

# Supervised Scoring of Protein Models using Kernels on Statistical Potentials

Alessio Ceroni<sup>1</sup>, Paolo Frasconi<sup>1</sup>, and Lawrence Kelley<sup>2</sup>

<sup>1</sup> Machine Learning and Neural Networks Group

Dipartimento di Sistemi e Informatica

Università degli Studi di Firenze, Italy

Web: <http://www.dsi.unifi.it/neural/>

<sup>2</sup> Structural Bioinformatics Group

Department of Biological Sciences

Imperial College, London SW7 2AY, UK

Web: <http://www.sbg.bio.ic.ac.uk/>

**Abstract.** We develop and test new algorithms for ranking alternative 3D protein conformations. The method enriches traditional approaches based on pairwise statistical potentials with supervised preference learning. This is obtained by (1) interpreting traditional pair potential modeling as maximum likelihood unsupervised learning, (2) by deriving kernels based on the parameters of the likelihood function that are used to measure the similarity between two conformations, and (3) by applying SVM ordinal regression to learn a ranking function over alternative conformations. Empirical tests on a realistic set of PDB proteins show the effectiveness of the proposed algorithm and significant improvements over the plain pair potential approach.

## 1 Introduction

Automatic prediction of protein structure is a challenging task when sequence or fold similarity with respect to proteins of known structure is low. Improvements towards solving the problem of “de-novo” prediction and remote homology detection have important implications and may help to narrow the gap between the exponentially increasing number of known sequences and the slowly increasing number of known structures.

State-of-the-art algorithms for the prediction of protein structures often produce multiple candidate three-dimensional models. In the case of fold recognition, tens or hundreds of candidates may be created by weak matches to known structures present in a fold library. Similarly, the most successful “de-novo” methods for structure prediction regularly produce tens of thousands of potential candidates. These methods use fragment-insertion approaches [1] where the protein conformational space is explored by assembling fragments extracted from available structures. Often these ensembles contain very good structures. The primary difficulty is to discriminate between correctly folded models (near-native structures) and incorrectly folded models (decoys) in such large ensembles. Theoretically, the estimation of the free energy associated with a protein structure is a natural solution for ranking alternative candidate structures. The free energy is a function of the spatial interactions of protein’s atoms between themselves and with solvent atoms. The native structure  $N(P)$  of a protein  $P$  is thought to correspond to the lower energy state accessible in equilibrium. In general, however, the real energy function cannot be evaluated and one needs to resort to an approximating scoring function  $U(X|M)$  defined according to an underlying statistical model  $M$ . In so doing, the predicted native structure can be retrieved as

$$\hat{N}(P) = \arg \min_{X \in \mathcal{X}(P)} U(X|M) \quad (1)$$

being  $\mathcal{X}(P)$  the set of alternative conformations of  $P$ .

Over the past 20 years, a wide variety of scoring functions have been published in the literature. By far the most popular and largely successful approach is based on collecting statistics about pairwise distances between residues [2, 1, 3, 4]. Interestingly (see details in Section 2), the energy model based on statistical potentials can be naturally interpreted as a generative probabilistic model trained on the set of available native structures. Generative models are usually trained by unsupervised learning algorithms and do not make use of “negative” examples. In the present context this means that decoys generated by structure prediction algorithms are not taken into account. In this paper we develop a novel alternative technique for deriving a suitable scoring function from data that is based on a *discriminant* learning process. Being informed about the bias due to the specific prediction algorithm in exploring the conformation space, supervised (discriminant) learning can be expected to produce more accurate solutions. The algorithm proposed in this paper is based on support vector machines (SVM) trained on a preference learning task where decoys should receive

lower preference scores when compared to good or native structures. To measure the similarity between structures we introduce new kernel functions derived from pair potentials. The general idea is that pair potentials can be seen as generative probabilistic models trained by maximum likelihood on the native structures. Several kernels have been proposed in the literature to exploit probability distributions fitted on the data in an unsupervised fashion. Fisher kernels were introduced by [5] in the context of remote homology detection, where the generative component consisted of a hidden Markov models trained on protein families. Probability product kernels [6] are based on the idea that each datum is mapped to a probability distribution and a measure of similarity between distributions (e.g. Bhattacharyya) is used to construct the kernel. Histogram intersection kernels [7] have been introduced in the context of image classification and builds on similar ideas replacing products by the minimum operator when comparing distributions. The kernels we propose here are based on the idea that a probability distribution can be naturally associated to the energy model of a protein structure according to Boltzmann law. In this way, kernels on probability distributions (such as those mentioned above) will compare two conformations by measuring the contributions of interacting pairs to energy potentials.

The remainder of this paper is organized as follows. In Section 2 we briefly review the pair potential approach and discuss our interpretation of it as a generative model for unsupervised learning. In Section 3 we present an ordinal regression solution to the problem of scoring protein structural models. In Section 4 we describe the kernels used to build the scoring function. In Section 5 we report an empirical evaluation of our method on a data set of 266 proteins. Finally some conclusions are drawn in Section 6.

## 2 Learning from natives with statistical potentials

We briefly review knowledge-based statistical potentials. Here we present the method in terms of maximum likelihood unsupervised learning, a formulation that is suitable for subsequent derivation of the kernels to be used in conjunction with supervised learning algorithms. The statistical model proposed in [2] is defined by a set of probability densities in which the range of distance values is split into  $S$  intervals  $[h_{s-1}, h_s), s = 1, \dots, S$ , with  $h_0 = 0$  and  $h_S = \infty$ . Specifically, the pair distribution  $\Pr(s|a, b, k)$  is the conditional probability that a pair of residues has distance in  $[h_{s-1}, h_s)$ , given that the two amino acids have type  $a$  and  $b$ , and they are separated by  $k$  positions along the sequence. The relationship between energy potentials  $E_k^{ab}(s)$  and pair distributions is expressed by Boltzmann law:

$$\Pr(s|a, b, k) = \frac{e^{-\frac{E_k^{ab}(s)}{kT}}}{Z_k^{ab}(s)} \quad (2)$$

with normalization factor  $Z_k^{ab}(s) = \sum_s e^{-E_k^{ab}(s)/kT}$ . By assuming  $Z_k^{ab}(s) \approx Z_k(s)$  for all  $a, b$ , the net potentials  $\Delta E_k^{ab}(s) = E_k^{ab}(s) - E_k(s)$  can be approximated

as

$$\Delta E_k^{ab}(s) \approx -kT \ln \left( \frac{\Pr(s|a, b, k)}{\Pr(s|k)} \right) \quad (3)$$

Given an amino acid sequence  $A = a_1, \dots, a_L$ , a conformation  $X$  consists of a sequence of  $C_\alpha$  atoms' coordinate vectors, denoted  $x_1, \dots, x_L$ . Let  $d_{ij} = \|x_i - x_j\|$  the distance between residues  $i$  and  $j$ . According to pair potentials model  $M_{pp}$  the score of  $X$  is defined as the total energy:

$$U(X|M_{pp}) = \Delta E(X) = \sum_i \sum_{j>i} \Delta E_{|i-j|}^{a_i a_j}(d_{ij}) \quad (4)$$

where, as a notational convenience, we write  $\Delta E_k^{ab}(d)$  instead of  $\Delta E_k^{ab}(s)$  if  $d \in [h_{s-1}, h_s)$ .

In order to estimate the probabilities  $\Pr(s|a, b, k)$  from the available protein native structures, we begin by noting that for a given type  $(a, b, k)$  the distances are associated with a discrete set of intervals and therefore follow a multinomial distribution parameterized as follows:

$$\Pr(d|a, b, k) = \prod_s (\theta_{ks}^{ab})^{z(d,s)} \quad (5)$$

where  $z(d, s) = 1$  iff  $d \in [h_{s-1}, h_s)$  is an indicator variable mapping the distances to their intervals, and the model parameters simply coincide with the pairwise densities, i.e.  $\theta_{ks}^{ab} = \Pr(s|a, b, k)$ . The contribution of a structure  $X$  to the log-likelihood can be calculated by combining eqs. (4) and (3):

$$\ell(X; \theta) = \sum_i \sum_{j>i} \sum_s z(d_{ij}, s) \ln \left( \theta_{|i-j|, s}^{a_i b_j} \right) \quad (6)$$

Assuming a data set  $\mathcal{D}$  of independent structures, the overall log likelihood function is simply  $\ell(\mathcal{D}; \theta) = \sum_X \ell(X; \theta)$ .

Sufficient statistics for the parameters  $\theta_{ks}^{ab}$  are collected from a training set of known structures. Specifically, they consists of the counts  $m_{ks}^{ab}$  of the number of occurrences of pairs of type  $(a, b, k)$  at distance interval  $s$  in the training set:

$$m_{ks}^{ab} = \sum_X m_{ks}^{ab}(X) = \sum_X \sum_i \sum_{j>i} \sum_s z(d_{ij}, s) \quad (7)$$

As observed in [2], in case of small data sets the normalized sufficient statistics may yield poor parameter's estimates because of data sparseness. One possible remedy in this case consists of estimating each density as a convex sum of a simplified model and the full model. In the simplified model amino acid types are neglected and parameters reduce to  $\theta_{ks}$ , which can be reliably estimated from sufficient statistics  $m_{ks}$ . The smoothed parameters are then computed as

$$\hat{\theta}_{ks}^{ab} = \frac{1}{1 + m\sigma} \hat{\theta}_{ks} + \frac{m\sigma}{1 + m\sigma} \frac{m_{ks}^{ab}}{\sum_s m_{ks}^{ab}} \quad (8)$$

where  $\sigma$  is a fixed smoothing factor (we use the value  $\frac{1}{50}$  as in [2]) and  $m$  is the total number of observed pairs. When  $m$  is small the mixing factor favors the simplified model which can be reliably estimated from a smaller number of observations.

### 3 Learning from decoys with ordinal regression

Due to the impossibility of a sufficient sampling of the conformations space, it is not feasible to learn a scoring function that is guaranteed to take its minimum value on the native structure. However, a suitable scoring function can be specifically built to discriminate between alternative conformations generated by a particular threading or “de-novo” structure prediction algorithm. In this case, the space of decoys is reduced by the bias of the structure prediction algorithm, a bias that can be exploited by the learning process.

This task could be defined as a problem of learning to discriminate between “good” and “bad” structural models, where the quality of a model is decided by a measure provided by the user (e.g. a structural measure of similarity with the native conformation). We may call this the binary classification model  $M_{bc}$ . Let us introduce the label  $y_j^i = 1$  if the  $j$ -th conformation  $X_j^i$  is a good model for the  $i$ -th chain in the training set and  $y_j^i = -1$  otherwise. A kernel machine would compute a real function of the conformations as a weighted average of the training set labels as follows:

$$U(X|M_{bc}) = f(X) = \sum_{i,j} \alpha_j^i K(X_j^i, X) y_j^i \quad (9)$$

where the weight of each example consists of a coefficient  $\alpha_j^i$  measuring the importance of the example as calculated by some optimization procedure (e.g. SVM) and a kernel function  $K(X_j^i, X)$  measuring the similarity between  $X_j^i$  and  $X$ .

Such a discriminative approach could find it difficult to estimate the decision function, because the resulting dataset would be highly unbalanced and contain only a small fraction of good models. Moreover, a continuous measure of quality should be unnaturally transformed in a discrete number of classes of structural models. Because of these considerations, we propose that the search of a suitable scoring function should be formulated as an ordinal regression task. In this formulation the training set consists of ordered pairs of alternative conformations for proteins of known structure. A preference relation  $\prec$  is used to distinguish between two conformations. The preference model is then trained to learn either a partial or a total ordering between alternative conformations of a protein.

Herbrich [8] proposed a large margin classifier to learn preference relations in the above setting. When learning to score protein models the examples are tuples  $\{(y_{jk}^i, X_j^i, X_k^i)\}$ , where  $X_j^i$  and  $X_k^i$  are two alternative conformations of protein  $i$ , and  $y_{jk}^i$  is

$$y_{jk}^i = \begin{cases} +1 & \text{if } X_j^i \prec X_k^i \\ -1 & \text{otherwise} \end{cases} \quad (10)$$

The labels  $y_{jk}^i$  can be determined for each training pair according to an evaluation measure that takes into account the known native structure. According to the preference model  $M_{pref}$ ,  $X \prec X' \Leftrightarrow f(X) < f(X')$  where

$$U(X|M_{pref}) = f(X) = \sum_{ijk} \alpha_{jk}^i (K(X_j^i, X) - K(X_k^i, X)) y_{jk}^i \quad (11)$$

Since  $f(X)$  can be written as the inner product  $\langle \mathbf{w}, \phi(X) \rangle$ , being  $\phi(X)$  the feature mapping induced by the kernel, and since the kernel is a inner product in feature space and therefore a bilinear operator, the preference model can be seen as a binary classifier on *pairs* of conformations with

$$f_p(X, X') = \sum \alpha_{jk}^i K_p((X_j^i, X_k^i), (X, X')) y_{jk}^i \quad (12)$$

being

$$K_p((X_j^i, X_k^i), (X, X')) = K(X_j^i, X) - K(X_j^i, X') - K(X_k^i, X) + K(X_k^i, X'). \quad (13)$$

In this way one can still apply the standard binary classification SVM algorithm to a data set of conformation pairs in order to obtain the coefficients  $\alpha_{jk}^i$ .

## 4 Computing similarity between structures

In order to train the preference model, a similarity measure between two conformations must be provided in the form of a kernel function  $K$  between conformations. In principle, a hyper-graph  $G(X)$  could be associated to the conformation  $X$ , whose nodes are residues (or atoms) and whose hyper-edges are  $r$ -wise interactions. Every hyper-edge has an associated measure of interaction strength (e.g. the distance in a pairwise interaction). In such an approach, the computation of the kernel between two conformations would be performed by computing the similarity between the corresponding hyper-graphs, thus effectively exploiting higher order relations among atoms as features. Many statistical potentials have terms for bond angles (involving three atoms) and dihedral angles (involving four atoms). In principle, statistics could be made for the properties of small polyhedra of any given size extracted from the protein structure. Moreover, any sort of graph similarity measure could be used to calculate the kernel between two conformation hyper-graphs. Unfortunately, exploitation of  $r$ -order relations with  $r$  greater than 2 poses computational and sparseness problems because of the exponential growth of the number of features. In the following we introduce two kernel functions that are inspired from pairwise interactions, with the aim of defining preference models that can be compared directly to the approach based on knowledge-based potentials.

### 4.1 Product kernel

The general form of a probability product kernel [6] is obtained by fitting a distribution  $p(x|X)$  on a each datum  $X$  and comparing two data points as

$$K(X, X') = \int p(x|X)^\rho p'(x|X')^\rho dx \quad (14)$$

being  $\rho$  a parameter. Inspired by this definition, the first kernel we introduce is based on a representation consisting of distance counts, where a conformation  $X$  is represented by the sufficient statistics  $m_{ks}^{ab}(X)$  defined in Section 2 for knowledge-based potentials. The distance counts can be seen as the (unnormalized) discrete distributions associated with  $X$  and the kernel can be written as

$$K(X, X') = \sum_{a,b,k,s} m_{ks}^{ab}(X)^\rho \cdot m_{ks}^{ab}(X')^\rho \quad (15)$$

We use absolute counts instead of normalized frequencies because the resulting features retain informations also on the dimensions of the three-dimensional model. In all subsequent work we use the simple form given by  $\rho = 1$  that can also be interpreted as a inner product between fixed-size vector representations of conformations. In this case it is immediate to see that the feature space consists of  $S$  distance bins for each triplet  $(a, b, k)$ .

## 4.2 Intersection kernel

The second form is inspired by previous work on measuring the similarity between images [9]. A histogram intersection kernel (HIK) is similar to a probability product kernel in the case of discrete distributions, but uses the minimum operator instead of product. In our case this idea yields the kernel

$$K(X, X') = \sum_{a,b,k,s} \min [m_{ks}^{ab}(X), m_{ks}^{ab}(X')] \quad (16)$$

HIKs have been proved more successful in computer vision tasks than common polynomial or Gaussian kernels. In this context, HIKs have an interesting interpretation in term of similarity between the interaction graph associated with conformations. In particular, if each edge is labeled with the tuple  $(a, b, k, s)$ , the HIK between two conformations measures similarity by counting how many edges with the same label are in common between the graphs derived from the two conformations.

Both kernels in eq. (15) and eq. (16) can be subsequently enriched by composition with standard polynomial or Gaussian kernels.

## 4.3 Fisher kernel

Fisher kernels were first introduced in [5] in the context of remote homology detection and can be applied to the present problem in order to combine the advantages of discriminant learning and knowledge-based potentials. The Fisher vector  $\phi(X)$  associated with a conformation  $X$  is defined as the gradient of the log-likelihood of  $X$  with respect to the model's parameters  $\theta$ :

$$\phi(X) = \nabla_{\theta} \ell(X; \theta) \quad (17)$$

From equations (5) and (6) follows:

$$\frac{\partial \ell(X; \theta)}{\partial \theta_{ks}^{ab}} = \sum_i \sum_{j>i} \sum_s z(d_{ij}, s) \frac{\partial \ln \left( \theta_{|i-j|,s}^{a_i b_j} \right)}{\partial \theta_{ks}^{ab}} = \frac{m_{ks}^{ab}(X)}{\theta_{ks}^{ab}} \quad (18)$$

The Fisher kernel  $K(X, x')$  between two conformations  $X$  and  $X'$  is then defined as the dot-product of the corresponding Fisher vectors:

$$K(X, X') = \langle \phi(X), \phi(X') \rangle = \sum_{a,b,k,s} \frac{m_{ks}^{ab}(X) m_{ks}^{ab}(X')}{(\theta_{ks}^{ab})^2} \quad (19)$$

Compared to the kernels of eq. (15) and eq. (16) we can immediately recognize that the Fisher kernel takes into account information from the entire data set used to compute the potentials when comparing two conformations. This information is used to scale each feature seen by those kernels by a value that indicates the significance of that particular contact.

## 5 Experimental results

A representative non-redundant subset of the Protein Data Bank was taken from PISCES [10], with resolution better than 1.8Å and R-factor less than 0.3. Standard pair potential approaches have been shown to be affected by protein length [11] such that improved performance is achieved when the size range of the proteins used for deriving a potential is the same as the size range of the proteins being assessed. We have restricted this initial analysis to a particular size range: small protein chains less than 100 amino acids in length. Filtering the PISCES-derived set based on this criteria resulted in 266 protein chains. Because of the high computational cost of producing large decoy sets for each of the 266 protein chains using a “de-novo” protocol, we have chosen instead to generate decoys using a threading algorithm. Each of the sequences was scanned against a fold library using a profile-profile fold recognition algorithm known as Phyre (manuscript in preparation <http://www.sbg.bio.ic.ac.uk/phyre>). For each sequence, 3D models were constructed based on the fold recognition alignments to the top 20 scoring matches in the fold library. Model quality was assessed using sequence-dependent structural alignment using the TM-score [12] of the model to the known experimental structure of the query sequence. Each protein chain was assigned a fold according to the SCOP hierarchy [13]. For each protein we label as *good* those models whose TM-score is greater than 0.2 and as *bad* models the rest.

Sufficient statistics  $m_{ks}^{ab}(X)$  were estimated for each structure  $X$  by extracting all the contacts with lengths between 0 and  $MaxDist$ . Sequence separation values between  $k = 0$  and  $k = 8$  were taken independently, while all the residues spaced more than 8 positions were assumed to be equivalent and conventionally assigned  $k = 9$ . The histograms of distances were constructed by dividing the interval  $[0, MaxDist]$  into  $S$  bins of equal size.

Using the 266 proteins we performed a set of experiments aiming to compare pair potentials (PP) to the alternative kernel based algorithms introduced in this paper, namely the product kernel between histograms of eq. (15), the histogram intersection kernel of eq. (16) and the Fisher kernel of eq. (19). For each method, performance was estimated using a five-fold cross validation procedure. The partition in five subsets was created by ensuring that chains in the same SCOP fold are all in the same subset. The scoring functions produced by the learning algorithms were then used to predict the ranking of unseen models. We took a particular care in eliminating any homology between train and test proteins to ensure that experimental results are indicative of the performances for “de-novo” prediction.

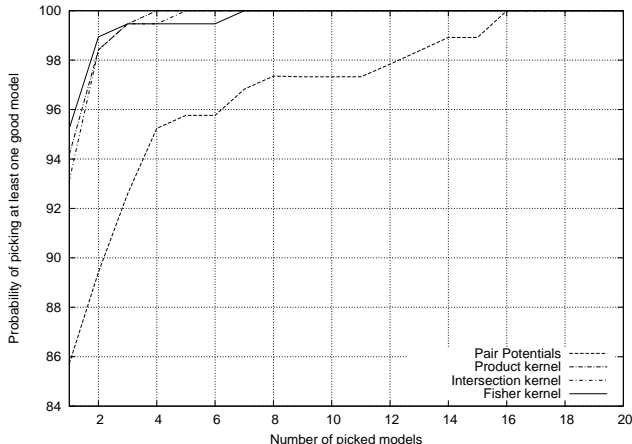
The results of the experiments are outlined in Table 1. Four different measures of performance are reported: *Accuracy* refers to the accuracy for the binary classification task associated with the comparison of two structures, i.e. the percentage of times the result of comparing two models is the same when using the TM-score and the scoring function generated by the learning algorithm; *RankCoeff* is the Kendall correlation coefficient computed between the rankings induced by the TM-score and by the estimated scoring function; *MinRank* is the average minimum rank of the good models; *BestPos* is the average rank of the model having highest TM-score. In Figure 1 we plot the percentage of times at least one good model was ranked within the first  $n$  positions. The results consistently indicate the superiority of discriminant training. When measuring performance using accuracy, all the kernels significantly outperform the pair potentials approach, with the null hypothesis of identical performance rejected at 99% confidence level in all cases. The comparison between the various kernel methods demonstrate a slight advantage of the Fisher on the Linear kernel, with a quite low confidence of 66%, while there is almost no difference between Fisher and Intersection kernel. Looking at the curves plotted in figure 1, it is worth noticing that all the discriminative approaches have a probability greater than 98% of getting a near native model within the best two choices, while PP would only have about 90% probability.

**Table 1.** Experimental results. Histograms were created by setting  $\text{MaxDist} = 20\text{\AA}$  and  $S = 20$  (bin size =  $1\text{\AA}$ ).

Method	<i>Accuracy</i>	<i>RankCoeff</i>	<i>MinRank</i>	<i>BestPos</i>
PP	74.7%	0.327	1.65	3.64
Product	84.0%	0.495	1.08	1.98
Intersection	85.2%	0.513	1.10	2.07
Fisher	85.1%	0.491	1.08	2.10

In a subsequent experiment we extended the data set used to train pair potentials in order to compare the accuracy of kernel based methods trained on small data sets to the accuracy of pair potentials trained on a large data set

**Fig. 1.** Percentage of times at least one good model was ranked within the first  $n$  positions.



of non redundant chains. We repeated the five fold cross validation procedure adding a second set of chains obtained as follows. A precompiled CulledPDB set of 512 native structures taken from PISCES, with chain’s length shorter than 100 residues, resolution better than  $2.5\text{\AA}$ , R-factor lower than 0.25 and homology lower than 90%. To enforce non-redundancy, for each of the five subsets in the cross validation procedure, the set of 512 native structures used to build the potentials was further reduced by taking out those proteins belonging to any SCOP fold that occurs in the test set. In this setting the pair potential method obtained  $Accuracy = 76.9\%$ ,  $RankCoeff = 0.351$ ,  $MinRank = 1.65$  and  $BestPos = 3.20$ . Kernels trained on small data sets outperformed pair potentials even when the latter were trained on larger data sets, maintaining a confidence level of 99% for this result.

## 6 Conclusions

The protein folding problem can loosely be divided into a) a conformational search problem, and b) the problem of deriving an appropriate discriminatory energy function. Conformational search is generally handled by using either a library of known structures (e.g. threading/inverse folding), or fragment insertion protocols. Energy function derivation has in the past been largely based on coarse-grained statistical potentials which are widely-used by the protein structure prediction community. However, such potentials invariably focus only on positive examples (known experimental structures), whilst ignoring the potentially rich source of information available from negative examples (computer-generated decoys). The principle of computational learning based on positive and negative examples is a well-established research domain. To-date, such tech-

niques have not, to our knowledge, been applied to the problem of discriminatory protein energy functions. The results presented here indicate that such information can be harnessed using kernel methods, and that this can lead to improved discriminatory power and thus more accurate predictions of protein structure.

## References

1. Simons, K., Ruczinski, I., Kooperberg, C., Fox, B., Bystroff, C., Baker, D.: Improved recognition of native-like protein structures using a combination of sequence-dependent and sequence-independent features of proteins. *Proteins* **34** (1999) 82–95
2. Sippl, M.J.: Calculation of conformational ensembles from potentials of mean force. *J. Mol. Biol.* **213** (1990) 859–883
3. Miyazawa, S., Jernigan, R.: An empirical energy potential with a reference state for protein fold and sequence recognition. *Proteins* **36** (1999) 357–369
4. Domingues, F., Koppensteiner, W., Jaritz, M., Prlic, A., et al., W.C.: Sustained performance of knowledge-based potentials in fold recognition. *Proteins* **37** (1999) 112–120
5. Jaakkola, T.S., Haussler, D.: Exploiting generative models in discriminative classifiers. In: *Proceedings of the 1998 conference on Advances in neural information processing systems II.* (1999) 487 – 493
6. Jebara, T., Kondor, R., Howard, A.: Probability product kernels. *J. Mach. Learn. Res.* **5** (2004) 819–844
7. Barla, A., Odone, F., Verri, A.: Histogram intersection kernel for image classification. In: *Proceedings of International Conference on Image Processing.* Volume 3. (2003) 513–516
8. Herbrich, R., Graepel, T., Obermayer, K.: Support vector learning for ordinal regression. In: *Proceedings of the Ninth International Conference on Artificial Neural Networks, 1999.* Volume 1. (1999) 97 – 102
9. Odone, F., Barla, A., Verri, A.: Building kernels from binary strings for image matching. *IEEE Transactions on Image Processing* **14** (2005) 169–180
10. Wang, G., Dunbrack Jr, R.: Pisces: a protein sequence culling server. *Bioinformatics* **19** (2003) 1589–1591
11. Melo, F., Sanchez, R., Sali, A.: Statistical potentials for fold assessment. *Prot. Sci.* **11** (2002) 430–448
12. Zhang, Y., Skolnick, J.: Scoring function for automated assessment of protein structure template quality. *Proteins* **57** (2004) 702–710
13. Murzin, A., Brenner, S., Hubbard, T., Chothia, C.: Scop: a structural classification of proteins database for the investigation of sequences and structures. *J. Mol. Biol.* **247** (1995) 536–540