distribution for these EHRs. The Y-axis represents the % of readmissions (positive class) for 30-day and 365-day readmission. The X-axis represents the indices of the EHRs. These EHRs were procured over successive readmissions of patients. The suffix next to EHR represents the index of readmission, for example EHR-i corresponds to the data about all patients readmitted for the i^{th} time. It can be observed that the number of patients in each of the longitudinal EHRs decreases with successive readmissions. In addition to the readmission datasets, we also use a basic cohort dataset (HF-cohort) which represents an aggregated dataset summarizing the readmission information for all individuals over 10 years. The feature groups that were included for our evaluation include medications, procedures, labs, demographics and comorbidities. We used several data pre-processing methods such as normalization, imputation and feature integration to create the final EHRs [25].

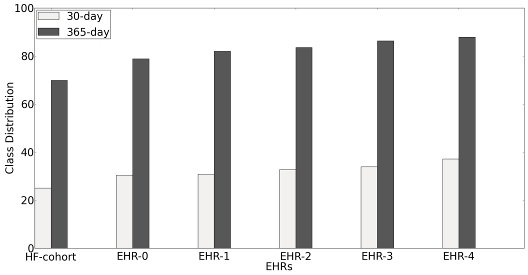


Figure 1: Class distribution for EHRs.

For our experiments, since we deal with the readmission risk prediction problem at two different thresholds, i.e., 30 days and 365 days, we determine the labels for each of these cases by calculating the difference between the readmission date and its preceding discharge date. In the case of 30-day readmission, if the difference is less than 30 days, we assign a label of 1 and if the difference is greater than 30 days, we assign 0. Following the same procedure for the 365-day readmission problem, we created two unique sets of binary prediction problems for each of the EHRs.

5.1.2 Synthetic Datasets

We now explain the generation of the synthetic datasets for evaluating the CRISP model. A simulation study will be examined in this section to evaluate the performance of our CRISP algorithm using synthetic datasets under various conditions. Based on a regression model $y=X\beta^*+\epsilon$, where $\beta^* \in \mathbb{R}^m$ and $\epsilon \sim N(0, \sigma^2 I)$, we consider three different scenarios and generate the synthetic datasets. These datasets are generated as per the guidelines given in [26] to encourage grouping and sparsity among the features. $X \sim N(0, C)$, where $C = [c_{ij}]$ is the covariance matrix, and the original feature coefficient values are given as follows.

1. In Syn-1, $n = 500$ and there are $m = 1000$ predictors. The parameters are generated as
$$\beta^* = [\underbrace{3, \cdots, 3}_{0.1m}, \underbrace{2, \cdots, 2}_{0.1m}, \underbrace{1.5, \cdots, 1.5}_{0.1m}, \underbrace{0, \cdots, 0}_{0.7m}]^T$$
and $\sigma = 3$, with covariance $c_{ij} = 0.7^{|i-j|}$.

2. In Syn-2, $n = 500$ and there are $m = 5000$ predictors. The parameters are generated as
$$\beta^* = [\underbrace{3, \cdots, 3}_{0.1m}, \underbrace{0, \cdots, 0}_{0.3m}, \underbrace{1.5, \cdots, 1.5}_{0.1m}, \underbrace{0, \cdots, 0}_{0.4m}, \underbrace{2, \cdots, 2}_{0.1m}]^T$$

3. In Syn-3, $n = 500$ and there are $m = 10000$ predictors. The parameters are generated as
$$\beta^* = [0.85, 0.85, \cdots, 0.85]^T$$

5.2 Implementation Details

In this section, we explain our experimental setup used for evaluating the CRISP algorithm. The CRISP algorithm was implemented using the R programming language. All the machine learning models used for comparison in our work were also implemented in R. Elastic net was implemented using the *glmnet* [23] R package for both the linear and logistic loss functions. Sparse Group Lasso (SGL) was implemented using the corresponding R package available in [27]. We implemented the hiernet-weak and hiernet-strong algorithms using the R package *hierNet* [19]. GAM and GA2M were implemented using the codes provided by the authors [16, 17]. While implementing the GA2M model, we only consider the top 50 interactions with lowest contribution to the overall error rate were considered. L_1 and L_2- SVR correspond to the L_2-regularized Support Vector Regression with the L_1 and L_2 loss functions, respectively. These were implemented using the LibLinear [28] R package.

We used the *SPAMS* [29] package to implement the L_0 and L_∞ models which are used to compare mean squared error (MSE) and coefficient of determination (R^2) values for all the three synthetic datasets. The performance results of all the models reported here are obtained using five-fold cross-validation. The model parameters (λ, γ) are tuned over the validation data to reduce overfitting, and the evaluation results are based on the test data. The R package *pROC* is used to calculate AUC values for all the models, and to calculate the MSE we used the *Metrics* [30] package. We now describe the procedure we used to select λ and γ values which generate different candidate models in our CRISP algorithm. In our experiments, while doing the parameter tuning, we generated a sequence of values for the regularization parameter λ and the non-convexity parameter γ and selected the model corresponding to the optimal values (λ^*, γ^*) which were then used for prediction on the test data.

5.3 Goodness of Prediction

We compare the performance of CRISP with various competing models for the 30-day readmission problem on all the longitudinal EHRs. Table 3 summarizes the performance comparison results using the AUC metric. The AUC values for CRISP algorithm in Table 3 are obtained from the optimal model parameters selected after applying the consensus criterion. For all of the datasets described in Table 2, the obtained AUC values evidently demonstrate that the proposed method CRISP provides significantly better results compared to the other methods. We also provide the p-values for CRISP to confirm the statistical significance of our results here. The p-value is calculated by comparing the performance of CRISP with respect to the second best performing model for each dataset. It should be noted that a result with a p-value of less than 0.05 is considered to be *statistically significant* and is interpreted as being small enough to justify the superiority of our approach over the methods used for comparison.

In Table 4, the mean squared error (MSE) along with the standard deviation for the 30-day readmission problem on all the datasets are provided. We observe that CRISP outperforms other competing methods. In Table 5 and Table 6, we also show the MSE along with standard deviations and the R^2 values for the three synthetic datasets using different regression models. It can be observed that CRISP performs better compared to the other regression models. This better performance is attributed to the fact that, in addition to using a sparse and efficient non-convex regularizer within CRISP, the algorithm generates several candidate models and then selects the best model using training data for prediction which gives a final model with good predictive ability. In Figure 2, we show the AUC values of CRISP model compared to other regression models using bar plots for the 365-day readmission problem. One can observe that CRISP gives better performance compared to other regression models on all the EHRs. The reason behind the better performance of CRISP lies in the fact that it builds an ensemble of non-convex regularized linear regression models and obtains a consensus among them to select the best set of model parameters.

5.4 Scalability Experiments

In this section, we perform experiments to evaluate the scalability of the MC+ penalty which is used within CRISP along with other well-known non-convex penalties mentioned in Table 7.

Smoothly Clipped Absolute Deviation (SCAD) [22] corresponds to a quadratic spline function with knots at λ and $\gamma\lambda$. This penalty function leaves large values of β_i not excessively penalized and makes the solution continuous. Log-Sum Penalty (LSP), due to its formulation, has the potential to guarantee more sparsity than the ℓ_1 norm. In Capped-ℓ_1 penalty, it treats all the β_i greater than γ equally, which makes it more robust to outliers than the ℓ_1 norm.

We use the Matlab package called Generalized Iterative Shrinkage and Thresholding (GIST) [31] to fit the above mentioned non-convex regularized linear regression models. These experiments were performed on a workstation with a quadcore CPU at 3.4GHz and 12 GB main memory.

Two high-dimensional synthetic datasets, Syn-2 and Syn-3, described in Section 5.1.2 were used in this experiment. Figure 3 measures the computational time for the MC+

Table 3: Performance comparison of CRISP with different models using AUC ± std for 30-day readmission problem in longitudinal EHRs.

Model	HF-cohort	EHR-0	EHR-1	EHR-2	EHR-3	EHR-4
Logit	0.5700±0.012	0.6060±0.013	0.5270±0.027	0.5490±0.013	0.6000±0.024	0.5960±0.035
GAM	0.6274±0.016	0.5944±0.015	0.5778±0.010	0.5990±0.040	0.6027±0.022	0.5728±0.019
GA2M	0.6192±0.013	0.5719±0.012	0.5546±0.032	0.5743±0.017	0.5894±0.015	0.5514±0.018
hiernet-weak	0.5980±0.011	0.5735±0.022	0.5657±0.010	0.5718±0.013	0.6163±0.038	0.5549±0.021
hiernet-strong	0.5887±0.010	0.5706±0.021	0.5628±0.026	0.5690±0.030	0.6055±0.041	0.5590±0.035
EN-linear	0.6181±0.009	0.6129±0.014	0.6185±0.026	0.6103±0.021	0.6351±0.025	0.6201±0.018
EN-logit	0.6184±0.021	0.6138±0.029	0.6192±0.018	0.6109±0.010	0.6350±0.050	0.6199±0.031
SGL	0.6233±0.010	0.6117±0.028	0.6095±0.016	0.5991±0.030	0.6222±0.050	0.5980±0.011
L_1-SVR	0.5171±0.016	0.5157±0.008	0.5070±0.018	0.5189±0.014	0.5919±0.013	0.5822±0.057
L_2-SVR	0.6269±0.017	0.6075±0.016	0.5892±0.013	0.6041±0.031	0.6258±0.033	0.5939±0.014
CRISP	**0.6504±0.008**	**0.6224±0.017**	**0.6194±0.025**	**0.6366±0.019**	**0.6433±0.033**	**0.6428±0.043**
(p-value)	(0.0013)	(7.85e-08)	(0.0003)	(5.725e-07)	(0.0031)	(0.0012)

Table 4: Performance comparison of CRISP with machine learning models using MSE ± std for the 30-day readmission problem in longitudinal EHRs.

Model	HF-cohort	EHR-0	EHR-1	EHR-2	EHR-3	EHR-4
Logit	0.2103±0.008	0.2056±0.004	0.2333±0.006	0.2254±0.010	0.2194±0.008	0.2283±0.011
GAM	0.1811±0.005	0.2122±0.010	0.2197±0.004	0.2246±0.009	0.2308±0.011	0.2488±0.014
GA2M	0.2238±0.010	0.2736±0.023	0.3154±0.048	0.3177±0.028	0.3089±0.038	0.3302±0.019
hiernet-weak	0.1914±0.008	0.2232±0.010	0.2226±0.002	0.2293±0.004	0.2309±0.010	0.2551±0.002
hiernet-strong	0.1933±0.007	0.2256±0.007	0.2297±0.005	0.2335±0.009	0.2250±0.008	0.2559±0.019
EN-linear	0.1832±0.003	0.2059±0.001	0.2075±0.002	0.2152±0.004	0.2145±0.004	0.2263±0.001
EN-logit	0.1833±0.006	0.2061±0.008	0.2077±0.009	0.2153±0.002	0.2146±0.004	0.2265±0.004
SGL	0.1816±0.003	0.2050±0.004	0.2065±0.008	0.2149±0.007	0.2151±0.009	0.2272±0.009
L_1-SVR	0.7166±0.022	0.9585±0.037	0.9756±0.027	1.0635±0.041	1.0861±0.053	1.0814±0.107
L_2-SVR	0.2402±0.007	0.2985±0.016	0.3104±0.026	0.3333±0.028	0.3441±0.049	0.3980±0.016
CRISP	**0.1775±0.002**	**0.2030±0.003**	**0.2050±0.003**	**0.2110±0.003**	**0.2083±0.004**	**0.2202±0.003**

Table 5: Performance comparison of CRISP with machine learning models using MSE ± std on synthetic datasets.

Model	Syn-1	Syn-2	Syn-3
L_0	0.3677±0.030	0.9598±0.156	1.0391±0.087
L_∞	0.2439±0.032	0.8806±0.140	1.0214±0.070
EN-linear	0.1892±0.032	0.7832±0.087	1.0020±0.138
SGL	0.1744±0.030	0.8392±0.097	**0.9028±0.059**
CRISP	**0.0861±0.012**	**0.7698±0.179**	1.0015±0.188

Table 6: Performance comparison of CRISP with machine learning models using R^2 metric on synthetic datasets.

Model	Syn-1	Syn-2	Syn-3
L_0	0.6269	0.1539	0.1602
L_∞	0.6197	**0.2510**	0.1269
EN-linear	0.8093	0.2064	0.1181
SGL	0.5046	0.1682	0.1038
CRISP	**0.9124**	0.2215	**0.2057**

Table 7: Non-Convex penalties used in our evaluation.

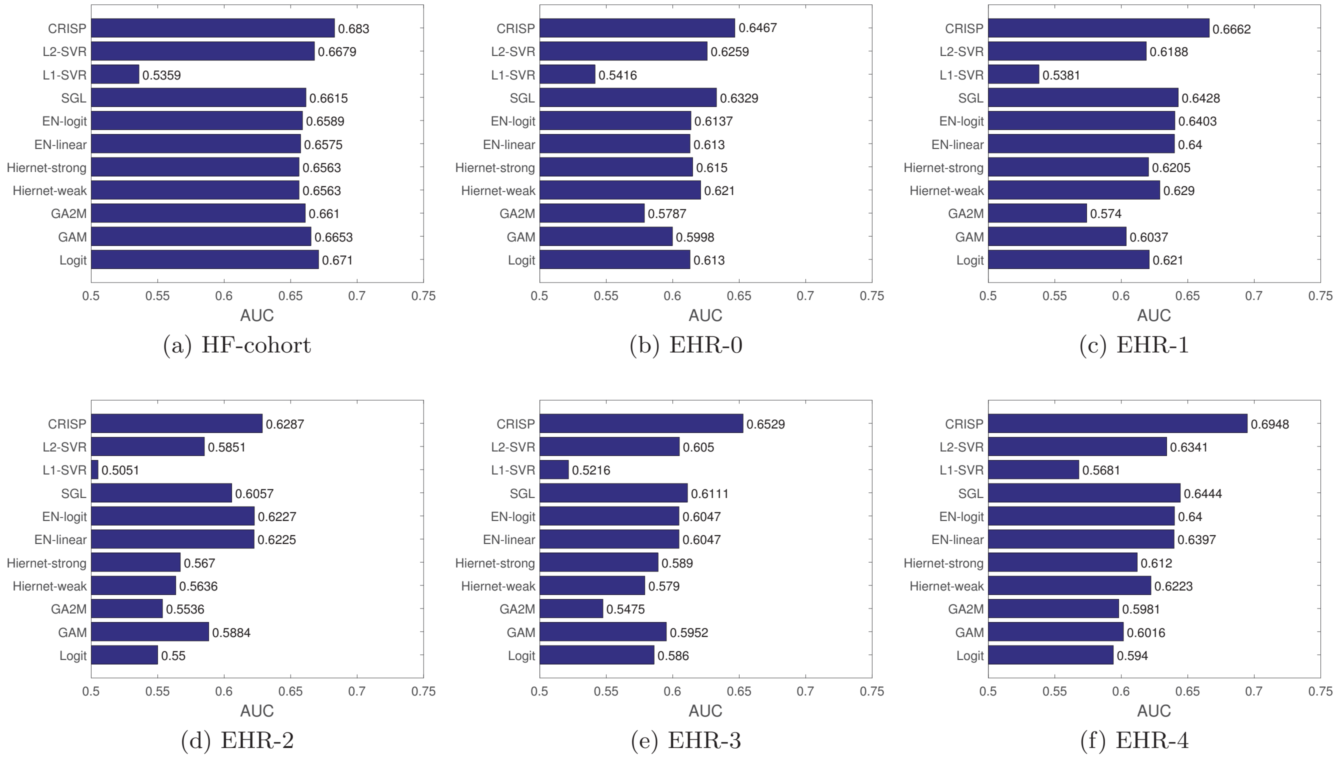
Name	$P(\beta_i)$
SCAD	$\lambda \int_0^{\lvert\beta_i\rvert} min(1, \frac{[\gamma\lambda - x]_+}{(\gamma-1)\lambda})dx \ (\gamma > 2)$
LSP	$\lambda log(1 + \lvert\beta_i\rvert/\gamma) \ (\gamma > 0)$
Capped-ℓ_1	$\lambda \min(\lvert\beta_i\rvert, \gamma) \ (\gamma > 0)$

penalty compared to three competing non-convex regularizers. In this plot, the Y-axis represents the time taken in seconds which was averaged over five runs. The X-axis represents the dimensionality of the features.

The scalability plot in Figure 3(a) for the Syn-2 dataset indicates that the MC+ penalty based model runs faster compared to the other three models. LSP penalty-based model takes highest time and the other two penalties, namely, SCAD and Capped ℓ_1 norm based models, are also slower than the MC+ penalty. Figure 3(b) shows the scalability plot for Syn-3 dataset, and it can be observed that even in this case the MC+ penalty runs faster compared to the other models. This shows that our CRISP method which uses the MC+ penalty can perform efficiently on high-dimensional datasets.

6. CONCLUSION AND FUTURE WORK

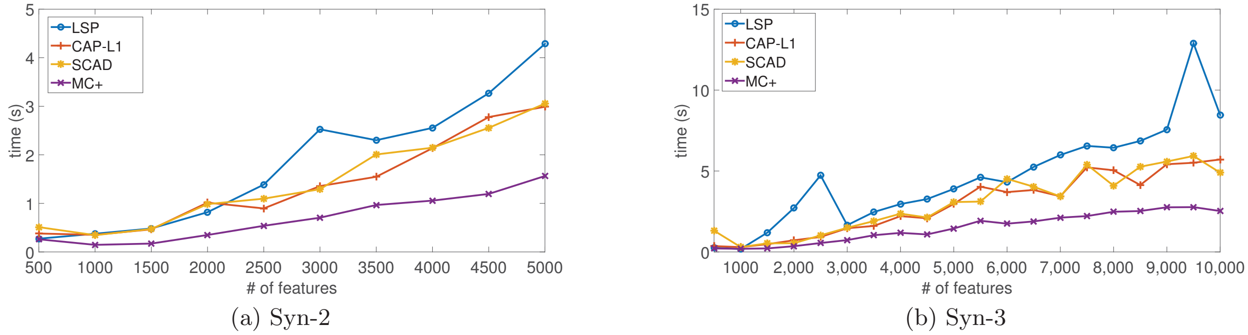
In this paper, we presented a method called CRISP for solving the consensus regularization problem for regression which has not been studied in the literature previously, due to the inherent difficulty associated with integrating and computing multiple regularizers efficiently within a unified framework. This method generates a committee of non-convex regularized linear regression models using the minimax concave plus (MC+) penalty, and it applies a consensus criterion to select the best model for prediction. This method is efficient because the problem of learning multiple candidate models within the committee is solved using a gen-



(a) HF-cohort (b) EHR-0 (c) EHR-1

(d) EHR-2 (e) EHR-3 (f) EHR-4

Figure 2: Performance comparison of CRISP with several state-of-the-art methods for the 365-day readmission problem in longitudinal EHR datasets.



(a) Syn-2 (b) Syn-3

Figure 3: Comparison of time taken in seconds for three different non-convex regularizers compared to MC+ with increasing dimensionality of the features.

eralized thresholding operator employed within a fast cyclic coordinate descent framework. Our method is also simple to interpret as it only selects the optimal model from all the candidate models for the final prediction. We evaluated this model using longitudinal EHRs collected at a large hospital and high-dimensional synthetic datasets using diverse metrics such as AUC, MSE and R^2. We also conducted experiments to assess the scalability of CRISP. Our results indicate that CRISP obtains higher AUC values compared to various other additive, interactions and sparse regression models. This work can be extended for building an active learning-based regression model [32] by querying the label for an instance after obtaining a consensus on including it in the training data using the multiple candidate models generated by CRISP.

7. ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation grants IIS-1231742, IIS-1527827 and IIS-1646881.

8. REFERENCES

[1] L. I. Kuncheva. A theoretical study on six classifier fusion strategies. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(2):281–286, 2002.

[2] D. Ruta and B. Gabrys. An overview of classifier fusion methods. *Computing and Information systems*, 7(1):1–10, 2000.

[3] S. Tulyakov, S. Jaeger, V. Govindaraju and D. Doermann. Review of classifier combination methods. *Machine Learning in Document Analysis and Recognition*, 33(1):361–386, 2008.

[4] N. Parikh, and S. P. Boyd. Proximal Algorithms. *Foundations and Trends in optimization*, 1(3):127–239, 2014.

[5] L. Condat. A generic proximal algorithm for convex optimization and its application to total variation minimization. *IEEE Signal Processing Letters*, 21(8):985–989, 2014.

[6] D. L. Donoho, and J. M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3):425–455, 1994.

[7] D. L. Donoho, I. M. Johnstone, G. Kerkyacharian, and D. Picard. Wavelet shrinkage: asymptopia?. *Journal of the Royal Statistical Society. Series B (Methodological)*, 57(2):301–369, 1995.

[8] Y. She. Thresholding-based iterative selection procedures for model selection and shrinkage. *Electronic Journal of statistics*, 3:384–415, 2009.

[9] C. H. Zhang. Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, 38(2):894–942, 2010.

[10] G. Gasso, A. Rakotomamonjy, and S. Canu. Recovering sparse signals with a certain family of nonconvex penalties and DC programming. *IEEE Transactions on Signal Processing*, 57(12):4686–4698, 2009.

[11] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288, 1996.

[12] M. R. Osborne, B. Presnell, and B. A. Turlach. On the lasso and its dual. *Journal of Computational and Graphical statistics*, 9(2):319–337, 2000.

[13] J. Friedman, T. Hastie, H. Höfling, and R. Tibshirani. Pathwise coordinate optimization. *The Annals of Applied Statistics*, 1(2):302–332, 2007.

[14] E. Candes, M. Wakin, and S. Boyd. Enhancing sparsity by reweighted ℓ_1 minimization. *Journal of Fourier analysis and applications*, 14(5):877–905, 2008.

[15] T. Zhang. Analysis of multi-stage convex relaxation for sparse regularization. *Journal of Machine Learning Research*, 11:1081–1107, 2010.

[16] Y. Lou, R. Caruana, J. Gehrke, and G. Hooker. Accurate intelligible models with pairwise interactions. *In Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 623–631. ACM, 2013.

[17] Y. Lou, R. Caruana, and J. Gehrke. Intelligible models for classification and regression. *In Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 150–158. ACM, 2012.

[18] R. Caruana, Y. Lou, J. Gehrke, P. Koch, M. Sturm, and N. Elhadad. Intelligible models for healthCare: predicting pneumonia risk and hospital 30-day readmission. *In Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1721–1730. ACM, 2015.

[19] J. Bien, J. Taylor, and R. Tibshirani. A lasso for hierarchical interactions. *Annals of statistics*, 41(3):1111–1141, 2013.

[20] P. Zhao, G. Rocha, and B. Yu. The composite absolute penalties family for grouped and hierarchical variable selection. *Annals of statistics*, 37(6):3468–3497, 2009.

[21] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *The Annals of statistics*, 32(2):407–499, 2004.

[22] J. Fan and R. Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association*, 96(456):1348–1360, 2001.

[23] J. Friedman, T. Hastie, and R. Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, 33(1):1–22, 2010, http://www.jstatsoft.org/v33/i01/.

[24] R. Mazumder, J. Friedman, and T. Hastie. SparseNet: Coordinate descent with nonconvex penalties. *Journal of the American Statistical Association*, 106 (495):1125–1138, 2012.

[25] B. Vinzamuri, and C. K. Reddy. Cox regression with correlation based regularization for electronic health records. *In Proceedings of the 2013 IEEE 13th International Conference on Data Mining*, pages 757–766, 2013.

[26] H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005.

[27] J. Friendman, T. Hastie, and R. Tibshirani. A note on the group lasso and the sparse group lasso. *arXiv preprint arXiv:1001.0736*, 2010.

[28] R. Fan, K. Chang, C. Hsieh, R. Wang, and C. Lin. LIBLINEAR: A library for large linear classification. *Journal of machine learning research*, 9:1871–1874, 2008.

[29] J. Mairal, F. R. Bach, and J. Ponce. Sparse Modeling for Image and Vision Processing. *Foundations and Trends in Computer Graphics and Vision*, 8(2):85–283, 2014.

[30] H. Ben. Metrics: Evaluation metrics for machine learning. 2012, https://cran.r-project.org/web/packages/Metrics/.

[31] P. Gong, C. Zhang, Z. Lu, J. Huang, and J. Ye. A General Iterative Shrinkage and Thresholding Algorithm for Non-convex Regularized Optimization Problems. *Proceedings of International Conference on Machine Learning (ICML)*, pages 37–45, 2013.

[32] B. Vinzamuri, Y. Li, and C. K. Reddy. Active learning based survival regression for censored data. *Proceedings of the ACM International Conference on Information and Knowledge Management (CIKM)*, pages 241–250, 2014.