Bioinformatics, 31(8), 2015, 1219–1225
doi: 10.1093/bioinformatics/btu794
Advance Access Publication Date: 8 December 2014
Original Paper

Structural bioinformatics

Clustering-based model of cysteine co-evolution improves disulfide bond connectivity prediction and reduces homologous sequence requirements

Daniele Raimondi[1,2,3,†], Gabriele Orlando[1,2,3,†] and Wim F. Vranken[1,2,3,*]

[1]Interuniversity Institute of Bioinformatics in Brussels, ULB-VUB, La Plaine Campus, Triomflaan, CP 263, [2]Structural Biology Brussels, Vrije Universiteit Brussel, Pleinlaan 2 and [3]Structural Biology Research Center, VIB, 1050 Brussels, Belgium

*To whom correspondence should be addressed.

[†]The authors wish it to be known that, in their opinion, the first two authors should be regarded as Joint First Authors.

Associate Editor: John Hancock

Received on June 18, 2014; revised on October 3, 2014; accepted on November 18, 2014

Abstract

Motivation: Cysteine residues have particular structural and functional relevance in proteins because of their ability to form covalent disulfide bonds. Bioinformatics tools that can accurately predict cysteine bonding states are already available, whereas it remains challenging to infer the disulfide connectivity pattern of unknown protein sequences. Improving accuracy in this area is highly relevant for the structural and functional annotation of proteins.

Results: We predict the intra-chain disulfide bond connectivity patterns starting from known cysteine bonding states with an evolutionary-based unsupervised approach called Sephiroth that relies on high-quality alignments obtained with HHblits and is based on a coarse-grained cluster-based modelization of tandem cysteine mutations within a protein family. We compared our method with state-of-the-art unsupervised predictors and achieve a performance improvement of 25–27% while requiring an order of magnitude less of aligned homologous sequences ($\sim 10^3$ instead of $\sim 10^4$).

Availability and implementation: The software described in this article and the datasets used are available at http://ibsquare.be/sephiroth.

Contact: wvranken@vub.ac.be

Supplementary information: Supplementary material is available at *Bioinformatics* online.

1 Introduction

Cysteine amino acids are extremely relevant for proteins from both structural and functional perspectives. Their side chain ends in a thiol group that, in an oxidative environment, may covalently bond with other cysteines to form disulfide bonds within the protein or to other proteins. These non-local interactions pose structural constraints that reduce the conformational freedom of proteins, thus affecting their stability, maturation and folding rate, inducing conformational changes and ultimately influencing their function (Inaba, 2010; Singh, 2008; Wedemeyer *et al.*, 2000).

Next-generation sequencing technology is producing a flood of new and unannotated protein sequences. There are commonly two prediction steps that can be taken to annotate cysteines in these sequences. The first one is to predict whether their thiol group is in a bound (oxidized) or free (reduced) state, which can help in the functional and structural characterization of proteins (Singh, 2008; Tsai *et al.*, 2007). The second one is to reconstruct the most probable disulfide bonding pattern, which represents the *connectivity* of the covalent bonds formed between the oxidized cysteines within one sequence (intra-chain). This prediction step is important, as knowledge of these connectivities provides clues to the protein fold and structural similarity to other proteins (Chuang *et al*., 2003; van Vlijmen *et al*., 2004), and in *ab-initio* structure prediction, it adds long-range constraints and reduces the conformational space to be searched (Tsai *et al*., 2007). Most of the state-of-the-art prediction tools follow this division in two steps; depending on the case, they can deal with the oxidation states alone (Martelli *et al*., 2002), the connectivity prediction alone (and thus assuming perfect knowledge of the oxidation states) (Fariselli and Casadio 2001; Rubinstein and Fiser 2008; Vullo and Frasconi 2004) or both (Ceroni *et al*., 2006; Ferrè and Clote 2005; Savojardo *et al*., 2011), which is the ultimate goal of this field.

Rubinstein and Fiser (2008) already observed that methods for the prediction of cysteine oxidation states have a long history of high performances: more than 10 years ago, 88% of accuracy was already reached (Martelli *et al*., 2002), and recently, 93% of cysteines were correctly classified, with 86% of proteins predicted correctly (Savojardo *et al*., 2011). These approaches are distinct from the prediction of disulfide bond connectivity, where typically accuracies of up to 50–60% are reached starting from known oxidation states. The machine learning (ML) methods commonly used to predict connectivity can be improved by coupling them with statistical approaches based on the analysis of correlated mutations between cysteine positions in multiple sequence alignments (MSAs). As shown in Rubinstein and Fiser (2008), the information produced by these statistical approaches is to a certain extent orthogonal and thus complementary to ML methods, which can natively integrate these predictions as new dimensions in the feature vectors. In particular, Savojardo *et al*. (2013) show that correlated cysteine mutation analysis can provide a 10% improvement over ML performances.

The principle underlying methods based on the statistical analysis of MSAs is that, generally, residues that experience spatial proximity in the three-dimensional protein structure tend to coevolve during the evolution of protein families, so preserving the propensity of their interaction (Ekeberg *et al*., 2013; Gobel *et al*., 1994; Marks *et al*., 2011; Morcos *et al*., 2011; Schug *et al*., 2009). Bonded cysteines are no exception to this behavior: their structural and functional relevance leaves a co-evolutionary trace among the cysteine-harboring positions in an MSA of homologous sequences in the form of 'tandem' (compensatory) mutations between contacting pairs (Rubinstein and Fiser 2008). Evolution thus provides MSAs with 'clues' of the protein three-dimensional structure, but this co-evolutionary signal is confounded by the entanglement of spurious (transitive network-mediated) correlations and entropic/phylogenetic biases, mostly due to the sampling of the sequences during alignment (Dunn *et al*., 2008).

Different algorithms have been developed to extract the valuable co-evolutionary signal from its noisy background (Ekeberg *et al*., 2013; Jones *et al*., 2012) and are routinely used in contact prediction (CP). Compared with the more general CP problem, the disulfide connectivity prediction presents a problem of lesser difficulty because (i) only a few positions in the MSA have to be analyzed and (ii) cysteines are expected to have a structural or functional impact and thus a strong correlation trace through evolution. Disulfide connectivities have been predicted from MSAs this way using correlated mutation analysis methods (Rubinstein and Fiser 2008) and using a mutual information (MI) approach combined with a sparse inverse covariance estimation method (Savojardo *et al*., 2013) derived from the CP tool PSICOV (Jones *et al*., 2012), so improving the performances of their full annotation predictor (Savojardo *et al*., 2011).

Although these approaches are effective, they do not take into account, for example, the evolutionary distances between the sequences or any biological reasoning except for the compensatory mutation process. Here, we address this problem through Sephiroth, an unsupervised predictor that infers the most likely intra-chain disulfide bond connectivity patterns starting from known or predicted cysteine bonding states. Sephiroth (i) relies on high-quality MSAs obtained with the fast and accurate hidden Markov model alignment method HHblits (Remmert *et al*., 2012) and (ii) is based on a novel coarse-grained cluster-based modelization of tandem cysteines mutations within a protein family. We compared our method with state-of-the-art unsupervised predictors on the same dataset and achieve a performance improvement of 25–27%, while requiring an order of magnitude fewer aligned homologous sequences (in the order of 10^3 instead of 10^4), thus extending the applicability of these tools in terms of increased range of predictable sequences at a lower computational cost. Sephiroth can be combined with ML methods and in this case provides a 15% improvement compared with other state-of-the-art supervised prediction methods.

2 Materials and methods

2.1 PDBCYS dataset

In this study, we adopted the PDBCYS dataset used in Savojardo *et al*. (2011, 2013), which is available at http://dislocate.biocomp.unibo.it/ dislocate/default/method. The dataset has been built from the Protein Data Bank release of May 2010 and contains 1797 proteins with at least two cysteines each. Only intra-chain disulfide bonds annotations are present in the datasets used in this article, as (i) inter-chain disulfide bonds are considerably rarer and (ii) their prediction is a radically different problem that regards also aspects related to the quaternary structure (Cheng *et al*., 2006).

In the PDBCYS dataset, 276 sequences have only bonded cysteines (15.4%), 1320 contain only free cysteines (73.5%) and the remaining 201 chains have both bond and free cysteine (11.2%). The total number of cysteines is 10 813, of which 7619 are free (70.5%) and 3194 are connected by disulfide bond (29.5%). From a taxonomic point of view, the dataset contains only eukaryotic sequences, of which 44.8% belong to *Homo sapiens*. The other most represented species are *Saccharomyces cerevisiae* (10%) and *Mus musculus* (7.7%). Finally, the dataset contains 100 proteins with two disulfide bonds, 85 proteins with three bonds, 41 with four, 37 with five and 17 with six.

2.2 SPX dataset

We also tested our predictor on another dataset taken from literature, called SPX (Cheng *et al*., 2006). It contains 1018 proteins with known 3D structures that are annotated for disulfide bonding and connectivity. Seventy proteins in SPX are also present in PDBCYS and we removed them obtaining a final independent dataset of 948 proteins ('IND-SPX').

In PDBCYS, the shortest protein is 40 amino acids long, whereas IND-SPX contains 161 peptides shorter than 40 residues. We built another dataset by removing from IND-SPX all the peptides shorter than 40 residues, obtaining the 'IND-noFrag' subset of 787 proteins.

2.3 Sephiroth

The pipeline starts from (i) the 'query' (or 'target') protein sequence for which we want to predict the disulfide connectivity and (ii) knowledge of the oxidation states of the cysteines in that

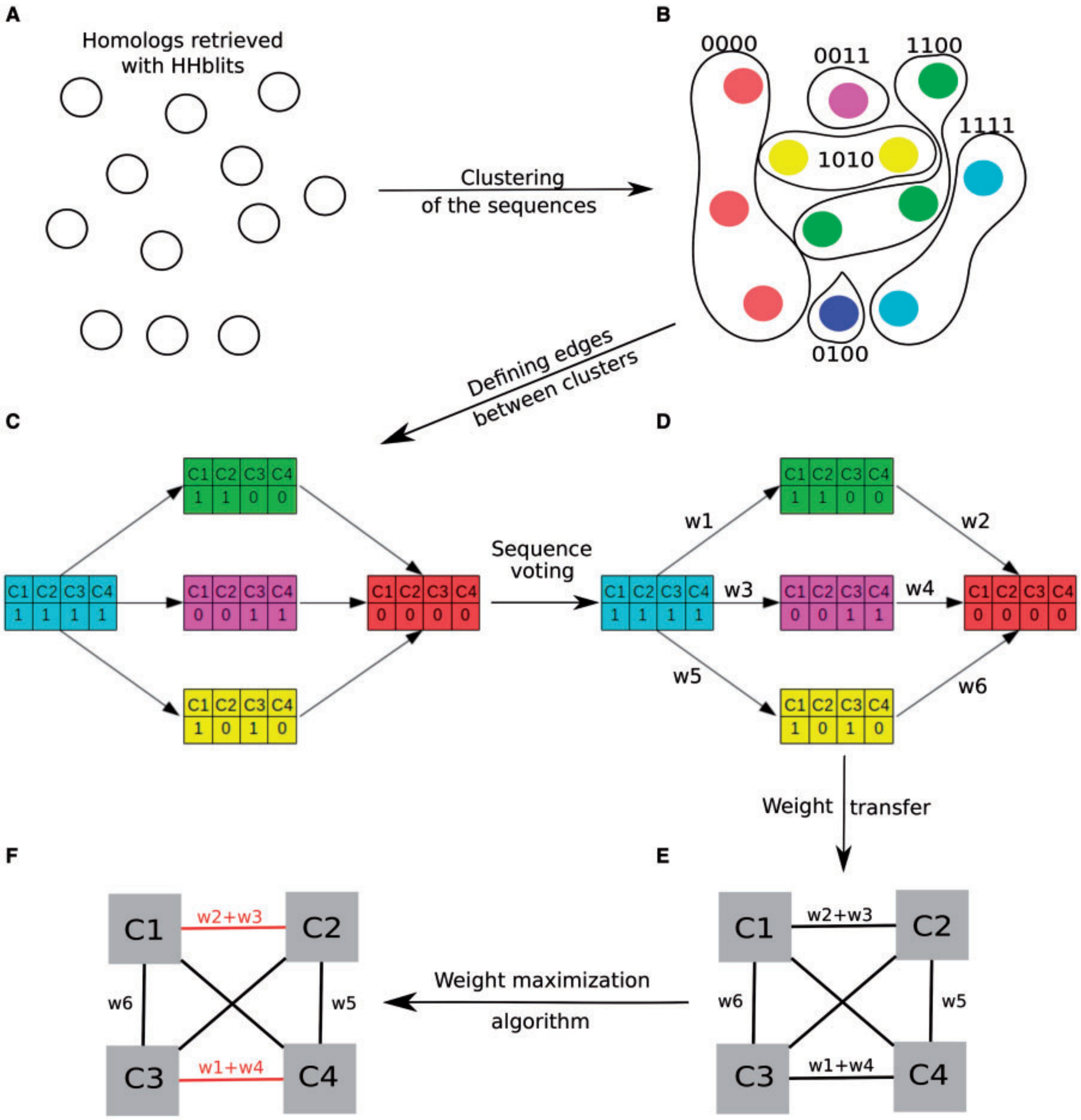

Fig. 1. Overview of the steps in the Sephiroth pipeline

query protein. It consists of the following steps, also shown in Figure 1.

2.3.1 MSA with HHblits

An MSA in which the query protein is aligned with homologous sequences is computed using HHblits (Remmert *et al.*, 2012) (Fig. 1A). The results are reformatted and trimmed to obtain a file containing one aligned sequence for each row. The iteration and *E* value parameters may vary and are discussed in Section 3.

2.3.2 Clustering the sequences

The sequences in the MSA obtained from the previous step are labeled in function of the presence/absence of cysteines at the positions of the query sequence that contain bonded cysteines, and, as shown in Figure 1B, these labels are used to partition the protein family into subsets. Any non-bonded cysteine in the query sequence is treated as any other amino acid and is thus ignored in this procedure. The simple algorithm described here ought not be confused with clustering algorithms based on evolutionary principles that are used for phylogenetic purposes on protein families. We will, however, refer to our subsets as *clusters*, because they provide a partitioning of the sequences (based on label similarity) that is crucial for the following steps of the prediction pipeline.

The clustering of the sequences is obtained by first assigning a binary label to each sequence that represents the ‘cysteine presence/absence footprint’. This binary label has a constant length equal to the total number of bonded cysteines in the query sequence and contains only ‘1’ (cysteine present) and ‘0’ (cysteine absent) symbols. For example, if the query sequence contains four bonded cysteines at positions $(C_1 = 14, C_2 = 29, C_3 = 56, C_4 = 90)$ and an homologous sequence S_1 has cysteines at positions C_2 and C_4 but different amino acids at C_1 and C_3, the binary label assigned to S_1 is ‘0101’.

The same binary label can be assigned to many sequences; sequences identified by the same binary label belong to the same cluster (Fig. 1B). The query protein always belongs to cluster containing only '1's, and each defined cluster contains at least one sequence.

2.3.3 Defining edges between clusters

The following step is to define edges between certain pairs of clusters i and j (with $j \neq i$) obtained in the previous step. Each cluster i contains n_i sequences and is identified by its binary label. We assign a direct edge (i, j) between clusters named i and j if j contains exactly two '0's *more* than i, obtaining the directed acyclic 'Cluster Graph' (CLG) shown in Figure 1C. We build this graph by allowing only edges between clusters that are likely to have experienced the disulfide-breaking 'tandem' cysteine mutations that are usually searched for by methods such as MI or Rubinstein and Fiser (2008). Instead of looking directly for these relationships, we *carve* them in the data structure underlying our modeled problem, obtaining a constrained CLG structure on which we can perform the search procedure.

The assumption of tandem mutations of bonded cysteines is a useful modelization, but it is not always respected: MSAs frequently contain sequences with an odd number of cysteines. Because the defined edges are always between nodes with the same type of parity, the clustering procedure actually may provide two disjoint CLGs that separately contain clusters representing sequences with even and odd numbers of cysteines. Both of them are considered and processed in the following steps, but we describe the ideal case (only sequences with an even number of cysteines) for sake of clarity.

2.3.4 Sequences voting and distance measures

Once we have defined the CLG for the clusters, we extract from this data structure information related to the cysteines paired in disulfide bonds. This is done in a two-step procedure explained below: first, we assign weights to any allowed transition (edge) between clusters (the weights w_n in Fig. 1D) and then we use these weights to build a new graph representing the possible disulfide bondings within the query protein (Fig. 1E).

The first step is done through a voting-like procedure. For each cluster i with exiting edges and for each protein sequence $k \in i$, a sequence-based distance function $\mathfrak{D}(\cdot,\cdot)$ is computed between k and each sequence q belonging to the clusters adj(i) adjacent to i (reachable via one edge from i). The distance function $\mathfrak{D}(\cdot,\cdot)$ can in principle be any function that takes as input two protein sequences and returns a measure of their similarity. Here, we adopted the sequence identity between the two proteins a and b, as it has a linear computational cost and we are dealing with already aligned sequences.

For each cluster i and sequence $k \in i$, the weight associated to the edge (i, j) is incremented by one for the cluster $j \in \mathrm{adj}(i)$ that contains the sequence q for which the distance $\mathfrak{D}(k, q)$ is minimal, compared with all other sequences in the adj(i) clusters. Analytically, the weight $w_{i,j}$ between the clusters i and j can thus be computed as

$$w_{i,j} = \sum_{\forall k \in i} \delta(\arg\min_{c \in \mathrm{adj}(i)} \mathrm{clustDist}(k, c), j) \qquad (1)$$

where $\mathrm{clustDist}(k, c) = \{\min(\mathrm{SI}(k, q)), \forall q \in c\}$, $c \in \mathrm{adj}(i)$ are the clusters adjacent to i, SI is the sequence identity function and $\delta(x, y)$ is the Kroenecker delta function that yields 1 if $x = y$ and 0 otherwise. The relative pseudocode is available in the Supplementary Material.

Every sequence in every cluster i with a non-empty adj(i) set therefore provides an unitary weight to the allowed ($j \in \mathrm{adj}(i)$) edges (i, j); the unitary contributions are then summed, obtaining the global edge weights. In this way, each cluster i 'distributes' its voting power (the number of clustered sequences) by weighting its exiting edges depending on the degree of similarity with the other connected clusters $j \in \mathrm{adj}(i)$. This procedure, coupled with the underlying cysteines tandem mutations assumption encoded in the CLG structure, indicates which cluster contains the closest homologs of i where loss of a disulfide bridge occurred and thus which cysteine pairs are more likely to experience a tandem mutation triggered by the functional disruption of a disulfide bond.

For the second step, we switch from the cluster-oriented CLG (Fig. 1C and D) to a cysteine-oriented graph by building a 'Connectivity Graph' (COG) (Fig. 1E and F) where the nodes represent the oxidized cysteines in the query sequence and the undirected edges between them represent the possible disulfide bonds. Recalling that each 1 or 0 in the label associate to each cluster represent an oxidized cysteine in the query sequence, the weights on the CLG edges (Fig. 1D) can now be transferred to the COG (Fig. 1E) by looking at which cysteine pairs are 'lost' when following each edge. For example, the transition from '0011' to '0000' involves the loss of cysteines in positions 3 and 4 in the binary label, so the weight of the edge between '0011' and '0000' in the CLG is added to the edge between nodes C_3 and C_4 in the COG. The weight for the CLG edge between '1111' and '1100' would equally be added to the same COG edge (The relative pseudocode is available in the Supplementary Material).

2.3.5 Edmonds–Gabow algorithm for maximum weight perfect matching

The weights obtained through the voting procedure on the CLG are then transferred to the COG, in which each edge represents a disulfide bond in the protein. Depending on the quality of the MSA and on the diversity of the sequences sampled from the evolutionary history of the protein family from which it has been generated, the CLG may be an *uninformative* graph, with few or just one node(s). This happens when all the sequences in the MSA belong to the same cluster or when they are grouped in few clusters that cannot be connected with edges following the rules shown in Section 2.3.3. In these pathological cases, it is not possible to assign weights to the CLG and thus to transfer them to the COG, since the retrieved homologous are not diverse enough to provide clues of the connectivity. When no connectivity information is available, the algorithm warns the user and returns a random prediction.

To (i) provide most likely connectivity prediction in the case of informative or partially informative graphs and (ii) pose constraints enforcing the biological meaning on the COG connectivity, we maximized the likelihood of the final inferred disulfide connectivity by applying the Edmonds–Gabow (EG) maximum weight perfect matching algorithm (Gabow, 1976) on the COG. This algorithm selects the edges with the highest weights and provides a final graph in which each node (cysteine) is the end point of one, and only one, edge representing a disulfide bond.

The entire method has been implemented in Python and it is publicly available at: http://ibsquare.be/sephiroth.

2.4 Scoring prediction performances

We measured the performances of our method using the common metrics adopted in the field that are generally derived from the precision scores $\mathrm{Pre} = \mathrm{TP}/(\mathrm{TP} + \mathrm{FP})$, where TP, the true-positive predictions and FP, the false-positive ones. We define these measures of the prediction performances for both (i) the connectivity pattern of the entire protein (Q_p) and (ii) the predictions of the single disulfide bridges (P_b):

- $Q_p = C_{patt}/N_{prot}$ is the fraction of proteins for which the disulfide connectivity has been correctly predicted; C_{patt} is the number of correctly predicted connectivity patterns and N_{prot} the total number of proteins analyzed.
- $P_b = C_{bonds}/N_{bonds}$ is the number of correctly predicted disulfide bridges divided by the total number of observed disulfide bridges.

3 Results and discussion

3.1 Performances in function of the alignment parameters and MSA sizes

We built MSAs against UniRef20 using HHblits (Remmert *et al.*, 2012).As the most direct way to influence the properties and information content of the MSAs used as inputs is by tuning the *E* value and the number of iterations chosen, in Table 1 we show the performance on the PDBCYS dataset in function of these parameters: for each fixed number of iterations we decreased the *E* value, providing more strict thresholds for the false positives and thus decreasing the total number of sequences. Table 1 presents that on average, higher Sephiroth performances are obtained by increasing the number of iterations, so including more distantly related homologs in the MSA. We performed a non-exhaustive search in the space of these parameters: for each number of iterations, we aimed at identifying the optimal *E* value. This heuristic search shows that three iterations with $E\,\text{value} = 10^{-2}$ is the best performing combination of values, but that 2 iterrations $E\,\text{value} = 10^{-5}$ or 4–5 iterrations $E\,\text{value} = 10^{-5}$ perform almost equally well.

The MSAs computed with HHblits and used by Sephiroth are roughly one order of magnitude smaller than the ones found to be optimal for MIp and ICOV (Savojardo *et al.* 2013): in the range of 10^3 sequences instead of 10^4. The performances of Sephiroth when tested on these bigger alignments obtained with JackHmmer (Eddy, 2011) are suboptimal with respect to HHblits and are shown in Supplementary Table S2.

In Supplementary Table S1, the average number of sequences retrieved and aligned by HHblits is shown for different *E* value and iteration parameters combinations. The most populated MSAs, obtained with three iterations and $E\,\text{value} = 10^{-2}$, contain on average 1406 sequences each, with a minimum of 1 sequence (no homologs found) and a maximum of 7966 sequences. The CP literature (Ekeberg *et al.* 2013; Jones *et al.* 2012) points out that the huge requirement in terms of homologous sequences is one of the main drawbacks of co-evolutionary statistical approaches. The possibility to lower these requirements by one order of magnitude opens new horizons for the applicability of these methods in terms of (i) reliably predicting sequences with few homologs and (ii) reduced computational cost of the procedure: if less homologs are needed, faster databases searches can be performed, obtaining overall faster and more accurate predictions.

Table 1. Prediction performances in function of number of iterations and *E* value threshold used to compute HHblits alignments

Alignments		Number of bonds								
		2		3		4		5		All
Iter	*E* value	P_b	Q_p	P_b	Q_p	P_b	Q_p	P_b	Q_p	Q_p
1	10^{+1}	76	76	57	47	57	44	35	16	53
1	10^{-1}	73	73	57	47	56	42	37	16	52
1	10^{-2}	71	71	56	47	60	44	41	19	52
1	10^{-3}	70	70	54	45	54	37	39	16	49
1	10^{-4}	67	67	53	42	55	39	39	11	47
1	10^{-5}	68	68	50	40	54	37	38	11	46
2	10^{-4}	73	73	56	47	62	49	38	13	52
2	10^{-5}	74	74	56	47	58	46	42	22	54
2	10^{-10}	74	74	56	46	53	41	37	13	51
3	10^{-2}	75	75	60	53	60	44	43	22	56
3	10^{-3}	74	74	58	49	57	41	39	16	53
3	10^{-4}	74	74	58	49	60	46	37	16	54
3	10^{-5}	74	74	57	46	58	46	39	22	53
3	10^{-10}	74	74	57	47	54	44	37	13	52
4	10^{-5}	75	75	58	48	59	49	39	22	55
5	10^{-5}	74	74	59	49	57	44	39	19	54

P_b and *Q*prot are presented as percentages for the sake of clarity.

3.2 Comparison with other methods

We benchmarked Sephiroth by calculating its performance on the PDBCYS dataset and comparing to the results obtained by MIp and ICOV, two state-of-the-art existing methods that were also validated on this dataset (Savojardo *et al.*, 2013). MIp is based on the corrected MI between two random variables, which is a non-negative symmetric measure of their dependence. MI estimates the degree of co-mutation between different positions in the MSA by considering the cysteine-containing positions in an MSA as random variables that can assume discrete values within the amino-acids alphabet, with probabilities empirically estimated from the sequence conservation profile information. This measure suffers from some drawbacks due to phylogenetic and entropic biases (Dunn *et al.*, 2008) and needs to be corrected for the background of spurious correlations. The second method, ICOV, is based on a more complex statistical method that takes the network of spurious interaction underlying the co-evolutionary signal into account. It estimates the inverse of a sparse covariance matrix (Jones *et al.*, 2012) to extract the *direct coupling* between the possibly correlated random variables (the positions in the MSA) by removing the confounding factors. We also included a random predictor that randomly guesses the bonds in each protein to obtain a baseline value for the prediction.

In the comparison of prediction performances in Table 2, we focus on the previously defined Q_p scores, which give the percentage of proteins whose connectivity has been predicted entirely correctly: this measure is the most prevalent scoring index in cysteine connectivity prediction. The 'All' Q_p scores averaged over proteins containing two, three, four and five bonds (last column of Table 2) show that Sephiroth performs 27% better than ICOV and 24.4% better than MIp, showing a clear improvement with respect to these methods. If we compare Sephiroth's Q_p scores with the maxima obtained by MIp or ICOV, but now subclassed by number of bonds, we obtain the following improvements: +10% for two bonds, +26% for three bonds, +51.7% for four bonds and +37.5% for five bonds. The difficulty of the prediction rapidly increases with the number of disulfide bonds: in particular, given the performance of the Random predictor, the five-bond prediction appears to be 330 times harder than for two bonds. Our method is, respectively, 127%, 657%, 4300% and 21 900% better than the random predictor for two, three, four and five bonds. Averaged over all the proteins, it performs 273% better than random guesses.

Table 2. Comparison between the prediction performances obtained by Sephiroth, MIp, ICOV and a random baseline predictor (Savojardo *et al.*, 2013) on the PDBCYS dataset, reproducing the same 20-folds scores averaging to account for sampling

Method	Number of bonds								
	2		3		4		5		All
	P_b	Q_p	P_b	Q_p	P_b	Q_p	P_b	Q_p	Q_p
Random	33	33	20	7	14	1	11	0.1	15
ICOV	62	62	53	42	52	27	40	16	44
MIp	68	68	48	38	49	29	34	14	45
Sephiroth[a]	75	75	60	53	60	44	43	22	56
Sephiroth[b]	77	77	61	53	63	45	44	24	57

P_b and Q_p are presented as percentages.

[a] Performances obtained predicting the entire set with three iterations and E value $= 10^{-2}$.

[b] Performances obtained with Sephiroth[a] reproducing the same 20-fold averaging used for ICOV and MIp.

Table 3. Sephiroth performances obtained on SPX dataset (Cheng *et al.*, 2006)

Dataset	Method	Number of bonds								
		2		3		4		5		All
		P_b	Q_p	P_b	Q_p	P_b	Q_p	P_b	Q_p	Q_p
IND-SPX	Random	33	33	20	7	14	1	11	0.1	15
	Sephiroth[a]	60	60	38	30	35	21	47	23	39
	Sephiroth[b]	58	58	32	25	36	22	40	18	36
IND-noFrag	Random	33	33	20	7	14	1	11	0.1	15
	Sephiroth[a]	78	78	54	45	38	23	47	23	51
	Sephiroth[b]	76	76	50	40	40	24	40	18	49

The first two rows show scores when predicting the IND-SPX, and the last two show performances on the subset of IND-SPX obtained after discarding peptide fragments shorter than 40 amino acids (IND-noFrag). P_b and Q_p are shown as percentages.

[a] Sephiroth performances obtained from MSAs calculated with three iterations and E value $= 10^{-2}$.

[b] Sephiroth performances obtained from MSAs calculated with three iterations and E value $= 10^{-5}$.

Table 4. Sephiroth's prediction performances when applied to ML methods with respect to state-of-the-art ML-based methods, SVR and SVR+MIp+ICOV (Savojardo *et al.*, 2013)

Method	Number of bonds								
	2		3		4		5		All
	P_b	Q_p	P_b	Q_p	P_b	Q_p	P_b	Q_p	Q_p
SVR[a]	75	75	60	48	57	44	46	19	54
SVR+MIp+ICOV[a]	76	76	63	55	68	51	59	32	59
skSVR+Sephiroth[b]	86	86	71	64	67	50	66	46	68

Comparison has been made on the same PDBCYS dataset, reproducing the cross-validation with the same 20-folds. P_b and Q_p are presented as percentages.

[a] Scores reported from Savojardo *et al.* (2013).

[b] Performances obtained by adding Sephiroth predictions to an in-house SVR tool for cysteine connectivity predictor during the 20-fold cross-validation.

Sephiroth is an *unsupervised predictor* based on statistical assumptions without learning steps; for this reason, it is not necessary to cross-validate the results to avoid overfitting. The ICOV and MIp performances were measured by averaging the scores in a 20-folds cross-validation-*like* procedure to provide results coherent with the other scores shown in that article. This procedure appears to provide a slight error in the estimation of the performances due to the sampling of the proteins in the different folds and to the subsequent averaging of the scores. To take into account also this effect when comparing the performances, we reproduced the same 20-folds sampling by using the same dataset division and averaging the scores. The positive sampling error appears to provide +2% estimation of the performances yielding approximately 27–30% improvement instead of approximately 25–27%.

3.3 Performances on independent dataset

We also ran Sephiroth on two independent datasets (IND-SPX and IND-noFrag) derived from the original SPX dataset (Cheng *et al.*, 2006) (see Section 2). The prediction performances we obtained are listed in Table 3: when predicting IND-SPX, we obtain a low 'Q_p All' score of 39%. However, 161 of IND-SPX entries are peptides with less than 40 residues, and most of these fragments are too short to (i) produce significant HHblits alignments (the average number of homologs found for these peptides is 81) and to (ii) provide a meaningful sequence identity-based voting procedure in Sephiroth. By removing these short fragments (IND-noFrag), we obtained a subset of 787 proteins with on average 1119 homologs each (the most populated MSA contains 6393 sequences). The prediction performances obtained on IND-noFrag are 31–36% higher than IND-SPX and comparable with the scores obtained on PDBCYS, showing that Sephiroth's performances are reproducible on different dataset and thus based on valid assumptions. Also in this case, Sephiroth is, on average, respectively 2.40 and 2.26 times better than the random predictions.

3.4 Combining Sephiroth with other ML methods

In Table 2, we showed that Sephiroth is able to outperform the other unsupervised prediction methods by providing approximately 25–27% improvement with respect to MIp and ICOV. A comparison between Tables 2 and 4 shows that Sephiroth also positively compares to ML methods based on hundreds of features (Savojardo *et al.*, 2013), as it performs 5.4% better than the support vector regression (SVR) approach.

Sephiroth predictions can also be added to these ML methods; we assessed its contribution by reproducing the SVR approach adopted in Savojardo *et al.* (2013), implementing it in Python with sklearn (Pedregosa *et al.*, 2011). The performance is represented under skSVR+Sephiroth in Table 4.

For each possible pair of cysteines, a window of 13 flanking residues representing the local sequence context of each cysteine is encoded in a feature vector with $2 \times 13 \times 20 = 520$ dimensions. Each flanking residue is represented by a 20-dimensional vector extracted from the sequence profile obtained from the MSA, containing the frequencies of occurrence of each amino acid at that position. We also added three further dimensions to each feature vector: the cysteines sequence separation distance and their relative order (Savojardo *et al.*, 2011).

We then added a single dimension containing Sephiroth bond/free prediction to each of the 523-dimensional feature vectors encoding each pair of cysteines in proteins with 2–5 bonds, and we ran a 20-fold cross-validation using the same fold division shown in (Savojardo *et al.*, 2013).

Adding a single dimension containing Sephiroth's predictions to our sklearn-SVR followed by EG algorithm (skSVR+Sephiroth) produces a 26% improvement with respect to SVR and +15% with respect to SVR+MIp+ICOV, showing the relevance of Sephiroth when it is integrated with ML methods.

We also assessed the performances of Sephiroth starting from predicted oxidations states by reproducing an approach similar to DISLOCATE (Savojardo *et al.*, 2011). The error introduced by the oxidation state prediction translates directly to the Sephiroth performance, showing that the two prediction steps are very close to being independent (see Supplementary Table S3).

4 Conclusion

In this article, we present a novel unsupervised method for the prediction of disulfide bond connectivity patterns from known or predicted cysteine bonding states. It is based on assumptions similar to other statistical methods (cysteine conservation and co-evolution) but approaches the problem from a different angle, directly encoding the cysteine tandem mutations assumption in the structure underlying the problem, thus providing *constraints* for the search of the optimal solutions.

The availability of homologous sequences is crucial for these predictors and we adopted a fast and accurate HMM-based iterative search tool, HHblits (Remmert *et al.*, 2012) to obtain small but highly informative MSAs. Compared with existing predictors, we obtain a significant improvement of the performances, while requiring one order of magnitude fewer aligned homologs. We also show that Sephiroth can provide a relevant improvement over state-of-the-art ML methods if used as extra feature in SVR pipelines for solving the disulfide connectivity prediction task.

The enhanced reliability of the connectivity predictions coupled with lower requirements in terms of MSA sizes (and thus time required by database search) can improve the dependability of bioinformatics methods for the structural and functional annotation of unknown sequences.

Acknowledgements

D.R. and G.O. thank Andrea Gazzo, Piero Fariselli, Castrense Savojardo, Anna Laura Mascagni and Angela Motz for the collaboration and helpful discussions.

Funding

This work was funded by a Ph.D. grant of the Agency for Innovation by Science and Technology (IWT) (to D.R.) and by the Brussels Institute for Research and Innovation (Innoviris) (grant BB2B 2010-1-12 to W.F.V.).

Conflict of Interest: none declared.

References

Ceroni,A. *et al*. (2006) DISULFIND: a disulfide bonding state and cysteine connectivity prediction server. *Nucleic Acids Res.*, **34**, W177–W181.

Cheng,J. *et al*. (2006) Large-scale prediction of disulfide bridges using kernel methods, two-dimensional recursive neural networks, and weighted graph matching. *Proteins*, **62**, 617–629.

Chuang,C.C. *et al*. (2003) Relationship between protein structures and disulfide-bonding patterns. *Proteins*, **53**, 1–5.

Dunn,S.D. *et al.* (2008) Mutual information without the influence of phylogeny or entropy dramatically improves residue contact prediction. *Bioinformatics,* **24**, 333-340.

Eddy,S.R. (2011) Accelerated profile HMM searches. *PLoS Comp. Biol.*, 7:e1002195.

Ekeberg,M. *et al.* (2013) Improved contact prediction in proteins: using pseudolikelihoods to infer Potts models. *Phys. Rev.,* **87**, 012707.

Fariselli,P. and Casadio,R. (2001) Prediction of disulfide connectivity in proteins. *Bioinformatics*, **17**, 957–964.

Ferrè,F. and Clote,P. (2005) Disulfide connectivity prediction using secondary structure information and diresidue frequencies. *Bioinformatics,* **21**, 2336–2346.

Gabow,H.N. (1976) An efficient implementation of Edmunds algorithm for maximum weight matching on graph. *J. ACM*, **23**, 221–234.

Gobel,U. *et al.* (1994) Correlated mutations and residue contacts in proteins. *Proteins,* **18**, 309-317.

Inaba,K. (2010). Structural basis of protein disulfide bond generation in the cell. *Genes Cells,* **15**, 935–943.

Jones,D.T. *et al.* (2012) PSICOV: precise structural contact prediction using sparse inverse covariance estimation on large multiple sequence alignments. *Bioinformatics*, **28**, 184-190.

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

Martelli,P.L. *et al*. (2002) Prediction of the disulfide-bonding state of cysteines in proteins at 88% accuracy. *Protein Sci.*, **11**, 2735–2739.

Morcos,F. *et al.* (2011) Direct-coupling analysis of residue coevolution captures native contacts across many protein families. *Proc. Natl Acad. Sci. USA*., **108**, E1293–E1301

Pedregosa,F. *et al.* (2011) Scikit-learn: machine learning in python. *J. Machine Learn. Res.,* **12**, 2825–2830.

Remmert,M. *et al.* (2012). HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment. Nat. Methods, **9**, 173–175

Rubinstein,R. and Fiser,A. (2008) Predicting disulfide bond connectivity in proteins by correlated mutations analysis. *Bioinformatics*, **24**, 498–504.

Savojardo,C. *et al.* (2011). Improving the prediction of disulfide bonds in eukaryotes with machine learning methods and protein subcellular localization. *Bioinformatics*, **27**, 2224–2230.

Savojardo,C. *et al.* (2013). Prediction of disulfide connectivity in proteins with machine-learning methods and correlated mutations. *BMC Bioinformatics*, **14** (Suppl. 1), S10.

Schug,A. *et al.* (2009) High resolution protein complexes from integrating genomic information with molecular simulation, *Proc. Natl Acad. Sci. USA*, **106**, 22124.

Singh,R. (2008). A review of algorithmic techniques for disulfide-bond determination. *Brief. Funct. Genomic. Proteomic.*, **7**, 157–172.

Tsai,C.H. *et al.* (2007) Bioinformatics approaches for disulfide connectivity prediction. *Curr. Protein Pept. Sci.*, **8**, 243–260.

van Vlijmen,H.W.T. *et al.* (2004) A novel database of disulfide patterns and its application to the discovery of distantly related homologs. *J. Mol. Biol.*, **335**, 1083–1092.

Vullo,A. and Frasconi,P. (2004) Disulfide connectivity prediction using recursive neural networks and evolutionary information. *Bioinformatics,* **20**, 653–659.

Wedemeyer,W.J. *et al.* (2000) Disulfide bonds and protein folding. *Biochemistry*, **39**, 7032.