Complex Networks Govern Coiled-Coil Oligomerization – Predicting and Profiling by Means of a Machine Learning Approach*[S]

Carsten C. Mahrenholz‡‖, Ingrid G. Abfalter§‖, Ulrich Bodenhofer§, Rudolf Volkmer‡, and Sepp Hochreiter§¶

Understanding the relationship between protein sequence and structure is one of the great challenges in biology. In the case of the ubiquitous coiled-coil motif, structure and occurrence have been described in extensive detail, but there is a lack of insight into the rules that govern oligomerization, *i.e.* how many α-helices form a given coiled coil. To shed new light on the formation of two- and three-stranded coiled coils, we developed a machine learning approach to identify rules in the form of weighted amino acid patterns. These rules form the basis of our classification tool, *PrOCoil*, which also visualizes the contribution of each individual amino acid to the overall oligomeric tendency of a given coiled-coil sequence. We discovered that sequence positions previously thought irrelevant to direct coiled-coil interaction have an undeniable impact on stoichiometry. Our rules also demystify the oligomerization behavior of the yeast transcription factor GCN4, which can now be described as a hybrid—part dimer and part trimer—with both theoretical and experimental justification. ***Molecular & Cellular Proteomics 10: 10.1074/mcp.M110.004994, 1–9, 2011.***

Fifty-nine years ago, in 1952, L. Pauling (1) and F. H. C. Crick (2) first described the structure of the α-helical coiled coil. Since then it has become a prototypical textbook example of a structural motif, being commonly described as consisting of between two and seven α-helices. Almost 6% of the proteins in the Protein Data Bank (PDB)[1] contain coiled-coil regions (3), of which more than 90% show dimeric or trimeric interactions. Because of their ability to oligomerize, coiled coils perform, either on their own or as part of larger protein complexes, a variety of important cellular functions (4). Their ubiquity and the stable interactions of their helices make coiled coils ideal building blocks for designing novel proteins. Furthermore, coiled-coil interactions have recently attracted attention as promising drug targets (5). Their use in successful inhibition of membrane fusion proteins of viruses such as HIV (6) and avian influenza (7) supports the concept of rational drug design based on coiled-coil proteins (8).

Today, a plethora of information about coiled coils is available, including their prevalence, sequence characteristics, and structures. They have in common a periodically recurrent sequence called a heptad repeat of the form $(abcdefg)_n$. Usually, the positions *a* and *d* in these repeats are occupied by hydrophobic amino acids located at the hydrophobic core crucial for tertiary structure, whereas positions *e* and *g* typically are charged residues (9). These obvious regularities and the clear and simple appearance of coiled-coil structures have made possible a large number of computational approaches to their analysis. These range from (i) simple sequence-based approaches using single (10) and pairwise residue distributions (11, 12) to (ii) approaches based on hidden Markov models without scanning windows (13), (iii) structure-based approaches detecting knobs-into-holes packing in helical bundles (14), and (iv) approaches based on matrices of residue frequencies that aim to distinguish different oligomeric tendencies (15, 16). (For a detailed comparative analysis of coiled-coil prediction methods see (17).)

Hence, one might expect our understanding of coiled coils to be complete. Most remarkably, however, the hidden and more complex rules for oligomeric formation, and thus the key to biological function, are poorly understood. A first but crude indicator of the oligomeric state of coiled coils may be their intra- and extracellular prevalence, which clearly does not provide any information about the sequence features that govern oligomerization. Despite extensive experimental and computational efforts such as mutation analysis, NMR, x-ray crystallography, and statistics (18–22), our knowledge of which oligomer a specific coiled coil forms has, until now, been limited to describing the phenomenon on the basis of a small number of protein samples. Now, as the amount of

From the ‡Institute of Medical Immunology, Charité Medical School, Hessische Str. 3-4, 10117 Berlin, Germany; §Institute of Bioinformatics, Johannes Kepler University, Altenberger Str. 69, 4040 Linz, Austria

Received September 13, 2010, and in revised form, January 18, 2011

⌘ *Author's Choice*—Final version full access.

Published, MCP Papers in Press, February 2, 2011, DOI 10.1074/mcp.M110.004994

[1] The abbreviations used are: PDB, Protein Data Bank; BLAST, Basic Local Alignment Search Tool; BLOSUM, BLOcks SUbstitution Matrix; FDR, false discovery rate; GCN4, yeast transcriptional activator protein; SVM, support vector machine.

available postgenomic sequence data is growing rapidly, the challenge is to explain coiled-coil oligomerization by extracting an actual set of rules from this data.

Experimentalists have recently discovered the first complex trimerization pattern by comparing trimeric coiled-coil sequences (23), and their latest results prove that it is in fact context-dependent (24). However, experimental approaches can never be exhaustive, making bioinformatics the method of choice (18) for identifying oligomerization rules embedded in the sequence data. To this end, we employ machine learning methods that distinguish dimers from trimers with high accuracy. Simultaneously, we extract rules that characterize each type of oligomer in the form of patterns to which we assign weights. We designed a sequence profiling tool, *PrOCoil*, that analyzes the patterns present in a given coiled-coil sequence to determine and visualize the contribution of each amino acid to the overall oligomer tendency using the rules in which it participates. *PrOCoil* elucidates, for example, the hitherto puzzling behavior of the yeast transcriptional activator GCN4, which can, as a result of minimal mutations in its amino acid sequence, switch from forming a dimer to forming a trimer (see, *e.g.* (19, 23)). Despite the challenge this borderline representative and its mutants pose when it comes to predicting their oligomeric states, it is exactly this property that identified GCN4 as the ideal test candidate to verify our findings both computationally and experimentally. Based on our double substitution analysis using SPOT synthesis (19) and a trigger-sequence-based mutated sequence (23), dimer- and trimer-forming GCN4 mutants were selected and examined using the previously revealed rules. The stoichiometry of each GCN4 mutant was analyzed and confirmed by biophysical methods (see supporting information online and (24)).

EXPERIMENTAL PROCEDURES

Data Preparation—We scanned the whole PDB (25) for dimeric and trimeric coiled-coil segments. Other oligomers were not considered in this work because they account for less than 10% of the structurally resolved coiled-coil structures. We used the program SOCKET (14) with a packing cutoff of 7.0 Å to scan the PDB (25) for knobs-into-holes packing between helices. The output was first parsed for dimeric and trimeric sequences and then divided into parallel and antiparallel samples. We refined our data set of parallel dimeric and trimeric coiled coils by removing identical (sub-)sequences, as they contribute no additional sequence information. Thus, we created a database of 385 dimeric and 92 trimeric coiled-coil sequences with heptad registers assigned by SOCKET.

We augmented the data set with coiled-coil sequences that were not yet structurally resolved and thus not listed in the PDB. To this end, we retrieved the complete amino acid chains containing coiled-coil segments from the PDB entries and masked the areas SOCKET had identified as coiled coils. Those chains that provided at least 40 unmasked amino acids were then used as inputs to Basic Local Alignment Search Tool (BLAST) (26) searches in the NR database. Subsequently, we removed BLAST output sequences that were less than 85% identical to the unmasked regions of the query sequences. Then we used the remaining sequences as input for the program MARCOIL (13) to confirm that they contain coiled-coil segments and to assign their heptad registers. Finally, only sequences that reached or exceeded a coiled-coil probability of 85% according to MARCOIL were selected and included in the data set. This resulted in a combined PDB and approved BLAST pool of 2043 dimers and 791 trimers.

In contrast to hitherto published approaches to coiled-coil analysis, we divided the PDB samples in our data set into clusters such that the maximum sequence identity between any two sequences from two different clusters was 60% (according to ungapped, heptad-specific pairwise alignments). Subsequently, we created an augmented data set by adding each sequence from the approved BLAST pool to the cluster of the query sequence from which it originated. We thus obtained two 60%-clustered data sets: one based exclusively on PDB samples and one augmented by BLAST. We chose an identity threshold of 60% because any lower level would have merged about half of the data set into a single cluster. This is due to the fact that coiled coils have a highly similar secondary structure and thus also have a priori a high level of sequence similarity.

Heptad-Specific Single Amino Acid Frequencies—For the clustered data set, each cluster was considered as a single coiled-coil sequence. This was accomplished by performing an ungapped, heptad-specific multiple alignment of all sequences in the cluster. Then, a cluster sequence was represented by the relative frequencies of amino acids at each of the aligned positions (analogous to the way clusters are treated when computing BLOcks SUbstitution Matrix (BLOSUM) matrices (27)). Finally, the overall single amino acid frequencies were computed as the sums of relative amino acid frequencies at all heptad positions in all clusters. The statistical significance of each single amino acid position was determined by Fisher's exact test (28), comparing the numbers of occurrences of a given amino acid at a given heptad position in trimers and dimers against the occurrences of other residues in the same heptad position. The overall numbers of occurrences of heptad positions in the sequences of the 60%-clustered data set amount to around 210 in trimers and around 800 in dimers. These sample sizes are large enough to have sufficient statistical power to detect even small differences in amino acid frequencies. Finally, we obtained nine single amino acid patterns that were significant according to the Benjamini-Hochberg false discovery rate (FDR) correction (29) with an FDR threshold of 0.05.

Statistical Significance of Amino Acid Pairs—In order to test whether patterns of pairs of amino acids provide a gain of information compared with single amino acid patterns, we considered all possible pairings of amino acids at specific heptad positions with at most six other residues in between. Again, we applied Fisher's exact test, this time comparing joint occurrences of two residues against occurrences of the first residue with other residues (again separately for trimers and dimers). Here, the overall sample size is the number of occurrences of the first single amino acid pattern. Extensive power calculations showed that we need at least 60 occurrences of a single amino acid pattern to detect statistical differences with sufficient certainty. Of the 4360 pair patterns fulfilling this criterion in the 60%-clustered data set, 130 pairs showed a p value of at most 0.05. After applying Benjamini-Hochberg FDR correction and a stringent FDR threshold of 0.05, two pair patterns remained significant. Thus, we indeed observe a gain in information with high statistical significance. Note, moreover, that we may have overlooked many potentially valuable pair patterns because the sample sizes were too small to detect a difference with sufficient significance.

Support Vector Machines (SVMs) and the Coiled-coil Kernel—The nontechnical reader may find these introductory tutorials (30–32) or standard literature (33–35) helpful to become familiar with the topic of support vector machines. We employed the well-established SVM implementation LIBSVM (36). Suppose we wish to perform a binary classification of samples x_i (in our case amino acid sequences with heptad registers assigned). Each sample can belong

either to the positive class with the label $y_i = 1$ (trimers) or to the negative class with the label $y_i = -1$ (dimers). For a given training set $\{(x_i,y_i)|1 \leq i \leq l\}$, the discriminant function value of the support vector machine for a sample x is given by

$$f(x) = b + \sum_{i=1}^{l} \alpha_i \cdot y_i \cdot k(x,x_i),$$

where b and α_i are optimized according to the training data. The two-place function k, referred to as kernel function, measures the similarity of two samples. We use our novel *coiled-coil kernel*, which can be written as

$$k(x,y) = \sum_{p} N(p,x) \cdot N(p,y).$$

A pair pattern p consists of two amino acids and a fixed number of up to m arbitrary amino acids in between. We indicate at which heptad position the first amino acid must occur. The pattern S.I*f*, for instance, matches a coiled-coil sequence if a Ser occurs at an *f* position and an Ile at the next *a* position (with an arbitrary amino acid at the *g* position in between). For a given pattern p and a sequence x, $N(p,x)$ denotes the number of occurrences/matches of pattern p in sequence x. The *coiled-coil kernel* calculates the number of coiled-coil patterns shared by two sequences, taking multiple occurrences into account. It bears some resemblance to the spatial sample kernel (37) and the kernel described in (38). However, in contrast to (37), the *coiled-coil kernel* has an additional position/heptad-specific property, and in contrast to (38), it considers pairs of residues from the same chain and is not restricted to a small set of pairs of positions. The kernel values were normalized to correct for variations in sequence length (32):

$$k'(x,y) = \frac{\sum_p N(p,x) \cdot N(p,y)}{\sqrt{\sum_p N(p,x)^2}\sqrt{\sum_p N(p,y)^2}}$$

Model Selection—The validity of our model selection was verified by nested cross-validation. In the outer cross-validation loop, the whole data set was split into 10 parts with a maximum sequence identity of 60% between parts. In each of 10 runs, the 10 parts were grouped differently to form a training data set (nine parts) and an unseen data set (one part). Model selection was performed by means of a ninefold inner cross-validation on the training data set. The resulting best model was then tested on the unseen data. The results in supplemental Table S7 show that our model selection procedure performs very well on independent test sets. Hence, we can safely apply cross-validation-based model selection to the entire data set. The best model in terms of accuracy obtained in this way was that trained with the BLAST-augmented data set using the normalized *coiled-coil kernel* with $m = 7$ and the SVM penalty parameter $C = 8$. Following retraining with the complete data set (*i.e.* with no data omitted), this became our PrOCoil model.

Pattern Extraction—Pattern extraction was performed by rearranging the discriminant function $f(x)$ as described in (39) to obtain the weights $w(p)$ of the patterns p given a support vector machine:

$$f(x) = b + \frac{1}{\sqrt{\sum_p N(p,x)^2}} \sum_p N(p,x) \cdot \underbrace{\sum_{i=1}^{l} \frac{\alpha_i \cdot y_i \cdot N(p,x_i)}{\sqrt{\sum_p N(p,x_i)^2}}}_{= w(p)}$$

Sequence Profiling—The discriminant function $f(x)$ was reformulated such that each position or amino acid i in the sequence x is attributed to the weight s_i (*i.e.* the sum over half of the weight of all patterns of which it is part) it contributes to the discriminant function. The base line of the resulting sequence profiling plot is given by $y = -b/L$.

$$f(x) = b + \sum_{i=1}^{L} s_i = \sum_{i=1}^{L} \left(s_i - \left(-\frac{b}{L} \right) \right)$$

RESULTS

Pattern Identification by Statistical Analysis is Insufficient for Predicting Oligomerization States—The first method we employed in search of oligomerization rules was a statistical analysis of the frequency of each amino acid at each position of the heptad register in dimers and trimers, in line with SCORER, Woolfson and Alber's first oligomerization predictor (15). The relative frequency results for the 60%-clustered data set are shown in supplemental Table S3 and supplemental Fig. S2. We calculated p values by Fisher's exact test (28) in order to identify those amino acids at specific heptad positions whose comparatively higher frequencies in one oligomer than in the other were statistically significant. Correction for the false discovery rate resulted in nine significant residues at specific heptad positions according to the Benjamini-Hochberg method (29) (see supplemental Table S3). Initially, it may seem that there is a clear preference for Ile at both hydrophobic core positions of trimers, as previously described in the literature (40). However, we identified a clear and significant preference for the amino acid Ile only in *a* positions of the hydrophobic cores of trimers. This β-branched amino acid also frequently occupies the *d* positions in trimers, but false discovery rate correction shows that the high prevalence in *d* positions is, in fact, not statistically significant. Arg, Asn, and Lys at *a* positions are the only amino acids in a hydrophobic core position that have a statistically higher prevalence in dimers. Interestingly, these residues in this particular heptad position are tolerated almost exclusively in dimers. In trimers, positions that usually form salt bridges show a comparatively higher prevalence of the small, uncharged amino acid Gly at *e* position. Additionally, Asn is more prevalent at this position in trimers. In dimers, Glu in *g* position, but not in *e* position, is statistically significant. In the literature (see, *e.g.* O'Shea *et al.* (9)), *e* and *g* positions are usually attributed the same characteristics. Our results, however, show that the amino acid distributions at these heptad positions are, in fact, different. For positions that do not participate in direct coiled-coil interaction, we found statistically significant amino acids only in trimers, namely Ala at *c*, and Ser at *f* positions. In summary, although there are obvious statistical differences in the occurrences of certain residues at certain heptad positions in dimers and trimers, these differences are insufficient to distinguish dimers from trimers. Specific residues at certain positions occur relatively infrequently, giving these occurrences a high specificity but low sensitivity, *i.e.* for the majority of sequences we cannot predict the oligomerization state because there is a lack of reliable indicators. The sensitivity can be improved by considering simultaneously multiple occur-

rences of single amino acids, but even this is insufficient to explain oligomerization. Hence, we shifted our focus to the dependences between amino acids within and beyond a heptad. We searched for a method that could draw on a maximum number of interactions and combine these into a network of rules that would allow us to predict and examine oligomerization. Support vector machines (SVMs) are ideally suited to this task and have previously been used in a different context to predict protein-protein interactions that are mediated by the coiled-coil motif (38).

Support Vector Machines Provide a Sound Basis for Finding Oligomerization Rules—In recent years, SVMs have become established as a standard tool in machine learning, and their popularity for biosequence classification has increased dramatically (41). SVMs provide mathematically sound classifications even if the data set is too small to achieve significant results with probabilistic techniques. In fact, SVMs are the method of choice both because they can be used to distinguish dimers from trimers and because, at the same time, they also provide the rules on which their decisions are based and which are so valuable for protein design purposes. In the context of classifying biological sequences, SVMs require a kernel that obtains two sequences as input and supplies a scalar value as a measure for their similarity. Inspired by earlier approaches that make use of pairwise residue co-occurrences (11, 12, 16, 38), we developed a new kernel, hereafter called *coiled-coil kernel,* that is tailored to classifying coiled-coil proteins. We verified (see supplemental Section S3 and supplemental Tables S5 and S6) that this kernel does indeed outperform significantly the currently most popular sequence kernels (*spectrum* (42) and *mismatch kernel* (43)). Using the *coiled-coil kernel*, an SVM generates rules by optimizing the pattern weights such that the combined rules achieve maximum discrimination between dimers and trimers.

Model Selection and Classification Results—To identify the SVM classifier (*i.e.* model) with the optimum SVM and *coiled-coil kernel* parameters, we had to assess which model performs best on future (previously unseen) data. For this purpose, we applied nested cross-validation according to state-of-the-art standards.

Our classification results are noteworthy: In the model selection procedure we were able to classify test (*i.e.* unknown) sequences with 86.9% average accuracy (see supplemental Table S7 for more details), even though they had only a maximum identity of 60% to any (known) coiled coil with which the SVM was trained. This is especially remarkable because the tests used all structurally resolved coiled-coil sequences and not the limited number of individual samples often used in experimental validation. Our calculations show that training the SVM with the augmented data set enhanced the classification. Using only structurally resolved PDB sequences in our test data sets ensured that this improvement was not due to the optimization of an "artificial" data set. Furthermore, our approach also ranks previously unknown

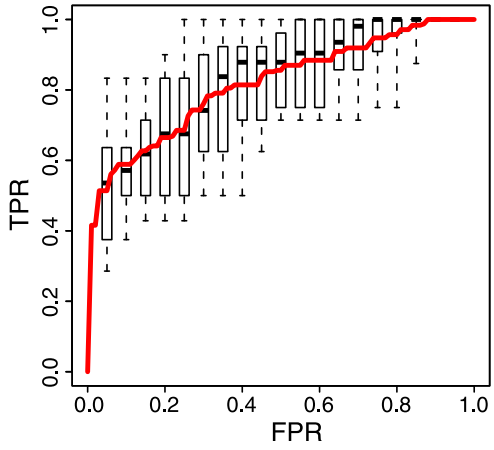


FIG. 1. **Average ROC curve of the PrOCoil model, obtained from 10-fold cross-validation on the 60%-clustered data set.** The superimposed error bars illustrate the deviations of the 10 ROC curves from which the average (red curve) was computed.

sequences very well, as shown by the ROC curve in Fig. 1. The average area under the curve obtained from 10-fold cross-validation on the 60%-clustered data set was 0.82 (see also supplemental Table S8).

Finally, the best setting identified with 10-fold cross-validation was used to train an SVM model on the entire BLAST-augmented data set, the result of which became our PrOCoil model. Machine learning theory states that the performance of the PrOCoil model on future data will be similar to the results achieved by the above model selection procedure (33, 44).

Supplemental Section S5 further illustrates that our approach indeed outperforms the state-of-the-art oligomerization predictor MultiCoil (16).

Pattern Extraction and Sequence Profiling—A Network of Interactions Determines Stoichiometry—Based on the rules constructed by our *coiled-coil kernel* approach, amino acid patterns were extracted from the augmented data set. These patterns comprise pairs of amino acids at certain heptad positions that are characteristic of each type of oligomer (see Fig. 2 and supplemental Fig. S4). We then used this information to implement a prediction and sequence profiling tool, *PrOCoil*, that characterizes the overall oligomeric tendency of a coiled-coil sequence by displaying each amino acid's contribution to the rules in which it participates in a specific sequence. Fig. 3 depicts the sequence profiling plots of a typical dimer (c-Jun, PDB entry 1jun) and a typical trimer (hemagglutinin, PDB entry 1htm) using *PrOCoil*. The sequences show a clear overall dimeric and trimeric tendency respectively. These tendencies are indicated by the areas above and below the base line, which equate respectively to the positive/trimeric contributions and the negative/dimeric contributions of the patterns involved. Fig. 4*A* depicts the same kind of plot for wildtype GCN4—a dimeric coiled-coil protein renowned for its ability to adopt easily a different

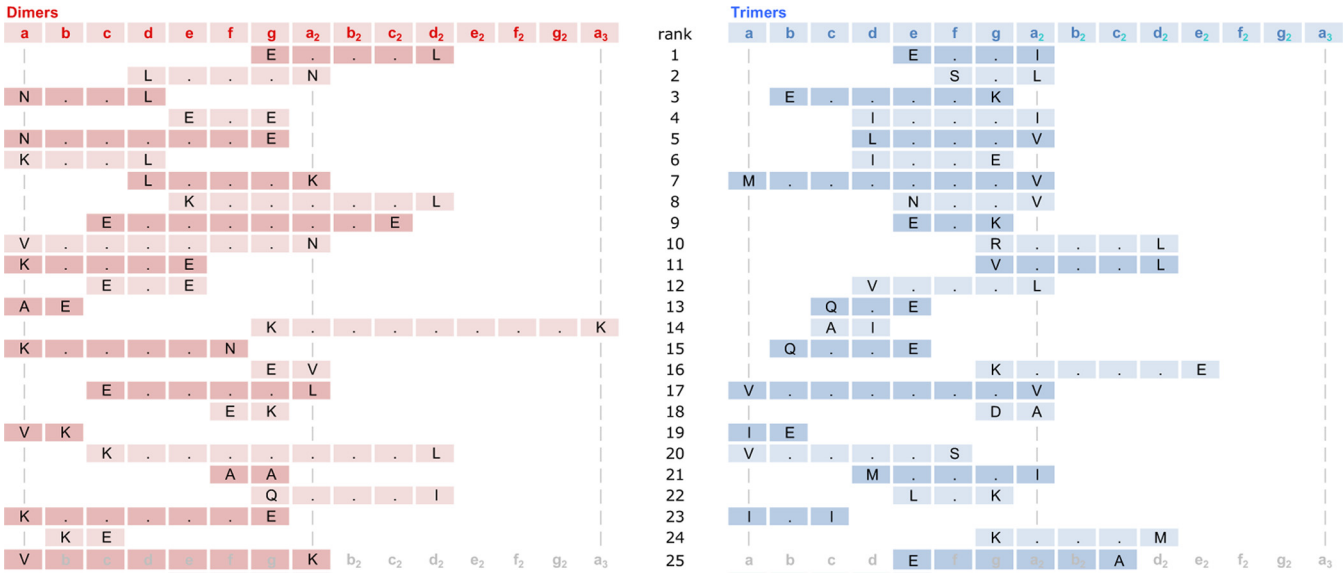


FIG. 2. **List of the 25 strongest pairwise patterns.** Dimer patterns are highlighted in pink, and trimer patterns are highlighted in blue. For instance, the top dimer pattern E . . . L, spanning columns g to d_2, describes a pattern with Glu at a *g* position, Leu at the next *d* position, and three arbitrary amino acids in between.

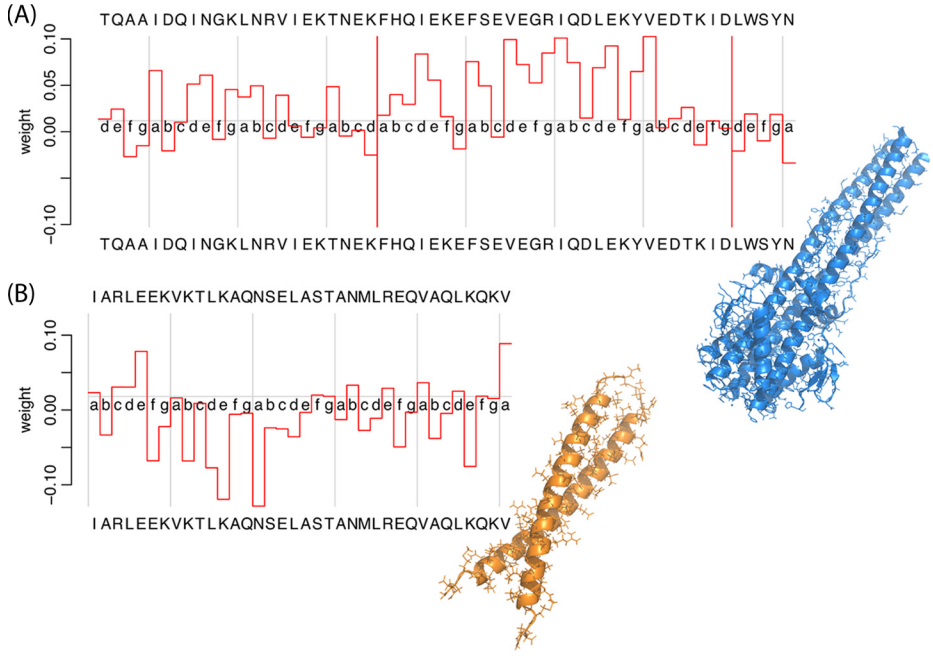
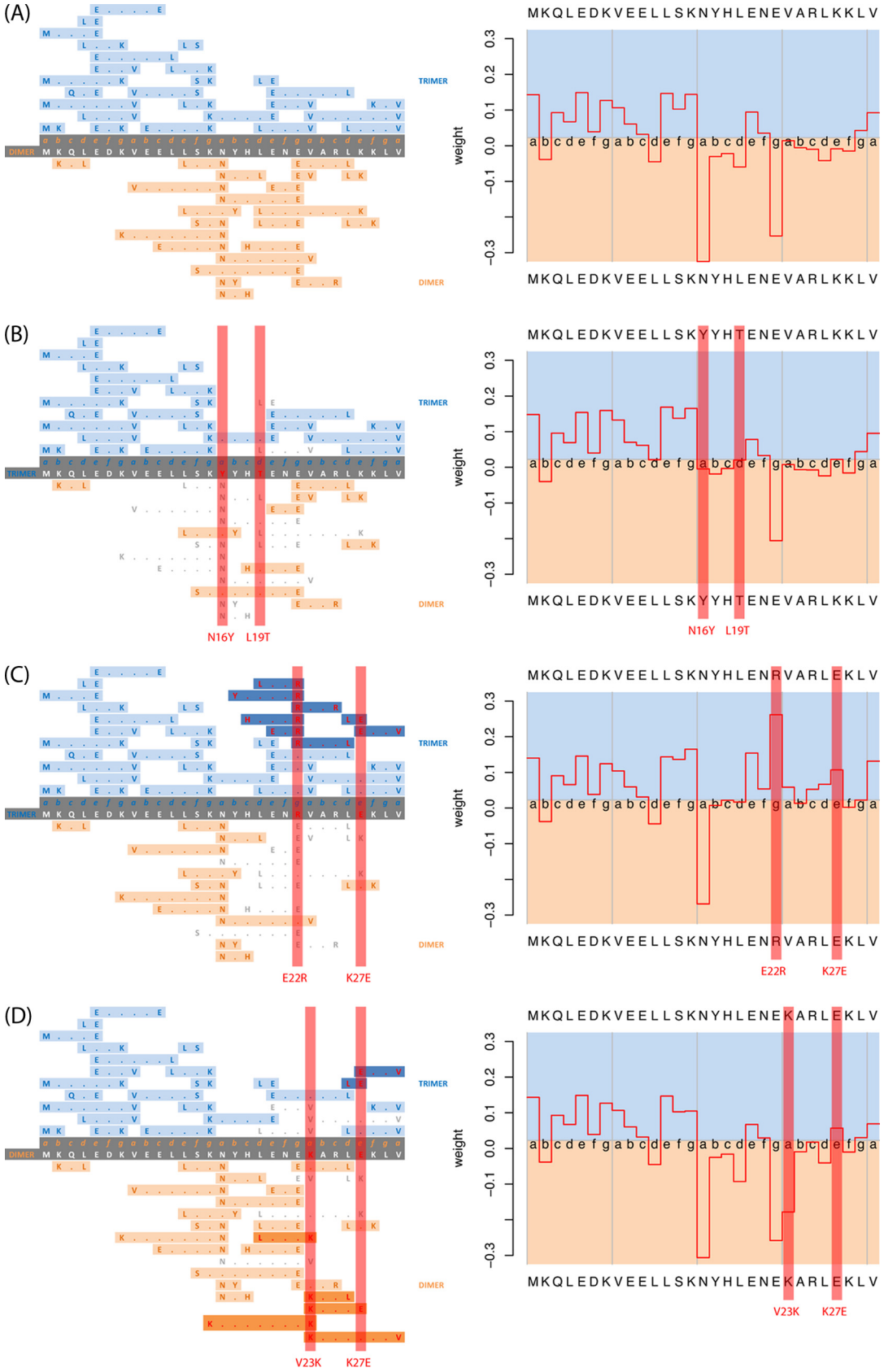


FIG. 3. **Sequence profiling and classification of a typical dimer and a typical trimer.** The plot shows (*A*) the hemagglutinin trimer and (*B*) the JUN-JUN dimer based on pairwise patterns of the PrOCoil model, visualizing the contribution of each amino acid position to the overall oligomeric tendency. The area above the base line equates to the positive/trimeric contributions, the area below corresponds to the negative/dimeric contributions. The three-dimensional structures were plotted using PyMOL (http://pymol.sourceforge.net/).

oligomerization state with very few mutations of its amino acid sequence (19). The plot shows how this is possible—unlike c-Jun and hemagglutinin, GCN4 does not display a clear tendency toward one oligomeric state. It is impossible to assess at first glance whether the area above or below the base line is larger. As this protein combines both dimeric and trimeric characteristics, it takes only a few selected mutations to tip the scales in one direction or the other. GCN4 was thus the ideal candidate to demonstrate that the PrOCoil model can not only be used to provide excellent classification of wild-type sequences, but can even be employed for mutation analysis, given that a sufficiently large set of (similar) samples is provided with which it can be trained.

Mutation Analysis of GCN4 Mutants Using PrOCoil—Solving the Puzzle—We chose two mutant GCN4 sequences from our double substitution analysis (19) for which the oligomeric state was assessed by analytical ultracentrifugation (see supplemental Tables S1 and S2) and a sample which was mutated using a trimerization motif (23). These mutants were not part of the data sets used for pattern extraction. Although predicting their oligomerization states is challenging because the amino acid sequences of the dimeric wild-type GCN4 and


(A)
TRIMER
DIMER
weight
MKQLEDKVEELLSKNYHLENEVARLKKLV
(B)
N16Y L19T
(C)
E22R K27E
(D)
V23K K27E

its dimeric and trimeric mutants differ only at very few positions, *PrOCoil* classified all samples correctly. The corresponding profiling plots are shown in Figs. 4*B*–4*D*. Additionally, we show the changes caused by the mutations in the pairwise patterns. It immediately becomes apparent that the Asn in position 16 contributes the most weight to the dimeric tendency of $GCN4_{wt}$ (Fig. 4*A*). This can be explained structurally by the fact that the Asn16 residues adopt an energetically more favorable conformation in dimers than in trimers because they point into the hydrophobic core and form a distinctive hydrogen bond that is critical for dimer formation in GCN4 (24). Asn16 participates in 10 of the top 100 dimer patterns (see supplemental Fig. S4), notably also in two of the three strongest of them all: L . . . N*d* and N . . L*a*, where the pattern L . . . N*d*, for example, denotes a Leu at *d* position, an Asn at the next *a* position, and three arbitrary amino acids in between. From this follows that Asn16 represents the ideal target for mutational analysis to switch oligomerization: Simple deletion of core-stabilizing dimer patterns by altering Asn16 to Tyr and Leu19 to Thr while not adding important trimeric patterns results in a correctly predicted change in oligomerization (Fig. 4*B*). That is, the overall effect of destroying strong dimer patterns is sufficient to tip the oligomeric tendency in the direction of trimerization. What our bioinformatics tool discovered and identified as a deletion of dimer patterns corresponds to dimer-destabilizing mutations described by experimentalists in the literature. Hu *et al.* showed that replacing Leu19 with Thr results in conditionally functional GCN4 (45). Furthermore, Potekhin *et al.* proposed that Asn16 desolvation is the driving force for GCN4 dimerization and that most substitutions at this position result in the preferential formation of a trimeric structure (46).

The next example (Fig. 4*C*) shows that, alternatively, trimerization can also be triggered without replacing Asn16 if strong trimer patterns are added instead. For this purpose, the trimerizer pattern Arg(*g*)-h(*a*)-x-x-h(*d*)-Glu(*e*') was inserted (23). As Steinmetz and Kammerer verified experimentally (24), and *PrOCoil* predicted correctly, oligomerization switches if mutations are inserted at positions that create strong trimer patterns. It has been shown that the Arg and Glu in this motif form a characteristic bifurcated interhelical salt-bridge network, and, because of tight packing interactions with neighboring residues, they play a role in the formation of the hydrophobic core (23). These effects are the basis for the trimerization driving force of the motif. Arbitrary mutations in this sequence region, however, do not change oligomerization. The substitutions have to be selected carefully to create additional trimer patterns. To prove this point, we chose a sequence in which Val23 was replaced by Lys, and Lys27 by Glu (Fig. 4*D*). This resulted in the loss of three important Val23-related trimer patterns whereas only two Lys-related ones were added. At the same time, a strong Lys- and Glu-related and four strong Lys-related dimeric patterns were created. This substitution with physicochemically similar amino acids added new dimer patterns and, as predicted, the mutations do not affect oligomerization.

Pairwise Patterns—The Building Blocks of Complex Networks—The 25 most influential amino acid pairings according to our PrOCoil model are shown in Fig. 2. On closer inspection, we are able to support general hypotheses of oligomerization that rely on defining the hydrophobic core positions *a* and *d* (40) and can extend the mainly GCN4-leucin-zipper-based knowledge concerning the core positions to the whole heptad. I . . . I*d*, for example, is a well known trimer pattern of Ile at core positions *d* and *a* that is also ranked high in our list (as number 4). However, patterns that combine a core position with a noncore position seem to be at least as important to trimerization. For example, patterns with Leu, Ile or Val at a core position and a tiny Ser at *f* position are ranked as numbers 2, 20, and 75, respectively. Ile at a core position combined with a charged Glu at *e* position is ranked as number one, and with Glu at *g* position as number six. In dimers, the most highly ranked patterns are combinations of amino acids with the γ-branched Leu in core positions *a* and *d*. Interestingly, highly ranked β-branched combinations with Ile in a core position also occur in dimers (*e.g.* pattern 22). The 100 most important pairwise patterns for each oligomer according to the PrOCoil model are listed in supplemental Fig. S4.

DISCUSSION

Inspired by approaches that sought to define and predict coiled coils based on statistics, we first attempted to find the rules for dimeric and trimeric oligomerization by examining the position-specific single amino acid frequencies in each oligomer. In contrast to hitherto published coiled-coil statistics, our single amino acid statistics are based on clustered data. This compensates for the (artificially) high prevalence of certain sequences in the PDB stemming from concentrated scientific interest in certain types of proteins. Our results show that a simple statistical analysis cannot provide an explanation or rules for a sequence's preference for a certain oligomeric state. Inspired by Berger *et al.*, who used

FIG. 4. **Sequence profiling and classification of GCN4 and its mutants.** The plot shows (*A*) the dimeric transcriptional activator protein $GCN4_{wt}$, (*B*) the trimeric $GCN4_{N16I,L19N}$ mutant, (*C*) the trimeric $GCN4_{E22R,K27E}$ mutant, and (*D*) the dimeric $GCN4_{V23K,K27E}$ mutant. The sequence profiling plots on the right side visualize the contribution of each amino acid position to the overall oligomeric tendency, based on the pairwise patterns from the PrOCoil model. The area above the base line (blue) equates to the positive/trimeric contributions, the area below (orange) corresponds to the negative/dimeric contributions. Red bars mark the positions that were mutated. Patterns of the top 90 pairwise pattern list found in GCN4 and its mutants are depicted on the left side to visualize which patterns are added (dark color) or lost (gray) through mutation.

pairwise residue correlations for predicting coiled coils (11, 12) and made progress in predicting dimer and trimer formation (16), we used the hypothesis that all amino acids in a given sequence influence each other as a basis for our approach.

We have shown that stepping up to a higher level of complexity, examining the relations between amino acids, is the key to predicting and understanding oligomerization. For the first time, a complete network of sequence parameters that influence oligomerization has been established, and the underlying rules of coiled-coil formation have been provided. We used an SVM with our new *coiled-coil kernel* as the method of choice enabling us to classify with outstanding accuracy new dimers and trimers from their amino acid sequences. The validity of our classification results was verified by means of stringent state-of-the-art testing methods and ensures that the valuable rules (*i.e.* the weighted patterns we subsequently extracted) which the machine learned to determine oligomeric preference are indeed based on significant patterns. Also, our tool outperforms the state-of-the-art oligomerization predictor MultiCoil (see supplemental Section S5).

Although the statistical approach identified similar individual amino acids as important, our method is much more powerful: (i) To understand and predict oligomerization, the patterns in a sequence must be viewed in context, as parts of a network of interactions spanning the whole sequence. Using the example of GCN4, we have demonstrated that we can merge this information in our sequence analyzing tool *PrOCoil* and draw an overall picture that explains the behavior of a sequence that, until now, seemed unclassifiable. (ii) We are able to provide a detailed picture of the influence of each amino acid on the overall structure by taking its neighborhood into account. *PrOCoil* can even be used to indicate which sequence positions contribute most to the dimeric or trimeric tendency. An amino acid at a certain position participates in various patterns; consequently, the patterns in a sequence are correlated. The factorization performed by *PrOCoil* is therefore essential to decorrelate the patterns and to reduce a coiled-coil-spanning network to its building blocks.

Our mutant GCN4 examples illustrate that the influence of mutations on oligomerization depends on the sequence context, *i.e.* on the overall effect of the change in the interdependent patterns caused by the mutations. Added trimer patterns and/or loss of strong dimer patterns results in an overall trimeric structure, whereas adding strong dimer patterns maintains dimerization of the protein.

The influence of the *b*, *c*, and *f* positions on oligomerization has long been underestimated because research has focused mainly on the positions in the hydrophobic core. Our results, however, indicate that all positions inside a heptad contribute to the oligomeric tendency of a coiled-coil sequence. In fact, nine out of the 25 most influential trimer patterns and 10 out of the 25 most influential dimer patterns are pairings with noncore positions (Fig. 2). The fact that a single amino acid approach is insufficient because individual positions must always be viewed in context becomes particularly obvious when comparing the respective patterns ranked as number two for dimers and as number five for trimers. Both patterns have a Leu at *d* position, but when combined with Asn at *a* position (L . . . N*d*), it counts in favor of dimers, whereas with Val in *a* position (L . . . V*d*) it is characteristic of trimers.

The patterns provided, together with our sequence profiling tool, will assist experimentalists in coiled-coil mutation analysis. We are confident that our findings will also improve rational coiled-coil design. In summary, the data we collected and the tool we designed in this work offer a new basis for coiled-coil prediction and design. Our findings shed light on the link between coiled-coil sequence and structure.

Outlook—We have shown that the coiled-coil data set can be augmented and classification enhanced with a sophisticated approach that incorporates a BLAST search followed by a strict selection process. This is of great interest in view of the vast quantity of data with which next generation sequencing techniques will soon provide us. Our BLAST approach can be used to tap this source of data for a wealth of new training sequences. Subsequent retraining of SVMs with our *coiled-coil kernel* should further refine our pattern set and further improve our prediction performance.

The next step will be to extend the current machine, which characterizes oligomerization states of coiled-coil proteins, into a machine that also predicts the non-coiled coil case. Mutations that are potentially coiled-coil-disrupting or extremely rare cases of high-order oligomerizations (tetramers, pentamers, etc.) have as yet to be checked by complementary experiments. Oligomers of higher order can at this time not be predicted by *PrOCoil* because the number of structurally verified samples is too small to produce meaningful results by training an SVM. However, our method can easily be extended to include their prediction should a sufficiently large set of such samples become available in the future.

Availability—A web version and an R package of our prediction and profiling software (PrOCoil) are available to the scientific community (see http://www.bioinf.jku.at/software/procoil/).

Acknowledgments—We gratefully acknowledge Michel Steinmetz, Philip Kim, Michal OrGuil, and Andrei Lupas for helpful discussions and comments on the manuscript.

* CCM supported by the Manchot Foundation, the GlaxoSmithKline Foundation, and the Charité Medical School, Berlin. IGA and SH supported by the City of Linz. RV supported by Deutsche Forschungsgemeinschaft (SFB449).

S This article contains supplemental Sections S1 to S5, Tables S1 to S9, and Figs. S1 to S5.

¶ To whom correspondence should be addressed: Institute of Bioinformatics, Johannes Kepler University, Linz, Austria 4040. Tel.: +43 732 2468 8880; Fax: +43 732 2468 9511; E-mail: hochreit@bioinf.jku.at.

‖ The first two authors contributed equally to this work.

REFERENCES

1. Pauling, L., and Corey, R. B. (1953) Compound helical configurations of polypeptide chains: Structure of proteins of the α-keratin type. *Nature* **171,** 59–61
2. Crick, F. H. C. (1952) Is α-keratin a coiled coil? *Nature* **170,** 882–883
3. Hadley, E. B., Testa, O. D., Woolfson, D. N., and Gellman, S. H. (2008) Preferred side-chain constellations at antiparallel coiled-coil interfaces. *Proc. Natl. Acad. Sci. U.S.A.* **105,** 530–535
4. Burkhard, P., Stetefeld, J., and Strelkov, S. V. (2001) Coiled coils: a highly versatile protein folding motif. *Trends Cell Biol.* **11,** 82–88
5. Strauss, H. M., and Keller, S. (2008) Pharmacological interference with protein-protein interactions mediated by coiled-coil motifs, in P*rotein-Protein Interactions as New Drug Targets* (Klussmann, E., and Scott, J., eds) vol. 186 of *Handbook of Experimental Pharmacology*, pp. 461–482, Springer, Berlin
6. Bianchi, E., Finotto, M., Ingallinella, P., Hrin, R., Carella, A. V., Hou, X. S., Schleif, W. A., Miller, M. D., Geleziunas, R., and Pessi, A. (2005) Covalent stabilization of coiled coils of the HIV gp41 N region yields extremely potent and broad inhibitors of viral infection. *Proc. Natl. Acad. Sci. U.S.A.* **102,** 12903–12908
7. Russell, R. J., Kerry, P. S., Stevens, D. J., Steinhauer, D. A., Martin, S. R., Gamblin, S. J., and Skehel, J. J. (2008) Structure of influenza hemagglutinin in complex with an inhibitor of membrane fusion. *Proc. Natl. Acad. Sci. U.S.A.* **105,** 17736–17741
8. McFarlane, A. A., Orriss, G. L., and Stetefeld, J. (2009) The use of coiled-coil proteins in drug delivery systems. *Eur. J. Pharmacol.* **625,** 101–107
9. O'Shea, E. K., Lumb, K. J., and Kim, P. S. (1993) Peptide 'Velcro': design of a heterodimeric coiled coil. *Curr. Biol.* **3,** 658–667
10. Lupas, A., Van Dyke, M., and Stock, J. (1991) Predicting coiled coils from protein sequences. *Science* **252,** 1162–1164
11. Berger, B., Wilson, D. B., Wolf, E., Tonchev, T., Milla, M., and Kim, P. S. (1995) Predicting coiled coils by use of pairwise residue correlations. *Proc. Natl. Acad. Sci. U.S.A.* **92,** 8259–8263
12. McDonnell, A. V., Jiang, T., Keating, A. E., and Berger, B. (2006) Paircoil2: improved prediction of coiled coils from sequence. *Bioinformatics* **22,** 356–358
13. Delorenzi, M., and Speed, T. (2002) An HMM model for coiled-coil domains and a comparison with PSSM-based predictions. *Bioinformatics* **18,** 617–625
14. Walshaw, J., and Woolfson, D. N. (2001) SOCKET: a program for identifying and analysing coiled coil motifs within protein structures. *J. Mol. Biol.* **307,** 1427–1450
15. Woolfson, D. N., and Alber, T. (1995) Predicting oligomerization states of coiled coils. *Protein Sci.* **4,** 1596–1607
16. Wolf, E., Kim, P. S., and Berger, B. (1997) MultiCoil: a program for predicting two- and three-stranded coiled coils. *Protein Sci.* **6,** 1179–1189
17. Gruber, M., Söding, J., and Lupas, A. N. (2006) Comparative analysis of coiled-coil prediction methods. *J. Struct. Biol.* **155,** 140–145
18. Lupas, A. N. (2008) The long coming of computational structural biology. *J. Struct. Biol.* **163,** 254–257
19. Portwich, M., Keller, S., Strauss, H. M., Mahrenholz, C. C., Kretzschmar, I., Kramer, A., and Volkmer, R. (2007) A network of coiled-coil associations derived from synthetic GCN4 leucine-zipper arrays. *Angew. Chem. Int. Ed. Engl.* **46,** 1654–1657
20. Gingras, A. R., Bate, N., Goult, B. T., Hazelwood, L., Canestrelli, I., Grossmann, J. G., Liu, H., Putz, N. S., Roberts, G. C., Volkmann, N., Hanein, D., Barsukov, I. L., and Critchley, D. R. (2008) The structure of the C-terminal actin-binding domain of talin. *EMBO J.* **27,** 458–469
21. Sheriff, S., Chang, C. Y., and Ezekowitz, R. A. (1994) Human mannose-binding protein carbohydrate recognition domain trimerizes through a triple alpha-helical coiled-coil. *Nat. Struct. Biol.* **1,** 789–794
22. Hu, J. C., Newell, N. E., Tidor, B., and Sauer, R. T. (1993) Probing the roles of residues at the e and g positions of the GCN4 leucine zipper by combinatorial mutagenesis. *Protein Sci.* **2,** 1072–1084
23. Kammerer, R. A., Kostrewa, D., Progias, P., Honnappa, S., Avila, D., Lustig, A., Winkler, F. K., Pieters, J., and Steinmetz, M. O. (2005) A conserved trimerization motif controls the topology of short coiled coils. *Proc. Natl. Acad. Sci. U.S.A.* **102,** 13891–13896
24. Ciani, B., Bjelic, S., Honnappa, S., Jawhari, H., Jaussi, R., Payapilly, A., Jowitt, T., Steinmetz, M. O., and Kammerer, R. A. (2010) Molecular basis of coiled-coil oligomerization-state specificity. *Proc. Natl. Acad. Sci. U.S.A.* **107,** 19850–19855
25. Bernstein, F. C., Koetzle, T. F., Williams, G. J., Meyer, E. F., Jr., Brice, M. D., Rodgers, J. R., Kennard, O., Shimanouchi, T., and Tasumi, M. (1977) The Protein Data Bank: a computer-based archival file for macromolecular structures. *J. Mol. Biol.* **112,** 535–542
26. Altschul, S. F., Gish, W., Miller, W., Myers, E. W., and Lipman, D. J. (1990) Basic local alignment search tool. *J. Mol. Biol.* **215,** 403–410
27. Henikoff, S., and Henikoff, J. G. (1992) Amino acid substitution matrices from protein blocks. *Proc. Natl. Acad. Sci. U.S.A.* **89,** 10915–10919
28. Fisher, R. A. (1922) On the interpretation of χ^2 from contingency tables, and the calculation of P. *J. Roy. Statist. Soc.* **85,** 87–94
29. Benjamini, Y., and Hochberg, Y. (1995) Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J. Roy. Statist. Soc. Ser. B* **57,** 289–300
30. Burges, C. J. C. (1998) A tutorial on support vector machines for pattern recognition. *Data. Min. Knowl. Discov.* **2,** 121–167
31. Müller, K.-R., Mika, S., Rätsch, G., Tsuda, K., and Schölkopf, B. (2001) An introduction to kernel-based learning algorithms. *IEEE Trans. Neural Networks* **12,** 181–201
32. Vert, J.-P., Tsuda, K., and Schölkopf, B. (2004) A primer on kernel methods, in *Kernel Methods in Computational Biology* (Schölkopf, B., Tsuda, K., and Vert, J.-P., eds) chapter 2, pp. 35–70, MIT Press, Cambridge, MA
33. Vapnik, V. N. (1998) *Statistical Learning Theory, Adaptive and Learning Systems*, Wiley Interscience, New York
34. Cortes, C., and Vapnik, V. N. (1986) Support-vector networks. *Machine Learning* **20,** 273–297
35. Christianini, N., and Shawe-Taylor, J. (2000) *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*, Cambridge University Press, Cambridge, MA
36. Chang, C.-C., and Lin, C.-J. (2001) *LIBSVM: a library for support vector machines.* Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm
37. Kuksa, P., Huang, P.-H., and Pavlovic, V. (2008) A fast, large-scale learning method for protein sequence classification, in *8th Int. Workshop on Data Mining in Bioinformatics*, pp. 29–37, Las Vegas, NV
38. Fong, J. H., Keating, A. E., and Singh, M. (2004) Predicting specificity in bZIP coiled-coil protein interactions. *Genome Biol.* **5,** R11
39. Bodenhofer, U., Schwarzbauer, K., Ionescu, M., and Hochreiter, S. (2009) Modeling position specificity in sequence kernels by fuzzy equivalence relations, in *Proc. Joint 13th IFSA World Congress and 6th EUSFLAT Conference* (Carvalho, J. P., Dubois, D., Kaymak, U., and Sousa, J. M. C., eds) pp. 1376–1381, Lisbon
40. Harbury, P. B., Zhang, T., Kim, P. S., and Alber, T. (1993) A switch between two-, three-, and four-stranded coiled coils in GCN4 leucine zipper mutants. *Science* **262,** 1401–1407
41. Schölkopf, B., Tsuda, K., and Vert, J.-P., eds (2004) *Kernel Methods in Computational Biology*, MIT Press, Cambridge, MA
42. Leslie, C., Eskin, E., and Noble, W. S. (2002) The spectrum kernel: a string kernel for SVM protein classification, in *Pacific Symp. on Biocomputing* 2002 (Altman, R. B., Dunker, A. K., Hunter, L., Lauderdale, K., and Klein, T. E. D., eds) pp. 566–575, World Scientific
43. Leslie, C., Eskin, E., Cohen, A., Weston, J., and Noble, W. S. (2004) Mismatch string kernels for discriminative protein classification. *Bioinformatics* **20,** 467–476
44. Luntz, A., and Brailovsky, V. (1969) On estimation of characters obtained in statistical procedure of recognition. *Technicheskaya Kibernetica* **3,** 563–575 (in Russian)
45. Hu, J. C., O'Shea, E. K., Kim, P. S., and Sauer, R. T. (1990) Sequence requirements for coiled-coils: analysis with λ repressor-GCN4 leucine zipper fusions. *Science* **250,** 1400–1403
46. Potekhin, S. A., Medvedkin, V. N., Kashparov, I. A., and Venyaminov, S. Yu (1994) Synthesis and properties of the peptide corresponding to the mutant form of the leucine zipper of the transcriptional activator GCN4 from yeast. *Protein Eng.* **7,** 1097–1101