Annealed Sparsity via Adaptive and Dynamic Shrinking

Kai Zhang[§], Shandian Zhe[†], Chaoran Cheng[♯], Zhi Wei[♯], Zhengzhang Chen[§]
Haifeng Chen[§], Guofei Jiang[§], Yuan Qi[†], Jieping Ye[‡]
[§]NEC Laboratories America, Princeton NJ
[†]Dept. Computer Science, Purdue University
[♯]Dept. Computer Science, New Jersey Institute of Technology
[‡]Dept. Computational Medicine & Bioinformatics, University of Michigan, Ann Arbor
[§]{kzhang,zchen,haifeng,gfj}@nec-labs.com, [†]{szhe,alanqi}@purdue.edu
[♯]{cc424,zhi.wei@njit.edu}, [‡]jpye@umich.edu

ABSTRACT

Sparse learning has received tremendous amount of interest in high-dimensional data analysis due to its model interpretability and the low-computational cost. Among the various techniques, adaptive ℓ_1-regularization is an effective framework to improve the convergence behaviour of the LASSO, by using varying strength of regularization across different features. In the meantime, the adaptive structure makes it very powerful in modelling grouped sparsity patterns as well, being particularly useful in high-dimensional multi-task problems. However, choosing an appropriate, global regularization weight is still an open problem. In this paper, inspired by the annealing technique in material science, we propose to achieve "annealed sparsity" by designing a dynamic shrinking scheme that simultaneously optimizes the regularization weights and model coefficients in sparse (multi-task) learning. The dynamic structures of our algorithm are twofold. Feature-wise ("spatially"), the regularization weights are updated interactively with model coefficients, allowing us to improve the global regularization structure. Iteration-wise ("temporally"), such interaction is coupled with gradually boosted ℓ_1-regularization by adjusting an equality norm-constraint, achieving an "annealing" effect to further improve model selection. This renders interesting shrinking behaviour in the whole solution path. Our method competes favorably with state-of-the-art methods in sparse (multi-task) learning. We also apply it in expression quantitative trait loci analysis (eQTL), which gives useful biological insights in human cancer (melanoma) study.

Keywords

Sparse regression, adaptive LASSO, multi-task LASSO, regularization path, sparse multi-task learning

1. INTRODUCTION

With the rapid development of data acquisition technologies in various science and engineering domains such as imaging, physics, biology, and computer networks, we are having access to digital data of unprecedented amount and quality. In this modern paradigm, a significant challenge for data discovery is the huge number of features in representing objects. For example, a high-resolution image is composed of millions of pixels; the micro-array data in human genomic study typically includes tens of thousands of gene expressions; in movie recommendation systems the number of movies can be tens of millions. How to identify truly relevant features in the huge feature pools for accurate learning and prediction has become one of the key challenges in data mining.

Sparse regression has recently emerged as a powerful tool for high-dimensional data analysis, especially in removing irrelevant variables and identifying a parsimonious subset of covariates for predicting the target [29, 38, 39]. Given a response vector $\mathbf{y} = [y_1, y_2, ..., y_n]^\top$ and predictors $\mathbf{X} \in \mathbb{R}^{n\times D}$, where without loss of generality the data is centered, sparse regression and in particular the LASSO solves the following regularized linear regression problem:

$$\min_{\beta} \|\mathbf{X}\beta - \mathbf{y}\|_2^2 + \lambda|\beta|_1, \tag{1}$$

where $\beta \in \mathbb{R}^{D\times 1}$ is the regression coefficient vector. The ℓ_1-norm $|\beta|_1$ is used to enforce the sparsity of solution, making the model easy to interpret. In the meantime, recent advances in solving the non-smooth, convex LASSO problem has made it computationally extremely efficient [14, 12]. Therefore the LASSO and related methods have been applied with great success in a number of domains including bioinformatics [17, 24, 35, 36], imaging and computer vision [33, 21, 9], and signal processing [5, 6].

It is shown that LASSO can perform automatic variable selection because the ℓ_1-penalty is singular at the origin [11]. It was also shown that variable selection with the LASSO is consistent only under certain conditions [22, 37]. Namely, there are scenarios in which the LASSO selection is not consistent. To fix this problem, [38] proposes to use adaptive weights to regularize the model coefficients along different features, as

$$\min_{\beta} \|\mathbf{X}\beta - \mathbf{y}\|_2^2 + \lambda \cdot |\hat{\mathbf{w}} \odot \beta|_1, \tag{2}$$

where $\hat{\mathbf{w}} \in \mathbb{R}^{D\times 1}$ is the regularization weight, and $|\hat{\mathbf{w}}\odot\beta|_1 = \sum_i \hat{w}_i|\beta_i|$, namely each dimension of β is penalized differently instead of sharing a single regularization parameter λ as in LASSO (1). The regularization weights $\hat{\mathbf{w}}$ can be chosen as $\hat{\mathbf{w}} = |\beta_{ols}|^{-\gamma}$, where β_{ols} is the ordinal least-square solution,

KDD '16, August 13–17, 2016, San Francisco, CA, USA.
© 2016 ACM. ISBN 978-1-4503-4232-2/16/08…$15.00
DOI: http://dx.doi.org/10.1145/2939672.2939769

and γ is a positive number. Such choice renders the oracle property of the adaptive LASSO estimator in simultaneous variable selection and prediction.

Besides improving the asymptotic behaviour of sparse model estimation, the adaptive LASSO can also be quite useful in imposing structured sparsity patterns, in particular in high-dimensional multi-task learning by sharing the same adaptive weight across different tasks. Therefore it has gained a lot of research interest from various domains [10, 17, 19]. However, choosing an optimal regularization weight vector $\hat{\mathbf{w}}$ (2) can be much more challenging than selecting a single regularization parameter λ (1). The former has a significantly larger search space, and is the key to the superior performance of adaptive LASSO.

In this paper, we propose a novel approach to simultaneously compute the model coefficients and adaptive regularization weights in ℓ_1-regularized regression, in comparison to most existing methods that address them separately. The basic idea is to adopt an alternating optimization framework to establish the closed-form relations between model coefficients and regularization weights (under an equality norm-constraint of the latter). By doing this, the two sets of parameters can then be optimized iteratively, until an equilibrium state is obtained upon convergence.

The interactive updating scheme can acquire greater flexibility in tuning the sparse model. In the meantime, to further improve its convergence and reduce the sensitivity on initial conditions, we borrow the idea from material science and propose an "annealed" shrinking procedure. Specifically, throughout the interactive updates between model coefficients and regularization weights, we gradually strengthen the global magnitude of ℓ_1-penalization by adjusting the equality norm-constraint on the regularization weight vector. Then, by starting from a dense solution, the system will go through a series of micro-stages that continuously sparsify and ameliorate itself. In this "annealing" process, features are like particles: in the "high-temperature" beginning, all features have the freedom to compete with each other and position themselves in the model; however, when the system gradually cools down, fewer and fewer features could preserve their energy; finally, only those features that survive the dynamic competing process will be selected.

We find that such a dynamic shrinking scheme leads to an interesting mechanism of feature selection and competition, which favors the choice of truly relevant features. Through extensive experiments, we compare our approach with a number of state-of-the-art techniques in sparse learning, and obtain promising results. The contribution of the paper is summarized as follows:

1. We introduce the concept of "annealing" in sparse learning, and propose an annealed, dynamic shrinking framework to improve the model selection in ℓ_1-regression;

2. We extend our approach to solve high-dimensional multi-task learning problems to improve state-of-the-art;

3. We apply our approach in eQTL, which successfully identifies significant and relevant pathways to help understand the P53 regulation mechanism for melanoma.

The rest of the paper is structured as follows. Section 2 discusses the proposed method. Section 3 extends it to multi-task learning scenario. Section 4 describes related methods. Empirical results are presented in Section 5, and the last section concludes the paper.

2. METHOD

Consider the following linear regression problem with the design matrix $\mathbf{X} \in \mathbb{R}^{n\times D}$, where n is the sample size and D is the dimensionality, and the target response vector $\mathbf{y} \in \mathbb{R}^{n\times 1}$. We use an adaptive weight vector $\mathbf{w} = [w_1,\ w_2,\ ..., w_D]^\top$ to regularize over different covariates, as

$$\min_{\mathbf{w},\mathbf{B}} \quad \|\mathbf{X}\beta - \mathbf{y}\|_2^2 + |\mathbf{w}^{-\gamma} \odot \beta| \tag{3}$$

$$s.t.\ \sum_d w_d = \omega,\ \ w_d \geq 0. \tag{4}$$

Here, $\beta \in \mathbb{R}^{D\times 1}$ is the model, $\mathbf{w} \in \mathbb{R}^{D\times 1}$ is the regularization weight vector, and $|\mathbf{w}^{-\gamma} \odot \beta| = \sum_{d=1}^{D} w_d^{-\gamma} \cdot |\beta_d|$. Both parameters will be optimized in our learning procedures.

The equality norm-constraint $\sum_d w_d = \omega$ is quite useful in controlling the global strength of regularization (in an average sense). To see this, note that the regularization imposed on the dth feature is $w_d^{-\gamma}$ (3) . Therefore, if γ is positive: then the larger the ω, the smaller the average strength of the ℓ_1-penalty; on the other hand, if γ is negative: then the larger the ω, the larger the average strength of ℓ_1-penalty. Later we shall see that, it is exactly because of this equality norm-constraint (4) that we acquire the flexibility of "annealing" the whole system to improve the state of solution. The power parameter γ can be chosen either as a positive or negative real number, in comparison to the power parameter that can only be positive in the adaptive LASSO [39].

2.1 Interactive Update

We first consider ω as a pre-defined constant. Then the problem (1) can be solved by alternating optimization. Namely we first fix $\mathbf{w}$ and solve β, and then fix β and solve $\mathbf{w}$, and keep iterating until convergence. Here we use β_d to denote the dth entry of β.

Fix w and solve β. Then this becomes an adaptive LASSO problem,

$$\min_{\beta} \|\mathbf{X}\beta - \mathbf{y}\|_2^2 + |\mathbf{w}^{-\gamma} \odot \beta|, \tag{5}$$

which can be computationally converted to a standard LASSO problem [38].

Fix β **and solve w**. This then become the following constrained optimization problem

$$\min_{\mathbf{w}} \sum_d \beta_d \cdot w_d^{-\gamma}, \tag{6}$$

$$\theta_d = |\beta_d|.$$

Problem (6) has a closed form solution,

$$w_d = \left(\frac{\theta_d^{\frac{1}{1+\gamma}}}{\sum_{j=1}^{D} \theta_j^{\frac{1}{1+\gamma}}} \right) \omega. \tag{7}$$

The derivations are in the appendix.

Choice of the γ Parameter. Based on equation (7), we can examine the relation between the actual regularization imposed in (3), $w^{-\gamma}$, and the (absolute) value of the model coefficient, θ_d (4). We discuss the following scenarios:

1. $\gamma > 0$: if θ_d is larger (compared with $\theta_{d'}, d' \neq d$), then w_d (7) will also be larger due to the positive power term

$\frac{1}{1+\gamma}$, and as a result the regularization term $w_d^{-\gamma}$ in (3) will be smaller, leading to a weaker regularization on the d feature in the next iteration;

2. $\gamma < -1$: if θ_d is larger (compared with $\theta_{d'}, d' \neq d$), then w_d will be smaller due to the negative power $\frac{1}{1+\gamma}$; as a result $w_d^{-\gamma}$ will also be smaller since $-\gamma > 0$, leading to a weaker regularization in the next iteration;

3. $-1 < \gamma < 0$: if θ_d is larger (compared with $\theta_{d'}, d' \neq d$), then w_d will be larger due to the positive power $\frac{1}{1+\gamma}$; so $w_d^{-\gamma}$ will be larger since $-\gamma > 0$, leading to a stronger regularization in the next iteration.

As can be seen, in case $\gamma > 0$ or $\gamma < -1$, the regularization term $w_d^{-\gamma}$ and the model coefficient θ_d are inversely related to each other: larger θ_d will lead to smaller regularization coefficient $w_d^{-\gamma}$, and vice versa. In practice, we update w_d and θ_d iteratively. As a result, important features in the current iteration will tend to be penalized less in the next iteration; on the contrary, less important features will be confronted with strong penalty in future iterations. The system reaches a stationary point upon convergence.

In case $-1 < \gamma < 0$, however, $w_d^{-\gamma}$ and θ_d will be favorably associated with each other. In other words, relevant features in the current step will be strongly penalized in the next iteration. This obviously leads to unstable iterations and therefore we will exclude it from our parameter choice.

In the adaptive LASSO [38], the regularization weight is also inversely related to some pre-computed model coefficient. The difference of our method is that, first, our weights are carefully tuned based on previous model coefficients through norm-regularization (7); second, we keep alternating instead of using a single update; third, as will be discussed, we have the freedom of annealing the strength of global regularization via the equality norm-constraint (4).

2.2 Multi-Stage Shrinking

The interactive updates between models and the adaptive weights mimic a self adapting process that is expected to drive the whole system to a desired state. However, this optimization problem is non-convex, therefore in case of bad initialization, the iterations might quickly get stuck into a local optimum. In this case, dimensions with large model coefficients will keep being dominant and and dimensions with small coefficients may never have a chance to regain their magnitudes.

In order to prevent pre-mature convergence, we propose a multi-stage shrinking procedure. The basic idea is to introduce strong perturbations in the beginning, such that all features have the chance to be selected and compete with each other. Then we gradually "cool down" the system by using stronger and stronger ℓ_1-penalties. Namely fewer and fewer features can survive in the progressive shrinking. By doing this, the system will go a series of self-adjusting micro-stages sequentially before reaching the final solution.

Suppose we initialize $\mathbf{w}$ with $|\mathbf{w}| = \omega^\tau$, $\tau = 0$. Then we interactively update β (5) and $\mathbf{w}$ (6) under this equality norm constraint until convergence. When this stage ends, we will start next stage of iterations with an updated norm constraint $|\mathbf{w}| = \omega^\tau$, $\tau = 1$, which imposes a stronger ℓ_1-penalty. Then we iterate between β and $\mathbf{w}$ until the second stage ends. By repeating this, we keep strengthening the global ℓ_1-norm regularization stage by stage, achieving an "annealing" effect. Here each stage is indexed by τ and is composed of iterations under $|\mathbf{w}| = \omega^\tau$.

Depending on the choice of γ, in order to guarantee that the global regularization strength $\mathbf{w}^{-\gamma}$ will gradually increase, we need different strategies in tuning the ω parameter. (1) $\gamma > 0$: ω will start from a large value (corresponding to a weak regularization) and gradually decrease; (2) $\gamma < -1$: ω will start from a small value and gradually increase throughout the iterations.

2.3 Relation with Annealing

In material science, annealing is a powerful heat treatment technique [31] to improve physical and chemical properties of a metal. It heats the metal to a high temperature, which gives the energy for its atoms to break the bond and diffuse actively within crystal lattices; a suitable temperature is then maintained and gradually cooled down, allowing the material to progress towards equilibrium state. Annealing can reduce the Gibbs-Free-Energy of the metal.

The dynamic shrinking method in Algorithm 1 very much resembles (and is indeed inspired by) an annealing process. The strength of the ℓ_1-regularization can be deemed as controlling the temperature of the system: in the beginning stages when regularization is weak, all features have the freedom to compete and position themselves in the model, meaning that the solution is dense and the system has a high energy. When the regularization gradually enhances, the system begins to cool down, model coefficients start shrinking progressively, and the system energy decreases as well. The norm-constraint $|\mathbf{w}| = \omega$ can be deemed exactly as the as controlling the "temperature" of the system: a larger ω imposes a weak regularization, meaning high temperature and energy status; a smaller ω, on the contrary, enforces low temperature and energy status.

The initial temperature of annealing should be higher than metal recrystallization temperature. Similarly, we also start from a high temperature, i.e., a weak regularization such that initial model parameters are dense. This allows different features to fully compete with each other; if the solution is already sparse in the beginning, the iterations will quickly get trapped into a poor local optima. In our context, the densest initial solution is the ordinary least-square solution, namely a sparse regression with vanishing ℓ_1-penalties.

It is worthwhile to point out the difference between our method and simulated annealing [15]. Simulated annealing is a probabilistic searching technique that can be applied to any pre-defined objective function to find its global optimum [25]; in comparison, we target on more effective sparse regression and feature selection by reformulating the adaptive LASSO with a progressive, multi-stage shrinking mechanism, thus bearing an analogy to the "annealing" process.

3. MULTI-TASK REGRESSION

Suppose we have a number of k tasks, each task is composed of the design matrix $\mathbf{X}^k \in \mathbb{R}^{n^k \times D}$ and target $\mathbf{y}^k \in \mathbb{R}^{n^k \times 1}$; we use shared adaptive weight $\mathbf{w} = [w_1, \ w_2, \ ..., w_D]^\top$ to regularize over all the K tasks, as

$$\min_{\mathbf{w},\mathbf{B}} \quad \sum_{k=1}^{K} \left(\left\| \mathbf{X}^k \beta^k - \mathbf{y}^k \right\|_2^2 + |\mathbf{w}^{-\gamma} \odot \beta^k| \right) \tag{8}$$

$$s.t. \ \sum_d w_d = \omega, \ \ w_d \geq 0. \tag{9}$$

```
Algorithm 1: Adaptive LASSO + dynamic shrinking
Input: multi-task data $\mathbf{Z} = \{\mathbf{X}^k, \mathbf{y}^k\}_{k=1}^K$; initial norm
       constraint $\omega^0$; shrinking factor $\delta$; $\tau = 0$; initial
       regularization weight $\mathbf{w}_0^0 = [\frac{\omega^0}{D}, \frac{\omega^0}{D}, ... \frac{\omega^0}{D}]$;
Output: solution path for all the $k$ tasks
begin
    while model is unempty do
        $t = 0$;
        while Convergence do
            $\mathbf{B}_{t+1}^\tau$ =ModelUpdate($\mathbf{w}_t^\tau, \mathbf{Z}$);
            $\mathbf{w}_{t+1}^\tau$ =WeightUpdate($\mathbf{B}_{t+1}^\tau, \omega^\tau$);
            $t = t + 1$;
        $\omega^{\tau+1} = \omega^\tau \cdot \delta$;
        $\mathbf{w}_0^{\tau+1} = \mathbf{w}_t^\tau$;
        $\tau = \tau + 1$
```

Here, $\beta^k \in \mathbb{R}^{D\times 1}$ is the model coefficients for the kth task for $k = 1, 2, ..., K$, and $\mathbf{B} = [\beta^1,\ \beta^2,\ ..., \beta^k]$. Through similar derivations, we have the following procedures.

Fix w and solve β^k's. Then this becomes K independent adaptive LASSO problems, for $k = 1, 2, ..., K$

$$\min_{\beta^k} \left\|\mathbf{X}^k\beta^k - \mathbf{y}^k\right\|_2^2 + |\mathbf{w}^{-\gamma} \odot \beta^k| \tag{10}$$

which can be easily converted to a standard LASSO.

Fix β^k's and solve w. This then becomes the following constrained optimization problem

$$\min_{\mathbf{w}} \sum_d \theta_d \cdot w_d^{-\gamma} \tag{11}$$

$$\theta_d = \textstyle\sum_{k=1}^K |\beta_d^k|. \tag{12}$$

Problem (11) has a closed form solution,

$$w_d = \left(\frac{\theta_d^{\frac{1}{1+\gamma}}}{\sum_{j=1}^D \theta_j^{\frac{1}{1+\gamma}}}\right)\omega. \tag{13}$$

As can be seen, the proposed method can conveniently handle multi-task learning scenarios, thanks to the flexibility of using an adaptive regularization weight. In the following we introduce two routines to simplify our presentation of the algorithm.

- **B** =ModelUpdate($\mathbf{w}, \mathbf{Z}$). This denotes training an adaptive LASSO with weights **w** (10) for each of the k tasks in $\mathbf{Z} = \{\mathbf{X}_k, \mathbf{y}_k\}_{k=1}^K$, independently, and obtaining the model coefficients $\mathbf{B} = [\beta^1, \beta_2, ..., \beta^K]$;
- **w** =WeightUpdate($\mathbf{B}, \omega$). This denotes the process of using current models **B** and a specified value of ω to update the regularization weights **w**, as in (11) to (13).

Using these notations, we summarize the algorithm in Algorithm 1, which is applicable to both single and multiple tasks. Here the upper index τ denotes outer iterations, where each iteration τ corresponds to a stage with distinct value of ω; the lower index t indexes the inner iterations inside each stage. The δ is a shrinking factor that is smaller than 1 when $\gamma > 0$, and a growing factor that is larger than 1 when $\gamma < -1$. The iteration will keep going until all features are removed from the model. Then a cross-validation can be used to select the best model along the solution path.

In case of classification tasks with high-dimensional features, one can consider the sparse logistic regression [27], $\min - \sum_{i=1}^n \ln\left(1 + \exp[-\beta^\top x_i \cdot y_i]\right) + |\mathbf{w}\odot\beta|_1$which can benefit from our dynamic shrinking approach as well. Similarly, the iterative procedures will decompose into two subproblems: when fixing **w**, it becomes a standard logistic regression with adaptive ℓ_1-regression; and when fixing β, the problem is identical to (6) and can be solved accordingly.

4. RELATED METHODS

4.1 Re-weighted LASSO

In [4], an interesting, re-weighted LASSO algorithm was proposed to improve the sparsity of LASSO. After solving a standard LASSO at time t (starting from $t = 0$), it will compute a set of adaptive regularization weights $w_i^{(t+1)} = 1/(|\beta_i^{(t)}| + \epsilon)$, and then use $w_i^{(t+1)}$'s to adaptively penalize the ℓ_1 regularization. Here ϵ is a small number to ensure that a zero component in β does not strictly prohibit a nonzero estimate at the next step. As can be seen, the algorithm repeatedly performs the adaptive LASSO by using the absolute value of the inverse of previous model coefficients as the regularization weights for the next iteration. Such iterations may easily get trapped in local optimal solution due to the sensitivity of the convergence on initial values. In comparison, our approach avoids pre-mature convergence by continuously adjusting the global regularization strength.

4.2 Mixed-norm Regularization

4.2.1 Univariate Regression Cases

In case there exists grouping structures among input variables, the LASSO algorithm has been extended to recover such grouping. For example, the elastic net algorithm penalizes both the ℓ_1 and ℓ_2 norm of the model [39], which encourages a grouping effect such that strongly correlated predictors tend to be in or out of the model together. When the groupings of the inputs are available as prior knowledge, the group LASSO [34] penalizes the ℓ_2-norm of each group as a unit for variable selection, using the following optimization framework,

$$\min \left\|\sum\nolimits_{l=1}^L \mathbf{X}_l\beta_l - y\right\|_F^2 + \lambda \sum\nolimits_{i=1}^L \cdot\sqrt{p_l}\|\beta_l\|_2.$$

Here, the predictors are assumed to have l groups with group size p_l; $\mathbf{X}_l$ represents predictors of the lth group, with corresponding coefficient β_l. The group LASSO achieves sparse feature selection at the group level: depending on λ, an entire group of predictors is either selected simultaneously in the model, or will be removed together.

4.2.2 Multi-variate/Multi-task Cases

Multi-task learning has drawn considerable interest in data mining [2, 3]. It assumes that different tasks share some common structures, and enforcing the task relatedness can help improve the learning performance. We focus on sparse multi-task learning [16, 23, 34, 17], namely joint feature selection in multiple tasks needs to be performed.

The ℓ_1/ℓ_2 penalty of group lasso has been used to recover inputs that are jointly relevant to all of the outputs, or tasks, by applying the ℓ_2-norm to outputs instead of groups of inputs.For example, [23] proposed to penalize the sum of the ℓ_q-norms of the blocks of coefficients associated with each feature across tasks, which is called mixed-norm or ℓ_1/ℓ_q regularization. One appealing property is that it encourages multiple predictors from different tasks to share similar parameter sparsity patterns. Let $\mathbf{B} = [\beta^1,\ \beta^2,\ ..., \beta^k]$, and define $\mathbf{B}^i \in R^{1\times K}$ as the ith row in the model coefficient matrix $\mathbf{B}$. Then the objective function of the ℓ_1/ℓ_q regularization is as follows:

$$\min_{\mathbf{B}} \sum_{k=1}^{K} \left\| \mathbf{X}^k \beta^k - \mathbf{y}^k \right\| + \lambda \sum_{i=1}^{D} \|\mathbf{B}\|_{\ell_1/\ell_q} .$$

Here $\|\mathbf{B}\|_{\ell_1/\ell_q} =$ is the block ℓ_1/ℓ_q norm

$$\|\mathbf{B}\|_{\ell_1/\ell_q} = \sum_{i=1}^{D} \left(\sum_{j=1}^{K} \mathbf{B}_{ij}{}^{q} \right)^{\frac{1}{q}} .$$

When $q = 2$, we have a block ℓ_1/ℓ_2 norm, which is identical to the group LASSO [34]. Other choices have also been studied such as ℓ_1/ℓ_∞ [30]. The mixed-norm regularization encourages simultaneous feature selection across tasks. Namely, a given feature is either selected as relevant for all the tasks' output simultaneously, or is excluded all-together for all the tasks. Such regularization is very effective if the underlying task relation satisfies such assumption. However it can be too restrictive in some other applications.

In [17], an adaptive multi-task LASSO framework was proposed which combines adaptive regularization with the mixed-norm regularization, as

$$\min_{\beta,\theta,\rho} \mathcal{L}(\beta) + \lambda_1 \sum_{j=1}^{D} \theta_j \sum_{i=1}^{K} |\beta_j^k| + \lambda_2 \sum_{j=1}^{D} \rho_j \|\beta_j\|_2 + \log Z(\theta, \rho).$$

Here $\mathcal{L}$ is the loss function; the second term is an adaptive LASSO that imposes ℓ_1-norm penalty with strength $\lambda_1 \cdot \theta_j$ on $|\beta_j|_1$ from all tasks; the third term is a mixed-norm regularization together with an adaptive weights, which imposes the penalty $\lambda_2 \cdot \rho_j\ \|\beta_j\|_2$; the last term is a normalization factor on the conditional probability $p(\beta|\theta, \rho)$. The whole framework has an elegant Bayesian interpretation. It achieves sparsity both across tasks and within each task. However, the regularization weights are assumed to be spanned by features from extra domains with prior knowledge, which might not be available in general multi-task learning; on the other hand, it separates the learning of the model and the regularization profile.

4.3 Regularization Path

The dynamic shrinking process of the proposed algorithm is illustrated in Figure 1, where the strength of the ℓ_1-regularization gradually increases, leading to a solution path. Due to the interplay between adaptive weights $\mathbf{w}$ and models coefficients $\mathbf{B}$, the whole solution path of $\mathbf{B}$ is connected: each solution $\mathbf{B}$ is affected by its predecessor. This means, the effect of system evolution is inherited from one stage τ to the next stage $\tau + 1$, or from one iteration t to the next iteration $t + 1$ inside a single stage. In other words, the solutions have to be obtained in a sequential manner. For the standard LASSO, in comparison, the solution path can actually be obtained by training a number of independent LASSO's with different λ's.

Note that the solution path of LASSO can also be obtained in a sequential manner by using the least angle regression [8], which fully exploits the piecewise linear structures of the solutions. However, an important difference is that, our approach will *re-define* the LASSO regression in each iteration. To see this, note that any adaptive LASSO problem min $\|\mathbf{X}\beta - \mathbf{y}\|_2^2 + |\mathbf{w} \odot \beta|_1$ can be converted to a LASSO min $\|\mathbf{X}\mathbf{W}^{-1}\overline{\beta} - \mathbf{y}\|_2^2 + |\overline{\beta}|_1$ where $\overline{\beta} = \mathbf{w} \odot \beta$ and $\mathbf{W} = diag(\mathbf{w})$. In our approach, since the regularization weight vector $\mathbf{w}$ keeps updating, therefore each iteration is equivalent to a LASSO problem with continuously rectified data $\mathbf{X}\mathbf{W}^{-1}$, making it different from traditional solution path. It will be a very interesting topic to explore the solution path structures of our dynamic shrinking approach, so as to make it more computationally efficient.

5. EMPIRICAL RESULTS

In this section, we perform extensive experiments to examine the performance of our approach, in both simulation data sets and real-world bioinformatics application.

5.1 Competing Methods

Altogether, we implement and compare the following algorithms:

1. Standard LASSO algorithm [29]: We use the LARS algorithm to generate the solution path;

2. Adaptive LASSO [38]: We use inverse of ridge regression coefficient to compute $\mathbf{w}$ for each task and average them as the shared regularization. Then we rescale $\mathbf{w}$ to generate the solution path;

3. Adaptive LASSO-II [13]: We use inverse of the marginal regression coefficient to compute $\mathbf{w}$ for each task and average them as the shared regularization, then we rescale the regularization to generate the solution path;

4. Multi-task LASSO [23]: We choose different values of the initial λ to compute the solution path;

5. Re-weighted LASSO [4]: We choose different values of the initial λ (each initiates a series of iterations till convergence) to generate the solution path;

6. Our approach: We simply choose $\gamma = 1$, an initial norm $\omega^0 = 1e8$, and shrinking factor $\delta = 0.8$; we can generate a solution path throughout the iterations until all features are removed. Results on using negative power $\gamma < -1$ is similar and therefore removed due to space limit.

We use the following measurements to evaluate the performance of different methods:

1. Specificity (SPC) versus true-positive-rate (TPR) (SPC VS TPR) curve based on solution paths from different algorithms;

2. Cross-validated mean-squared-error (CV-MSE): we report 5-fold cross-validated error of different methods;

3. Cross-validated F-score (CV-Fscore): we comput the 5-fold cross-validated F-score for different methods;

We use the SLEP sparse learning package [18] to implement our approach. All codes are written in Matlab and run on a cluster server with 2.2 ∼ 2.8 GHz CPU.

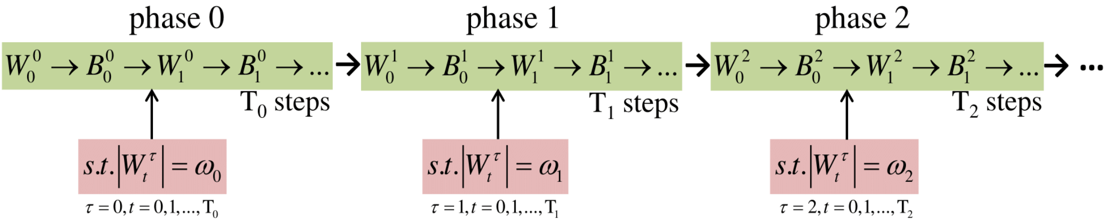

phase 0
phase 1
phase 2
$W_0^0 \to B_0^0 \to W_1^0 \to B_1^0 \to ... \to W_0^1 \to B_0^1 \to W_1^1 \to B_1^1 \to ... \to W_0^2 \to B_0^2 \to W_1^2 \to B_1^2 \to ... \to \cdots$
T_0 steps
T_1 steps
T_2 steps
$s.t. |W_t^\tau| = \omega_0$
$\tau = 0, t = 0,1,...,T_0$
$s.t. |W_t^\tau| = \omega_1$
$\tau = 1, t = 0,1,...,T_1$
$s.t. |W_t^\tau| = \omega_2$
$\tau = 2, t = 0,1,...,T_2$

Figure 1: Illustration of the regularization path of our approach. Here, $\{\omega_0, \omega_1, \omega_2, ...\}$ is a sequence such that the global strength of ℓ_1-regularization grows stronger.

5.2 Single Task Regression

First we use single task sparse regression problem to test the performance of different methods. Following the details in [4], we simulate the data set of $n = 100$ samples with dimensionality $D = 256$, and the design matrix is an n-by-d i.i.d. Gaussian entries. Among the 256 features, only $p = 20$ are relevant features with randomly chosen non-zero β entries from a zero-mean unit-variance Gaussian distribution. We then use the linear relation $\mathbf{y}_i = \mathbf{x}_i\beta + \mathcal{N}(0, \sigma^2)$ to generate the response $\mathbf{y}$.

Results are shown in Figure 2, where each algorithm is marked by their indexes specified in Section 5.1. In this data set, multi-task LASSO (method (4)) is identical to standard LASSO (method (1)) and therefore is removed. As can be seen, our approach is superior in terms of picking out relevant covariates throughout the whole solution path, demonstrating the effectiveness of annealed sparsity in improving the sparse model selection. In the meantime, the cross-validated mean-squared-error and F-score of our approach are also the best among competing methods.

5.3 Multi-task Regression

In this experiment we simulate data with $K = 5$ tasks, each task has $n = 40$ samples with dimension $D = 100$. For each task design matrix is an i.i.d Gaussian distribution, and we assume the linear relation $\mathbf{y}_i^k = \mathbf{x}_i^k \beta^k + \mathcal{N}(0, \sigma^2)$, and for the relevant features, the corresponding entries in β^k's are randomly chosen from the distribution $3 + \mathcal{N}(0, 1)$. Here we generate two types of multi-task data.

1. Multitask-I: strict group-wise sparsity. We choose $p = 20$ relevant features for all tasks, and each row of the model coefficient matrix $\mathbf{B}$ is either all zeros or all non-zeros, meaning that one feature is either relevant to all tasks, or excluded from all tasks;

2. Multitask-II: mixed sparsity patterns. We then introduce a perturbation on the model coefficients $\mathbf{B}$: for each non-zero row of $\mathbf{B}$, we randomly pick one entry and set it to zero; in this case, each row of $\mathbf{B}$ can have both zero and non-zero entries. Namely it has a mixed sparsity pattern (across-group and within-group).

We report the results in Figure 3 and Figure 4. We can observe that our approach has the best performance in terms of both feature selection (F-score) and regression (predicting error), on both types of multi-task data sets. The adaptive LASSO-II [13] using the inverse of the marginal regression coefficients as adaptive weights seems to perform better than the adaptive LASSO using ordinary-least-squares coefficients. The LASSO considers each task separately and can be less accurate. Another observation is that, in case of mixed sparsity patterns, all algorithms perform worse than in the case of strict group-wise sparsity, in particularly judged by feature selection accuracy (F-score). Nevertheless, our approach still performs the best among competing methods.

We also experiment with different noise levels to test the noise tolerance shown in Figure 5. As can be observed, our approach is competitive under different noise levels.

5.4 Algorithm Behaviors

In this section, we study properties of the proposed method from different perspectives.

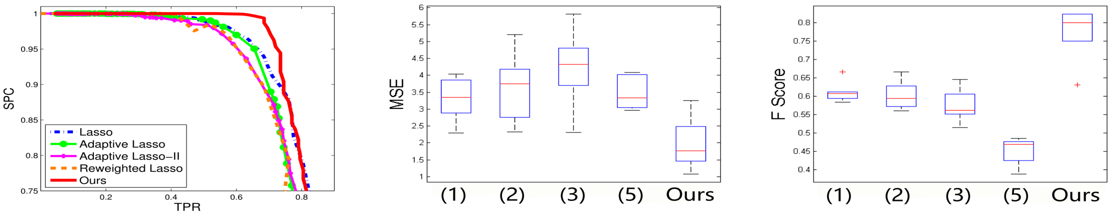
5.4.1 *Solution Path*

First, to have a direct picture on the shrinking behaviour of our method, we plot the solution path of our approach in Figure 6. Here we use the multi-task simulation data with group wise sparsity under the highest noise level ($\delta = 3$). To prevent visual cluttering, we only plot the solution path for one task, and we only demonstrate 10 of the 20 relevant features and all the rest 80 irrelevant features.

We have several interesting observations. First, note that when the regularization is relatively weak, the solution paths are all smooth; when the regularization becomes stronger, solution paths begin to show clear stage-wise behaviour: the coefficient value is relatively stable within each stage, but may change significantly across stages (due to the change of ω), indicating that the system state goes through significant changes. Second, the solution path is quite non-monotonic. With the growing strength of regularization ($\frac{1}{\omega}$), we can observe that the model coefficients first expand and then gradually shrink. This is in sharp contrast to the solution path of the LASSO, whose solution path almost monotonically shrinks with growing regularization.

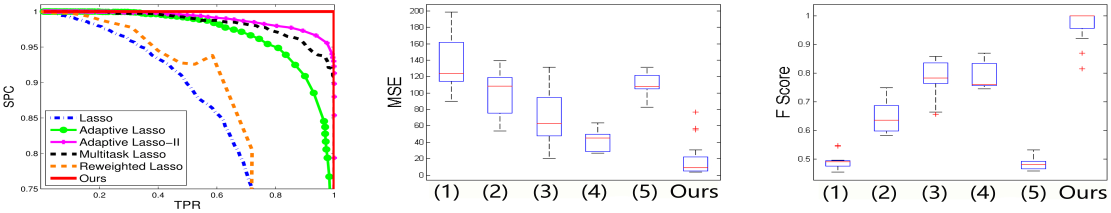
The non-monotonic shrinking can be quite beneficial in practice. Note that in the beginning stage, both relevant and irrelevant features have large model coefficients, meaning that they are difficult to differentiate. When the regularization grows stronger, interestingly, we can see that most relevant features begin to expand, while most irrelevant features begin to shrink. This becomes particularly obvious around $\frac{1}{\omega} \approx 10^{-4.5}$, where the majority of irrelevant features suddenly shrink to zero, while relevant features have a jump increase in their coefficients. This is quite beneficial in practical feature selection problems.

Competing mechanism of annealing. In the beginning, under weak global ℓ_1-penalty, model coefficients are dense, indicating that the "energy" of the system is distribut-



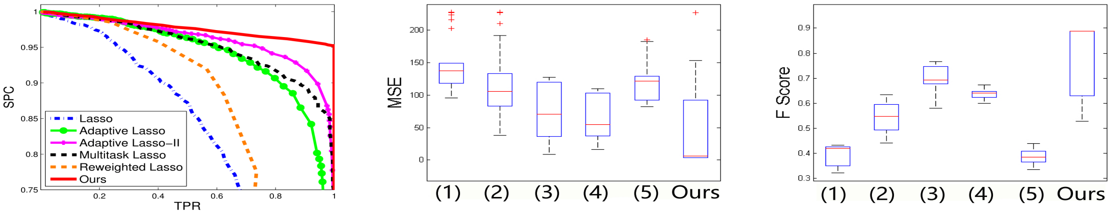
(a) SPC VS TPR (b) Cross-validated MSE (c) Cross-validated F-score

Figure 2: Performance for different methods on single task data, with noise $\sigma = 1$.



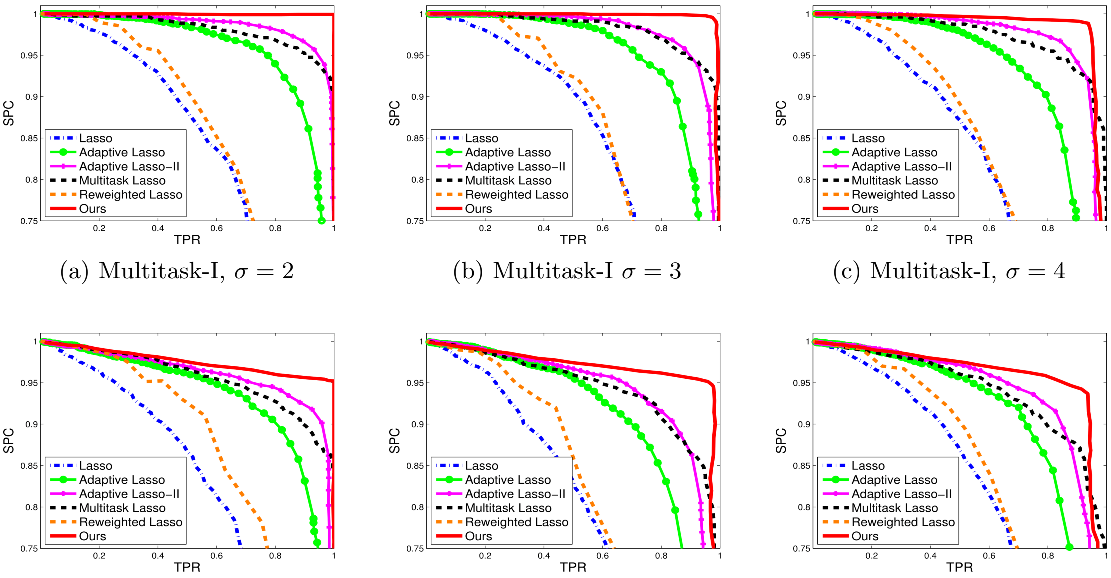
(a) SPC VS TPR (b) Cross-validated MSE (c) Cross-validated F-score

Figure 3: Results on multitask-I data (strict group sparsity), with noise $\sigma = 1$.



(a) SPC VS TPR (b) Cross-validated MSE (c) Cross-validated F-score

Figure 4: Results on multitask-II data (mixed sparsity pattern), with noise $\sigma = 1$.



(a) Multitask-I, $\sigma = 2$ (b) Multitask-I $\sigma = 3$ (c) Multitask-I, $\sigma = 4$

(d) Multitask-II, $\sigma = 2$ (e) Multitask-II $\sigma = 3$ (f) Multitask-II, $\sigma = 4$

Figure 5: Results on different noise levels for multitask-I (1st row) and multitask-II (2nd row).

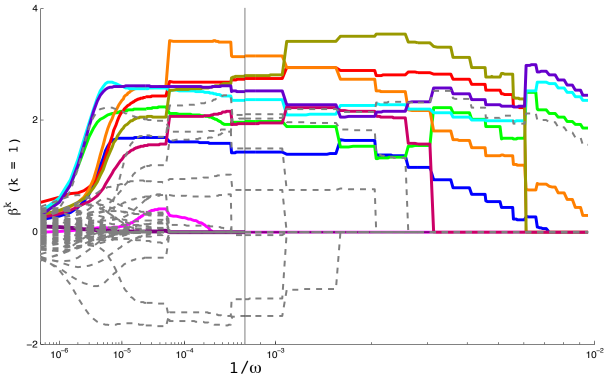
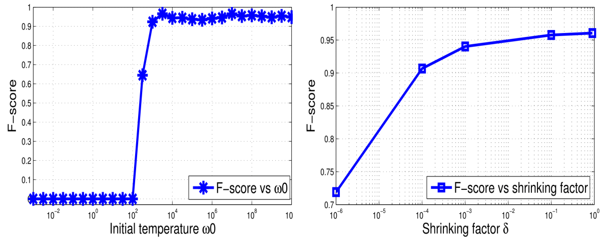


Figure 6: Solution path for dynamic shrinking. Thick colored lines represent relevant features, and dashed lines for irrelevant features. Vertical line in the middle marks change of display scales (for visual clarity). Regularization increases from left to right.

ed somewhat uniformly among competing features. As the regularization grows, the system begins cooling toward a lower-energy state; in the meantime, energy distribution becomes more concentrated. That is, competitive features (in terms of better prediction in the least-square) will attract more energy from irrelevant features, making the latter shrink. This is why we observe significant growth of some feature magnitude even though the global sparsity enhances. Such energy re-allocation through annealing solves the feature selection problem in an effective manner.

5.4.2 *Parameter Selection*

In this section, we study how the performance of our approach is affected by the following two parameters: the ω^0 that controls the initial "temperature" of the system; the shrinking factor δ that controls the "cooling rate" of the system. The performance is measured by the F-score using the multitask-I data set with $\sigma = 1$.



(a) Initial temperature (ω^0) (b) Shrinking factor (δ)

Figure 7: Performance of our method versus different parameters.

First, we examine the performance w.r.t. ω^0 chosen from some grid points $\{10^{10}, 10^9, ..., 10^{-3}\}$. As can be seen from Figure 7(a), in a wide range of high initial temperatures, the performance of our approach is quite satisfactory; when the initial temperature is below a certain value, the performance quickly drops. This coincides with our expectation, since a low initial temperature fails to start the whole system with sufficient energy and as a result the iterations could quickly stop at a local optima. In practice, we simply choose ω^0 as a large value such as $1e8$.

In Figure 7(b), we examine the performance of our approach w.r.t. the shrinking factor. As can be observed, more aggressive shrinking scheme ($\delta \to 0$) makes the performance worse; in comparison, milder shrinking scheme ($\delta \to 1$) allows the system to evolve slowly such that the "annealing" is sufficient, but it is computationally more expensive. In practice, we find that $0.2 < \delta < 0.8$ can strike a balance between efficiency and the quality of annealing.

5.5 Bioinformatics Application

In this section, our task of expression quantitative trait loci (eQTL) is to identify genes whose DNA copy (DNA copy-number data as *input*) are associated with the mRNA expression level of six P53 target genes (normalized expression data as *response*). Note that P53 is a well-known tumor suppressor gene. The data set is obtained from Cancer Cell Line Encyclopedia (CCLE) project[1], with DNA copy-number of 23316 genes across 1011 samples. The six target genes include CDKN1A, PMAIP1, BBC3, MSH2, PML and PRKAA2, which are of particular relevance to melanoma as suggested by biological experts [32]. The regulatory genes identified through our regression analysis will then help understand the whole P53 regulation mechanism for cancer, and in particular melanoma.

In the application, we treat the eQTL of six P53 target genes as six tasks, since we believe that the regulating processes on all these melanoma-related genes should share some underlying mechanism. We have used the 5-fold cross-validated error to select the best model. Table 1 reports the 5-fold CV-MSE for all competing methods, from which we can see that our method achieves the lowest fitting error. This fully illustrates the superior performance of the proposed dynamic shrinking scheme in high-dimensional, real-world multi-task learning problems.

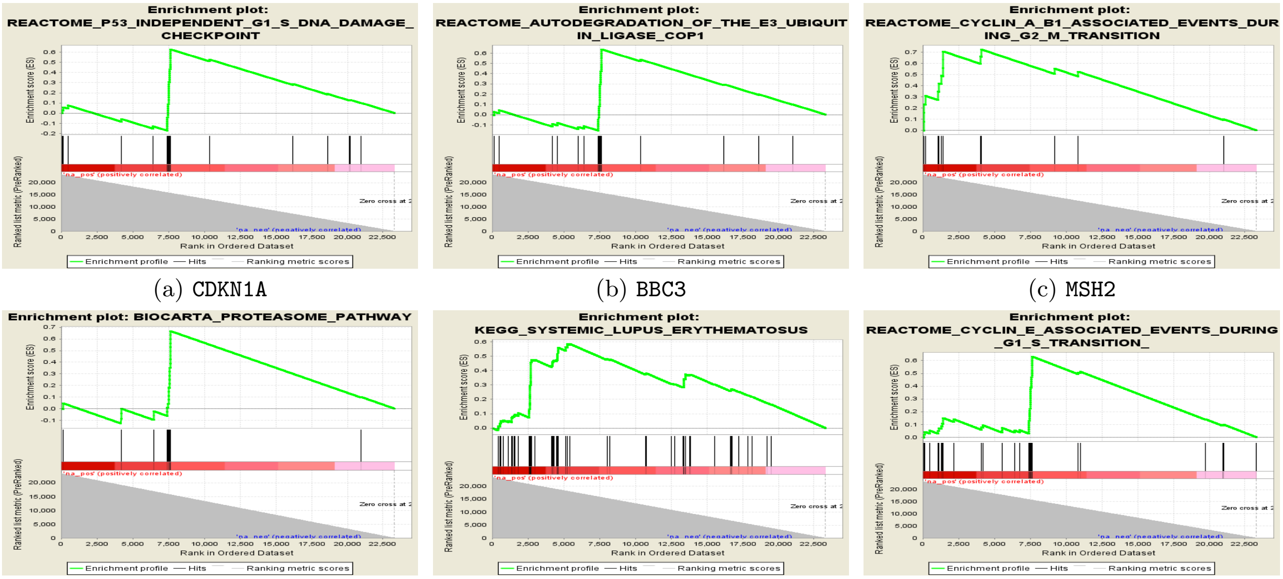
We further explore whether the selected genes by our method makes biological sense, by following the common practice of gene set enrichment analysis (GSEA) [28]. Specifically, we rank the selected genes in each task based on the regression coefficients and feed the ranking to GSEA2-2.2.2 software. We consider the canonical pathways/gene sets provided by the Molecular Signatures Database[2]. For each task, GSEA returns a number of significant pathways/gene-sets under false discovery rate (FDR) 5%, and we pick one example pathway to illustrate in Figure 8. Here, the pathway name is marked on top of each figure; the red bar denotes the ranking of the β-coefficient for each task, and the black lines mark the genes belonging to selected pathway. Our approach identifies more than 100 significant pathways for each task, which is much larger than other methods.

These significant pathways based on our computed gene ranking are very relevant to cancers and/or melanoma, as discussed below:

- The gene CDKN1A, cyclin-dependent kinase inhibitor 1A, itself is relevant to cell cycle. The significant pathway "REACTOME_P53_INDEPENDENT_G1_S..." includes genes in p53-Independent G1/S DNA damage checkpoint, which has been shown to be quite relevant to dysfunctional cell cycle causing cancer [20].

[1] http://www.broadinstitute.org/ccle/home

[2] http://software.broadinstitute.org/gsea/msigdb/collections.jsp.



(a) CDKN1A (b) BBC3 (c) MSH2

(d) PMAIP (e) PML (f) PRKAA2

Figure 8: Our ranking of the genes, as well as one example of the identified pathways for each task based on this ranking, via gene enrichment analysis on the CCLE data.

Table 1: CV-MSE on CCLE data set

Method	5 fold Cross-validated MSE
LASSO	3.2530
Adaptive LASSO	1.3078
Adaptive LASSO-II	1.3701
Multitask LASSO	3.2451
Re-weighted LASSO	1.5507
Ours	1.1165

- The gene BBC3 is a protein that cooperates with direct activator proteins to induce mitochondrial outer membrane permeabilization and apoptosis. The pathway "REACTOME_AUTODEGRADATION_OF_THE..." include genes involved in autodegradation of the E3 ubiquitin ligase COP1. Destruction of COP1 results in abrogation of the ubiquitination and degradation of p53 [7].

- The gene MSH2 is involved in cyclin A/B1 associated events during G2/M transition and is a protein coding gene. In literatures, its related pathways are all about cancer and cell cycle, or checkpoint control. The pathway we find, "REACTOME_CYCLIN_A_B1_ASSOCIATED...", is also a cell cycle gene set. It is responsible for phosphorylation of nuclear lamins and histones [26], which in turn regulates G2/M transition, thus controlling cell cycle progression by cyclin-dependent protein kinases in G1/S and G2/M transitions.

- The gene PAMIP promotes activation of caspases and apoptosis. It contributes to p53/TP53-dependent apoptosis after radiation exposure. In the proteasome pathway "BIOCARTA_PROTEASOME_PATHWAY", the regulated proteolysis of proteins by proteasomes removes damaged or improperly translated proteins from cells, and aids caspases and apoptosis [1].

The above results show that our approach not only predicts target gene expressions more accurately, but also identifies biologically meaningful molecular predictors.

6. CONCLUSIONS

In this paper, we propose a dynamic shrinking framework to compute adaptive regularization in sparse (multi-task) regression. Our key contribution is to introduce the concept of annealing in sparse model estimation and feature selection, through an iterative, self-adapting and self-competing mechanism. Empirically, the annealing process can improve the accuracy of models in particular in multi-task problems. In the future, we will study how to explore underlying structures of the dynamic solution path to make it computationally more efficient; we also want to incorporate explicit, task-level constraints to make the learned model coefficients more useful for subsequent learning tasks. Finally, we are trying to build a more rigorous, mathematical connection between our approach and annealing so as to fully characterize the behaviour of system evolutions.

Appendix

To derive (7), we use the Lagrangian of (6). We first drop the non-negativity constraint. Then the Lagrangian can be written as

$$J = \sum_d \theta_d w_d^{-\gamma} + \alpha \left(\sum_d w_d - \omega \right).$$

By setting $\frac{\partial J}{\partial w_d} = 0$, we have

$$\alpha = \frac{\theta_d \gamma}{w_d^{1+\gamma}}. \tag{14}$$

Plugging the above relation in the constraint $\sum_d w_d = \omega$, then we have

$$\alpha = \left(\frac{\sum_d (\theta_d \gamma)^{\frac{1}{1+\gamma}}}{\omega} \right)^{1+\gamma},$$

so we have

$$w_d^{1+\gamma} = \frac{\theta_d\gamma}{\alpha} = \frac{\theta_d\gamma \cdot \omega^{1+\gamma}}{\left(\sum_d (\theta_d\gamma)^{\frac{1}{1+\gamma}}\right)^{1+\gamma}}.$$

Plugging the above equation in (14), we finally have

$$w_d = \frac{\theta_d^{\frac{1}{1+\gamma}}}{\sum_d \theta_d^{\frac{1}{1+\gamma}}}\omega.$$

Since $\theta_d = \sum_k |\beta_k^d| \geq 0$, and $\omega \geq 0$, the solution will satisfy the non-negative constraints automatically. This completes the proof of solution (13).

References

[1] I. Amm, T. Sommer, and D. H. Wolf. Protein quality control and elimination of protein waste: The role of the ubiquitin–proteasome system. *Biochimica et Biophysica Acta (BBA)-Molecular Cell Research*, 1843(1):182–196, 2014.

[2] A. Argyriou, T. Evgeniou, and M. Pontil. Convex multi-task feature learning. *Machine Learning*, 73(3):243–272, 2008.

[3] S. Ben-david and R. Schuller. Exploiting task relatedness for multiple task learning. In *Proceedings of the 16th Annual Conference on Learning Theory*, pages 567–580, 2003.

[4] E. J. Candes, M. B. Wakin, and S. Boyd. Enhancing sparsity by reweighted l1 minimization. *Journal of Fourier Analysis and Applications, special issue on sparsity*, 14(5):877–905, 2008.

[5] S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Review*, 43(1):129–159, 2001.

[6] D. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1304, 2006.

[7] D. Dornan, H. Shimizu, A. Mah, T. Dudhela, M. Eby, K. O'Rourke, S. Seshagiri, and V. M. Dixit. Atm engages autodegradation of the e3 ubiquitin ligase cop1 after dna damage. *Science*, 313(5790):1122–1126, 2006.

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

[9] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image Processing*, 37(4):3736 – 3745, 2006.

[10] J. Fan, Y. Feng, and Y. Wu. Network exploration via the adaptive lasso and scad penalties. *The Annals of Applied Statistics*, 3(2):521–541, 2009.

[11] J. Fan and R. Li. Variable selection via nonconcave penalized like- lihood and its oracle properties. *Journal of the American Statistical Association*, 96:1348–1360, 2001.

[12] M. Figueiredo, R. Nowak, and S. Wright. Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems. *IEEE Journal on Selected Topics in Signal Processing*, 1(4):586 –597, 2007.

[13] J. Huang, S. Ma, and C.-H. Zhang. Adaptive lasso for sparse high-dimensional regression models. *Statistica Sinica*, pages 1603–1618, 2008.

[14] S. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky. An interior-point method for large-scale l1-regularized least squares. *IEEE Transactions on Signal Processing*, 1(4):606–617, 2007.

[15] S. Kirkpatrick, C. Gelatt, and M. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1993.

[16] M. Kowalski. Sparse regression using mixed norms. *Applied and Computational Harmonic Analysis*, 27(3):303–324, 2009.

[17] S. Lee, J. Zhu, and E. P. Xing. Adaptive multi-task lasso: with application to eqtl detection. In *Advances in Neural Information Processing Systems*, pages 1306–1314, 2010.

[18] J. Liu, S. Ji, and J. Ye. Slep: Sparse learning with efficient projections. *Arizona State University*, 2009.

[19] A. C. Lozano and G. Świrszcz. Multi-level lasso for sparse multi-task regression. In *Proceedings of the International Conference on Machine Learning*, 2012.

[20] N. Mailand, J. Falck, C. Lukas, R. G. Syljuåsen, M. Welcker, J. Bartek, and J. Lukas. Rapid destruction of human cdc25a in response to dna damage. *Science*, 288(5470):1425–1429, 2000.

[21] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Supervised dictionary learning. In *Advances in Neural Information Processing Systems*, 2009.

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

[23] G. Obozinski, M. J. Wainwright, and M. I. Jordan. High-dimensional union support recovery in multivariate regression. In *Neural Information Processing Systems*, pages 1217–1224, 2008.

[24] S. Puniyani, S. Kim, and E. P. Xing. Multi-population gwa mapping via multi-task regularized regression. *Bioinformatics*, 26(12):208–216, 2010.

[25] S. Raman and V. Roth. Sparse point estimation for bayesian regression via simulated annealing. *Lecture Notes in Computer Science*, 7476:317–326, 2012.

[26] B. M. Sefton and S. Shenolikar. Overview of protein phosphorylation. *Current Protocols in Protein Science*, pages 13–1, 2001.

[27] S. K. Shevade and S. S. Keerthi. A simple and efficient algorithm for gene selection using sparse logistic regression. *Bioinformatics*, 19(17):2246–2253, 2003.

[28] A. Subramanian et al. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences*, 102(43):15545–15550, 2005.

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

[30] J. Tropp, A. Gilbert, and M. Strauss. Algorithms for simultaneous sparse approximation, part ii: Convex relaxation. *Signal Processing*, 86:572–588, 2006.

[31] L. H. V. Vlack. *Elements of Materials Science and Engineering*. Addison-Wesley, 1985.

[32] C. Wei et al. A global map of p53 transcription-binding sites in the human genome. *Cell*, 124(1):207–219, 2006.

[33] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2009.

[34] M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society Series B*, 68(1):49–67, 2006.

[35] J. Zhang, W. Cheng, Z. Wang, Z. Zhang, W. Lu, G. Lu, and J. Feng. Pattern classification of large-scale functional brain networks: Identification of informative neuroimaging markers for epilepsy. *PLoS ONE*, 7(5):e36733, 2012.

[36] K. Zhang, J. Gray, and B. Parvin. Sparse multitask regression for identifying common mechanism of response to therapeutic targets. *Bioinformatics*, 26(12):97 – 105, 2010.

[37] P. Zhao and B. Yu. On model selection consistency of lasso. *Journal of Machine Learning Research*, 7:2541–2563, 2007.

[38] H. Zou. The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, 101(476):1418–1429, 2006.

[39] 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.