

Pcons5: combining consensus, structural evaluation and fold recognition scores

Björn Wallner^{1*} and Arne Elofsson¹

¹Stockholm Bioinformatics Center, Stockholm University, SE-106 91 Stockholm, Sweden.

ABSTRACT

Motivation: The success of the consensus approach to the protein structure prediction problem, has led to development of several different consensus methods. Most of them only rely on a structural comparison of a number of different models. Even though, there are other types of information that might be useful such as the score from the server and structural evaluation.

Results: Pcons5 is a new and improved version of the consensus predictor Pcons. Pcons5 integrates information from three different sources: consensus analysis, structural evaluation and the score from the fold recognition servers. We show that Pcons5 is better than the previous version of Pcons and that it performs better than using only consensus analysis. In addition, we also present a version of Pmodeller based on Pcons5, that performs significantly better than Pcons5.

Availability: Pcons5 is the first Pcons version available as a standalone program from: <http://www.sbc.su.se/~bjorn/Pcons5>. It should be easy to implement in local meta-servers.

Contact: bjorn@sbcsu.se

INTRODUCTION

The use of many different methods to predict the structure of a protein is now state-of-the-art in protein structure prediction. This is facilitated by the different meta or consensus predictors that are available, e.g. through the meta-server at <http://bioinfo.pl/Meta/> [4]. The consensus predictors use the result from different prediction methods to select the best protein model. In principle they are all based on the simple approach of selecting the most abundant representative among the set of high scoring models. Pcons [20] was the first fully automated consensus predictor, followed by several others [8, 9]. All benchmarking results obtained in the last two years, both at CASP [21] and in LiveBench [22] indicate that consensus prediction methods are more accurate than the best of the independent fold recognition methods [10], e.g. in CAFASP3 the performance was 30% higher than that of the best independent fold recognition methods and comparable to the best 2-3 best human CASP predictors [8]. The main strength of the consensus analysis is coupled to the structural comparison. However, there are also other factors that can be used in the selection process. The score from the fold recognition method and a structural evaluation of the model are such parameters. A problem with using the score from the fold recognition methods directly is that the scoring scheme might change at any time, leading to a frequent re-optimization of the parameters. In this paper we describe the newest version of Pcons, Pcons5. It consists of three parts: the consensus analysis, structural evaluation and

a final part dependent on the score from the fold recognition server. In addition, we also present an improved version of Pmodeller [29] based on Pcons5 and ProQ [27].

METHODS

Data sets

All data sets used in the development of Pcons5 were constructed from different versions of LiveBench [3]. These sets contain models that are possible to be obtained for unknown targets and that show a range of quality differences. LiveBench is continuously measuring the performance of different fold recognition web-servers by submitting the sequence of recently solved proteins structures, with no obvious close homolog (10^{-3} BLAST cutoff [1]) to a protein in the Protein Data Bank [2].

The structural evaluation module were trained on the same data set as used in ProQ [27] (LiveBench-2). The parameters for the consensus analysis and score evaluation as well as the final combination was performed on a data set constructed from LiveBench-4. The LiveBench-4 data set was collected during the period 2001-11-07–2002-04-25 and contains protein structure predictions for 107 targets from 14 individual servers and 3 consensus servers. In total 10,974 protein models for these eleven servers were used: PDB-BLAST [23], FFAS [23], Sam-T99 [16], mGenTHREADER [14], INBGU [7], three FUGUE servers [25], 3D-PSSM [18], Orfeus [11] and Superfamily [13]. The models used were simple backbone copies of the aligned residues from the template.

METHOD DEVELOPMENT

Pcons5 consists of three different modules: consensus analysis, structural evaluation and score evaluation. It has been developed with the goal to be independent of the use of a fixed set of methods/servers, i.e. it should work with any number of methods and with any number of models. Each of the modules in Pcons5 produce two scores reflecting different aspects of model quality. These scores are combined to produce the final score using a weighted sum. In the following the three different modules will be described.

Consensus Analysis

The consensus analysis is performed in a similar way as in 3D-Jury [9], with the only difference being that LGscore [5] is used to compare the models. The comparison is done for all and for the first ranked models only, as in the different versions of 3D-Jury. This results in two scores one reflecting the average similarity to all other models (Eq. 1) and one reflecting the similarity to all first ranked models (Eq. 2)

$$C_i^{all} = \frac{1}{N} \sum_{\forall j \notin \text{method}(i)} \text{sim}(i, j) \quad (1)$$

*to whom correspondence should be addressed

$$C_i^{first} = \frac{1}{M} \sum_{\substack{\forall j \in \text{rank}(j)=1 \\ j \neq i}} sim(i, j) \quad (2)$$

where N is the number of comparisons to other methods, M the number of comparisons to first ranked models and $sim(i, j)$ the similarity between model i and j . Here, we used LGscore but any structural similarity measure should most likely work.

Structural Evaluation

The structural evaluation is done using a backbone version of ProQ [27]. ProQ uses distribution of atom-atom contacts, residue-residue contacts, secondary structure information and surface accessibility for different amino acids to assess the quality of protein models. The original version was developed for protein models with all atoms. The version used in Pcons5 uses only the backbone atoms, usually obtained by copying the aligned coordinates from the template. This version of ProQ is not as accurate as the original ProQ version, but since there is no need to build all-atom models, the overall method is considerably faster.

Since the structural evaluation is performed on backbone models it is not possible to use exactly the same structural information as in the all-atom version of ProQ, e.g. using contacts between different types of atoms was no longer possible. However, the same six residue types as in ProQ were used, but the residue-residue contact cutoff had to be increased to 14Å to compensate for the non-existing side-chains. The cutoff was chosen by trying different cutoffs in the range 6Å-20Å. The calculation of surface accessibility also had to be changed. We chose to use a reduced representation defining buried and exposed residues based on number of neighboring CA atoms. Residues with less than 16 atoms within 10Å were defined as exposed and residues with more than 20 atoms within 10Å as buried. These definitions showed the largest agreement with Naccess [19]. The secondary structure information was the fraction of agreement between predicted secondary structure using PSIPRED [15] and the actual secondary structure in the model. As in the original ProQ version, neural nets were trained to predict LGscore [5] and MaxSub [26], based on the structural features. A final correlation coefficient of 0.65 was obtained, in comparison with 0.76 for the all-atom ProQ.

Score evaluation

A good indicator of model quality is the score from the fold recognition method or server, a high score is usually connected to a good model. However, since Pcons5 should be independent of a fixed number of servers it is impossible to include the raw score from the server directly in Pcons5. Instead, each raw score is scaled to a common scale based on the reliability of the score. If the reliability is not known Pcons5 will not use the score to compute the final score.

The score evaluation was designed to be easy to update and facilitate the inclusion of new methods, without the need to re-optimize all parameters. Further, if the scoring of a method suddenly changes it should not impact the result too much. To limit the impact on the final score, all scores were scaled using two levels, "good" and "very good". In principle it works as follows: if the score is good the model will obtain one extra point and if the score is even better (very good) the model have the possibility to get one additional point. Thus, even an extremely high score could only yield two additional points.

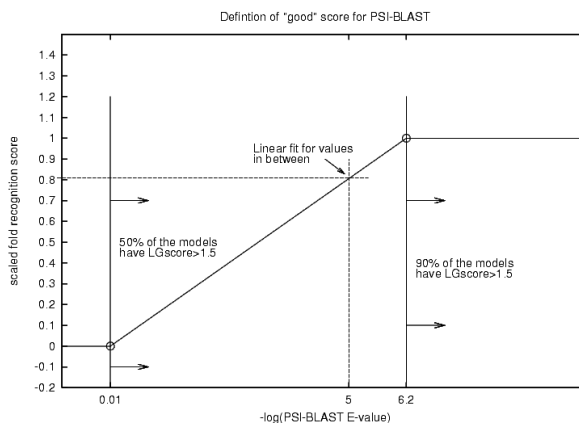


Fig. 1. Scaling of PSI-BLAST scores for the definition of "good" models.

The reliability of each server score was assessed by correlate fold recognition score with model quality from LiveBench-4. For each server two cutoffs were used to define "good" models and two cutoffs were used to define "very good" models. These cutoffs were decided by analyzing the quality of models associated with a certain score. In more detail, the first cutoff, was set to the score for which 50% of all models had an LGscore > 1.5 and the second, to the score for which 90% of all models had LGscore > 1.5. The "very good" models was defined in a similar manner but with LGscore > 3. For scores falling between these cutoffs a linear fit was used. The process is exemplified here by the PSI-BLAST method, which has a familiar E-value score (Figure 1 and 2). 50% of all models with PSI-BLAST E-value below 10^2 have LGscore above 1.5 and 90% of the models with a score above $10^{-6.2}$ have LGscore above 1.5. And for LGscore above 3 the cutoffs are $10^{-20.9}$ and $10^{-124.3}$ respectively. For scores in between the cutoff values a linear fit is used, e.g. a score of 10^{-5} would yield a scaled score of 0.81 on the "good" scale and 0 on the "very good" (Figure 1). In a way this is a reflection of the reliability of a hit in the database with a certain score. An E-value of 10^{-5} is mostly likely correct but the alignment is probably not optimal. If on the other hand the E-value is $10^{-72.6}$ the model is very likely to be of high quality and this is also reflected by a significantly higher scaled score of 1 on the "good" scale and 0.5 on the "very good" scale (Figure 2).

Compiling the final Pcons5 score

The final Pcons5 score is a combination of the six different scores from the three different modules described above. They were combined using multiple linear regression to fit the LGscore quality measure, with the following coefficients

$$Pcons5 = 0.53C^{all} + 0.20C^{first} + 0.64ProQ^{MX} + 0.27ProQ^{LG} + 0.32Score^{good} + 0.13Score^{very\ good} \quad (3)$$

where C^{all} and C^{first} is the score from the consensus analysis, $ProQ^{MX}$ and $ProQ^{LG}$ the predicted MaxSub and LGscore score respectively, $Score^{good}$ and $Score^{very\ good}$ the scaled fold recognition score for *good* and *very good* respectively.

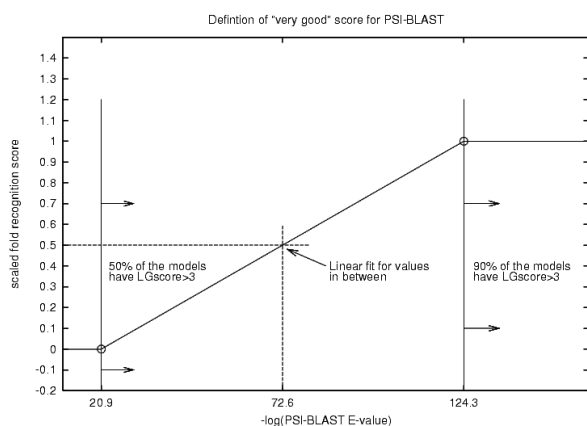


Fig. 2. Scaling of PSI-BLAST scores for the definition of "very good" models.

Table 1. Performance of the individual modules in Pcons5.

Score	R^2
C^{all}	0.81
C^{first}	0.77
$ProQ^{MX}$	0.44
$ProQ^{LG}$	0.52
$Score^{good}$	0.55
$Score^{very\ good}$	0.18
Pcons5	0.86

The performance was measured by the squared correlation coefficient, R^2 , also called the coefficient of determination.

The fit is rather good explaining 86% of the variance in the data (Table 1). If the range of the parameters are known their influence on the final score can be assessed directly from the size of the coefficients. The range of C^{all} , C^{first} and $ProQ^{LG}$ are all comparable in size (as they are trained to predict the LGscore of the model), $ProQ^{MX}$ needs to be multiplied by ten to put it on the same scale, $Score^{good}$ and $Score^{very\ good}$ are roughly one third of the three first parameters. Thus, the consensus analysis is the most important factor, followed by the structural evaluation using $ProQ^{LG}$, while $ProQ^{MX}$ and the two server specific scores influence the final score to a lesser degree. This is also in agreement with the R^2 values in Table 1.

Pmodeller5

For the previous Pcons version we have developed a corresponding Pmodeller version [29]. Pmodeller uses Modeller [24] to build all-atom models which are assessed using ProQ and the final Pmodeller score is combination of the Pcons score and ProQ score. In CASP5 it was shown that Pmodeller performs better than its corresponding Pcons version.

In Pmodeller5, we have used a slightly different approach. Instead of a linear combination of the Pcons and ProQ score, the combination is done in two steps. First Pcons5 is used to find the best scoring models, then all-atom models are built using Modeller6v2 [24] for

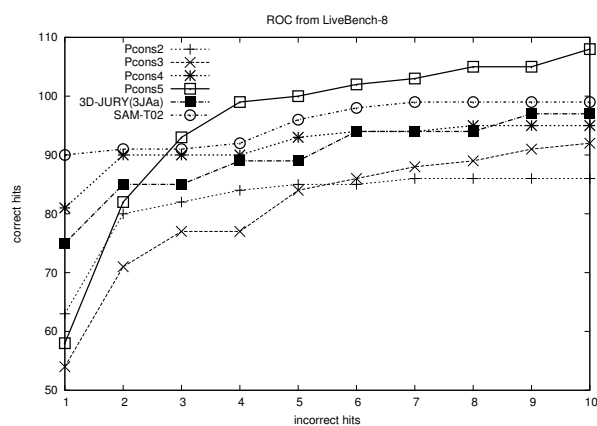


Fig. 3. Receiver Operator Characteristics on all targets from LiveBench-8 for the different Pcons methods and a 3D-Jury method (3D-JuryA-all). The evaluation was done using MaxSub with 0.1 as cutoff for correct model. As a reference the best performing independent server is also included. (data taken from <http://bioinfo.pl/LiveBench/>)

all models with a score within 10% from the highest. These models are then subjected to a re-ranking using the original all-atom version of ProQ [27], which is significantly better than the backbone ProQ module used in Pcons5. The final Pmodeller score consists only of the ProQ score. The use of only the top 10% scoring models will ensure that only the best models are included in the final ranking. At the same time the algorithm gets significantly faster, since there is no need to build all-atom models for low scoring models.

RESULTS AND DISCUSSION

It is important that new methods are benchmarked and compared to existing methods. Pcons5 has been thoroughly benchmarked in LiveBench and both Pcons5 and Pmodeller5 participated in CASP6.

Performance in LiveBench

ROC analysis from LiveBench-8 is seen in Figure 3, with the exception of two incorrect hits, Pcons5 consistently performs better than the previous versions of Pcons. In addition, it also performs better than the 3D-Jury method that uses consensus from eight selected server (more or less the same servers that Pcons5 was using), i.e. this 3D-Jury method correspond to the consensus analysis module in Pcons5. Consequently, the structural evaluation using ProQ and the score from the server is the reason for the 10% improvement relative to the 3D-Jury method. The performance of the best independent server, SAM-T02 [17], used by Pcons5 in LiveBench-8 is remarkable. Pcons5 is only marginally better (for > 2 incorrect) and it outperforms all previous Pcons versions and even 3D-Jury which also use SAM-T02 in its consensus analysis. However, one problem with Pcons5 on LiveBench is that we have no control over which servers that are used. Sometimes the results from certain independent servers are missing e.g. if the pressure of the servers is high. Therefore, the comparison to the independent servers is better done on the CASP6 data, where it was guaranteed that the results from as many servers as possible were used as input to Pcons5.

Performance in CASP6

Pcons5 use a number of independent servers as its input, normally these are submitted through the meta-server (<http://bioinfo.pl/meta/>). During CASP6 there was a problem running Pcons5 as an automatic server using the meta-server. Instead two different versions of Pcons5 participated in CASP (“Pcons5” and “SBC-Pcons5”). “Pcons5” used a limited number of independent servers and was run through the *genesilico.pl* meta-server (<http://genesilico.pl/meta/>), while “SBC-Pcons5” used more (and better) servers from the *bioinfo.pl* meta-server once this data was available. Unfortunately this data was not always ready within the time limit to participate as a server in CASP. This forced us to have “SBC-Pcons5” registered as a manual group, even though it was run without any human intervention. For each of the Pcons version a corresponding Pmodeller version also participated in CASP (“Pmodeller5” and “SBC-Pmodeller5”).

As expected the “SBC-*” versions of Pcons and Pmodeller using more and better servers performed significantly better with 10% higher sum of GDT_TS than the versions using fewer servers. This shows that the success of the consensus approach is dependent on a good set of individual servers. It can be used on a limited number of servers, but the performance can only be expected to be as good as the models it can choose between. The following analysis will be on the “SBC-*” versions of Pcons and Pmodeller. The relative performance of Pcons and Pmodeller is similar for the versions using fewer servers (data not shown).

To compare the performance of Pmodeller5, Pcons5, to the server they are using and to the other groups participating in CASP6, the GDT_TS [31] score for the first ranked models were used. Other scoring schemes like MaxSub [26] or TM-score [32] produce similar results (data not shown). Identical to our previous analysis of CASP5 results [29] we made two assumptions in our analysis. First, insignificant differences in performances were ignored, by consider two models with a difference <0.05 GDT_TS score to be of similar quality. Second, models where none of the compared methods made a “correct” prediction were also ignored. This was done by ignoring all targets where none of the compared methods could align >30 residues, i.e. where the GDT_TS multiplied by the length of the target is <0.30 . Targets were also divided into Comparative Modeling targets (CM) by concatenating CM easy and CM hard and to Fold Recognition/New Fold (FR/NF) by combining FR-homologous, FR-analogous and New Fold as defined by the CASP assessors (see <http://predictioncenter.org>). This resulted in a total of 59 domains, divided into 41 CM targets and 18 FR targets, after filtering out models where no predictor made a “correct” prediction.

Pcons5 vs Pmodeller5

Pmodeller5 performed significantly better than Pcons5 for 10 targets and only significantly worse for 3, see Table 2. For the FR&NF models it did not make any model worse than the corresponding Pcons5 model. Since Pmodeller5 uses the result from Pcons5 it is possible that the predictions are based on the same alignment. This is the case for 20% of the Pmodeller5 models, but in none of these cases is the model significantly improved by the homology modeling procedure, thus the main reason for the increase in performance is the re-ranking of the models using ProQ [28]. This was also the main conclusion from comparisons of the previous versions of Pcons and Pmodeller [29].

Table 2. Comparison between Pcons5 and Pmodeller5 in CASP6.

	Pcons5	Pmodeller5
All (59)	3	10
Comparative modeling (41)	3	7
Fold recognition and new fold (18)	0	3

The numbers represent the number of times the method is significantly better than the other, as measured by a difference >0.05 in GDT_TS.

Table 3. Servers used by Pcons5 and Pmodeller5

Server	Pcons5	Pmodeller5
SAM-T02	–	1
FUGUE3	6	4
SUPFAM_PP	1	–
FFAS03	5	7
BasC	7(2)	2(1)
PDB-BLAST	–	2
ORFEUS	5(1)	4
mGenTHREADER	12	2
BLAST	2	2
3D-PSSM	1	2
RAPTOR	16	15(1)
PROSPECT2	3	1
Eidogen-SFST	2	12(4)
INBGU	1	5(4)
ARBY	–	1
total	61(3)	60(10)

The individual servers used as input to Pcons5 in CASP6 and the number of times a particular server was ranked highest by Pcons5 and Pmodeller5 respectively. There were 64 official targets Pcons5 produced results for 61 and Pmodeller5 for 60. The values within parenthesis is the number of models that are significantly improved.

Comparison to servers used by Pcons5 and Pmodeller5

The servers used by Pcons5 and Pmodeller5 are listed in Table 3 together with the number of times it was selected by either Pcons5 and Pmodeller5. RAPTOR [30] is the most frequently selected method both by Pcons5 and Pmodeller5. The main differences in server preference are that Pmodeller5 selects Eidogen-SFST [6] and INBGU [7] models more frequently than Pcons5, while Pcons5 seem to prefer mGenTHREADER [14] and BasC [12] models. In fact, all four INBGU models selected by Pmodeller5 significantly increase the model quality compared to the Pcons5 model for the same targets. Four of the selected Eidogen-SFST models are also better than the corresponding Pcons5 model.

For each group an “average rank” was also calculated using the following formula:

$$\frac{N}{\sum \frac{1}{Rank}}$$

where N is the number of targets and $Rank$ is the number of predictions that are >0.05 GDT_TS units better than the current model. For a group that always makes (one of) the best predictions,

this average rank will be 1, whereas if a group always makes the worst prediction, the average rank will be identical to the number of groups in the comparison. One advantage with this measure compared to a simple average is that it is less sensitive to one (or a few) bad prediction. In addition, the number of times a certain method did one of the best predictions (i.e. no prediction had a GDT_TS 5 units better) and the number of times a prediction was better or worse than average was also calculated.

In Table 4 the performance of the all groups participating in CASP6 sorted by average rank is shown. Unfortunately not all servers used by Pcons5 participated in CASP6, e.g. the INBGU server and some servers hosted by bioinfo.pl only participated in CAFASP4. However, according to the CAFASP4 evaluation the best independent server was Eidogen-EXPM. Thus, the servers only participating in CAFASP4 could be expected to be ranked slightly below Eidogen-EXPM.

Pmodeller5 show the highest average rank of all servers. Pcons5 performs significantly worse compared to Pmodeller5. One of the servers used by Pcons5, Eidogen-SFST, is actually ranked slightly higher than Pcons5. Even though it is disappointing that Pcons5 is not ranked higher than Eidogen-SFST, it is even more impressive that the ProQ re-ranking managed to make Pmodeller5 the best server.

The sum of the GTD_TS for the first ranked model from each group was also used to measure performance, see Table 5. Here, Pcons5 performs better than the best individual server, in particular for the harder targets. Pmodeller5 is better than Pcons5 for both hard and easy targets, but the real improvement is observed for the easy targets, where it performs almost as well as the best manual groups.

One advantage with a consensus method is that it most often selects model that is better than the average model and seldom a model that is worse than average. However, even though it usually makes a good choice, it often misses the best possible model, i.e. the best model is ranked high but not at the top. Here, a more specific method or energy function can be used to evaluate the top hits. In our case, by using ProQ we increased the number of top hits produced from 13% to 24% of all targets.

CONCLUSIONS

We have developed a new version of Pcons, Pcons5, that uses structural evaluation and reliability assessment of the server score on top of the consensus analysis. This add on improves the performance with 10% compared to only using consensus on the LiveBench-8 data set. The performance compared to previous versions is also slightly higher. The new version is easy to update and works even for “unseen” methods.

In addition to the development of Pcons5 a new version of Pmodeller has also been developed, Pmodeller5. This method uses ProQ to evaluate the best hits from Pcons5. Pmodeller5 was among the best servers in CASP6 and consistently ranked higher than Pcons5.

Pcons5 is the first Pcons version available as a standalone program from: <http://www.sbc.su.se/~bjorn/Pcons5>. It should be easy to implement in local meta-servers. The model evaluation module in Pmodeller, ProQ, is also available as a standalone program from: <http://www.sbc.su.se/~bjorn/ProQ>.

Table 4. Results from CASP6 – average rank

Group	average rank	Top (%)	Better (%)	Worse (%)
Ginalski	1.4	62	83	1
Skolnick-Zhang	1.8	45	75	2
KOLINSKI-BUJNICKI	2.0	39	80	1
GeneSilico-Group	2.2	37	76	3
BAKER	2.2	33	82	2
CBRC-3D	2.3	33	64	6
CHIMERA	2.4	28	70	2
TOME	2.5	30	60	11
BAKER-ROBETTA_04	2.7	24	64	3
SAM-T04-hand	2.7	25	70	2
SBC-Pmodeller5	2.8	24	69	5
Jones-UCL	2.8	23	71	2
MCon	2.8	24	66	7
ACE	3.0	23	61	8
SBC	3.0	22	67	5
CMM-CIT-NIH	3.0	26	53	37
honiglab	3.1	25	52	32
ZHOUSPARKS2	3.2	22	57	6
BAKER-ROBETTA	3.2	22	62	6
LTB-Warsaw	3.2	21	56	14
Eidogen-EXPM	3.2	20	59	11
VENCLOVAS	3.3	26	39	61
Sternberg	3.3	16	68	1
zhousp3	3.3	20	55	7
CAFASP-Consensus	3.4	17	62	6
FISCHER	3.4	15	68	2
UGA-IBM-PROSPECT	3.5	18	52	6
CaspIta	3.6	16	52	13
Eidogen-SFST	3.7	16	54	17
Shortle	3.7	16	51	24
SAMUDRALA	3.7	17	53	10
Eidogen-BNMX	3.7	16	53	16
WATERLOO	3.9	16	56	14
Bilab	4.1	14	41	28
MacCallum	4.1	11	56	16
3D-JIGSAW	4.1	14	54	2
SBC-Pcons5	4.1	13	66	8
LOOPP_Manual	4.1	16	45	24
Rokko	4.1	14	55	13
mGenTHREADER	4.2	15	39	28
rohl	4.3	16	47	25
RAPTOR	4.3	11	49	8
CBSU	4.4	11	46	13
BioInfo_Kuba	4.5	16	41	34
Pan	4.5	15	40	22
agata	4.7	14	41	28
FUGMOD_SERVER	4.8	11	46	29
fams	4.9	11	39	28
hmmspectr3	4.9	14	39	18
famd	4.9	11	41	28

Servers are marked in boldface. (Top) is the fraction of prediction that were among the best, (Better) is the fraction of prediction that were significantly better than average and (Worse) is the fraction that were significantly worse than the average prediction. All groups an average rank > 5 are removed. The complete table is available at <http://www.sbc.su.se/~bjorn/Pcons5/extras/>.

Table 5. Results from CASP6 – sum of GDT_TS

Group	sum of GDT_TS for first ranked model (rank)		
	All	CM	FR&NF
Ginalski	50.25(1)	31.62(1)	18.62(1)
KOLINSKI-BUJNICKI	46.58(2)	30.40(3)	16.18(3)
BAKER	46.43(3)	29.01(17)	17.42(2)
Skolnick-Zhang	46.07(4)	30.92(2)	15.15(9)
GeneSilico-Group	45.73(5)	29.92(4)	15.81(7)
CHIMERA	45.55(6)	29.45(5)	16.10(4)
CBRC-3D	44.98(7)	29.12(14)	15.86(6)
SAM-T04-hand	44.83(8)	28.87(19)	15.96(5)
Jones-UCL	44.70(9)	29.22(10)	15.48(8)
FISCHER	44.01(10)	29.24(9)	14.78(11)
Sternberg	43.81(11)	29.14(13)	14.67(12)
BAKER-ROBETTA_04	43.26(12)	28.21(25)	15.05(10)
SBC	42.94(13)	29.20(11)	13.75(20)
TOME	42.92(14)	29.15(12)	13.78(19)
MCon	42.89(15)	28.68(21)	14.21(16)
SBC-Pmodeller5	42.80(16)	29.26(7)	13.54(21)
BAKER-ROBETTA	42.69(17)	28.11(26)	14.58(13)
3D-JIGSAW	42.53(18)	28.68(21)	13.85(17)
CAFASP-Consensus	42.43(19)	29.05(16)	13.38(23)
ACE	42.03(20)	28.84(20)	13.19(24)
zhousp3	41.74(21)	29.00(18)	12.74(28)
SBC-Pcons5	41.53(22)	28.60(23)	12.93(27)
RAPTOR	40.38(29)	28.21(25)	12.17(34)
<i>Eidogen-SFST</i>	39.55(34)	28.60(23)	10.96(53)
<i>mGenTHREADER</i>	36.70(49)	26.46(43)	10.24(65)
<i>FUGUE_SERVER</i>	35.67(61)	26.35(44)	9.32(82)
<i>SAM-T02</i>	34.35(69)	26.02(49)	8.33(99)
<i>Sternberg_3dpssm</i>	33.65(74)	24.59(64)	9.06(88)
<i>Arby</i>	30.24(88)	20.41(93)	9.82(69)
<i>FFAS03</i>	25.15(104)	17.52(105)	7.63(111)

Servers are marked in boldface, servers in italics are used in Pcons5. The sum of the GDT_TS score for the first ranked model from each group, for All, CM and FR&NF.

ACKNOWLEDGMENTS

This work was supported by grants from the Swedish Natural Sciences Research Council, and a grant by the Graduate Research School in Genomics and Bioinformatics.

REFERENCES

- [1] S.F. Altschul, T.L. Madden, A.A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and D.J. Lipman. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res*, 25(17):3389–3402, 1997.
- [2] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne. The protein data bank. *Nucleic Acids Res*, 28(1):235–242, Jan 2000.
- [3] J.M. Bujnicki, A. Elofsson, D. Fischer, and L. Rychlewski. Livebench-2: large-scale automated evaluation of protein structure prediction servers. *Proteins*, 45(Suppl 5):184–191, 2001.
- [4] J.M. Bujnicki, A. Elofsson, D. Fischer, and L. Rychlewski. Structure prediction meta server. *Bioinformatics*, 17(8):750–751, 2001.
- [5] S. Cristobal, A. Zemla, D. Fischer, L. Rychlewski, and A. Elofsson. A study of quality measures for protein threading models. *BMC Bioinformatics*, 2(5), 2001.
- [6] Eidogen. company web-site. <http://www.eidogen.com>, 2005.
- [7] D. Fischer. Hybrid Fold Recognition: Combining sequence derived properties with evolutionary information. In *Pac. Symp. Biocomput.*, pages 119–130. World Scientific, Singapore, 2000.
- [8] D. Fischer. 3D-SHOTGUN: a novel, cooperative, fold-recognition meta-predictor. *Proteins*, 51(3):434–441, May 2003.
- [9] K. Ginalski, A. Elofsson, D. Fischer, and L. Rychlewski. 3d-jury: a simple approach to improve protein structure predictions. *Bioinformatics*, 19(8):1015–1018, May 2003.
- [10] K. Ginalski, N. V. Grishin, A. Godzik, and L. Rychlewski. Practical lessons from protein structure prediction. *Nucleic Acids Res*, 33(6):1874–1891, 2005.
- [11] K. Ginalski, J. Pas, L. S. Wyrwicz, M. von Grothuss, J. M. Bujnicki, and L. Rychlewski. ORFeus: Detection of distant homology using sequence profiles and predicted secondary structure. *Nucleic Acids Res*, 31(13):3804–3807, Jul 2003.
- [12] K. Ginalski, M. von Grothuss, N. V. Grishin, and L. Rychlewski. Detecting distant homology with meta-BASIC. *Nucleic Acids Res*, 32(Web Server issue):W576–81, Jul 2004.
- [13] J. Gough and C. Chothia. SUPERFAMILY: HMMs representing all proteins of known structure. SCOP sequence searches, alignments and genome assignments. *Nucleic Acids Res*, 30(1):268–272, Jan 2002.
- [14] D.T. Jones. GenTHREADER: an efficient and reliable protein fold recognition method for genomic sequences. *J Mol Biol*, 287(4):797–815, 1999.
- [15] D.T. Jones. Protein secondary structure prediction based on position-specific scoring matrices. *J Mol Biol*, 292(2):195–202, 1999.
- [16] K. Karplus, C. Barrett, and R. Hughey. Hidden markov models for detecting remote protein homologies. *Bioinformatics*, 14:846–856, 1998.
- [17] K. Karplus, R. Karchin, J. Draper, J. Casper, Y. Mandel-Gutfreund, M. Diekhans, and R. Hughey. Combining local-structure, fold-recognition, and new fold methods for protein structure prediction. *Proteins*, 53 Suppl 6:491–496, 2003.
- [18] L.A. Kelley, R.M. MacCallum, and M.J. Sternberg. Enhanced genome annotation using structural profiles in the program 3D-PSSM. *J Mol Biol*, 299(2):523–544, 2000.
- [19] B. Lee and F.M Richards. The interpretation of protein structures: estimation of static accessibility. *J Mol Biol*, 55(3):379–400, 1971.
- [20] J. Lundström, L. Rychlewski, J. Bujnicki, and A. Elofsson. Pcons: a neural-network-based consensus predictor that improves fold recognition. *Protein Sci.*, 10(11):2354–2362, 2001.
- [21] J. Moul, K. Fidelis, A. Zemla, and T. Hubbard. Critical assessment of methods of protein structure prediction (CASP)-round V. *Proteins*, 53 Suppl 6:334–339, 2003.
- [22] L. Rychlewski, D. Fischer, and A. Elofsson. Livebench-6: large-scale automated evaluation of protein structure prediction servers. *Proteins*, 53 Suppl 6:542–547, 2003.
- [23] L. Rychlewski, L. Jaroszewski, W. Li, and A. Godzik. Comparison of sequence profiles. strategies for structural predictions using sequence information. *Protein Sci.*, 9(2):232–241, 2000.
- [24] A. Sali and T.L. Blundell. Comparative modelling by satisfaction of spatial restraints. *J Mol Biol*, 234(3):779–815, 1993.
- [25] J. Shi, T.L. Blundell, and Mizuguchi K. Fugue: sequence-structure homology recognition using environment-specific substitution tables and structure-dependent gap penalties. *J Mol Biol*, 310(1):243–257, 2001.
- [26] N. Siew, A. Elofsson, L. Rychlewski, and D. Fischer. Maxsub: An automated measure to assess the quality of protein structure predictions. *Bioinformatics*, 16(9):776–785, 2000.
- [27] B. Wallner and A. Elofsson. Can correct protein models be identified? *Protein Sci*, 12(5):1073–1086, May 2003.
- [28] B. Wallner and A. Elofsson. Can correct protein models be identified? *Protein Sci*, 12(5):1073–1086, May 2003.
- [29] B. Wallner, H. Fang, and A. Elofsson. Automatic consensus-based fold recognition using Pcons, ProQ, and Pmodeller. *Proteins*, 53 Suppl 6:534–541, 2003.
- [30] J. Xu, M. Li, D. Kim, and Y. Xu. RAPTOR: optimal protein threading by linear programming. *J Bioinform Comput Biol*, 1(1):95–117, Apr 2003.
- [31] A. Zemla, C. Veclavas, J. Moul, and K. Fidelis. Processing and analysis of CASP3 protein structure predictions. *Proteins*, Suppl 3:22–29, 1999.
- [32] Y. Zhang and J. Skolnick. Scoring function for automated assessment of protein structure template quality. *Proteins*, 57(4):702–710, Dec 2004.