Abstract

Motivation: Apart from meta-predictors, most of today’s methods for residue–residue contact prediction are based entirely on Direct Coupling Analysis (DCA) of correlated mutations in multiple sequence alignments (MSAs). These methods are on average ~40% correct for the 100 strongest predicted contacts in each protein. The end-user who works on a single protein of interest will not know if predictions are either much more or much less correct than 40%, which is especially a problem if contacts are predicted to steer experimental research on that protein.

Results: We designed a regression model that forecasts the accuracy of residue–residue contact prediction for individual proteins with an average error of 7 percentage points. Contacts were predicted with two DCA methods (gplmDCA and PSICOV). The models were built on parameters that describe the MSA, the predicted secondary structure, the predicted solvent accessibility and the contact prediction scores for the target protein. Results show that our models can be also applied to the meta-methods, which was tested on RaptorX.

Availability and implementation: All data and scripts are available from http://comprec-lin.iiar.pwr.edu.pl/dcaQ/.

Contact: malgorzata.kotulska@pwr.edu.pl

Supplementary information: Supplementary data are available at Bioinformatics online.

1 Introduction

Prediction of residue–residue contacts from sequence information alone has been a challenge in protein structure bioinformatics for a long time ( Göbel et al., 1994). Most often such contacts are predicted to support modelling of the three dimensional structure of a protein (Bohr et al., 1993; Konopka et al., 2014; Saitoh et al., 1993; Skolnick et al., 1997; Vendruscolo et al., 1997), but other fields like protein interface prediction (Du et al., 2016; González et al., 2013; Horn et al., 1998), protein structure alignment (Terashi and Takeda-Shitaka, 2015), or even experimental protein structure determination and validation (Olive et al., 1999; Wozniak et al., 2017) can benefit from contact prediction too. The importance of residue–residue contact prediction is further underlined by the fact that it is one of the challenges in the bi-annual CASP (Critical Assessment of Techniques for Protein Structure Prediction) (Monastyrskyy et al., 2014) event since 1996 (Lesk, 1997). Typically, a contact is defined as two amino acids that have either their Cα or Cβ atoms within a specific distance, usually between eight and twelve Ångström. These definitions gave the best results in protein structure reconstruction from contact map predictions (Duarte et al., 2010).

Today’s contact prediction methods all start with the analysis of correlated mutations in a multiple sequence alignment (MSA). An MSA is a set of homologous sequences in which evolutionary related residues are lined up at the same position. The main assumption of
methods based on correlated mutations is that if two sequence positions together are important for protein structure and the residue at one of the two positions mutates, then the one at the other position should mutate as well, to maintain proper interactions (Morcos et al., 2013). Unfortunately, two residue positions that together are important for the protein function will show such behaviour as well (Oliveira et al., 2003), and so will residues that do not interact but both contact the same third residue (transitivity of correlations (Marks et al., 2011)). Contact prediction methods need to cope with such problems. Many of the methods that have been proposed over the years use machine learning techniques such as support vector machines (Cheng and Baldi, 2007; Gonzalez et al., 2013), neural networks (Di Lena et al., 2012; Ding et al., 2013; Kukic et al., 2014; Tegge et al., 2009; Xue et al., 2009), hidden Markov models (Bjorkholm et al., 2009; Bystroff et al., 2000; Pollastri and Baldi, 2002), or random forests (Li et al., 2011; Wang, 2011; Wang and Xu, 2013). Today’s methods with the highest residue-residue contact prediction accuracy, though, are based on Direct Coupling Analysis (DCA) which infers interactions from observations of correlated mutations using so called model learning and inference (Kappen and Rodriguez, 1998; Wainwright and Jordan, 2008).

Most of the DCA-based methods deduce direct interactions between sequence positions using a Potts model (Cocco et al., 2013; Ekeberg et al., 2013; Feinauer et al., 2014; Morcos et al., 2011; Skwark et al., 2013; Zhang et al., 2016), but other statistical approaches proved their usefulness too, e.g. the sparse inverse covariance estimation in PSICOV (Jones et al., 2012).

Neither machine learning, nor DCA-based methods work well enough yet for reliable ab initio protein structure prediction, albeit that the performance occasionally is impressive: e.g. in transmembrane protein structure prediction (Hopf et al., 2012; Wang and Barth, 2015), RNA secondary structure prediction (De Leonards et al., 2015), or protein-protein interaction prediction (Guo et al., 2015; Horn et al., 1998; Isete et al., 2015). Excluding meta-servers, the accuracy of today’s best methods is on the order of 40% for the 100 contacts that are predicted with the greatest confidence (Feinauer et al., 2014). Nevertheless, a recent study indicated great potential for modelling completely unknown protein folds only from contacts obtained from correlated mutation analyses (Ovchinnikov et al., 2017). Several studies have shown that even scarce correct contact information obtained from a known structure can be enough to reconstruct that structure with good quality (Duarte et al., 2010; Dyrka et al., 2016; Konopka et al., 2014; Sathyapriya et al., 2009). Obviously, an unofficial competition is ongoing to increase the prediction accuracy but results presented in new publications often differ only little from previous ones (Cocco et al., 2013; Ekeberg et al., 2013; Feinauer et al., 2014; Morcos et al., 2011; Skwark et al., 2013; Zhang et al., 2016).

A major problem for the end-user of DCA-based methods is that the reported percentage of correctly predicted contacts is an average over a large test set. This implies that the number of correctly predicted contacts can be much higher but also much lower than 40%. Especially the latter can be a problem (if unnoticed) for scientists who use contact predictions to steer their research on one individual protein. We therefore designed a method to forecast the accuracy of residue-residue contact predictions for individual proteins.

We analyzed a large number of parameters that describe the MSA, the predicted secondary structure, the predicted solvent accessibility, and the contact prediction scores for the target protein. We built regression models to test how well (combinations of) these parameters forecast the accuracy of two well-working, readily accessible, representative DCA-based methods: gpmlDCA (Feinauer et al., 2014) and PSICOV (Jones et al., 2012). gpmlDCA is representative for a large set of methods based on the Potts model, while PSICOV uses sparse inverse covariance estimation to deduce direct information from correlated mutations. When only information independent of the DCA-based method is used, the contact prediction accuracy can be forecasted with an average error of about 11 percentage points. When parameters that describe the contact prediction output are included, this becomes about 7 percentage points.

A forecast of the prediction accuracy will be helpful for research in all fields where DCA-based contact prediction methods play a role, including research on the contact prediction methods themselves.

2 Materials and methods

We used regression models on a large number of parameters that describe the MSA, the predicted secondary structure, the predicted solvent accessibility, and the contact prediction scores for the target protein. We analyzed which parameters should be combined to obtain the best model to forecast the quality of contact prediction. The regression models were trained and tested on multiple sets of proteins with known structures.

The 24 parameters that contributed most to the performance of the regression models are described in this Methods section. Other parameters that were tested, but for various reasons were rejected, are listed in the Supplementary Materials (SM)—Methods.

2.1 Training and test data

The training set consisted of 1128 domains selected from SCOP (Murzin et al., 1995). These domains were superfamily representatives that had no alternate side chain conformations, no missing residues inside the chain, and only contained the 20 canonical amino acid types. The domains in the training set were structurally diverse with an average pair-wise TM-score (Zhang and Skolnick, 2004; Zhang and Skolnick, 2005) of 0.34 ± 0.08.

The test set was a subset of the 804 protein chains used by the gpmlDCA authors to test their method (Feinauer et al., 2014). The list is available online at http://gpmldca.aurell.org/input-data. All PDB chains that were structurally similar to any of the domains in the training set (TM-score 0.7 or higher) were removed from this list. The final test set contained 446 PDB chains. The average pair-wise TM-score for this final test set was 0.32 ± 0.06. The average pair-wise TM-score between the training set and the test set was 0.33 ± 0.06.

2.2 Contact definition

Two residues are in contact when their Cα-Cα distance is less than 8.0 Å. However, if they are separated in the sequence by less than four positions they are rejected from the calculations.

2.3 Secondary structure and solvent accessibility

Secondary structure and solvent accessibility were calculated for each domain in the training set with DSSP.2.2.1 (Kabsch and Sander, 1983; Touw et al., 2015). Secondary structure and solvent accessibility for each protein chain in the test set were predicted with Spro and ACCpro20, respectively, that are both part of the SCRATCH 5.2 software package (Magnan and Baldi, 2014). We used the get_abinitio_predictions script that is provided with this software. This script ensures that the software does not make use of homologous structures.
ACCpro2D predicts relative solvent accessibilities. Relative solvent accessibilities were therefore also used for the training set instead of the absolute solvent accessibilities as provided by DSSP. The relative solvent accessibility is defined as the observed solvent accessible surface area divided by the maximally possible solvent accessible surface area and is given as a percentage. The maximal solvent accessible surface area for each residue type was set at the largest value observed (for each residue type) in the DSSP files of the training set (see Supplementary Table S26 in SM—Results). Residues are considered buried when their relative solvent accessibility is less than 20% (Chen and Zhou, 2005) otherwise they are called exposed.

2.4 MSA
MSAs for the test set have been made available online by the authors of gplmDCA (Feinauer et al., 2014) (http://gplmdca.aurell.org/input-data). MSAs for the training set were calculated following the same procedure as for the test set (Feinauer et al., 2014) except that HHblits (Remmert et al., 2011) from the HHsuite 2.0.16, used a slightly older version of Uniprot. Five search iterations with an E-value cut-off of 1.0 were performed. MSA filtering was switched off during the search using the -all parameter. The HHblits commands used are given in the SM—Methods.

2.5 Correlation scores
Correlation scores (also called co-evolutionary scores), as given to residue pairs by gplmDCA or PSICOV, reflect the chance that a pair of residues is in contact (Feinauer et al., 2014; Jones et al., 2012; Marks et al. 2011; Morcos et al. 2011).

Correlation scores calculated with gplmDCA for the training set were obtained by running gplmDCA with default parameters (\(\lambda_h = 0.01, \lambda_f = 0.01, \lambda_{chi} = 0.001\), reweighting_threshold = 0.1 and M = -1). The scores calculated with gplmDCA for the test set are available online (http://gplmdca.aurell.org/input-data).

Correlation scores calculated with PSICOV for the training and test sets were obtained using default parameters with a target precision matrix density of 3% (-d) and output estimated rather than raw scores (-p). The minimum sequence separation was set to 1 (-s), albeit that pairs with a sequence separation less than 4 were not treated as contacts. PSICOV normally requires a minimal level of sequence diversity in the MSA to calculate correlation scores. We forced PSICOV calculations to also be performed on insufficiently diverse MSAs by setting the MINEFSEQS parameter to override (-o).

2.6 Contact prediction procedure
For each domain and chain, predicted contact pairs were sorted in descending order of correlation score. We call the top \(N\) pairs the strongest predicted contacts. We tried \(N = 20, 50, 100, 200, L/2, L\) and \(3L/2\), in which \(L\) is the length of the target sequence.

2.7 Contact prediction evaluation
To assess the performance of the contact prediction method the positive predictive value (PPV) was used:

\[
PPV = \frac{TP}{(TP + FP)} \times 100\% \quad (1)
\]

in which:

\(TP\) — number of properly predicted contacts (true positives),
\(FP\) — number of non-contacts predicted as contacts (false positives).

This PPV will be referred to as the true PPV, while the PPV that was forecasted will just be called PPV or forecasted PPV.

2.8 Regression models
The regression models built to forecast the PPV of gplmDCA and PSICOV were: Lasso (LASSO), regression trees (TREE), Random Forest (FOREST), Support Vector Machine (SVM) and linear regression models in two setups: with only single parameters (primary terms) as input (LM1) and with all possible products of two parameters included as input too (LM2). Combinations of the most successful models that were also tested were: (i) average of Lasso and Random Forest (avLF), (ii) average of SVM and Random Forest (avSF) and (iii) average of Lasso, SVM and Random Forest (avLSF). Detailed information on the training and testing of regression models is available in the SM—Methods.

2.9 Input parameters
The parameters that were used as input for the regression models can be grouped in 4 sets as:

A. MSA;
B. secondary structure;
C. solvent accessibility;
D. correlation score.

The 24 parameters that were used to train regression models are listed below. Descriptions of rejected parameters are available in the SM—Methods.

A. MSA parameters:
1. \(seqNmb\) — target sequence length;
2. \(MSAseq\) — number of sequences in the MSA;
3. \(gapArea\) — number of gaps in the MSA defined as:

\[
gapArea = \frac{gapNmb}{(L \cdot seqNmb)} \quad (2)
\]

in which:

\(gapNmb\) — number of gaps in the MSA,
\(seqNmb\) — number of sequences in the MSA,
\(L\) — target sequence length;
4. \(aveBlock\) — average gap length in the MSA defined as:

\[
aveBlock = \frac{(\sigma_{blocks} \cdot L) \cdot 100}{seqNmb} \quad (3)
\]

in which:

\(\sigma_{blocks}\) — average gap length in the MSA,
\(seqNmb\) — number of sequences in the MSA,
\(L\) — target sequence length;
5. \(identScoreAve\) — average sequence identity between the target sequence and all sequences in the MSA (Bartona, 2008).

B. Secondary structure parameters:
1. \(helixAll\) — number of occurrences of helical residues in the top \(N\) pairs divided by the total number of helical residues in the target sequence (note that a residue can be counted more than one time);
2. \(otherAll\) — number of occurrences of non-\(\alpha\) and non-\(\beta\) residues in the top \(N\) pairs divided by the total number of non-\(\alpha\) and non-\(\beta\) residues in the target sequence (note that a residue can be counted more than one time);
3. \(betaUnique\) — fraction of \(\beta\)-strand residues that participate in the top \(N\) pairs;

Forecasting residue–residue contact prediction accuracy 3407

Downloaded from https://academic.oup.com/bioinformatics/article-abstract/33/21/3405/3892393 by guest on 01 May 2019
4. \textit{otherUnique}—fraction of non-\(\alpha\) and non-\(\beta\) residues that participate in the top \(N\) pairs.

C. Solvent accessibility parameters:

1. \textit{accBuriedAll}—number of occurrences of buried residues in the top \(N\) pairs divided by the total number of buried residues in the target sequence (note that a residue can be counted more than one time);

2. \textit{accBuriedUnique}—fraction of buried target sequence residues that are involved in the top \(N\) pairs;

3. \textit{accExposedUnique}—fraction of exposed target sequence residues that are involved in the top \(N\) pairs;

4. \textit{accBurBur}—fraction of the top \(N\) pairs in which both residues are buried;

5. \textit{accExpExp}—fraction of the top \(N\) pairs in which both residues are exposed;

6. \textit{accExpBur}—fraction of the top \(N\) pairs in which one residue is buried and the other is exposed;

7. \textit{accCV}—coefficient of variation for the average relative solvent accessibility in the top \(N\) pairs defined as:

\[
\text{accCV} = \frac{\sigma_{\text{RSA}/2}}{\mu_{\text{RSA}/2}}
\]  

in which:

\(\sigma_{\text{RSA}/2}\)—standard deviation of average relative solvent accessibility calculated for each of the top \(N\) pairs, 
\(\mu_{\text{RSA}/2}\)—average of average relative solvent accessibility calculated for each of the top \(N\) pairs.

The average relative solvent accessibility for a predicted contact is defined as:

\[
\text{aveRSA} = \frac{(\text{RSA}_1 + \text{RSA}_2)}{2}
\]

in which:

\(\text{RSA}_1\)—relative solvent accessibility of the first residue in the predicted contact, 
\(\text{RSA}_2\)—relative solvent accessibility of the second residue in the predicted contact.

8. \textit{accGlobal}—average relative solvent accessibility of all residues in the target sequence.

D. Parameters that describe the variation and distribution of correlation scores calculated for the top \(N\) pairs:

1. \textit{standDev}—standard deviation of the correlation scores;

2. \textit{corSpread}—difference between the maximal and minimal correlation score divided by the maximal correlation score:

\[
\text{corSpread} = \frac{(\text{Cor}_{\text{max}} - \text{Cor}_{\text{min}})}{\text{Cor}_{\text{max}}}
\]  

in which:

\(\text{Cor}_{\text{max}}\)—correlation score for the strongest predicted contact in the top \(N\) pairs, 
\(\text{Cor}_{\text{min}}\)—correlation score for the \(N\)th strongest predicted contact;

3. \textit{IQR}—half of the interquartile range defined as:

\[
\text{IQR} = \frac{(Q_1 - Q_3)}{2}
\]

in which:

\(Q_1\)—first quartile of the correlation scores, 
\(Q_3\)—third quartile of the correlation scores;

4. \textit{robIQR}—robust coefficient of half of the interquartile range defined as:

\[
\text{robIQR} = \sum_{i=1}^{N} |\text{Cor}_i - \text{IQR}| / (N \cdot \text{IQR})
\]

in which:

\(\text{Cor}_i\)—correlation score for the \(i\)-th strongest predicted contact, 
\(\text{IQR}\)—half of the interquartile range as defined in Eq. 7;

5. \textit{CV}—coefficient of variation for correlated scores defined as:

\[
\text{CV} = \frac{\text{standDev}}{\mu}
\]

in which:

\(\text{standDev}\)—standard deviation of the correlation scores, 
\(\mu\)—average of the correlation scores;

6. \textit{skewness}—skewness of the distribution of correlation scores;

7. \textit{sepReduction}—number of strongest predicted contacts that were skipped because of the sequence separation threshold while going down the list to select the top \(N\) pairs.

2.10 Model performance evaluation

The root mean square error (RMSE) and R-squared statistics \((R^2)\) \cite{James2013} were used to measure the performance of regression models. RMSE describes the average difference between the estimated and the true value of the response variable:

\[
\text{RMSE} = \sqrt{\frac{1}{S} \sum_{i=1}^{S} (y_i - f(x_i))^2}
\]

in which:

\(S\)—number of samples in the test set (here number of protein chains in the test set), 
\(y_i\)—true value of the response variable (here true PPV) for the \(i\)-th sample, 
\(x_i\)—vector of input variables (here the 24 parameters) for the \(i\)-th sample, 
\(f(x_i)\)—value predicted by the model (here forecasted PPV) for \(x_i\).

The true PPV and forecasted PPV are both represented using percentages. Therefore, the RMSE value is expressed in percentage points. \(R^2\) indicates the amount of the total variance in the data that is explained by the model. In general \(R^2\) falls between 0 and 1. In case the model forecasts values worse than the sample average, \(R^2\) can adopt negative values. \(R^2\) is defined as:

\[
R^2 = \frac{TSS - RSS}{TSS} = 1 - \frac{RSS}{TSS}
\]

in which:

\(\text{RSS}\)—residual sum of squares defined as:

\[
\text{RSS} = \sum_{i=1}^{S} (y_i - f(x_i))^2
\]  

\(\text{TSS}\)—total sum of squares defined as:

\[
\text{TSS} = \sum_{i=1}^{S} (y_i - y_{\text{ave}})^2
\]
and:

\[
\begin{align*}
S & \text{— number of samples in the test set (here number of protein chains in the test set),} \\
y & \text{— true value of the response variable (here true PPV) for the } i\text{-th sample,} \\
x & \text{— vector of input variables (here 24 parameters) for the } i\text{-th sample,} \\
y_{\text{ave}} & \text{— average of true values of the response variable (here average of true PPVs)} \\
f(x_i) & \text{— value predicted by the model (here forecasted PPV) for } x_i.
\end{align*}
\]

2.11 Parameter significance evaluation

The significance of parameters was determined as follows:

\[
\% \text{IncRMSE} = \frac{\text{RMSE}_{\text{feat_perm}} - \text{RMSE}_{\text{all_data}}}{\text{RMSE}_{\text{all_data}}} \times 100 \%
\]

in which:

\[
\text{RMSE}_{\text{all_data}} \text{— denotes RMSE based on all input data,} \\
\text{RMSE}_{\text{feat_perm}} \text{— denotes RMSE based on all data but with values of a single feature shuffled randomly among all samples.}
\]

The \%IncRMSE was calculated with 5-fold cross-validation. This means that the model was trained based on 4/5th of the training set and then for every feature \%IncRMSE was calculated based on the remaining 1/5th. This resulted in a single \%IncRMSE value for each input feature. After five repetitions the five \%IncRMSE values were averaged resulting in the finally reported \%IncRMSE for each input feature. \%IncRMSE is very similar to \%IncMSE in the R package randomForest (Liaw and Wiener, 2002).

2.12 Data and code availability

Data collection and parameter calculations were performed with scripts written in Java using BioJava 3.0.7 (Prlić et al., 2012), commons-math3.3.6 and mrs-ws-client 1.0.1 (Hekkelman and Vriend, 2005) libraries. Regression models were built using the R programming language and a series of freely available R packages (Friedman et al., 2010; Liaw and Wiener, 2002; Meyer et al., 2015; Therneau and Atkinson, 2015; Varmuza and Filzmoser, 2009). All scripts and datasets generated during the study are available online at http://comprec-lin.iiar.pwr.edu.pl/dcaQ/.

3 Results

Figure 1 shows the RMSE values for the nine regression models used to forecast the prediction accuracy of gplmDCA. The best results were obtained with the avLSF regression model that calculates the average of the LASSO, SVM and FOREST results. avLSF forecasts the PPV of gplmDCA for \( N = 200 \) with an RMSE of 7.3 percentage points. For \( N = 200 \) the LASSO regression model gives a similar RMSE of 7.6 percentage points. The avLSF model, though, is the most robust of all regression models with regard to different values of \( N \) (see Supplementary Tables S1 and S3 in SM—Results). Figure 2A presents the scatter plot of the PPVs forecasted with the avLSF regression model as a function of the gplmDCA true PPV for the whole test set. The \( R^2 \) value is 0.86 (Supplementary Table S2 in SM—Results). Figure 2B shows, as an example, that in 70% of all cases the forecasted PPV is less than 7.3 percentage points different from the true PPV. It can also be seen that in half of all cases the forecasted PPV error is less than 5 percentage points.

Figure 3 shows the average true PPV of gplmDCA for seven values of \( N \). The true PPV of gplmDCA, like for every DCA-based method, decreases when the number of predicted contacts increases. Figure 3 also shows that the more contacts are predicted the better is the avLSF forecast performance as the RMSE clearly decreases when more contacts are predicted. The same effects are observed when \( N \) is a function of the sequence length \( L \). The RMSEs for \( N = 200 \) and \( N = 3L/2 \) are nearly the same (7.3 and 7.2 percentage points, respectively). The RMSE and \( R^2 \) values for all regression models and all values of \( N \) are presented in Supplementary Tables S1–S4 in the SM—Results.

We examined how our best regression model, which has been trained for gplmDCA, performs with mfDCA—one of the oldest DCA implementations (Morcos et al., 2011) and which applies a mean field approximation. We used an in-house Java implementation of mfDCA with MSAs from Pfam (see SM—Methods) (Finn et al., 2016). Our models, trained for \( N = 200 \) on the contacts predicted with gplmDCA, forecasted the PPV of mfDCA with RMSE = 7.3 percentage points, which is similar to the results obtained for gplmDCA. The \( R^2 \) values, on the other hand, are lower.
for mfDCA than for gplmDCA (see Supplementary Tables S2 and S4 in SM—Results). mfDCA rarely predicts contacts with a PPV better than 50%, so that its TSS value (see Eq. 13) is lower than for the gplmDCA method. If $R^2$ is calculated for gplmDCA using only protein chains for which the true PPV is lower than 50% then the $R^2$ is almost identical to the $R^2$ for the mfDCA case.

These observations suggested that the regression model trained for gplmDCA can be also applied to forecast the PPV of other methods but when we tried to forecast the PPVs of PSICOV, however, the gplmDCA-based regression model performed poorly (Supplementary Tables S1–S4 in SM—Results). This problem is likely caused by the radically different ways by which PSICOV and gplmDCA cope with the transitivity of contacts, and thus by the information the regression models can extract from the method’s outputs. We therefore decided to train a separate regression model for PSICOV. The results of this training are shown in Figure 4. The best results, for $N = 200$, were obtained with the avLF and avLSF models. The avLSF regression model forecasts PSICOV’s PPV with RMSE $= 6.5$ percentage points. The best model for PSICOV, avLF, has RMSE $= 6.4$ percentage points, which is marginally better. For most other values of $N$ the avLSF model works better than the avLF model (see Supplementary Tables S5–S8, S11 and S12 in SM—Methods). We therefore continue the discussion using the avLSF results, also to allow for a better comparison of the results obtained for gplmDCA and PSICOV.

Figure 5A shows the correlation between the true PPVs for PSICOV and the avLSF forecasted PPVs. The $R^2$ value equals 0.86 (Supplementary Table S6 in SM—Results). Figure 5B shows that the regression model forecasts the PPV of PSICOV with an accuracy better than its RMSE (of 6.5 percentage points) for 72% of the proteins in the test set.

The ability to forecast the PSICOV results is slightly better than for gplmDCA. The avLSF model performs better on PSICOV data (RMSE $= 6.5$ percentage points) than on gplmDCA data (RMSE $= 7.3$ percentage points). Figure 5B reveals that avLSF forecasts the PPV of PSICOV with an accuracy better than 7.3 percentage points (the RMSE of gplmDCA) for 79% of all cases.

Figure 6 shows the performance of avLSF models built for PSICOV as function of $N$. This bar plot is highly similar to the one presented in Figure 3 for the gplmDCA results. The RMSE and $R^2$ values for all regression models and all variants of $N$ are presented in Supplementary Tables S5–S8 in the SM—Results.

We examined the importance of individual parameters for the performance of the regression models built for gplmDCA or PSICOV. Table 1 presents the values of $\%\text{IncRMSE}$ for each parameter, each method, and the LASSO, FOREST, SVM and avLSF models. Supplementary Table S13 in the SM—Results contains also the results for other regression models. The higher the value of $\%\text{IncRMSE}$, the more important the parameter is for the model’s performance. Parameters that describe solvent accessibility are highly important for the performance of the gplmDCA and PSICOV regression models. The coefficient of variation for the average

![Fig. 3. Average true PPV of gplmDCA for seven values of $N$: 20, 50, 100, 200, $L/2$, $L$ and $3L/2$, where $L$ is the target sequence length. Error bars indicate the RMSE for forecasting the PPV of gplmDCA using the avLSF model. Average true PPV and error bars are expressed in percentages and percentage points, respectively.](image1)

![Fig. 4. RMSE for forecasting the PPV of PSICOV. Results are for models trained for $N = 200$. RMSE is in percentage points. Note that the ordinate starts at 5](image2)

![Fig. 5. Results for the PSICOV avLSF model trained for $N = 200$. Axes and lines have the same meaning as in Figure 2](image3)

![Fig. 6. Average true PPV of PSICOV for $N$ is 20, 50, 100, 200, $L/2$, $L$ and $3L/2$. Error bars indicate RMSEs for forecasting the PPV of PSICOV using the avLSF model. Average true PPV and error bars are expressed in percentages and percentage points, respectively](image4)
relative solvent accessibility (accCV) is usually found among the five most important parameters. The importance of all other parameter types differs between gplmDCA and PSICOV. For gplmDCA models, secondary structure and MSA parameters are among the five most important ones much more often than for PSICOV models. The length of the target sequence (seqLen) is almost always one of the most significant parameters for gplmDCA models (see also Supplementary Table S13 in SM—Results). For FOREST regression model, the %IncRMSE of seqLen is low, but, on the other hand, it is high for the number of sequences in the MSA (MSAseq). For LASSO, FOREST, SVM and avLSF models, parameters which describe correlation scores are clearly more often important for PSICOV than for gplmDCA models. Especially, the half of the interquartile range (IQR), standard deviation (standDev) and skewness are always among the five most important parameters for PSICOV models. These differences in parameter importance between gplmDCA and PSICOV are likely to be the reason for the necessity to build regression models for PSICOV separately from the gplmDCA models.

Besides the %IncRMSE, for N = 200 we also examined the RMSE of our models when only one single parameter was used, and the correlation between all parameters. The results are presented in Supplementary Tables S14–S17 in SM—Results. accGlobal and accExpExp are among the most important single features (Supplementary Tables S14 and S15 in SM—Results), which was not seen in the %IncRMSE test. This might be caused by their high correlation score (Supplementary Tables S16 and S17 in SM—Results). Similarly, %IncRMSE did not place MSAseq and sepReduction among the important parameters in either model. Finally, for PSICOV’s model permutation of betaUnique lead to an increase of 20% in RMSE (Table 1). In the single feature analysis the importance of betaUnique was not high (Supplementary Table S15 in SM—Results).

We investigated if our regression models are able to forecast the PPV for a target protein independently of the DCA-based method in use. The parameters which are not based on the results of the DCA-based method are the average relative solvent accessibility of all residues in the target sequence (accGlobal in Table 1) and all MSA parameters (grey cells in Table 1). When regression models are trained using only these parameters they perform much worse than when all parameters are used. For example, with N = 200 the gplmDCA avLSF model results in a RMSE of 7.3 (see Fig. 1 and Supplementary Table S1 in SM—Results), which in the absence of other parameters becomes 11.4 percentage points (see Supplementary Table S9 in SM—Results). This test gave similar results for the regression models built for gplmDCA and PSICOV (see Supplementary Tables S9–S12 in SM—Results).

### 4 Discussion

We designed a method to forecast the residue–residue contact prediction accuracy of gplmDCA and PSICOV. The method works well resulting in RMSE values of 7.3 and 6.5 percentage points, respectively, which means that the percentage of correct contacts among the 200 strongest predicted ones can be forecasted with an average error of about 7 percentage points. The best performing regression model is avLSF which is the average of results of LASSO, SVM and FOREST models. This is also the most robust model with regard to different values of N.

The models were trained on SCOP domains and tested on protein chains. Domains and chains often are part of larger proteins with multiple domains or multiple chains in a quaternary structure so that a
(limited) number of residues in both the training set and the test set will have the wrong accessibility value assigned, and Table 1 indicates that accessibility values are important for the performance of the regression models. Fortunately, the SCOP definition of domains (Gough et al., 2001; Murzin et al., 1995) is such that this error is not very big, e.g. in 720 out of 1128 domains in the training set all residues have the same accessibility in the domain as in the full protein. Nevertheless this stays a point of interest for future work.

The avLF model trained for gpmlDCA performed just as well for mDCA despite that there are differences between the mDCA and gpmlDCA algorithms. For example, we trained the models with SCOP domains while Pfam domains were used by the authors of mDCA. SCOP domains might be closer to Pfam domains than to complete chains used in our test set. Also, Pfam alignments usually contain fewer sequences than the alignments by HHblits. Usually, a lower number of sequences in the MSA leads to a lower PPV (Cocco et al., 2013; Feinauer et al., 2014) and our regression models forecast a low PPV better than a high one (see e.g. Fig. 3). Finally, gpmlDCA returns corrected Frobenius norms, which were used as correlation scores to calculate parameters, while mDCA returns direct information values.

Models trained with gpmlDCA forecast the accuracy of mDCA very well, but, on the other hand, they perform poorly for PSICOV. The PSICOV algorithm is so much different from the other DCA algorithms that PSICOV needed its own regression model. Building separate regression models for gpmlDCA and PSICOV showed that different types of parameters were important for these two methods (see Table 1). Two Potts model-based methods and one sparse matrix based method, obviously, provide insufficient examples to allow for extrapolation, but we nevertheless believe that our method generalizes for families of prediction methods, and we believe that our procedure can be applied to other prediction methods that still will be developed. It might be interesting to see if better contact predictions can be obtained when the PSICOV way of dealing with the transitivity of contacts can be combined with the gpmlDCA way of dealing with this problem, and it might be interesting to see how our regression models will perform on such a hybrid method.

Meta-methods perform better than the single methods they combine (Jones et al., 2013; Wang et al., 2017). To make meta-methods perform better beyond a certain limit, though, the underlying non-meta methods should be improved, which is why we focused on gpmlDCA and PSICOV. Nevertheless, we also built models for RaptorX (Wang et al., 2017) which was one of the best methods in CASP11 and CASP12 (Monastyrskyy et al., 2016). RaptorX is a deep learning-metat-method that integrates information such as sequence conservation, residue co-evolution, predicted secondary structure and solvent accessibility. We trained and tested our models following the same procedures as for gpmlDCA and PSICOV. We predicted the PPV of RaptorX for N = 200 and obtained the best RMSE for the avLF model (RMSE = 9.67) and the avLSF model (RMSE = 9.72). Although these results show the potential of our approach for meta-methods, the results are slightly worse than for gpmlDCA or PSICOV, which suggests that meta-methods behave yet differently from the two non-meta methods. One problem might be that RaptorX uses more data types than we cover in the gpmlDCA and PSICOV models, but RaptorX also does certain things different from the other methods (e.g. three-state versus two-state accessibility descriptions) so that we cannot use all information RaptorX uses. The %IncRMSE and single parameter analyses—shown in the Supplementary Tables S22–S24 in SM—Results—show that parameters based on correlation scores have the largest importance for forecasting the RaptorX PPV. Compared to the non-meta methods, RaptorX does not return an actual correlation score but the probability of a residue–residue pair to create a contact. That makes the output more intuitive because the more the distribution of correlation scores is shifted toward 1.0, the higher is the true PPV of RaptorX. This probably causes the corSpread, skewness and CV parameters to be the most significant. All the results for RaptorX are provided in SM—Results (Supplementary Tables S18–S25).

The parameters that describe the solvent accessibility of the top N pairs were most often observed among the most important parameters for the gpmlDCA and PSICOV models (see Table 1). This is very much related to the fact that if a contact is predicted between buried residues, this contact is more likely to be a true contact (due to the geometry of the protein core). Parameters related to correlation scores are also important, especially, for forecasting the performance of PSICOV. When the model has to forecast the contact prediction accuracy from a low number of predicted contacts, it receives fewer data about the distribution of correlation scores, which is bad for the forecasting performance. That is in agreement with Figure 3 and Figure 6 showing that the performance of models depends on the number of predicted contacts. The more contacts were predicted, the better was the estimation of the forecasting accuracy.

We trained our models using solvent accessibility and secondary structures provided with DSSP and tested them using values predicted with the Sspro and ACCpro20 methods. The performance of such predictors is continuously increasing and each research group working on DCA-based methods and interested in using our method will use their own predictors for this purpose. Therefore, we decided to use the perfect, DSSP-based values for training, and any well-performing method available for testing. We did investigate the performance when both training and testing were done with Sspro and ACCpro20, or when training and testing were done both using DSSP, and obtained marginally better RMSE values of 7.1 and 6.5, respectively.

We have shown that the quality of DCA-based contact predictions can be forecasted remarkably well from a series of parameters that can be extracted from the MSA, the predicted secondary structure, the predicted solvent accessibility, and the contact prediction scores for the target protein. We believe that knowledge about the quality of contact predictions will greatly enhance their usefulness in many areas of the life sciences, and might even contribute to improving the DCA-based methods themselves.

Author Contributions
B.M.K. and P.P.W. came up with general concept of prediction methods and designed input parameters for regression models. P.P.W. implemented Java code for processing the data and calculating input parameters for regression models and prepared the draft of the manuscript. B.M.K. implemented R-package code for building, training and testing regression models. J.X. provided the RaptorX data. M.K. and G.V. supervised the project. All authors participated in the study design, data analysis and final manuscript writing.

Acknowledgements
We thank Monika Kurczyńska for producing the SCOP domain training set. Wroclaw Centre for Networking and Supercomputing at Wroclaw University of Science and Technology is also acknowledged.

Funding
The work of P.P.W., B.M.K. and M.K. was partially supported by Statutory Funds. P.P.W acknowledges partial funding from the Polish National Science
References


James, G. et al. (2013) An Introduction to Statistical Learning with Applications in R, Springer-Verlag New York.


Marks, D. S. et al. (2011) Protein 3D structure computed from evolutionary sequence variation. PLoS One, 6, e28766.


