Deep autoencoders for additional insight into protein dynamics

Mihai Teletin[1], Gabriela Czibula[1], Maria-Iuliana Bocicor[1], Silvana Albert[1], and Alessandro Pandini[2]

[1] Babeş-Bolyai University, Cluj-Napoca
[2]Institute of Environment, Health and Societies, Brunel University London
tmic1334@scs.ubbcluj.ro, {gabis, iuliana, albert.silvana}@cs.ubbcluj.ro, alessandro.pandini@brunel.ac.uk

Abstract. The study of *protein dynamics* through analysis of conformational transitions represents a significant stage in understanding protein function. Using molecular simulations, large samples of protein transitions can be recorded. However, extracting functional motions from these samples is still not automated and extremely time-consuming. In this paper we investigate the usefulness of unsupervised machine learning methods for uncovering relevant information about protein functional dynamics. *Autoencoders* are being explored in order to highlight their ability to learn relevant biological patterns, such as structural characteristics. This study is aimed to provide a better comprehension of how protein conformational transitions are evolving in time, within the larger framework of automatically detecting functional motions.

Keywords: Protein molecular dynamics, Autoencoders, Unsupervised learning.

1 Introduction

Proteins are large biomolecules having crucial roles in the proper functioning of organisms. They are synthesized using information contained within the ribonucleic acid (RNA), when by means of the process known as translation, building blocks, the amino acids, are chained together in a sequence. Although this sequence is linear, the protein acquires a complex arrangement in its physiological state, as intramolecular forces between the amino acids and the hydrophobic effect lead to a folding of the protein into its three dimensional shape, which determines the protein's function [27]. The stable three dimensional structure of a protein is unique, however this shape undergoes significant changes to deliver its biological function, according to various external factors from the protein's environment (e.g. temperature, interaction with other molecules). Thus, a protein will acquire a limited number of conformations during its lifetime, having the ability to transition between alternative conformations [26].

The study and prediction of conformational transitions represents a significant stage in understanding protein function [21]. In this paper we investigate protein molecular motions and conformational transitions starting from the structural alphabet devised by Pandini et al., a representation which provides a highly informative encoding of proteins [22]. In this description, each fragment consists of 4 residues and is defined by three

internal angles: two pseudo-bond angles between the C^α atoms (C^α is the first carbon atom that attaches to a functional group) of residues 1-2-3 and 2-3-4 and one pseudo-torsion angle formed by atoms 1-2-3-4 [22]. These internal angles entirely define each structural fragment which can be also encoded as a letter from a Structural Alphabet (SA) [22]. In addition to the previously mentioned representation based on angles, we investigate whether enhancing the structural alphabet states (represented by the three angles) with relative solvent accessibility information might bring further insight into the matter at hand. *Relative solvent accessibility* (RSA) of amino acid residues is a value indicating the degree to which the residue is exposed [20], being able to characterize the spatial distribution of amino acids in a folded protein. RSA is significant for predicting protein-interaction sites [20] and it is used in protein family classification [1]. The intuition is that, even if RSA values independently do not offer a unique characterization of a protein, being individually non-specific, new structural states defined by the three angles together with RSA values could bring additional information.

Using molecular simulations, large samples of protein transitions can be recorded. However, extracting functional motions from these samples is still not automated and extremely time-consuming. Therefore, we consider that computational methods such as unsupervised learning could be a well suited solution for better understanding protein dynamics. We are investigating the usefulness of deep autoencoder neural networks to acquire a clearer sense of proteins' structure, with the long term goal of learning to predict proteins' conformational transitions. Several approaches in the literature were proposed for analyzing and modeling protein structural conformations using both supervised and unsupervised machine learning techniques. Support vector machine's performance was tested in [14] by classifying gene function from heterogeneous protein data sets and comparing results with various kernel methods. In [28], a Radial Basis Function Network (RBFN) is proposed for classifying protein sequences. Fifteen supervised learning algorithms were evaluated in [9] by automating protein structural classification from pairs of protein domains and *Random Forests* were proven to outperform the others. Additional insight into protein *molecular dynamics* (MD) is gained in [16] by employing L1-regularized reversible Hidden Markov Models. Self-organizing maps have also been used alongside hierarchical clustering in [6], for the purpose of clustering molecular dynamics trajectories. A methodology for detecting similarity between three dimensional structures of proteins was introduced by Iakavidou et al. in [8].

The contribution of the paper is twofold. Our first main goal is to investigate the capability of unsupervised learning models, more specifically of *autoencoders*, to capture the internal structure of proteins represented by their conformational transitions. Secondly, we propose two internal representations for a protein (one using the structural alphabet states defined by three angle values, as introduced in [22] and one in which these states are extended with RSA information) with the aim of analyzing which of them is more informative and would drive an autoencoder to better learn structural relationships between proteins. The experiments performed are aimed at evaluating the extent by which the combination of a reduced representation and an autoencoder is suitable to compress the complex MD data into a more interpretable representation. With this aim we propose a proof of concept that considers only two similar but unrelated proteins where learning on one can be used on the other. The literature regarding protein data

analysis reveals that a study similar to ours has not been hitherto performed. The study can be further extended on a large scale where evolutionary relationship are considered, with the goal of answering how much the "closeness" of proteins in evolutionary space can affect the efficiency of the encoding. To sum up, in this paper we seek answers to the following research questions: **RQ1** What is the potential of *autoencoders* to unsupervisedly learn the structure of proteins and how does the internal representation for a protein impact the learning process?; and **RQ2** Are autoencoders able to capture biologically relevant patterns? More specifically, are our computational findings obtained by answering **RQ1** and **RQ2** correlated with the biological perspective?

The remainder of the paper is organized as follows. The autoencoder model used in our experiments is described in Section 2. Section 3 provides our methodology and Section 4 contains the results of our experiments, as well as a discussion regarding the obtained results, both from a computational and biological perspective. The conclusions of our paper and directions for future work are summarized in Section 5.

2 Autoencoders

Autoencoders were successfully applied in different complex scenarios such as image analysis [13] and speech processing [5]. An autoencoder [7] is a feed forward neural network. The input of the network is a real numbered vector $x \in \mathbb{R}^n$.

An autoencoder is composed of two main components: (1) an encoder: $g : \mathbb{R}^n \rightarrow \mathbb{R}^m, g(x) = h$ and (2) a decoder: $f : \mathbb{R}^m \rightarrow \mathbb{R}^n, f(h) = \widehat{x}$. The two components are stacked together, hence the goal of the autoencoder is to model a function: $f(g(x)) \approx x$. We notice that the input and the label of the model are the same vector. Thus the autoencoders may be considered *self-supervised learning* techniques. If $m < n$ then the autoencoder is called *undercomplete*.

We consider the learning process of autoencoders as minimizing a loss function $L(\widehat{x}, x) = \frac{1}{n}\sum_{i=1}^{n}(\widehat{x}_i - x_i)^2$. The optimization is performed using stochastic gradient descent with backpropagation. One may notice that the goal of the autoencoder is to copy the input x into the output value. However, such a model would not be useful at all. In fact, the goal of the autoencoder is to come up with useful representation of data in the hidden state, h. Good encoded values may be useful for various tasks such as information retrieval and data representation. A *sparse* autoencoder is a technique used to help the model avoid the simple copying of the input to the output by introducing a sparsifying penalty to the loss function. Usually this sparsing penalty is the L1 regularization on the encoded state. The penalty term is scaled using a small real number denoted as λ. Thus the employed loss becomes $L(\widehat{x}, x) = \frac{1}{n}\sum_{i=1}^{n}(\widehat{x}_i - x_i)^2 + \lambda\sum_{i=1}^{m}|h_i|$.

Denoising autoencoders represent another technique to avoid the mere copying of the input data to the output layer, forcing the hidden layers to learn the best defining, most robust features of the input. To achieve this, a denoising autoencoder is fed stochastically corrupted input data and tries to reconstruct the original input data. Thus, in the case of denoising autoencoders the loss function to be minimized is $L(g(f(\widetilde{x})), x)$, where the input given to the autoencoder is represented by $\widetilde{x}$ - input data corrupted by some form of noise [7]. Therefore, the autoencoder will not simply elicit the input data, but will learn a significant representation of it. Various experiments proved that autoencoders are better than *Principal Component Analysis* (PCA) [7]. This is mainly because

autoencoders are not restricted to perform linear mapping. One can consider that a single layer autoencoder with linear activation function has the same capacity as PCA. However, the capacity of autoencoders can be improved by tuning the complexity of the encoder and decoder functions.

3 Methodology

In this section we present the experimental methodology used in supporting our assumption that *autoencoders* can capture, from a computational viewpoint, biologically relevant patterns regarding structural conformational changes of proteins. With the goal of answering the first research questions formulated in Section 1, the experiments will investigate the ability of an *autoencoder* to preserve the structure of a protein. Two types of representations will be considered in order to identify the one that is best suited for the analysis we are conducting. These representations will be detailed in Section 3.1.

3.1 Protein representations

A protein is a macromolecule with a very flexible and dynamic innate structure [18] that changes shape due to both external changes from its environment and internal molecular forces. The resulting shape is a different conformation. For each conformation of a protein, two different representations of the local geometry of the molecule will be used in our study.

The *first* representation for a protein's conformation, which we call the *representation based on angles* (**Angles**), consists of conformational states given by the three types of angles mentioned in Section 1 [22]. In this representation, a conformation of k fragments (letters from the structural alphabet [22]) is represented as $3k$ dimensional numerical sequence. This sequence contains three angles for each fragment from the conformation. The *second* way to represent a protein conformation, named in the following the *combined representation* (**Combined**) is based on enhancing the conformational states given by angles with the RSA values of the amino acid residues (see Section 1). In our second representation, a conformation of k states is visualized as a $4k$ dimensional numerical vector. The first $3k$ positions from this vector contain the conformation's *representation based on angles*, whereas the following k positions contain the RSA values.

3.2 *Autoencoder* architecture

In the current study we use sparse denoising autoencoders to learn meaningful, lower-dimensional representations for proteins' structures, considering their conformational transitions. Hence, the loss function will be computed as shown in Section 2, where $\widehat{x} = g(f(\widetilde{x}))$ and $\widetilde{x}$ represents the corrupted input data. We chose a denoising autoencoder in our experiments, because experimental measurements of biological processes and information generated by particle methods (e.g. MD simulations) can be noisy or subject to statistical errors. We are going to use such an autoencoder in order to reduce the dimensionality of our data. Considering that one of our purposes is to be able to visualize our data sets, all the techniques implied are going to encode the protein representations into 2 dimensional vectors.

The sparse denoising autoencoder learns a mapping function from an n-dimensional space (where n can have different values, according to the employed representation) to

a 2 dimensional hidden state. We performed several experiments, with variable numbers of hidden layers and using various activation functions, in order to reduce dimensionality. More specifically, the activation functions we employed for the hidden layers are: rectified linear unit (ReLU), exponential linear unit (ELU) [4] and scaled exponential linear unit (SELU) [12]. As a regularization strategy, we use the *dropout* technique [24], with dropout rates in $\{0.1, 0.2, 0.3\}$. Since we have only 2 values in the encoded state we are going to use a small value for λ hyperparameter: 10^{-6}. The encoded values are then reconstructed using a similar decoding architecture.

Optimization of the autoencoder is achieved via stochastic gradient descent enhanced with the *adam* optimizer [11]. We employ the algorithm in a minibatch perspective by using a batch size of 16. The batch size affects the performance of the model. Usually, large batch sizes are not recommended since it may reduce the capacity of the model to generalize. Adam is a good optimizer since it also deals with the adjustment of the learning rate. The data set is shuffled and 10% is retained for validation. We keep the best performing model on the validation phase by measuring the validation loss. The loss obtained on the validation set was 0.555 for 1P1L and 0.378 for 1JT8 for the ReLU activation function, with 0.2 dropout rate. Regarding the encoding architecture, we experimented with 2 and 3 hidden layers, containing different numbers of neurons (depending on the size of the input data), and each of the hidden layers benefit from batch normalization. The decoding architecture is similar, having the same dimensions for the hidden layers, but in reverse.

3.3 Evaluation measures

In order to determine whether the representation learned by the autoencoder preserves the similarities found in the original protein data we define the intra-protein similarity measure, *IntraPS*, which evaluates the degree of similarity between conformations within a protein and we will use this as an indication of how well the intra-protein conformational relations are maintained in the lower-dimensional representation learned by the autoencoder. $IntraPS$ is based on the cosine similarity measure, which is employed to evaluate the likeness between two conformations of a protein.

Cosine similarity (*COS*) is widely used as a measure for computing the similarity between gene expression profiles. It is a measure of the direction-length similitude between two vectors and is defined as the cosine of the angle between the high-dimensional vectors. To define the intra-protein similarity measure, we consider that a protein p is represented as a sequence of n conformations, i.e. $p = (c_1^p, c_2^p, \cdots, c_n^p)$. Each conformation c_i^p of the protein is visualized as an m-dimensional numerical vector (i.e the *representation based on angles* or the *combined representation* previously described).

The *Intra-protein similarity* of a protein $p = (c_1^p, c_2^p, \cdots, c_n^p)$, denoted as IntraPS($p$), is defined as the average of the absolute cosine similarities between two consecutive conformations, i.e. $IntraPS(p) = \dfrac{\sum_{i=1}^{n-1} |COS(c_i^p, c_{i+1}^p)|}{n-1}$.

In computing the $IntraP$ measure, we decided to use the absolute values for the cosine between two conformations, since our assumption was that for protein data the relative strengths of positive and negative cosine values between RSA vectors is the

same. This was experimentally confirmed in our experiments. For computing the similarity/dissimilarity between two protein conformational transitions, different methods were investigated (Euclidian distance, Pearson correlation, Biweight midcorrelation) and the *cosine similarity* has proven to be the most appropriate. Since the dimensionality of the original protein conformations is significantly reduced by the autoencoder (i.e. two dimensions), Euclidian, Pearson and Biweight midcorrelation are not good options for measuring the similarity: the Euclidean distance is larger between points in a high dimensional space than in a two dimensional one; Pearson and Biweight are not suitable in 2D (the correlation between two dimensional points is always 1).

4 Results and discussion

The experiments we performed for highlighting the potential of deep autoencoders to capture the proteins' structure will be further presented, using the experimental methodology presented in Section 3.

The proteins used in our study are described in Table 1 which shows a brief depiction of the proteins together with their superfamily and sequence length. The proteins from Table 1 were chosen based on data availability (conformational transitions *and* RSA values), the fact that they have the same sequence length (which enables us to carry out our investigations related to RQ2 from Section 1.

Protein	Description	Superfamily	Sequence Length
1P1L	Component of sulphur-metabolizing organisms	3.30.70.120	102
1JT8	Protein involved in translation	2.40.50.140	102

Table 1: Proteins selected for analysis [2].

For both these proteins, 10000 conformational transitions were recovered from the MoDEL database [17] (i.e. n=10000), where each transition consists of a sequence of 99 fragments of the structural alphabet [22]. Thus, as described in Section 3.1, in the *representation based on angles*, a conformation has a length of 297, whereas in the *combined representation* a conformation is visualized as a 396-dimensional point. For both proteins, the two representations proposed in Section 3.1 will be further used. Before applying the autoencoder, the protein data sets are standardized, i.e. transformed to mean 0 and standard deviation 1. Furthermore, considering that the employed technique is a denoising autoencoder, the input data is corrupted by adding noise (random samples from a standard normal distribution).

4.1 Results

The experiment described below is conducted with the aim of answering our first research question RQ1 and of investigating if and how the internal representation for a protein impacts the learning process. For each protein data set, we trained a number of denoising sparse autoencoders (Section 3.2). For the autoencoder we have employed the Keras implementation available at [3]. The autoencoders presented in Section 3.2 are used to reduce the dimensionality of our data and to visualize the protein data sets. Figures 1 and 2 depict the visualization of the proteins from our data set using trained sparse denoising autoencoders. The axes on Figures 1 and 2 represent the range of values obtained within the 2-dimensional encoding of the input data set (the values of

the two hidden nodes representing the encoder output). Colours were added to better emphasize the representations of successive conformations.

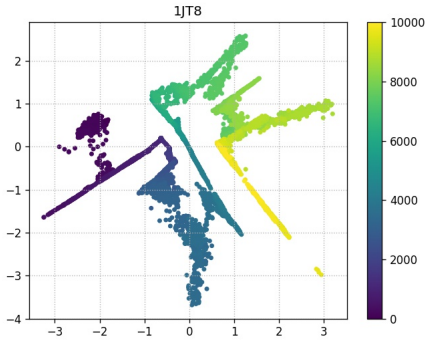

Fig. 1: Visualization of protein 1JT8.

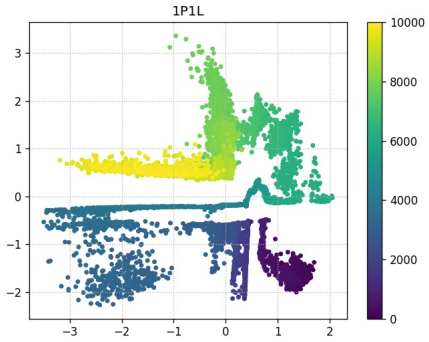

Fig. 2: Visualization of protein 1P1L.

The original data fed to the autoencoder for each protein represents a timely evolution of the protein's structure (albeit for an extremely small interval of time - nanoseconds), considering its transitional conformations. From one conformation to another, the protein might remain unchanged, or certain parts of it might incur minor modifications. The autoencoders used to obtain these representations were trained on original data in its *combined representation*, they employ 6 hidden layers (3 for encoding and 3 for decoding), with ReLU as activation function, batch normalization and a dropout rate of 0.2. Nevertheless, we experimented with the *representation based on angles*, as well as with various combinations of parameters (number of neurons, layers, dropout rate, activation functions), as described in Section 3.2 and all resulting plots denote an evolution of the data output by the autoencoder (henceforth referred to as *encoded data*), thus suggesting that autoencoders are able to identify the most relevant characteristics of the original representations.

The two dimensional representations of the proteins as captured by the autoencoders, illustrated in Figures 1 and 2, reflect the autoencoder's ability to accurately learn biological transitions. Successive conformations in the original data are progressively chained together in the autoencoder's output data thus denoting a visual evolution. Figures 1 and 2 also show that the considered protein data are relevant for machine learning models, as it correctly captures biological chained events, by encoding successive conformations into points that are close in a 2-dimensional space.

Further, to decide whether the autoencoder maintains the relationships found within the original data, we use the *IntraPS* measure. Thus, first we compute these similarities for the original data and then for the two-dimensional data output by the autoencoder, for both considered representations. The results are shown in Table 2. For each protein, in addition to the values for the *IntraPS* measure, we also present the *minimum* (**Min**), *maximum* (**Max**) and *standard deviation* (**Stdev**) of the absolute values of cosine similarities between two consecutive conformations, for both representations. We mention that Min, Max and Stdev were computed using batches of 100 successive conformations. These results are also illustrated in Figures 3 and 4, which show the comparative

evolution of average *IntraPS* values for each 100 conformations in the 10000 conformations that characterize each considered protein. We notice that for both proteins 1JT8 and 1P1L the results output by the autoencoder (denoted by "Encoded data" in the images) are slightly larger, but, on average, particularly similar to the values computed for the original data. All these results suggest that the original proteins' conformations have a high degree of cosine similarity (highlighted in Table 2), which is still preserved in the data resulted from the autoencoder. One observes from Figure 3 that there is a spike in the encoded data, which is not visible in the original data. Analyzing protein 1JT8, we observed that there is an event in the protein structure, but it happens with about 100 conformations before the spike, thus it needs further investigation.

Protein		**Angles**	**Combined**	**Min/Max/Stdev (COS)**	
				Angles	**Combined**
1JT8	Original	**0.9960**	**0.9913**	0.9894/0.9995/0.0023	0.9843/0.9962/0.0022
	Encoded	**0.9939**	**0.9985**	0.9213/0.9999/0.0161	0.9573/0.9999/**0.0044**
1P1L	Original	**0.9779**	**0.9573**	0.9593/0.9896/0.0064	0.9464/0.9695/0.0054
	Encoded	**0.9912**	**0.9962**	0.9315/0.9999/0.0119	0.9661/0.9999/**0.0052**

Table 2: *IntraPS* for proteins 1JT8 and 1P1L, using the two considered representations.

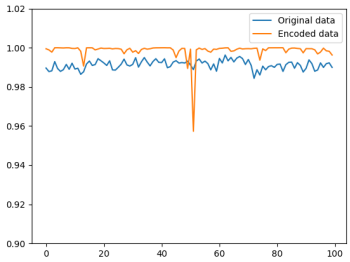

Fig. 3: Protein 1JT8 (combined representation).

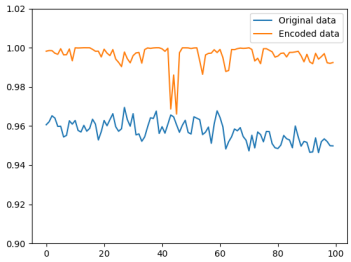

Fig. 4: Protein 1P1L (combined representation).

With regard to the used internal representations, we conclude that these do not seriously influence the learning process. This may be due to the significant reduction of data dimensionality (two dimensions). Still, for the *combined representation* which is richer in information than the *representation based on angles*, slightly better results were obtained. As highlighted in Table 2, for both proteins, *IntraPS* values are larger for the encoded data and the standard deviation of the cosine similarities between two consecutive conformations is smaller, as well. If the data were reduced to a higher dimensional space, the RSA values might bring additional improvements, which induces an interesting matter for future investigations.

With the aim of answering research question RQ2, we are analyzing in the following the biological relevance of the above presented computational results. The molecular dynamics sampled by the ensemble of structures in the two data sets is consistent with small consecutive changes in the protein structure occurring on the nanosecond time scale. These changes are typical of the first stages of the functional motions and they are generally dominated by local transitions and significant resampling of the conformational space. The autoencoder is able to capture both these features, as demonstrated

by the obtained results: changes are encoded in chained events that resample the conformational space effectively. In addition, there is evidence that evolutionary related proteins are also similar in their functional motions [23].

The study performed in this paper with the aim to highlight the ability of *autoencoders* to uncover relevant information about protein dynamics is new. *Autoencoders* have been previously used in the literature for protein structure analysis, but from perspectives which differ from ours.

Autoencoders were proven to be effective for analysis of protein internal structure in [15] where the authors initialized weights, refined them by backpropagation and used each layer's input back to itself in order to predict backbone C^{α} angles and dihedrals. In [10], autoencoders were employed for improving structure class prediction by representing the protein as a "pseudo-amino acid composition" meaning the model consisted of normalized occurrences of the each of the 20 amino acids in a protein, combined with the order of the amino acid sequence. The algorithm called DL-Pro [19] is designed for classifying predicted protein models as good or bad by using a stacked sparse autoencoder which learns from the distances between two C^{α} atoms residues. Sequence based protein to protein interaction was also predicted using a sparse *autoencoder* in [25].

5 Conclusions and further work

We have conducted in this paper a study towards applying *deep autoencoders* for a better comprehension of protein dynamics. The experiments conducted on two proteins highlighted that *autoencoders* are effective unsupervised models able to learn the structure of proteins. Moreover, we obtained an empirical evidence that autoencoders are able to encode hidden patterns relevant from a biological perspective.

Based on the study performed in this paper and on previous investigations regarding protein data analysis, we aim to advance our research towards predicting protein conformational transitions using supervised learning models. Furthermore, we plan to continue our work by using a two-pronged strategy: from a biological viewpoint we will consider other proteins and examine how their evolutionary relationships are reflected within the resulting data; computationally, we will investigate different architectures for the *sparse autoencoder* used in our experiments (e.g. model's architecture, different optimizers for the gradient descent) and we will apply *variational* and *contractive* autoencoders instead of sparse ones.

References

1. Asgari, E., Mofrad, M.: Continuous Distributed Representation of Biological Sequences for Deep Proteomics and Genomics. Plos One (2015). https://doi.org/https://doi.org/10.1371/journal.pone.0141287
2. Berman, H., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T., Weissig, H., Shindyalov, I., Bourne, P.: The Protein Data Bank. Nucleic Acids Res. **28**, 235–242 (2000)
3. Chollet, F., et al.: Deep learning for humans. `https://github.com/fchollet/keras` (2015)
4. Clevert, D.A., Unterthiner, T., Hochreiter, S.: Fast and accurate deep network learning by exponential linear units (elus). arXiv preprint arXiv:1511.07289 (2015)
5. Deng, J., Zhang, Z., Marchi, E., Schuller, B.: Sparse autoencoder-based feature transfer learning for speech emotion recognition. In: ACII. pp. 511–516. IEEE (2013)

6. Fraccalvieri, D., Pandini, A., Stella, F., Bonati, L.: Conformational and functional analysis of molecular dynamics trajectories by self-organising maps. Bioinformatics **12** (2011)
7. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016)
8. Iakovidou, N., Tiakas, E., Tsichlas, K., Manolopoulos, Y.: Going over the three dimensional protein structure similarity problem. Artificial Intelligence Review **42**(3), 445–459 (Oct 2014)
9. Jain, P., Garibaldi, J.M., Hirst, J.: Supervised machine learning algorithms for protein structure classification. Computational Biology and Chemistry **33**, 216–223 (2009)
10. Jian-wei, L., Guang-hui, C., Ze-yu, L., Yuan, L., Hai-en, L., Xiong-Lin, L.: Predicting protein structural classes with autoencoder neural networks. In: CCDC. pp. 1894–1899 (2013)
11. Kingma, D., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
12. Klambauer, G., Unterthiner, T., Mayr, A., Hochreiter, S.: Self-Normalizing Neural Networks. In: NIPS (2017)
13. Le, Q.: Building high-level features using large scale unsupervised learning. In: ICASSP. pp. 8595–8598. IEEE (2013)
14. Lewis, D., Jebara, T., Noble, W.S.: Support vector machine learning from heterogeneous data: an empirical analysis using protein sequence and structure. Bioinformatics **22**(22), 2753–2760 (2006)
15. Lyons, J., Dehzangi, A., Heffernan, R., Sharma, A., Paliwal, K., Sattar, A., Zhou, Y., Yang, Y.: Predicting backbone Ca angles and dihedrals from protein sequences by stacked sparse auto-encoder deep neural network. J. Comput. Chem. **35**(28), 2040–2046 (2014)
16. McGibbon, R., Ramsundar, B., Sultan, M., Kiss, G., Pande, V.: Understanding Protein Dynamics with L1-Regularized Reversible Hidden Markov Models. In: ICML. pp. 1197–1205 (2014)
17. Meyer, T., D'Abramo, M., Hospital, A., Rueda, M., Ferrer-Costa, C., Pérez, A., Carrillo, O., Camps, J., Fenollosa, C., Repchevsky, D., Gelpí, J., Orozco, M.: MoDEL: A database of atomistic molecular dynamics trajectories. Structure **18**(11), 1399 – 1409 (2010)
18. Moon, K.K., Jernigan, R.L., Chirikjian, G.S.: Efficient generation of feasible pathways for protein conformational transitions. Biophysical Journal **83**(3), 1620–1630 (2002)
19. Nguyen, S., Shang, Y., Xu, D.: Dl-pro: A novel deep learning method for protein model quality assessment. In: IJCNN. pp. 2071–2078. IEEE (2014)
20. Palmieri, L., Federico, M., Leoncici, M., Montangero, M.: A High Performing Tool for Residue Solvent Accessibility Prediction. In: ITBAM. pp. 138–152 (2011)
21. Pandini, A., Fornili, A.: Using Local States To Drive the Sampling of Global Conformations in Proteins. Journal of Chemical Theory and Computation **12**, 1368–1379 (2016)
22. Pandini, A., Fornili, A., Kleinjung, J.: Structural alphabets derived from attractors in conformational space. BMC Bioinformatics **11**(97), 1–18 (2010)
23. Pandini, A., Mauri, G., Bordogna, A., Bonati, L.: Detecting similarities among distant homologous proteins by comparison of domain flexibilities. Protein Eng Des Sel. **20**(6), 285–299 (2007)
24. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: A simple way to prevent ANNs from overfitting. J. Mach. Learn. Res. **15**(1), 1929–1958 (2014)
25. Sun, T., Zhou, B., Lai, L., Pei, J.: Sequence-based prediction of protein protein interaction using a deep-learning algorithm. BMC Bioinformatics **18**(1), 277 (2017)
26. Tokuriki, N., Tawfik, D.: Protein dynamism and evolvability. Science **324**(9524), 203–207 (2009). https://doi.org/10.1126/science.1169375
27. Voet, D., Voet, J.: Biochemistry. Wiley, 4 edn. (2011)
28. Wang, D., Lee, N., Dillon, T.: Extraction and optimization of fuzzy protein sequences classification rules using GRBF neural netw. Neural Information Processing-Lett. and Reviews **1**(1), 53–57 (2003)