2011 11th IEEE International Conference on Data Mining

Learning Protein Folding Energy Functions

Wei Guan*, Arkadas Ozakin†, Alexander Gray*,
Jose Borreguero‡, Shashi Pandit‡, Anna Jagielska‡, Liliana Wroblewska‡, and Jeffrey Skolnick‡
*College of Computing, Georgia Institute of Technology, Atlanta, Georgia 30332
Email: wguan@gatech.edu, agray@cc.gatech.edu
†Georgia Tech Research Institute, Atlanta, Georgia 30318
Email:arkadas.ozakin@gtri.gatech.edu
‡Department of Biology, Georgia Institute of Technology, Atlanta, GA
Email:jeffrey.skolnick@biology.gatech.edu

***Abstract*—A critical open problem in *ab initio* protein folding is protein energy function design, which pertains to defining the energy of protein conformations in a way that makes folding most efficient and reliable. In this paper, we address this issue as a weight optimization problem and utilize a machine learning approach, learning-to-rank, to solve this problem. We investigate the ranking-via-classification approach, especially the RankingSVM method and compare it with the state-of-the-art approach to the problem using the MINUIT optimization package. To maintain the physicality of the results, we impose non-negativity constraints on the weights. For this we develop two efficient non-negative support vector machine (NNSVM) methods, derived from L2-norm SVM and L1-norm SVMs, respectively. We demonstrate an energy function which maintains the correct ordering with respect to structure dissimilarity to the native state more often, is more efficient and reliable for learning on large protein sets, and is qualitatively superior to the current state-of-the-art energy function.**

***Keywords-ab initio* protein folding, energy function, learning-to-rank, support vector machine, non-negativity constrained SVM optimization**

I. Introduction

Proteins are polymers assembled from 20 naturally occurring amino acids, which fold to unique, biologically active, three-dimensional conformations called *native structure*s. Their biological functions are governed by their three-dimensional structures, which in turn are fully determined by their amino acid sequences. Predicting the native structure of a protein from its amino acid sequence is one of the most important and challenging scientific problems in contemporary biology and chemistry [1]. The capability to reliably make such predictions would allow biochemists to design drugs more efficiently, understand biological processes in detail, and answer fundamental questions about biological systems, diseases, immune response, and more.

The experimental determination of protein structure is a time consuming and expensive process. Hence, computational methods play an essential role in the prediction of the native structures of proteins. There are three classes of computational approaches to protein structure prediction: homology modeling, threading, and *ab initio* folding. Homology modeling and threading methods utilize proteins with known structure that are evolutionarily related to the target protein with unknown structure [2]. If one can not find such proteins in the available library of experimentally resolved protein structures, the only remaining approach to predicting the native structure is *ab initio* folding.

Ab initio folding attempts to find the native structure of a protein “from scratch”. The fundamental assumption in *ab initio* folding is the existence of a *free energy function* that assigns an energy value to each three-dimensional structure the protein can in principle assume. The native structure is assumed to be the one with the lowest energy [3]. Thus, there are two main ingredients in *ab initio* folding: The design of a reliable energy function, and the development of an efficient approach to search the space of all possible conformations for the one with the lowest energy. In this paper, we focus on the first problem.

The energy functions used in *ab initio* folding are physics-based: for a given three dimensional configuration of a protein, one first calculates various terms contributing to the total energy such as electrostatic energy, covalent bonding energy, Van der Waals energy, etc., and then adds these terms to obtain the total energy [4]. While these terms are based on physics, their functional forms are sometimes approximate, and the coefficients that appear are obtained by various fitting procedures. In this work, we represent the total energy of a configuration as a linear combination of these physics-based energy terms, and optimize the coefficients.

The fitness of a given energy function for a given protein can be visually inspected by plotting the total energy versus the structural dissimilarity from the native structure. In order to do this, one generates many possible conformations and computes the total energy and dissimilarity from the native structure for each. [1] Fig. 1 shows such a plot for a desirable energy function. As can be seen, the energy value is higher for conformations that have large dissimilarities from the native structure, with a roughly monotonic trend. Due to

[1] There are various notions of structural similarity used in the literature, the most basic one being the root mean squared distances (RMSD) [5] between the building blocks (e.g., atoms) of the protein as represented in two candidate structures aligned in three-dimensional space.

1550-4786/11 $26.00 © 2011 IEEE
DOI 10.1109/ICDM.2011.88

the monotonic trend, reducing the energy corresponds to getting closer to the native structure during *ab initio* folding procedure. If one can construct an energy function that has energy vs. structural dissimilarity plots like that of Fig. 1, one can hope to reproduce a similar trend for proteins with unknown structure.

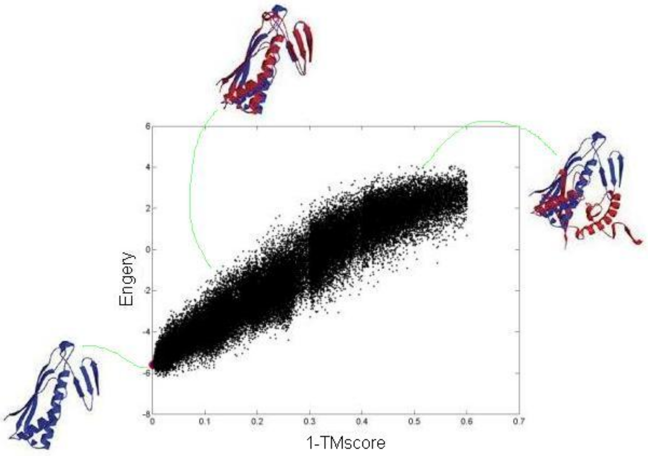


Figure 1. Energy versus Structural Dissimilarity Plot: each dot represents a non-native conformation, and the red square represents the native structure

For a given protein, we represent the total energy of a conformation s_i as $f(s_i) = w^T x_i = E_i$, where $x_i \in R^n$ represents the collection of the energy terms for s_i, and $w \in R^n$ are the weight coefficients. Treating w as the unknown, the task of learning an *ab initio* protein folding energy function becomes a weight optimization problem. Much of the literature on this problem is based on maximizing correlation (or related quantities) between the total energy and the dissimilarity [4],[6]. In this paper, we propose a ranking-based approach to this problem. Namely, given m conformations for each protein, we search for the weights w such that for each protein a meaningful *subset* of the constraints below are satisfied.

- Total energy of the native structure is the minimum, that is, $E_0 < E_j$ for all $j = 1, \cdots, m$.
- Energy of random conformations with smaller structural dissimilarity are smaller than those with larger dissimilarity, that is, if $r_j < r_i$, then $E_j < E_i$.

The paper is organized as follows. We begin in Section 2 by converting the weight optimization problem into a learning-to-rank task and then describe RankingSVM, a ranking-via-classification method that we utilize. Due to physicality constraints, we restrict the problem to non-negative weights. Section 3 describes two efficient algorithms to solve the constrained ranking problem. Section 4 summarizes the experiment results. Section 5 concludes the study and discusses future work.

II. Approach: Weight Learning By Ranking

The problem of learning protein energy function can be reduced into a learning-to-rank problem if we consider the ordering derived from structure dissimilarity as the true ordering over the protein conformations, and the ordering derived from the energy function as the predicted ordering. The reduced problem seeks to find a ranking function $f(s) = w^T x$ that optimally approximates the true ordering. That is, for each protein, we expect the predicted ordering to satisfy the following requirements as closely as possible:

i) Rank the native structure above other conformations.
ii) Rank conformations with lower structural dissimilarities above those with higher dissimilarities.

Current machine learning approaches in learning-to-rank tasks can be divided into three classes: pointwise approach [7],[8]; pairwise approach [9],[10]; listwise approach [11],[12]. Pointwise and pairwise approaches have the advantage that the existing theories and algorithms on regression and classification can be readily applied into the learning task. Moreover, pairwise approaches generally outperform pointwise approaches and have been successfully applied to various information retrieval applications [13],[14]. Therefore, we adopt the pairwise ranking-via-classification approach, to solve our problem.

We next describe RankingSVM method, which is the basis of our proposed methods.

A. Ranking Via Support Vector Machines

RankingSVM finds a ranking function $f(\cdot)$ that maximizes the expected Kendall τ statistic on training dataset $S = \{S_1, \cdots, S_N\}$. Kendall's τ statistic $\tau_{S_q}(o_q^*, \hat{o}_q) = \frac{\text{concordant \#−discordant \#}}{\text{concordant \#+discordant \#}}$ [15], where an object pair $s_i \neq s_j$ is called discordant if the orderings o_q^* and $\hat{o}_q$ do not agree in how they order s_i and s_j, and called concordant otherwise. In our study, $S_q = \{(x_i^q, r_i^q)\}_{i=0}^{m_q}$ contains the energy data of the 3D confirmations of qth protein. Ranking o_q^* denotes the true ordering of the protein conformations derived from structure dissimilarity, and $\hat{o}_q$ denotes the ordering determined by the ranking function.

For strict orderings on m instances, we have $\frac{m(m-1)}{2} =$ concordant # + discordant #. Maximizing the expected Kendall's τ statistic of a linear ranking function on a data set S is equivalent to maximizing the pairwise agreement (concordant #). This optimization problem can be formulated as a search for the weight vector w that maximizes the number of inequalities of form $\mathrm{Sign(r_i - r_j)w^T(x_i - x_j) \geq 1}$ that hold true. It can be approximately solved by learning the SVM classifier [16] on the transformed data set, $S^{diff} = \{z_{ij} = x_i - x_j, y_{ij} = \mathrm{Sign}(r_i - r_j)\}$, where z_{ij} is the pairwise difference vector, y_{ij} is the sign of the rank difference of objects s_i, s_j, and ξ_{ij} are the slack variables.

$$\begin{array}{ll} \min & \frac{1}{2} w^T w + \pi \sum_{ij} \xi_{ij} \\ \text{s.t.} & y_{ij} w^T z_{ij} \geq 1 - \xi_{ij}, \xi_{ij} > 0, \forall i, j \end{array} \quad (1)$$

III. Method: Non-Negativity Constrained Weight Learning

A. Non-Negativity Constraints

The energy terms used in our optimization represent "costs", in the sense that the natural physical tendency of the protein is to decrease each one of these values. Each energy term, taken separately, represents a uniquely defined physical tendency. For the case of electrostatic interactions, two positive charges move away from each other in order to lower their interaction energy. Reversing the sign of this interaction energy would turn the repulsive force to an attractive one, hence resulting in an unphysical interaction. If we sacrifice the physicality of the energy function by picking negative weights for some terms, it may be possible to obtain a better ranking on the collected set of conformations. Unfortunately, experience shows that such unphysical energy functions, while performing well on the chosen set of existing proteins, perform poorly when predicting new physical structures. This is partly because it is impossible to sample the whole set of possible conformations for a given protein, and the methods used to generate the conformations in the training set start from special, compact conformations that already satisfy various physicality properties. Dropping the positivity constraints could improve the ranking for these special conformations, but there will be very large, unsampled subsets of the set of possible conformations where the negative coefficients would result in incorrect foldings/orderings. Thus, one enforces a positivity constraint on the weights in order to avoid overfitting to the (small) set of sampled conformations.

We next describe two approaches to non-negative support vector machines.

B. Non-Negative L2-norm SVM

In this section, we propose a non-negative version of SVM by using an L_2 norm approach, and solve it through the exponential gradient (EG) algorithm [17].

Due to the characteristics of our problem, we formulate the optimization in primal form. Adding the non-negativity constraints to the standard SVM formulation gives the optimization problem,

$$\min_{w \geq 0} \frac{\nu}{2} w^T w + \frac{1}{l} 1_l^T (1_l - DAw)_+ \tag{2}$$

where $(u)_+ = \max(u, 0)$ sets the negative elements of the vector u to zero, A denotes the data matrix with rows given by the z_{ij}s, $D = \mathrm{diag}(\mathrm{y}_1, \cdots, \mathrm{y}_\mathrm{l})$ is the label matrix, $1_l = [1, 1, 1, \ldots, 1]^T$ is an l-dimensional vector of 1s, and l is the total number of data points (i.e. total number of pairwise difference vectors in our study).

The objective function in (2) is non-differentiable, hence typical optimization methods cannot be directly applied to this problem. To address this issue, we use the L_2-norm of the hinge loss variables in the objective function. This type of SVM has gained popularity in large scale classification because the resulting objective function $J(w)$ is a piecewise quadratic and strongly convex function, and efficient algorithms such as coordinate descent can be applied. The Non-Negative L_2-norm SVM (NNL2SVM) objective function is,

$$J(w) = \min_{w \geq 0} \frac{\nu}{2} w^T w + \frac{1}{2l} \mid\mid (1_l - DAw)_+ \mid\mid^2 \tag{3}$$

We use the exponential gradient (EG) algorithm [17] to solve this NNL2SVM problem because its optimization is naturally constrained to the non-negative space R_+^n. The algorithm is summarized in Table I.

Table I
EG Algorithm for NNL2SVM Problem

```
Initialize $w^0 = \frac{1}{n} 1_n$ so that $\mid\mid w^0 \mid\mid_1 = 1$
For $t = 0, 1, 2, \ldots$
    Compute $\bigtriangledown J(w^t) = \nu w^t - \frac{1}{l} A^T D (1_l - DAw)_+$
    For all $j = 1, \cdots, n$
        Update $w_j^{t+1} = w_j^t e^{-\eta \bigtriangledown w_j^t}$
    Normalize $w^{t+1}$
```

The standard normalization sets $\mid\mid w^{t+1} \mid\mid_1 = 1$. We also investigate another normalization rule that enforces $\mid\mid w^{t+1} \mid\mid_1 \leq \mid\mid w^t \mid\mid_1$ by keeping w^{t+1} unchanged if $\mid\mid w^{t+1} \mid\mid_1$ is less than $\mid\mid w^t \mid\mid_1$, and setting its norm to $\mid\mid w^t \mid\mid_1$ otherwise. In our study, we set the learning rate $\eta = \frac{1}{R}$, where $R = \max_{ij}(\max_k z_{ij,k} - \min_k z_{ij,k})$ is the largest value over the sample set of the maximum difference between the components of a feature vector z_{ij}.

C. Non-Negative L1-norm SVM

Another approach to the NNSVM problem is to add non-negativity constraints to the L1-SVM formulation and extend the existing L1SVM algorithm [18] to solve the resulting NNL1SVM problem. The optimization problem is,

$$\begin{array}{lrcl} \min & \multicolumn{3}{l}{1_n^T w + \pi 1_l^T \xi} \\ \text{s.t.} & DAw & \geq & 1_l - \xi \\ & w, \xi & \geq & 0 \end{array} \tag{4}$$

We solve this problem using an approach described in [18]. Proposition 1 in [18] states that for any $\epsilon \in (0, \bar{\epsilon}]$ for some $\bar{\epsilon} > 0$, the optimal solution of the exterior penalty problem gives an exact solution to the original, primal problem. The corresponding exterior penalty problem can be derived by assigning quadratic penalty terms to the constraints of the dual problem. The exterior penalty problem of (4) minimize the following objective function,

$$\begin{array}{rcl} J(\mu) & = & -\epsilon 1_l^T \mu + \frac{1}{2}(\mid\mid (A^T D \mu - 1_n)_+ \mid\mid^2 \\ & & + \mid\mid (\mu - \pi 1_l)_+ \mid\mid^2 + \mid\mid (-\mu)_+ \mid\mid^2) \, . \end{array} \tag{5}$$

Problem (5) is an unconstrained optimization problem. We solve it using a generalized Newton method.

Table II
NEWTON METHOD FOR NNL1SVM PROBLEM

```
Initiate $t = 0$ and $\mu^1 = 1_l$
Repeat
    $t = t + 1$
    $\mu^{t+1} = \mu^t - \zeta_t(\delta I_l + \partial^2 J(\mu^t))^{-1} \bigtriangledown J(\mu^t)$
        $\zeta_t$ is the largest number in $\{1, \frac{1}{2}, \frac{1}{4}, \cdots, \}$
            such that $J(\mu^t) - J(\mu^t + \zeta^t d^t) \geq -\frac{\zeta^t}{4} \bigtriangledown J(\mu^t) d^t$
              where $d^t = -(\delta I_l + \partial^2 J(\mu^t))^{-1} \bigtriangledown J(\mu^t)$
Until $t \geq$ max_iter or $|| \mu^t - \mu^{t+1} ||_2 \leq$ tol
$w = \frac{1}{\epsilon}(A^T D\mu - 1_n)_+$
```

Following the definition of generalized Hessian in [18], the gradient and hessian for (5) are given as,

$$\begin{aligned} \bigtriangledown J(\mu) &= -\epsilon 1_l + DA(A^T D\mu - 1_n)_+ \\ &\quad +(\mu - \pi 1_l)_+ - (-\mu)_+ \\ \partial^2 J(\mu) &= DA\mathrm{diag}\{(\mathrm{A^T D}\mu - 1_\mathrm{n})_*\}\mathrm{A^T D} \\ &\quad +\mathrm{diag}\{(\mu - \pi 1_\mathrm{l})_* + (-\mu)_*\} \end{aligned}$$

where $u_* = \mathrm{Sign}(\mathrm{u}_+)$, with Sign being applied element-wise on the vector. Notice that at each Newton iteration, we need to invert the matrix $Q = \delta I_l + \partial^2 J(\mu)$. This is computationally expensive when the total number of data points l is large ($l > 1000$). We address this issue by using the Sherman-Morrison-Woodbury formula [19]. We decompose the hessian matrix as $Q = F + H * H^T$, where diagonal matrix $F = \mathrm{diag}(\rho)$ with $\rho = \delta 1_l + (\mu - \pi 1_l)_* + (-\mu)_*$ and $\rho > 0$, and matrix $H = DAE$ with $E = (\mathrm{diag}(\mathrm{A^T D}\mu - 1_\mathrm{n})_*)^{\frac{1}{2}}$. The inversion can be computed as $Q^{-1} = F^{-1} - F^{-1}H(I_l + H^T F^{-1} H)^{-1} H^T F^{-1}$ and the time complexity is reduced from $O(l^3)$ to $O(ln^2) + O(n^3)$.

IV. RESULTS AND DISCUSSION

A. Data Set Description

The dataset used in this study consists of the values of various energy terms for a non-redundant set of 171 proteins that fall into the *ab initio* folding class. This set is representative of the "hard target" protein sequences in the Protein Data Bank with up to 200 residues, meaning that current homology search tools fail to identify proteins with an evolutionary relationship with proteins in this class.

For each protein, a large set of non-native random conformations (over $50,000$ per protein) are generated in the manner described in [4]. The energy terms for the native structure and each one of the generated conformations are collected. The energy terms are obtained from the CABS (C_α-C_β-Side chain) force field [4], which is used in the protein structure prediction tool TASSER [20]. We include 20 different energy terms from this force field, briefly summarized in Table III. The structural similarity of conformations is measured by the 1-(TM-score) [21], which is intended as a more accurate similarity measure than the commonly used RMSD [5].

Table III
ENERGY TERMS USED IN TASSER

$E_{*,1}$	pairwise interaction of C_α-SC (side chain)
$E_{*,2}$	pairwise interaction for non-parallel C_α-C_α
$E_{*,3}$	excluded volume of SC-SC
$E_{*,4}$	pairwise interaction of SC-SC
$E_{*,5}$	quarsi3 for SC-SC
$E_{*,6}$	enhance good piece
$E_{*,7}$	-1/r for parallel contact of C_α-C_α
$E_{*,8}$	hydrogen bond interactions on the alpha helix
$E_{*,9}$	hydrogen bond interactions on the beta sheet
$E_{*,10}$	bury potential for SG (side group)
$E_{*,11}$	bias2,3 : $v(i) - v(i+4)$ anti/parallel $c(i) - c(i+2)$ anit/paralel
$E_{*,12}$	crumpling
$E_{*,13}$	bias4 to predicted secondary structure
$E_{*,14}$	bias1 to possible secondary structure
$E_{*,15}$	correlation of E13 of C_α
$E_{*,16}$	correlation of E14
$E_{*,17}$	correlation of E15
$E_{*,18}$	environment potential
$E_{*,19}$	deviation from predicted contact order
$E_{*,20}$	deviation from predicted contact number

B. Previous Approach

In an earlier optimization study [4], the authors proposed to use an objective function related to the correlation $\mathrm{corr}(r(q), E(q))$ between the structural dissimilarity and the total energy of the generated conformations. Namely, they used the product of two quantities $G1$ and $G3$, given by,

$$\begin{aligned} G1 &= \frac{1}{1+\frac{1}{N}\sum_{q=1}^{N} \mathrm{corr(r(q),E(q))}} \\ G3 &= \frac{1}{1+\frac{1}{N}\sum_{q=1}^{N} Z_n(E(q))} \end{aligned}$$

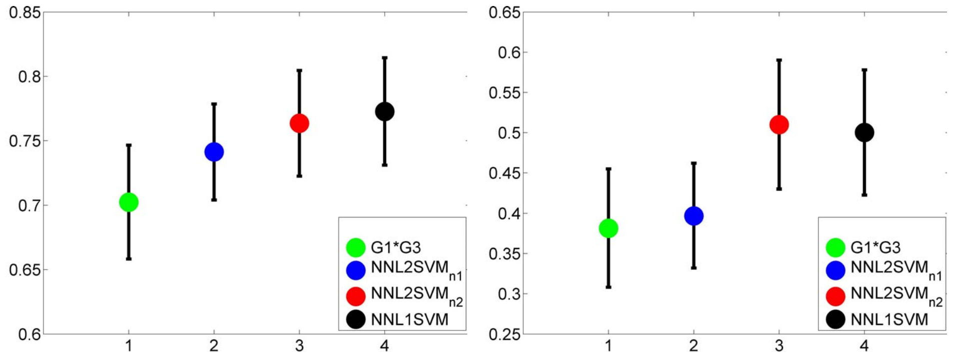
where Z-score of the mean of the total energy $Z_n(E(q)) = (\bar{E}(q) - E_0(q))/(\sqrt{\bar{E}^2(q) - (\bar{E}(q))^2})$.

Using the CERN MINUIT package [22] to optimize the weights, they achieved significant results in CASP [20]. Their study employed proteins from all homology modeling, threading, and *ab initio* prediction classes.

C. Experiment Design

The number of all pairwise difference vectors $z_{ij} = x_i - x_j$ is quadratic in the number of data points (conformations). In addition to this computational issue, it is not realistic to expect the energy function to rank all conformations according to their dissimilarity from the native structure. Therefore, in our experiments, we use the following sampling scheme to generate the training data set.

- For the first class, $C_1 = \{z_{i0} = x_i - x_0 \mid y_{i0} = \mathrm{sign(r_i - r_0)} = 1\}$, we uniformly sample 100 non-native conformations according to their structural dissimilarity and include their comparisons with the native structure.



(a) Pairwise Agreement

(b) Correlation

Figure 2. Error Plot of the Performance of the Learned Energy Functions

- For the second class, $C_2 = \{z_{jk} = x_j - x_k \mid y_{jk} = \mathrm{sign(r_j - r_k)} = -1\}$, we generate pairs of comparisons between non-native conformations. If two conformations have close values of dissimilarity from the native structure, it may not be reasonable to require the energy function to rank them according to the dissimilarity. We thus restrict the second class to pairs whose dissimilarities from the native structure are sufficiently different. In particular, we first partition the set of non-native conformations into 6 subsets, $S_{(0,0.1)}$, $S_{[0.1,0.2)}$, $\cdots$, $S_{[0.4,0.5)}$, $S_{[0.5,0.6]}$, where $S_{(0,0.1)}$ contains conformations with dissimilarity from the native structure in the range $(0, 0.1)$, etc. We then uniformly sample 25 conformations $\{s_i^{(j)}\}_{i=1}^{25}$ according to dissimilarity from each subset $S_{[a_j,b_j)}$. The comparisons we include are then, $(s_1^{(1)} - s_1^{(3)})$, $(s_2^{(1)} - s_2^{(3)})$, $\ldots$, $(s_{25}^{(1)} - s_{25}^{(3)})$, $(s_1^{(2)} - s_1^{(4)})$, $\ldots$, $(s_{25}^{(2)} - s_{25}^{(4)})$, $\ldots$, $(s_{25}^{(4)} - s_{25}^{(6)})$.

By the sampling method described above, we generate 100 data points in each class, for each protein. This gives a total of $34,200$ data points of dimension 20.

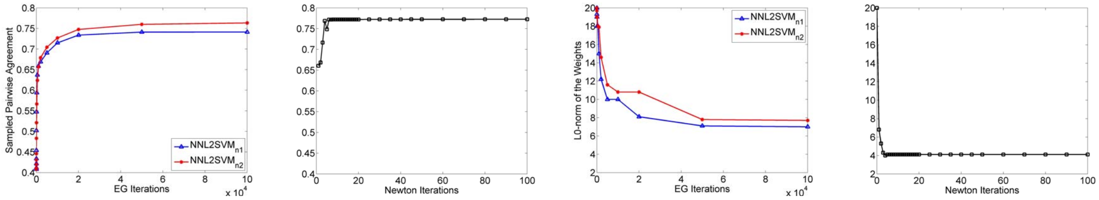
D. Results Analysis

We evaluate the capability of our Non-Negative RankingSVM (NN-RankingSVM) approach to learning protein energy functions through 10-fold cross validation. We randomly partition the 171 proteins into 10 folds. For each fold i, we learn an energy function from the energy data of the other 9 folds using each method, and evaluate the learned energy functions on the data of fold i. We employ the grid search procedure during cross-validation for parameter tuning of the NNSVM methods. We denote ranking via NNL2SVM with the normalization rule $\| \ w^t \ \|_1 = 1$ as $\mathrm{NNL2SVM}_{n1}$, ranking via NNL2SVM with the normalization rule $\| \ w^{t+1} \ \|_1 \leq \| \ w^t \ \|_1$ as $\mathrm{NNL2SVM}_{n2}$, and ranking via the NNL1SVM approach as $\mathrm{NNL1SVM}$. The baseline method (see Section 4.C) is denoted as $\mathrm{TASSER}^{\mathrm{MINUIT}}$ (with randomly generated initial weights).

1) NN-RankingSVM versus TASSER$^{\mathrm{MINUIT}}$*:* We first compare the protein folding energy function learned by our proposed NN-RankingSVM approach with those learned by the baseline method. We use two criteria to evaluate the fitness of the learned energy functions: Kendall τ rank statistics (approximated by sampled pairwise agreement) and Pearson's correlation. Fig. 2(a) lists the sampled pairwise agreement, measured by the average testing accuracy on the labeled pairwise difference data. Fig. 2(b) lists the average correlation coefficients between the (1-TMscore) value and the energy, which are computed using the learned energy function during each cross-validation.

Comparing to the energy functions learned by baseline method, energy functions learned using the NN-RankingSVM approach generally achieve better performance in both sampled pairwise agreement and correlation. On average, energy functions learned using the NNL2SVM methods can obtain around 9% increase in the sampled pairwise agreement and around 34% increase on the correlation values. While those output by NNL1SVM method have around 10% and 28% improvement on those values, respectively. In addition, the average computation time of a 10-fold cross validation for NNL1SVM is about 7 seconds, which is much more efficient comparing to the baseline method (around 2 minutes) and the NNL2SVM methods (around 20 minutes). In summary, the experiments show that NNL1SVM method outperform the other methods in terms of both learning performance and computational efficiency.

2) NNL2SVM versus NNL1SVM: We then analyze the trend of the sampled pairwise agreement (measured by testing accuracy) and the sparsity of the weight vector during the algorithm optimization of the proposed NNSVMs. As shown in Fig. 3, NNL2SVM methods generally converges to the optimal solutions after about 5000 EG iterations. $\mathrm{NNL2SVM}_{n_1}$ method obtains sparser solutions than those of $\mathrm{NNL2SVM}_{n_2}$ method on average, but at the cost of classification accuracy. Overall, NNL1SVM method demonstrates robust performance in classification accuracy and sparsity



(a) NNL2SVM Accuracy (b) NNL1SVM Accuracy (c) NNL2SVM Sparsity (d) NNL1SVM Sparsity

Figure 3. NNSVM Optimization Method Comparison

while enjoying fast convergence.

V. Conclusion

A critical open problem in *ab initio* protein folding is protein energy function design. In this paper, we addressed this problem as a weight optimization problem, and demonstrated a machine learning approach using the ranking-via-classification paradigm. Comparing with state-of-the-art approach that based on maximizing the correlation between the total energy and the structural dissimilarity, our learning-to-rank approach was able to learn energy functions that maintain the correct ordering of the conformations more often, and give higher correlations with the structural dissimilarity from the native structure. We believe that this new approach for learning protein energy functions presents a new avenue of exploration with potential. We will investigate generalization of our methods to the problem of learning nonlinear energy functions. We also expect the capability to learn SVMs with non-negative weights to have diverse applications beyond protein structure prediction.

References

[1] D. Baker and A. Sali, "Protein structure prediction and structural genomics," *Science*, vol. 294(5540), p. 93, 2001.

[2] J. Skolnick, D. Kihara, and Y. Zhang, "Development and testing of the PROSPECTOR 3.0 threading algorithm," *Proteins*, vol. 3, pp. 502–518, 2004.

[3] C. Anfinsen *et al.*, "Principles that govern the folding of protein chains," *Science*, vol. 181(96), pp. 223–230, 1973.

[4] Y. Zhang, A. Kolinski, and J. Skolnick, "Touchstone II: A new approach to ab initio protein structure prediction," *Biophysical Journal*, vol. 85, pp. 1145–1164, 2003.

[5] W. Kabsch, "A solution for the best rotation to relate two sets of vectors," *Acta Crystallographica Section A*, vol. 32(5), pp. 922–923, 1976.

[6] B. Kuhlman and D. Baker, "Native protein sequences are close to optimal for their structures," *Proceedings of the National Academy of Sciences*, vol. 97(19), p. 10383, 2000.

[7] R. Caruana, S. Baluja, and T. Mitchell, "Using the future to sort out the present: Rankprop and multitask learning for medical risk analysis," in *NIPS'95*, pp. 959–965.

[8] K. Crammer and Y. Singer, "Pranking with ranking," in *NIPS'02*, pp. 641–648.

[9] R. Herbrich, K. Obermayer, and T. Graepel, "Large margin rank boundaries for ordinal regression," in *Advances in Large Margin Classifiers*. MIT Press, 2000, pp. 115–132.

[10] Y. Freund, R. D. Iyer, R. E. Schapire, and Y. Singer, "An efficient boosting algorithm for combining preferences," *Journal of Machine Learning Research*, vol. 4, pp. 933–969, 2003.

[11] Z. Cao, T. Qin, T. Liu, M. Tsai, and H. Li, "Learning to rank: from pairwise approach to listwise approach," in *Proceedings of the 24th ICML conference*, 2007, pp. 129–136.

[12] Y. Yue, T. Finley, F. Radlinski, and T. Joachims, "A support vector method for optimizing average precision," in *Proceedings of the 30th ACM SIGIR conference*, 2007, pp. 271–278.

[13] R. Iyer, D. Lewis, R. Schapire, Y. Singer, and A. Singhal, "Boosting for document routing," in *Proceedings of the 19th CIKM conference*, 2000, pp. 70–77.

[14] T. Joachims, "Optimizing search engines using clickthrough data," in *Proceedings of the 8th ACM SIGKDD conference*, 2002, pp. 132–142.

[15] M. Kendall, "A new measure of rank correlation," *Biometrika*, vol. 30(1-2), pp. 81–93, 1938.

[16] V. Vapnik, *The Nature of Statistical Learning Theory*. Springer, 1995.

[17] J. Kivinen and M. Warmuth, "Exponentiated gradient versus gradient descent for linear predictors," *Information and Computation*, vol. 132(1), pp. 1–63, 1997.

[18] O. Mangasarian, "Exact 1-norm support vector machines via unconstrained convex differentiable minimization," *Journal of Machine Learning Research*, vol. 7(2), pp. 1517–1530, 2006.

[19] G. Golub and C. Van Loan, *Matrix computations*. Johns Hopkins University Press, 1996.

[20] Y. Zhang and J. Skolnick, "TASSER: An automated method for the prediction of protein tertiary structures in CASP6," *Proteins*, vol. 61(S7), pp. 91–98, 2005.

[21] Y. Zhang and J. Skolnick, "Scoring function for automated assessment of protein structure template quality," *Proteins*, vol. 57(4), pp. 702–710, 2004.

[22] MINUIT, "http://seal.web.cern.ch/seal/snapshot/work-pack ages /mathlibs/minuit/."