High Accuracy Prediction of β-Turns and Their Types Using Propensities and Multiple Alignments

Patrick F.J. Fuchs[1]* **and Alain J.P. Alix**[2]
[1]*Equipe de Bioinformatique Génomique et Moléculaire (EBGM), INSERM EMI 0346, Université Paris 7, Paris, France*
[2]*Laboratoire de Spectroscopies et Structures BioMoléculaires (LSSBM), Université de Reims Champagne-Ardenne FR53 "Biomolécules," Faculté des Sciences, BP 1039, 51687 Reims Cedex 2, France*

ABSTRACT **We have developed a method that predicts both the presence and the type of β-turns, using a straightforward approach based on propensities and multiple alignments. The propensities were calculated classically, but the way to use them for prediction was completely new: starting from a tetrapeptide sequence on which one wants to evaluate the presence of a β-turn, the propensity for a given residue is modified by taking into account all the residues present in the multiple alignment at this position. The evaluation of a score is then done by weighting these propensities by the use of Position-specific score matrices generated by PSI-BLAST. The introduction of secondary structure information predicted by PSIPRED or SSPRO2 as well as taking into account the flanking residues around the tetrapeptide improved the accuracy greatly. This latter evaluated on a database of 426 reference proteins (previously used on other studies) by a sevenfold crossvalidation gave very good results with a Matthews Correlation Coefficient (MCC) of 0.42 and an overall prediction accuracy of 74.8%; this places our method among the best ones. A jackknife test was also done, which gave results within the same range. This shows that it is possible to reach neural networks accuracy with considerably less computional cost and complexity. Furthermore, propensities remain excellent descriptors of amino acid tendencies to belong to β-turns, which can be useful for peptide or protein engineering and design. For β-turn type prediction, we reached the best accuracy ever published in terms of MCC (except for the irregular type IV) in the range of 0.25–0.30 for types I, II, and I′ and 0.13–0.15 for types VIII, II′, and IV. To our knowledge, our method is the only one available on the Web that predicts types I′ and II′. The accuracy evaluated on two larger databases of 547 and 823 proteins was not improved significantly. All of this was implemented into a Web server called COUDES (French acronym for: Chercher Où Une Déviation Existe Sûrement), which is available at the following URL: http://bioserv.rpbs.jussieu.fr/Coudes/index.html within the new bioinformatics platform RPBS. Proteins 2005;59:828–839.**

Key words: β-turns; prediction; β-turn type prediction; propensities; multiple alignments COUDES

INTRODUCTION

β-Turns are small elements of secondary structure that play an important role both in peptides and proteins. They are made of four residues (denoted i to $i + 3$) with a distance between Cα(s) of residues i and $i + 3$ that has to be smaller than 7 Å. They are part of a more general category known as tight turns that also contains γ- and α-turns (constituting of three and five residues, respectively);[1] in any case, they remain the most numerous category of tight turns and represent about 25% of all residues in proteins The restrictive distance of 7 Å implies a particular geometry to the backbone, which can therefore turn back on itself or more generally change direction. Therefore, they are usually described as an orienting structure because they orient α-helices and β-strands, thus, playing a critical role in the topology of proteins. So as not to confuse with α-helices (which can obviously be considered as a succession of turns), the two central residues of β-turns ($i + 1$ and $i + 2$) must not be helical. They can sometimes be stabilized by a hydrogen bond between the N—H of residue i and the C═O of residue $i + 3$, and if they are not, one talks about an open β-turn. Because there are many possibilities to satisfy these simple criteria, β-turns are classified into types according to the φ/ϕ values of the central residues $i + 1$ and $i + 2$. Up to now, the accepted nomenclature is that described by Hutchinson and Thornton[2] (cf Table I), which contains types I, II, VIII, I′, II′, VIa, VIb, and IV. A maximum deviation of ±30° from these canonical values is allowed on three of these angles, while the fourth can deviate of ±45°. If this criterion is not respected, the β-turn is classified as type IV. It has to be noted that type VI contains a *cis* Pro at position $i + 2$.[3]

β-Turns play many biological roles in proteins and peptides.[4] Within proteins they tend to be more solvent exposed than buried into the hydrophobic core; thus, they are usually involved in molecular recognition as an acces-

The Supplementary Materials referred to in this article can be found at http://www.interscience.wiley.com/jpages/0887-3585/suppmat

*Correspondence to: Patrick F.J. Fuchs, Equipe de Bioinformatique Génomique et Moléculaire (EBGM), INSERM EMI 0346, Université Paris 7, Case Courrier 7113, 2 place Jussieu, 75251 Paris Cedex 05, France. E-mail: Patrick.Fuchs@ebgm.jussieu.fr, alain.alix@univ-reims.fr

Received 30 July 2004; Accepted 16 December 2004

Published online 8 April 2005 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/prot.20461

TABLE I. List of the Dihedral Angle Values for the Common Types of β-Turns

Type	φ_{i+1}	ψ_{i+1}	φ_{i+2}	ψ_{i+2}	Remark
Type I	−60	−30	−90	0	
Type II	−60	120	80	0	
Type VIII	−60	−30	−120	120	
Type I′	60	30	90	0	
Type II′	60	120	−80	0	
Type VIa1	−60	120	−90	0	*i + 2* is a *cis* Pro
Type VIa2	−120	120	−60	0	*i + 2* is a *cis* Pro
Type VIb	−135	135	−75	160	*i + 2* is a *cis* Pro
Type IV			all other values		

sible site. Furthermore, the predominance of functional side chains (i.e., Asp, Ser, Thr, Lys, etc.) favors this recognition role. For instance, in bent linear epitopes, β-turns can allow the side chains of their two central residues (*i* + 1 and *i* + 2) to be exposed against an antibody. Moreover, this tendency to be solvent exposed gives them more flexibility, which is critical in proteins. This latter property is also enhanced by their intrinsic facility of formation and deformation. β-Turns are also largely involved in the biological activity of peptides, often being the bioactive structure that interacts with another molecule (e.g., a receptor, an enzyme, an antibody, etc.) within a conformational population. These last years there has been great interest in mimicking β-turns for the synthesis of medicines in the field of medical and pharmacological chemistry because of their important role in molecular recognition.[5,6] Therefore, it is particularly useful to know whether or not one tetrapeptidic sequence may form a β-turn, and which type(s) is(are) likely to occur. This first step linked to a molecular modeling study using techniques such as molecular dynamics or Monte Carlo simulations,[8] can save time and money by avoiding useless chemical synthesis. Last, some studies have demonstrated that β-turns might be involved in protein folding.[4] For example, some authors proposed a zip-up mechanism for the folding of a β-hairpin based on experimental[7] or numerical simulation[8] data: it consists of forming first the β-turn and then to expand hydrogen bonds to the termini of the hairpin. On such a mechanism, the β-turn plays a critical role on the initiation of the folding process.

Because of the biological importance of β-turns, and as the number of solved structures is one step behind the number of known sequences, there has been a great deal of interest in predicting their presence in proteins. We can briefly divide all the published methods into two categories: those based on pure statistical methods that came out first, and those based on neural networks, which appeared later. An early attempt was made by Lewis,[9] who discovered the high predictability of β-turns based on simple probabilities observed in 3 proteins. Later, Chou and Fasman proposed some stronger methods based on propensities,[10–13] or Garnier et al.[14] on the information theory, within the prediction of other classical states[10,11,14] (i.e., helix, strand, and random coil), or not.[12,13] Rose[15] also tried a method based on a hydrophobicity scale, starting from the idea that β-turns tend to be solvent exposed. Wilmot and Thornton[16,17] applied propensities to types I and II prediction (because the other types the databases were too small to derive significant propensities). In 1994, Hutchinson and Thornton recalculated propensities on a larger database,[2] and were able to derive fairly significant propensities for the additional types VIII, I′, and II′. Recently, those propensities have been recalculated on a database of 426 proteins,[18] and some of them were refined (especially those of either rare residues or rare types). The second generation of statistical methods were based on conditional probabilities (using Markov chains of the first order) with the 1-4 and 2-3 residue–correlation model,[19] or the residues coupled model[20] applied to β-turns as well as for types.[21] On the other hand, neural netwoks have been used, with a first method published by McGregor et al.[22] in the late 1980s. In 1999, Shepherd et al.[23] added secondary structure information predicted by PHD[24] to their neural networks and made available a Web server called BTPRED. Their work compared to that of McGregor[22] showed that the addition of secondary structure information dramatically improved the accuracy. To compare the main methods without any bias, Kaur and Raghava[25] evaluated them on the same data set and using a secondary structure information (for old statistical techniques, the idea was to predict turns within regions predicted as random coil by PROF[26]). They showed that neural networks present a better accuracy than statistical methods, and that BTPRED reached the best score with a Matthews Correlation Coefficient (MCC) of 0.39. Since this work, two new methods have been published that used techniques shown to be efficient for "classical" secondary structure prediction: the first one combined neural networks and multiple alignments,[27] and reached an MCC of 0.43, while the second used the nearest-neighbor technique[28] and presented an MCC of 0.40. Both used a secondary structure predicted by PSIPRED.[29]

In 2003, another work based on support vector machines was published;[30] however, the authors neither used the same data set nor the same tools for evaluating their method (especially the MCC); thus, it is impossible to compare it with the previous ones. Very recently, Kaur and Raghava[31] used the same approach as they did in 2003 (secondary structures from PSIPRED and multiple alignments), but this time for type prediction. They significantly improved the accuracy compared to Shepherd's work.

In the present work, we wanted to address that it was possible to reach a very high accuracy, as good as those of neural networks techniques, using a simple statistical approach based on propensities and multiple alignments. Furthermore, we have incorporated in our method the prediction of all types (except type VI, which is very poorly populated and thus very hard to predict); we have developed concomitantly a Web server called COUDES, which performs the prediction from a protein sequence and which is the only available platform predicting all types (except type VI). Furthermore, we have evaluated the influence of many factors such as the database size, the method of β-turns extraction, the method of secondary structure prediction, and the use of multiple alignments and the specific parameters to our method (window size and various thresholds). The significance of the parameters used to assess the accuracy (rate of good prediction and Matthews correlation coefficient) has also been tested. Last, our work has been compared to the other prediction methods for turn/nonturn prediction as well as for type prediction.

MATERIALS AND METHODS

Databases

To compare with other works, we used a database of 426 chains of protein (95,289 residues) that has been widely used in β-turn literature.[18,25,27,28,31] This database contains chains solved by X-ray crystallography with a resolution better than 2.0 Å, and is nonredundant at 25%. To test the effect of the database size, we constructed two other ones of 547 and 823 chains (for a total number of residues of 104,522 and 150,969, respectively), with crystal structures presenting a resolution better than 2.0 Å and containing at least one β-turn. These two additional databases were constructed with the pdbselect[32,33] of June 2000 and October 2003 (available at http://homepages.fh-giessen.de/~hg12640/pdbselect/ or on our Web site). It should be noted that each chain of these three databases presents at least one beta-turn (all those that did not contain any β-turn at all were removed).

Extraction of β-Turns

We tested two ways of extracting β-turns: (1) using PROMOTIF[34] via the pdbsum Web server (http://www.biochem.ucl.ac.uk/bsm/pdbsum/); (2) using an in-house program called *extract_turn* that we made available on the Web at the following URL: http://bioserv.rpbs.jussieu.fr/RPBS/cgi-bin/Ressource.cgi?chzn_lg=an&chzn_rsrc=extract-Turn. *extract_turn* extracts β-turns with the same criteria as described by Hutchinson[2] using secondary structures generated by DSSP:[35] the distance between the Cα of residues i and $i + 3$ has to be smaller than 7 Å, and the central residues $i + 1$ and/or $i + 2$ must not be helical. We also excluded the tetrapeptides totally assigned as extended (EEEE) by DSSP as proposed previously.[2] Types were determined according to the φ/ψ dihedral angle values of the two central residues ($i + 1$ and $i + 2$) compared to the canonical values (see Table I); a maximum deviation of ±30° was allowed on three of these angles, while the fourth could deviate by ±45°. Because COUDES does not predict types VI, they were not extracted explicitly but classified as types IV. All tetrapepides with exotic residues were not taken into account.

With three databases and two methods of β-turn extraction, we finally got six data sets of propensities that we called EXTR426, PROM426, EXTR547, PROM547, EXTR823, and PROM823 (the first four letters represent the method of extraction and the last three numbers represent the database size in number of chains). The list of chains of each database is available on our Web site.

Calculation of Propensities

Propensities[10] applied to β-turns were calculated, as described previously.[2]

β-Turn positional propensity:

$$P_j^{\,i} = (n_j^i/n_j)/(N^i/N) \tag{1}$$

where n_j^i is the number of residues j (j = 1 to 20) located at position i ($i = -m$ to $m + 3$, where m is the number of flanking residues, see next section) of a β-turn, n_j the number of residues j in the database, N^i the number of residues located at position i of a β-turn, and N the number of residues in the database.

β-Turn positional propensity for type k:

$$P_j^{\,i}(k)=(n_j^{\,i}(k)/n_j)/(N^i(k)/N) \tag{2}$$

where $n_j^i(k)$ is the number of residues j located at position i of a β-turn of type k, $N^i(k)$ the number of residues located at position i of a β-turn of type k.

Non β-turn propensity:

$$\overline{P_j^{\,i}} = (\overline{n_j^{\,i}}/n_j)/(\overline{N^i}/N) \tag{3}$$

where $\overline{n_j^i}$ is the number of residues j located at position i of a non β-turn, $\overline{N^i}$ the number of residues located at position i of a non-β-turn.

These propensities describe the tendency of a residue to be part of a β-turn (or non-β-turn) and vary around 1; for example, for P_j^i, a value higher than 1 means that the residue is more likely to be at position i of a β-turn, and vice versa, while a value around 1 means it is indifferent.

Evaluation of a Score for Prediction

After some preliminary tests, we found that it was better to divide in two steps the prediction of the presence of β-turns and then their type, as observed before.[23] Some first attempts were done using a simple product of each individual propensity on a given window, made of a central tetrapeptide (residues i to $i + 3$) with m flanking residues on the left ($i - m$ to $i - 1$) and on the right ($i + 4$ to $i + 3 + m$). For example, if one considers $m = 4$ flanking residues on each side of the central tetrapeptide, one obtains: $i - 4$, $i - 3$, $i - 2$, $i - 1$, $\boldsymbol{i, i + 1, i + 2, i + 3}$, $i + 4$, $i + 5$, $i + 6$, $i + 7$. Hence, the score to be in turn and the one to be in nonturn were calculated this way:

$$S = \prod_{j=i-m}^{j=i+3+m} P_k^{\,j} \tag{4}$$

$$\overline{S} = \prod_{j=i-m}^{j=i+3+m} \overline{P_k^{\,j}} \tag{5}$$

where k is the residue present at position j (in or around a β-turn), m the number of flanking residues around the central tetrapeptide, P_k^j the propensity of residue k at position j (in or around a β-turn), and $\overline{P_k^j}$ the propensity of residue k at position j (in or around a non-β-turn).

The prediction of a β-turn at the central position of the window was then done by this straightforward rule if these two conditions were true: $S > \overline{S}$ and $S > S_{\text{threshold}}$. If they were not, the tetrapeptide at the central position of the window was predicted as a non-β-turn. The value of

$S_{\mathrm{threshold}}$ was optimized to get the best accuracy. Different sizes of window from $m = -1$ (window of two residues) to $m = 8$ (window of 20 residues) were also tested.

Use of Position-Specific Score Matrices for Weighting Propensities

Position-specific score matrices (PSSMs) were generated using PSI-BLAST[36] with three rounds against the last up-dated NR (nonredundant) database and an E-value threshold for multipass model of 10^{-3}. These are the same parameters as those used in PSIPRED;[29] therefore, in a single PSI-BLAST run, we could generate the PSSM for both COUDES and PSIPRED (if this latter was used, see next section). This PSSM is a matrix of $20 \times M$ elements, where M is the number of residues of the query sequence. The basic idea with these PSSMs was to take into account all the residues present at a given position in the multiple alignment instead of taking only that of the query sequence. This is indeed a decisive information to take into consideration whether a residue is conserved or occurs by chance, and which other ones are likely to occur. PSSMs contain this information for each residue at each position of the alignment in the form of a log-likelihood time a factor, that is, $[\ln(Q_k/P_k)]/\lambda_u$, where Q_k is the estimated probability for residue k to be found at this position of the alignment, P_k is the probability of occurrence for residue k in the database, and λ_u is a scaling factor;[36] let us call this log-likelihood ψ_k^i for a residue k at position i. Because the presence of a residue is reflected in the ψ_k^i value, our idea was to use it for weighting propensities. So, in a given window, a score by residue is evaluated as the mean of each residue presenting ψ_k^i higher than a given threshold ($\psi_{\mathrm{threshold}}$) weighted by the corresponding ψ_k^i. To extend the score to the whole window, we just took the simple product of the individual scores corresponding to each residue of that window:

$$S = \prod_{j=i-m}^{j=i+3+m} \left[\frac{\sum\limits_{\text{all residue k with } \psi_k > \psi \text{ threshold}} P_k^j \times \psi_k^j}{\sum\limits_{\text{all residue k with } \psi_k > \psi \text{ threshold}} \psi_k^j} \right] \tag{6}$$

where j is the position in the window ($i - m$ to $i + 3 + m$), m the number of flanking residues, P_k^j the propensity of residue k at position j of the window, and ψ_k^j the PSI-BLAST log-likelihood of residue k at position j of the window.

We also noticed that some residues with negative ψ_k^i values (but quite close to 0) could improve the accuracy [in other words, residues that may occur with a smaller evaluated probability (Q_k) than that in the database (P_k)]. To use these residues (when $\psi_{\mathrm{threshold}} \leq 0$), we needed to avoid a division by zero, so in this case we just shifted the weights by $\psi_{\mathrm{threshold}}$:

$$S = \prod_{j=i-m}^{j=i+3+m} \left[\frac{\sum\limits_{\text{all residue k with } \psi_k > \psi \text{ threshold}} P_k^j \times (\psi_k^j + |\psi_{\mathrm{threshold}}|)}{\sum\limits_{\text{all residue k with } \psi_k > \psi \text{ threshold}} (\psi_k^j + |\psi_{\mathrm{threshold}}|)} \right] \tag{7}$$

To sum up, we used Equation (6) for the case $\psi_{\mathrm{threshold}} > 0$ and Equation (7) for $\psi_{\mathrm{threshold}} \leq 0$. For both cases a score was evaluated for β-turns and non-β-turns. We then applied the same rules as described in the previous section, that is, $S > \bar{S}$ and $S > S_{\mathrm{threshold}}$. We optimized $S_{\mathrm{threshold}}$, $\psi_{\mathrm{threshold}}$ and m to get the best accuracy.

Use of Secondary Structure Information

We tested the effect of using a secondary structure information to improve the accuracy of our method. We used some of the best methods such as PSIPRED[29] already utilized in recent articles of β-turn literature,[27,28,31] SSpro2,[37] and PROF[26] (this latter has already been used for statistical β-turn prediction methods[25]). Any tetrapeptide containing helical or extended residues on either one or at both central positions ($i + 1$ and/or $i + 2$), was considered as a non-β-turn. To assess an upper and lower bound reachable by our method, we also tested with observed secondary structure (defined by DSSP[35]) or without any secondary structure information.

Trivial Strategies

Two trivial strategies were also tested in this article. The first one was by predicting every residue not helical or extended as a β-turn, either with or without secondary structure information (with no secondary structure information, we then predict all the residues as a β-turn); this strategy was called "all." The second one was by predicting randomly β-turns according to their frequency in proteins either with or without secondary structure information; this strategy was called "random."

Prediction of Types

β-Turn types were predicted more simply using only propensities and no multiple alignments. For each tetrapeptide predicted as a β-turn, a score for each type was calculated using Equation (4) with β-turn types propensities. We applied then the simple strategy "winner takes all," that is, the predicted type was the one with the highest score.

Evaluation of the Method

To assess the accuracy of our method, we used the classical tools of β-turn literature.[23,25,27,28,31]

The percentage of correct prediction (Q_{total}) for each type of β-turn has been calculated as follows:

$$Q_{\mathrm{total}} = \frac{a + b}{\mathrm{a} + b + c + d} \times 100 \tag{8}$$

where a is the number of residues observed and predicted as turn (true positive), b the number of residues not observed and not predicted as turn (true negative), c the number of residues observed but not predicted as turn (false negative, underprediction), and d the number of residues not observed but predicted as turn (false positive, overprediction).

To describe the accuracy, Q_{total} tends to raise an overestimation of predictive performances.[23,38] This is especially evident when true negative are numerous, false negative, and positive are really "drowned," giving a highly misleading impression of good performances via the Q_{total} (e.g., in

TABLE II. Percentage and Number of β-Turns Extracted from the Different Databases[a]

	EXTR426		PROM426		EXTR547		PROM547	
N_{total}[b]	95,278		95,289		104,429		104,522	
	N	%	N	%	N	%	N	%
Type I	2974	37.5	2452	34.3	3253	37.0	2697	34.1
Type II	968	12.2	920	12.9	1080	12.3	1020	12.9
Type VIII	847	10.7	666	9.3	968	11.0	758	9.6
Type I′	318	4.0	302	4.2	373	4.3	350	4.4
Type II′	182	2.3	166	2.3	205	2.3	186	2.3
Type IV	2644	33.3	2649	37.0	2913	33.1	2901	36.7
Total	7933	100.0	7155	100.0	8792	100.0	7912	100.0
$N_{\beta\text{-turn}}$[c]	25,411	26.7	23,222	24.4	28,081	26.7	25,577	24.5

	EXTR823		PROM823	
N_{total}[b]	150,733		150,969	
	N	%	N	%
Type I	4574	37.4	3889	34.5
Type II	1507	12.3	1436	12.8
Type VIII	1261	10.3	991	8.8
Type I′	533	4.4	518	4.6
Type II′	294	2.4	276	2.5
Type IV	4055	33.2	4147	36.8
Total	12,224	100.0	11,257	100.0
$N_{\beta\text{-turn}}$[c]	39,561	26.0	36,263	24.0

[a] EXTR426, EXTR547, EXTR823 are data sets with turns extracted using extract_turn, and PROM426, PROM547, PROM823 using PROMOTIF.[34]

[b] Total number of residues within the database.

[c] Total number and percentage of residues in β-turns within the database.

the case of sparsely populated types such as I′ and II′). Thus, it is better to use the Matthews Correlation Coefficient (MCC),[39] which takes more into account both under- and overprediction and gives a better idea of the accuracy:

$$MCC = \frac{ab - cd}{\sqrt{(a+c)(a+d)(b+c)(b+d)}} \qquad (9)$$

To be evaluated separately, underprediction can be assessed using Q_{obs}, which is the fraction of observed β-turns that are correctly predicted:

$$Q_{\mathrm{obs}} = \frac{a}{a+c} \times 100 \qquad (10)$$

Overprediction can be assessed using Q_{pred}, which is the fraction of predicted β-turns that are correct:

$$Q_{\mathrm{pred}} = \frac{a}{a+d} \times 100 \qquad (11)$$

Sevenfold Crossvalidation and Jackknife Test

Two ways have been used to assess the accuracy. The first was with a sevenfold crossvalidation: each database was divided into seven subsets, six of them used for calculating propensities and the seventh one for validation. The process was repeated seven times, to have a different subset for validation each time. For the database of 426 proteins, we used the same subsets as Kaur and Raghava.[25] For the other databases, the partition into seven subsets was done to get a right balance in the number of β-turns and residues. The advantage of crossvalidation is that it gives an idea of the standard deviation of the parameters MCC, Q_{total}, Q_{obs}, and Q_{pred}. Thus, each time a crossvalidation is used in this study, the standard deviation is also given.

The second way of assessing the accuracy was with a jackknife test which, in theory, is the most precise one. It consists in taking one chain of protein for validation and all the others (i.e., $n - 1$; n being the number of proteins) for calculating propensities (more generally for the learning phase); this process is then repeated n times with a different chain each time.

RESULTS AND DISCUSSION

Distribution of β-Turns and Updated Propensities

In Table II the numbers and percentages of β-turns are reported, as well as their types, extracted from the six data sets (EXTR426, PROM426, EXTR547, PROM547, EXTR823, and PROM823). If we focus on data sets elaborated with one specific method of β-turn extraction, the global percentage of residues in β-turn (around 26% for *extract_turn* and 24% for PROMOTIF) and the percentage for each type show a strong stability from one database to another, with differences up to 0.2%. It shows that our three databases are nonredundant, and any of them is suitable for prediction. However, if we now compare the results generated with the two methods of β-turns extraction, the differences are significant. In fact, *extract_turn*

TABLE III. Effect of PSSM and Different Methods of Secondary Structure Prediction on the Accuracy

Secondary structure method	Use of PSSM	MCC	Q_{total} (%)	Q_{obs} (%)	Q_{pred} (%)
No SS	No	0.289 ± 0.014	64.4 ± 0.7	70.3 ± 1.7	40.4 ± 0.7
	Yes	0.361 ± 0.010	66.9 ± 0.8	77.8 ± 1.4	43.3 ± 1.2
PSIPRED	No	0.377 ± 0.011	73.1 ± 0.5	64.2 ± 2.2	49.7 ± 0.6
	Yes	0.410 ± 0.007	73.3 ± 0.4	70.7 ± 1.8	50.0 ± 1.0
SSpro2	No	0.369 ± 0.012	73.0 ± 0.5	62.6 ± 2.0	49.6 ± 0.4
	Yes	0.406 ± 0.009	72.8 ± 0.5	71.3 ± 1.7	49.3 ± 0.8
PROF	No	0.338 ± 0.011	72.2 ± 0.4	58.5 ± 1.7	48.3 ± 1.1
	Yes	0.381 ± 0.010	72.5 ± 0.6	67.0 ± 1.4	48.9 ± 1.2
DSSP	No	0.466 ± 0.010	78.5 ± 0.6	64.0 ± 1.4	59.0 ± 0.5
	Yes	0.513 ± 0.009	79.3 ± 0.6	72.8 ± 1.4	59.2 ± 0.7

[a]These runs have been tested on the data set EXTR426 with $m = 4$, $S_{\text{threshold}} = 1.5$ and $\psi_{\text{threshold}} = -125$.

assigns more β-turns than PROMOTIF (about 2% more), and the type of these extra β-turns is mainly type I. This may be explained by two main causes. First, *extract_turn* uses DSSP for secondary structure assignments while PROMOTIF uses a slightly modified algorithm (but still based on DSSP) that tries to extend strands and helices when possible.[34] This has important consequences, especially for helices, because a β-turn is defined with nonhelical central residues. Second, PROMOTIF may use the helices defined by the crystallographers as they did before.[2] Now, why are these extra β-turns type I? Because such a type is the closest one to the helix conformation (especially for residue $i + 1$ with $\varphi = -60°$ and $\psi = -30°$). Therefore, if a fragment of polypeptide is not assigned as helix by DSSP, but it is by PROMOTIF, it is very likely that *extract_turn* assigns it to type I. For the other remaining types, the differences are very small, and also come from the secondary structure assignment variety, but to a lesser extent because these other types possess different dihedral angles from the α-helix. This shows how important the way of assigning a secondary structure is, especially for prediction approaches, as recently highlighted.[40] We discuss the effect of β-turns extraction on prediction accuracy in a following section (see Influence of the Database Size and of the Method of β-Turn Extraction on the Accuracy).

All the calculated propensities are available on our Web site (http://bioserv.rpbs.jussieu.fr/Coudes/index.html). For nearly all propensities, we retrieve the tendencies observed in previous works.[2,18] Only rare residues (especially His, Trp, and Met) and rare types are really refined.

Influence of Secondary Structures and PSSMs on Prediction Accuracy

Among the many parameters, we tested first the effect of secondary structure and PSSMs on the first propensities data set we built: EXTR426 (the database of 426 proteins with β-turns extracted using *extract_turn*). It should be noted that the values of m, $S_{\text{threshold}}$, and $\psi_{\text{threshold}}$ we used here were optimized to give the best results with PSIPRED (see Influence of the Parameters m, $S_{\text{threshold}}$, and $\psi_{\text{threshold}}$).

Before going on to the analysis we give an idea of the accuracy of the secondary structure methods used in this study. For that, we evaluated the $Q3$ (percentage of correct prediction in three states: Helix, Extended, and Coil) on the EXTR426 data set (95,278 residues) compared to the observed secondary structures assigned by DSSP (this $Q3$ might be different compared to the original methods because we used a different NR database, different alignments, etc.). We obtained the following $Q3$: 81.8% for PSIPRED, 81.8% for SSpro2, and 75.2% for PROF.

In Table III the effect of using secondary structures and PSSMs on prediction accuracy is presented. It is clear that the use of PSI-BLAST PSSMs greatly improves the accuracy of prediction whatever the secondary structure method. This effect is particularly high when no secondary structure information is used as the MCC evolves from 0.289 to 0.361 and the Q_{total} from 64.4 to 66.9%. When secondary structures are predicted by one of the three methods, the positive effect is large on MCC but no longer on Q_{total}; in this case, we see that PSSMs improve the underprediction with higher Q_{obs} but the overprediction is not affected, with values of Q_{pred} almost equal. It is one first example showing that MCC seems to be more affected by underprediction while Q_{total} is by overprediction, and also Q_{obs} and Q_{pred}, respectively (some other examples will be met on this study). The positive effect of PSSMs (more exactly multiple alignments) has already been shown for a long time for classical secondary structure[29,37,41] and database growth is expected to improve more and more prediction accuracy;[42] we see here that it works as well on β-turns (or more generally on tight turns) as shown in a recent study.[27] The reason for that is obvious: the information contained in a multiple alignment tells us whether a residue is conserved in the related known sequences. This decisive knowledge allows one to give more or less weight on the information represented by the residue itself (or to those appearing at the same position). Almost all the methods use this information as an entry of a neural network; interestingly, we see here that it is also suitable in a straightforward way, just by weighting propensities.

In Table III, we also see that the use of secondary structure information has a dramatic effect on the accuracy; for example, we pass from an MCC of 0.361 to 0.410 and from a Q_{total} of 66.9 to 73.3% by using PSIPRED. This time, the improvement comes from a better behavior of underprediction as Q_{pred} is increased (e.g., from 43.3 to 50.0% with PSIPRED). On the other hand, Q_{obs} is decreased

TABLE IV. Results on Trivial Strategies[a]

Method	MCC	Q_{total} (%)	Q_{obs} (%)	Q_{pred} (%)
"all" (PSIPRED)	0.376 ± 0.007	62.8 ± 1.0	88.5 ± 1.2	40.9 ± 1.1
"all" (noSS)	0.002 ± 0.010[b]	26.7 ± 1.0	100.0 ± 1.0	26.7 ± 1.0
"random" (PSIPRED)	0.147 ± 0.012	72.1 ± 0.6	19.3 ± 0.8	44.7 ± 1.8
"random" (noSS)	0.000 ± 0.010	60.8 ± 0.6	26.7 ± 0.8	26.7 ± 1.3

[a] These runs have been tested on the data set EXTR426.
[b] To avoid a division by 0, one arbitrary residue has been set to nonturn.

(from 77.8 to 70.7%). The explanation for this is quite simple: when we use secondary structure information, we no longer predict β-turns inside helices or strands if and only if these latter are correctly predicted. In this case, we predict fewer turns that do not exist, thus decreasing overprediction. However, if these helices and strands are wrongly predicted, we can sometimes miss some β-turns that exist. In this case, underprediction increases. In any case, we see that the effect is globally very favorable as observed in other studies.[25,27,28] Among the tested methods, PSIPRED gives the best results on turn/nonturn prediction with an MCC of 0.410; SSpro2 does not give as good results, but even so, in the same range, according to the standard deviation, with an MCC of 0.406. These two methods give significantly better results than PROF, which raises an MCC of 0.381, which can be explained by their respective $Q3$ on our database. Using observed secondary structures by DSSP, we get the upper bound our method can reach with an MCC of 0.513 and a Q_{total} of 79.3%.

Last, it is worth noting that the way of using secondary structure has an important effect on the accuracy. Until now, every tetrapeptide containing helical or extended residues (predicted by PSIPRED) at any of the two central positions ($i + 1$ and/or $i + 2$) of the turn was predicted as a non-β-turn. Some tries were also done by applying the same rule but on any of the four positions (i, $i + 1$, $i + 2$, and or $i + 3$), which gave less good results (data not shown). This decrease is due to the fact that residues i and $i + 3$ of β-turns usually overlap helices or strands in proteins. In the light of these results, this simple rule may increase the accuracy of methods that exclude all helical or extended residues on the four residues like Refs. 25 and 28.

Results on Trivial Strategies

To get an idea whether our method gives better results than trivial strategies, we present in Table IV the accuracy by predicting every residue as β-turn or randomly according to their frequency in proteins. It allowed us to clarify the meaning of MCC, Q_{total}, Q_{obs}, and Q_{pred}, that is, what were the set of minimal values that any method should overcome to be better than random, as a kind of lower bound. With no secondary structure, we get in either case an MCC around 0 compared to 0.361 with COUDES. With PSIPRED, the "all" strategy seems to work well in terms of MCC with a value of 0.376; however, the Q_{total} is rather poor with 62.8%. For the "random" strategy it is the contrary: we get a rather good Q_{total} (72.1%) but a very bad MCC of 0.147. This shows that it is important to evaluate prediction with both measures not to get a misleading idea of good prediction quality. Both MCC and Q_{total} have to be maximized to get the best prediction of a method (and, consequently, Q_{obs} and Q_{pred} as well). In any case, COUDES outperforms by far these two trivial strategies.

Because it gives the best results, we will always use from now to the end of this article predicted secondary structure by PSIPRED, PSSMs from PSI-BLAST, as well as tetrapeptides excluded if one of the two central residues is either helical or extended.

Influence of the Database Size and of the Method of β-Turn Extraction

In Table Va the results on the different data sets of propensities measured by a sevenfold crossvalidation are presented; it should be noted that the parameters were chosen to get the best MCC/Q_{total} balance (see next section). COUDES presents a very good accuracy with an MCC up to 0.42 and a Q_{total} up to almost 75%. The differences between results obtained with data sets PROM* and EXTR* are within one standard deviation; thus, they are not significant even if they seem better in the case of PROMOTIF. Concerning the database, increasing its size to 547 or 823 chains has no significant effect, and the small differences might be due to a random fluctuation. It shows that it is important to use the same database to compare two different methods (see section Comparison of the Accuracy with Other Methods) if one wants to get rid of this fluctuation as suggested in Ref. 25.

In Table Vb, the accuracy measured with a jackknife test (using the same parameters) is given. In each case, we get a value extremely close to that obtained with crossvalidation, which shows that crossvalidation gives a good estimate of prediction accuracy.

To resume, COUDES presents a strong stability whatever the database size, the method of turn extraction, or the way to evaluate performances. Furthermore, it reaches a very good accuracy with an MCC of 0.42 and a Q_{total} of almost 75% with the data set PROM547. It places COUDES among the best methods (see section Comparison of the Accuracy with Other Methods).

Influence of the Parameters m, $S_{\text{threshold}}$, and $\psi_{\text{threshold}}$

In Figures 1 and 2 the variation of MCC and Q_{total} versus the window size (m), the threshold score ($S_{\text{threshold}}$), and the threshold value used in PSSMs ($\psi_{\text{threshold}}$) tested on the data set PROM426 (because it was the one used in recent studies) are presented. In practise, all the values were tested ranging from −1 to 10 for m, 0.0 to 3.0 for $S_{\text{threshold}}$, and −200 to +200 for $\psi_{\text{threshold}}$. For clarity, only

TABLE V. Effect of the Database Size and the Method of β-Turn Extraction on the Accuracy Measured by Crossvalidation (a) or Jackknife Test (b)[a]

Data set	MCC	Q_{total} (%)	Q_{obs} (%)	Q_{pred} (%)
(a)				
EXTR426	0.406 ± 0.008	74.5 ± 0.5	65.7 ± 1.8	51.8 ± 1.2
EXTR547	0.408 ± 0.020	74.3 ± 0.9	66.6 ± 2.3	51.8 ± 1.3
EXTR823	0.386 ± 0.019	73.6 ± 0.4	65.3 ± 2.4	49.4 ± 2.0
PROM426	0.417 ± 0.006	74.8 ± 0.6	69.7 ± 1.5	49.0 ± 1.6
PROM547	0.417 ± 0.018	74.6 ± 0.8	70.4 ± 2.3	48.7 ± 1.8
PROM823	0.405 ± 0.017	74.2 ± 0.2	69.6 ± 2.3	47.5 ± 1.8
(b)				
EXTR426	0.402	74.3	65.5	51.5
EXTR547	0.409	74.4	66.7	51.8
EXTR823	0.387	73.6	65.4	49.4
PROM426	0.416	74.7	69.8	48.8
PROM547	0.419	74.7	70.5	48.9
PROM823	0.404	74.2	69.5	47.4

[a] All runs were done with the following parameters: $m = 4$, $S_{threshold} = 1.8$ and $\psi_{cutoff} = 0$, which gave the best MCC/Q_{total} balance.

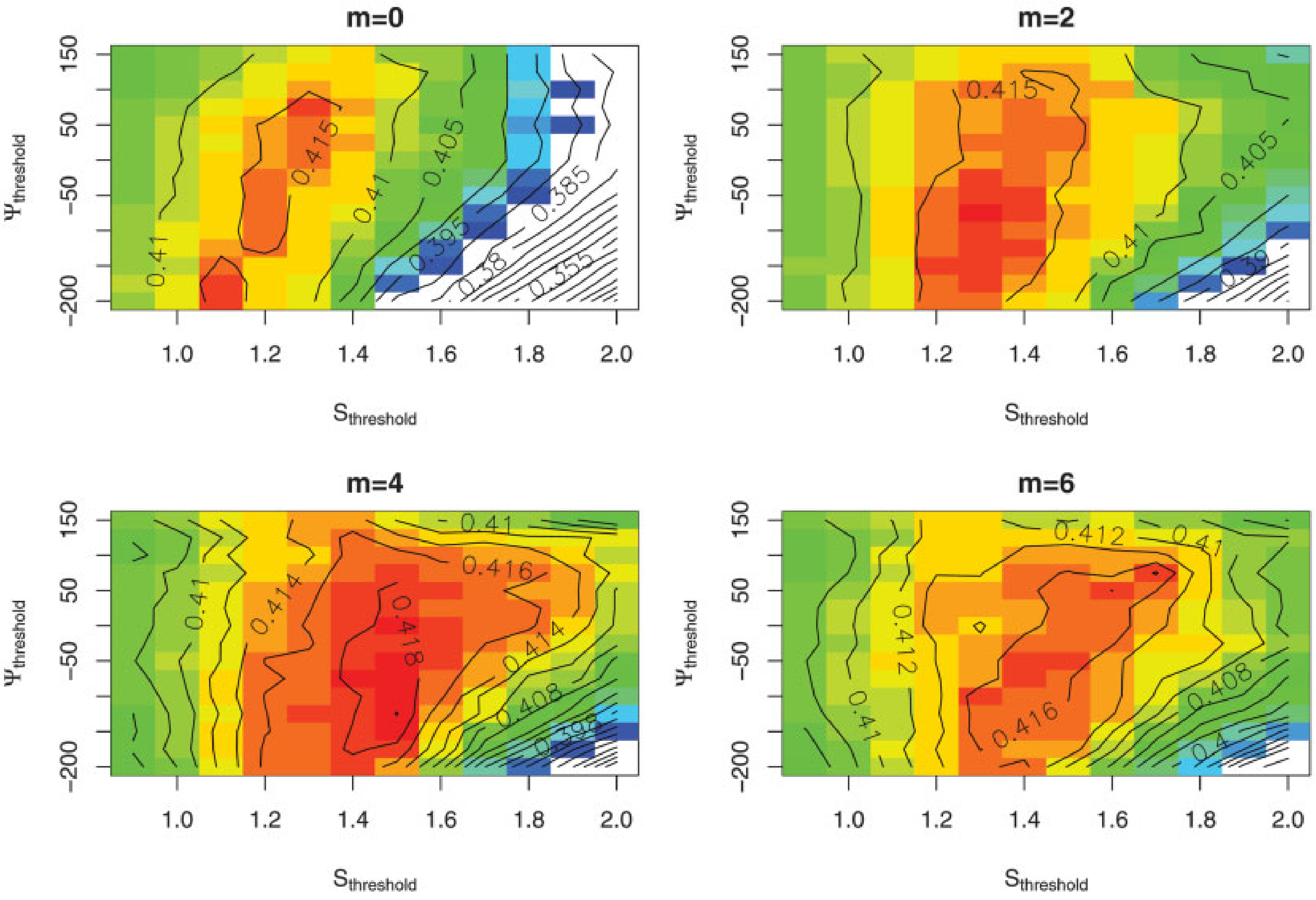

Fig. 1. Effect of the different parameters (m, $S_{threshold}$, $\psi_{threshold}$) on turn/nonturn prediction accuracy measured by MCC. For clarity, only the relevant values of m are shown. The color code goes from blue (~0.39) to red (~0.42); all values below 0.39 are plotted white).

a few examples are shown in these figures. The problem was to maximize the MCC as well as the Q_{total}, knowing that when one of these parameters increases, the other generally decreases (and so does Q_{obs} and Q_{pred}).

The window size does not have a strong effect on Q_{total}, but it does have a nonneglecting effect on MCC. The higher the value of m, the larger the area in which the MCC is maximum. This is true until $m = 4$. For larger sizes ($m = 6$

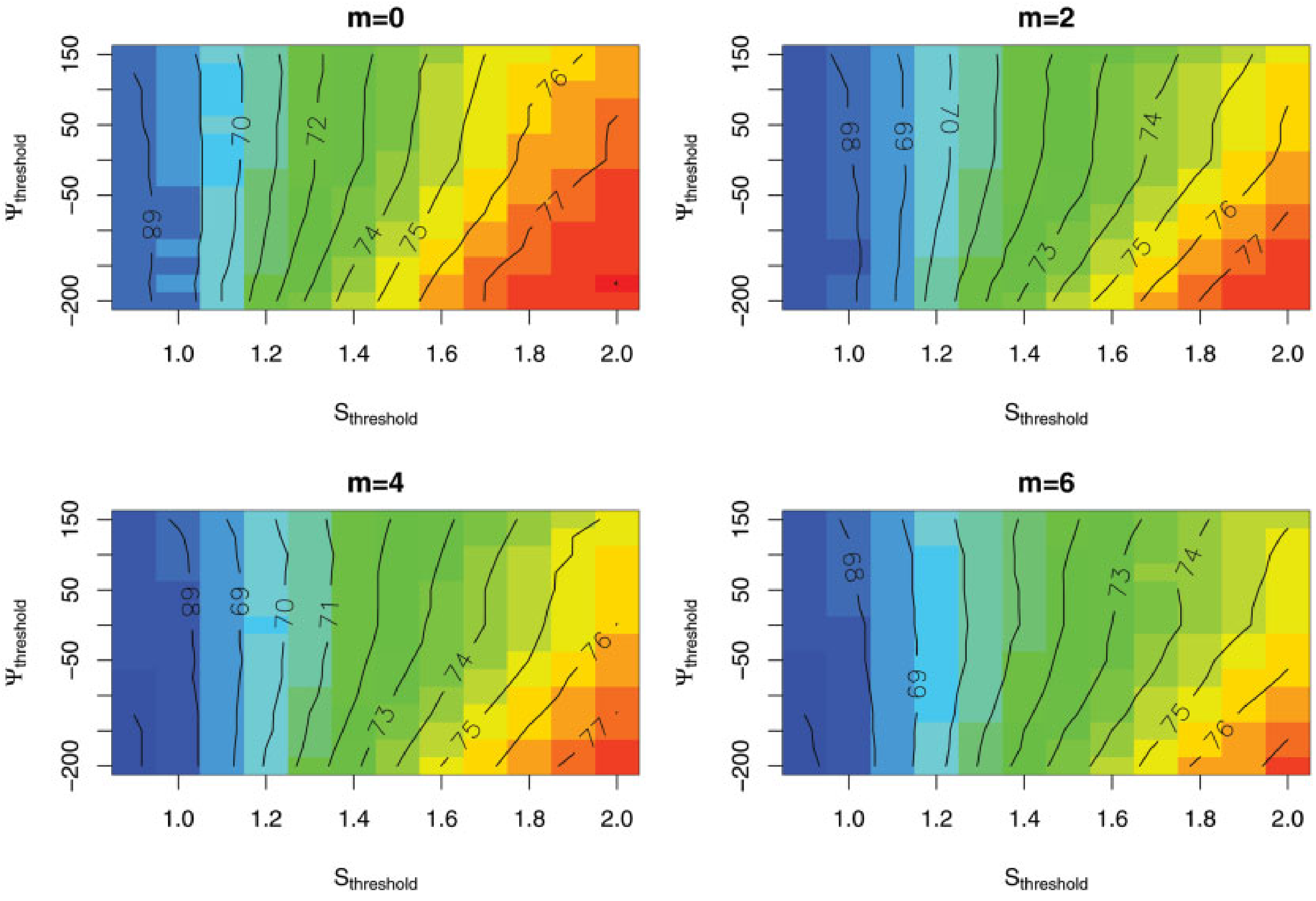

Fig. 2. Effect of the different parameters (m, $S_{\text{threshold}}$, $\psi_{\text{threshold}}$) on turn/nonturn prediction accuracy measured by Q_{total}. For clarity, only the relevant values of m are shown. The color code goes from blue (~68%) to red (~77%). [Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

and more), the accuracy reaches a saturation with values that no longer evolve; it shows that increasing the value of m too much is useless except for a waste of computer time. For short sizes, the results are significantly worst. We see that the area in which MCC is less than 0.39 is fairly large, and the maxima are not as good as for larger windows. This is logically explained by the fact that when $m = 0$, the window is as large as a β-turn, we thus do not take into account of the neighboring residues around it. The best value is definitely for $m = 4$, because it maximizes the area of high MCC and shifts this latter towards high $S_{\text{threshold}}$; it allows one to get higher Q_{total} (Q_{total} always increases linearly with $S_{\text{threshold}}$) for a given MCC to obtain a better balance MCC/Q_{total}.

The value of $S_{\text{threshold}}$ has a large effect both on MCC and Q_{total}. MCC presents a maximum for a particular value of $S_{\text{threshold}}$ that depends on the window size (e.g., for $m = 4$ MCC is maximum when $S_{\text{threshold}}$ ~1.5). For Q_{total}, the higher the $S_{\text{threshold}}$, the higher its value. Both parameters (MCC and Q_{total}) are less sensitive to $\psi_{\text{threshold}}$ except within the particular area at a high $S_{\text{threshold}}$ and a low $\psi_{\text{threshold}}$ (at the bottom right corner of graphs in Figs. 1 and 2): it presents very high Q_{total} and very low MCC. It comes from the fact that under these conditions overprediction is lowered by a high $S_{\text{threshold}}$, which affects positively the Q_{total} and negatively the MCC. What is also interesting is that it occurs at a low $\psi_{\text{threshold}}$: lowering this parameter "dilutes" the original sequence (the lower the $\psi_{\text{threshold}}$, the larger the number of residues we take into account for calculating the score giving thus a kind of "dilution").

From Figures 1 and 2, the best compromise between MCC and Q_{total} would be the following values: $m = 4$, $S_{\text{threshold}} = 1.8$ and $\psi_{\text{threshold}} = 0$ (MCC = 0.416 and $Q_{\text{total}} = 74.8\%$). Nevertheless, it is important to emphasize that around these optimized values, a small variation of any of these three parameters has only little effect on the accuracy, which shows somehow the relative stability of COUDES. A ROC plot (where specificity is plotted against sensitivity, see Refs. 27 or 31 for a definition) was obtained by varying $S_{\text{threshold}}$ (with m and $\psi_{\text{threshold}}$ fixed at 4 and 0, respectively) and is available as supplementary data.

Comparison of the Accuracy with Other Methods

The accuracy of COUDES versus the other methods published so far (those for that we have MCC and Q_{total} data) is shown in Table VI. The first line of the table presents the highest reached MCC/Q_{total} balance for

TABLE VI. Comparison of COUDES Accuracy with Some Other Methods[a]

Method	MCC	Q_{total} (%)	Q_{obs} (%)	Q_{pred} (%)
COUDES (1)[b]	0.42	74.8	69.9	48.8
COUDES (2)[c]	0.41	75.5	66.6	49.8
BETATPRED2[d]	0.43	75.5	72.3	49.8
KNN[e]	0.40	75.0	66.7	46.5
BTPRED (2)[f]	0.40	76.0	63.0	50.9
BTPRED (1)[g]	0.35	74.9	55.3	48.0
Chou-Fasman[h]	0.34	74.3	54.3	47.7
Thornton[h]	0.31	75.2	44.9	49.3
1–4 & 2–3 corr[h]	0.31	73.4	51.5	46.2
Seq Coupl Mod[h]	0.33	75.2	60.0	45.0
GORBTURN[h]	0.28	75.4	37.7	49.6

[a] All runs were done on the data set *PROM426* proteins and by using PSIPRED secondary structure information (except for g and h).

[b] Best compromise between MCC and Q_{total} achieved by using the following parameters: $m = 4$, $S_{\text{threshold}} = 1.8$, and $\psi_{\text{threshold}} = 0$ (the MCC value is exactly 0.417).

[c] score achieved by setting the parameters in order to get the same Q_{total} than ref. 27: with $m = 4$, $S_{\text{threshold}} = 1.8$, and $\psi_{\text{threshold}} = -100$ (the MCC value is exactly 0.410).

[d] Value taken from ref. 27.

[e] Value taken from ref. 28, "KNN" stands for *k*-nearest neighbor.

[f] Value taken from ref. 25 (improved by the use of secondary structures predicted by PSIPRED with multiple alignments, compared to PHD in the original method of ref. 23).

[g] Value taken from ref. 23, the data set is different.

[h] Value taken from ref. 25, "1–4 & 2–3 corr" stands for 1–4, 2–3 correlation model,[19] "Seq Coupl Mod" stands for sequence coupled model[20] (see ref. 25 for a definition of all these methods); here, secondary structure information comes from PROF.[26]

COUDES, while the second shows the accuracy when we choose $S_{\text{threshold}}$ to get the same Q_{total} as for BETATPRED2. It is worth noting that all the results [except the line BTPRED(1)] were tested on the same database. BETATPRED2 slightly outperforms COUDES, although it uses the same basic information (multiple alignments and secondary structure). If we compare the difference in accuracy between older statistical methods versus BTPRED,[25] with COUDES versus BETATPRED2, this difference is considerably smaller in our case. It shows that it is possible to lead a statistical-based method as far as neural networks. The addition of multiple alignments in COUDES makes it outperform all other methods in which this information is lacking (BTPRED, KNN, etc.). In conclusion, COUDES is the second best method published so far in terms of turn/nonturn prediction.

Prediction of β-Turn Types

The last part of this work is devoted to type prediction. This is not an easy task, because each type behaves differently: therefore, one set of parameter (m, $S_{\text{threshold}}$, and $\psi_{\text{threshold}}$) may work for a given type, whereas it may not for another type. Because it would be inconvenient to change the parameter every time, we have searched the optimal ones that give reasonable results for each type but with a priority for turn/nonturn prediction: we thus used the following parameters $m = 4$, $S_{\text{threshold}} = 1.8$, and $\psi_{\text{threshold}} = 0$ (the best parameters for turn/nonturn prediction). Hence, it is important to bear in mind that the reported values are not the highest ones we could obtain, but the best compromise. In Table VII we report these results as well as a comparison with those of BTPRED[23] and BETATURNS.[31] We will first focus on COUDES results, and then we will compare them to the two others methods.

Types I and II raise reasonable results, with MCC slightly higher than 0.3. Their Q_{obs} is around 50%, which means that one of this type is detected in the two that exist. Q_{pred} is around 30% for type I, which means that among 10 residues predicted in such a type, three of them are correct (Q_{pred} is equal to 20% for type II). The other types of β-turn are harder to predict, especially types VIII and II′, with MCC inferior or equal to 0.1. The problem comes mainly from overprediction with Q_{pred}, which drops under 10%. One reason for the poor performance for type II′ comes from its very low frequency: indeed, a large number of chains do not contain any type II′ at all. It complicates prediction considerably, as we do not have a well-balanced set for such a rare type; if we calculated MCC and Q_{total} on solely sequences that do contain at least one type II′, the accuracy would be better. For instance, with a sevenfold crossvalidation on the 120 chains that contain at least one type II′ we could obtain much better results: MCC = 0.163, $Q_{\text{total}} = 92.6\%$, $Q_{\text{obs}} = 35.8\%$, $Q_{pred} = 10.4\%$. For type VIII, the low accuracy may be explained by its conformational heterogeneity because it cannot be stabilized by any backbone hydrogen bond; therefore, the amino acid conformational preferences are fuzzier and complicate its prediction. Type IV is a bit better treated with an MCC close to 0.11, mainly thanks to its better behavior with overprediction with a Q_{pred} of 20%. However, underprediction is not as good as for types I and II (Q_{obs} ~20%). The problem comes from the fact it is ill-defined; thus, there is no clear φ/ψ preference such as in the other types (in fact, type IV is close to any other type). Accordingly, when tested with other parameters that maximize the information (high m, low $S_{\text{threshold}}$, and $\psi_{\text{threshold}}$), we could obtain better Q_{obs} (up to 45%) and an MCC (higher than 0.15); however, this set of parameters does not necessarily work well for the other types. Last, Type I′ reaches a reasonable MCC of 0.226, but still suffers from overprediction (with a Q_{pred} around 12%) because it is sparsely populated (as type II′ but to a lesser extent).

Compared with the other methods, COUDES obtains a slightly better accuracy in terms of MCC for types I, II, and VIII; furthermore, it is the only one that predicts types I′ and II′. Only type IV is better predicted by BETATURNS with an MCC of 0.230 compared to 0.110 for COUDES and 0.033 for BTPRED. This large difference is due to the behavior of BETATURNS for underprediction with a particularly high Q_{obs} (72%). To conclude our type prediction, it is clear that our method is suitable for type prediction, and it can do better than neural networks.

TABLE VII. Comparison of β-Turn Type Prediction Accuracy with Other Methods[a]

	MCC			Q_{total}		
	COUDES	BTPRED	BETATURNS	COUDES	BTPRED	BETATURNS
Type I	0.309	0.219	0.29	84.5	91.2	74.5
Type II	0.302	0.253	0.29	91.0	95.5	93.5
Type VIII	0.071	0.062	0.02	90.7	95.7	96.5
Type I′	0.226	—	—	94.4	—	—
Type II′	0.106	—	—	94.6	—	—
Type IV	0.109	0.033	0.23	84.9	96.8	67.9

	Q_{obs}			Q_{pred}		
	COUDES	BTPRED	BETATURNS	COUDES	BTPRED	BETATURNS
Type I	50.0	46.6	74.1	30.8	13.9	22.1
Type II	52.8	58.4	52.8	22.2	12.2	25.5
Type VIII	18.7	18.0	2.8	6.9	3.3	7.2
Type I′	51.8	—	—	11.6	—	—
Type II′	32.8	—	—	4.6	—	—
Type IV	17.7	2.2	72.0	20.7	9.3	18.6

[a] Results obtained with the following parameters: $m = 4$, $S_{\text{threshold}} = 1.8$, and $\psi_{\text{cutoff}} = 0$. For BTPRED, values taken from ref. 23; for BETATURNS, values taken from ref. 31. It should be noted that the testing set of BTPRED is not the same; furthermore, they used PHD[24] for predicting secondary structures, whereas we, and Kaur and Raghava, used PSIPRED.[29]

CONCLUSION

In this article, we showed that it was possible to build a strong method of β-turn predictions with simple concepts such as amino acid propensities, with the help of predicted secondary structure information and multiple alignments. We tested many parameters that have a direct influence on prediction quality. First, it is sensitive to the way of extracting β-turns, and actually the best one is with PROMOTIF.[34] The database size has only a little influence on the accuracy, and we observed that this latter stays within the same range when we pass to larger databases (up to 823 chains). On the other hand, the secondary structure prediction method is one of the most important features for a good prediction quality. We obtained the best results with PSIPRED[29] and thereafter with SSpro2.[37] The neighboring residues have also an influence on prediction quality, and we found that the optimal window size is 12 residues long ($m = 4$). This means that β-turns are largely influenced by their local environment. The score threshold from which we consider the presence of a β-turn ($S_{\text{threshold}}$) has a strong influence: the higher its value, the lower overprediction but the higher the underprediction. We found that the best compromise was with a value of 1.8. The other threshold, the one from which we consider to pick up a residue in PSSMs ($\psi_{\text{threshold}}$), has a lower influence except at a high $S_{\text{threshold}}$. The optimal value in accordance to the window size and $S_{\text{threshold}}$ would be of 0. It should be noted that if the window size and $S_{\text{threshold}}$ change, it might no longer be the optimal one.

We also showed in this article that trivial strategies like predicting every (or randomly) residues as a turn outside the secondary structure could give either a quite good MCC or Q_{total}. This is why it is strictly necessary to evaluate methods with both measures. Recently, Kaur and Raghava[43] highlighted the problem, stating that one can get very high Q_{total} by choosing a very high threshold, but at the price of a low MCC. Therefore, if both measures are not given, the accuracy might be overestimated. To get rid of this problem, the ROC plot approach can give an idea of the method performance independently of the threshold (see supplementary data).

COUDES reached an MCC of 0.42 and a rate of good prediction of 74.8% for β-turn/non-β-turn prediction, which places it among the best two methods published so far. COUDES also predicts the different types of β-turn (except type VI), with a slightly higher accuracy than neural networks for types I, II, VIII, except type IV. Furthermore, it is the only available method that predicts rare types I′ and II′. One other advantage compared to neural networks is that it allows one to get the tendency of residues by means of propensities. COUDES is usable the other way round, that is, knowing a specific type we can propose some candidate sequences. This may be useful in peptide and/or protein design. Overall, the main limitation appearing in β-turn prediction whatever the method is overprediction. In general, the best methods present a Q_{pred} around 50%, meaning that one residue out of two (from those we predict) is mispredicted. This is even worst for type prediction (Q_{pred} drops down to 10% for certain types) and remains a severe limitation. However, it can still be useful to predict types to get a tendency; a specific sequence may form more than one particular type depending on the conditions under which it is considered (pH, temperature, concentration, solvent, other interacting molecules, etc.).

One way to improve turn prediction would be to use a consensus prediction between the highest accurate methods (e.g., neural networks, k-nearest neighbors, and COUDES). The other way will be by using more and more large sequence databases for generating multiple alignments.[42]

ACKNOWLEDGMENTS

The authors gratefully thank the members of the EBGM for useful discussions. They also thank G. Pollastri for having provided SSpro2, and A. Clarke for the support on PROF.

REFERENCES

1. Chou KC. Prediction of tight turns and their types in proteins. Anal Biochem 2000;286:1–16.
2. Hutchinson EG, Thornton JM. A revised set of potentials for beta-turn formation in proteins. Protein Sci 1994;3:2207–2216.
3. Richardson JS. The anatomy and taxonomy of protein structure. Adv Protein Chem 1981;34:167–339.
4. Rose GD, Gierasch LM, Smith JA. Turns in peptides and proteins. Adv Protein Chem 1985;37:1–109.
5. Müller G, Hessler G, Decornez HY. Are beta-turn mimetics mimics of beta-turns? Angew Chem Int Ed Engl 2000;39:894–896.
6. Kee KS, Jois SD. Design of beta-turn based therapeutic agents. Curr Pharm Des 2003;9:1209–1224.
7. Munoz V, Thompson PA, Hofrichter J, Eaton WA. Folding dynamics and mechanism of beta-hairpin formation. Nature 1997;390: 196–199.
8. Bonvin AM, van Gunsteren WF. beta-Hairpin stability and folding: molecular dynamics studies of the first beta-hairpin of tendamistat. J Mol Biol 2000;296:255–268.
9. Lewis PN, Momany FA, Scheraga HA. Folding of polypeptide chains in proteins: a proposed mechanism for folding. Proc Natl Acad Sci USA 1971;68:2293–2297.
10. Chou PY, Fasman GD. Conformational parameters for amino acids in helical, beta-sheet, and random coil regions calculated from proteins. Biochemistry 1974;13:211–222.
11. Chou PY, Fasman GD. Prediction of protein conformation. Biochemistry 1974;13:222–245.
12. Chou PY, Fasman GD. Beta-turns in proteins. J Mol Biol 1977;115: 135–175.
13. Chou PY, Fasman GD. Prediction of beta-turns. Biophys J 1979;26: 367–373.
14. Garnier J, Osguthorpe DJ, Robson B. Analysis of the accuracy and implications of simple methods for predicting the secondary structure of globular proteins. J Mol Biol 1978;120:97–120.
15. Rose GD. Prediction of chain turns in globular proteins on a hydrophobic basis. Nature 1978;272:586–590.
16. Wilmot CM, Thornton JM. Analysis and prediction of the different types of beta-turn in proteins. J Mol Biol 1988;203:221–232.
17. Wilmot CM, Thornton JM. Beta-turns and their distortions: a proposed new nomenclature. Protein Eng 1990;3:479–493.
18. Guruprasad K, Rajkumar S. Beta- and gamma-turns in proteins revisited: a new set of amino acid turn-type dependent positional preferences and potentials. J Biosci 2000;25:143–156.
19. Zhang CT, Chou KC. Prediction of beta-turns in proteins by 1-4 and 2-3 correlation model. Biopolymers 1997;41:673–702.
20. Chou KC, Blinn JR. Classification and prediction of beta-turn types. J Protein Chem 1997;16:575–595.
21. Chou KC. Prediction of beta-turns. J Pept Res 1997;49:120–144.
22. McGregor MJ, Flores TP, Sternberg MJ. Prediction of beta-turns in proteins using neural networks. Protein Eng 1989;2:521–526.
23. Shepherd AJ, Gorse D, Thornton JM. Prediction of the location and type of beta-turns in proteins using neural networks. Protein Sci 1999;8:1045–1055.
24. Rost B. PHD: predicting one-dimensional protein structure by profile-based neural networks. Methods Enzymol 1996;266:525–539.
25. Kaur H, Raghava GP. An evaluation of beta-turn prediction methods. Bioinformatics 2002;18:1508–1514.
26. Ouali M, King RD. Cascaded multiple classifiers for secondary structure prediction. Protein Sci 2000;9:1162–1176.
27. Kaur H, Raghava GP. Prediction of beta-turns in proteins from multiple alignment using neural network. Protein Sci 2003;12:627–634.
28. Kim S. Protein beta-turn prediction using nearest-neighbor method. Bioinformatics 2004;20:40–44.
29. Jones DT. Protein secondary structure prediction based on position-specific scoring matrices. J Mol Biol 1999;292:195–202.
30. Cai YD, Liu XJ, Li YX, Xu XB, Chou KC. Prediction of beta-turns with learning machines. Peptides 2003;24:665–669.
31. Kaur KS, Raghava GP. A neural network method for prediction of beta-turn types in proteins using evolutionary information. Bioinformatics 2004;16:2751–2758.
32. Hobohm U, Scharf M, Schneider R, Sander C. Selection of representative protein data sets. Protein Sci 1992;1:409–417.
33. Hobohm U, Sander C. Enlarged representative set of protein structures. Protein Sci 1994;3:522–524.
34. Hutchinson EG, Thornton JM. PROMOTIF—a program to identify and analyze structural motifs in proteins. Protein Sci 1996;5: 212–220.
35. Kabsch W, Sander C. Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. Biopolymers 1983;22:2577–2637.
36. Altschul SF, Madden TL, Schaffer AA, Zhang J, Zhang Z, Miller W, Lipman DJ. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res 1997;25: 3389–3402.
37. Pollastri G, Przybylski D, Rost B, Baldi P. Improving the prediction of protein secondary structure in three and eight classes using recurrent neural networks and profiles. Proteins 2002;47:228–235.
38. Baldi P, Brunak S, Chauvin Y, Andersen CA, Nielsen H. Assessing the accuracy of prediction algorithms for classification: an overview. Bioinformatics 2000;16:412–424.
39. Mathews BW. Comparison of the predicted and observed secondary structure of T4 phage lysozyme. Biochim Biophys Acta 1975;405:442–451.
40. Fourrier L, Benros C, de Brevern AG. Use of a structural alphabet for analysis of short loops connecting repetitive structures. BMC Bioinformatics 2004;5:58.
41. Rost B, Sander C. Prediction of protein secondary structure at better than 70% accuracy. J Mol Biol 1993;232:584–599.
42. Przybylski D, Rost B. Alignments grow, secondary structure prediction improves. Proteins 2002;46:197–205.
43. Kaur H, Raghava GP. Prediction of alpha-turns in proteins using PSI-BLAST profiles and secondary structure information. Proteins 2004;55:83–90.