Supplemental Material

Mutational paths with sequence-based models of proteins: from sampling to mean-field characterisation

Eugenio Mauri, Simona Cocco, Remi Monasson

October 27, 2022

1 Path sampling algorithm

1.1 Detailed description

We introduce below the details of the sampling procedure we use to obtain paths connecting two fixed sequences in a landscape described by the probability distribution P_{model}. Starting from a path $\{\mathbf{v}_t\}$, we look at intermediate sequences (starting from $t=1$) and propose a mutation with the constraint that the Hamming distance between $\mathbf{v}_{t-1}$ and $\mathbf{v}_{t+1}$ is not greater than 1. We accept this move with a probability fixed to ensure detail balance. Different cases have to be considered, depending on the Hamming distance D_H between the new attempted sequence and existing ones (note that hereafter we define $\pi(\mathbf{v},\mathbf{v}') = \exp\left[-N\Phi(\frac{1}{N}\sum_i \delta_{v_i,v'_i})\right]$):

- $D_H(\mathbf{v}_{t-1},\mathbf{v}_{t+1}) = 0$. In this case the new sequence $\mathbf{v}'_t$ can have a single mutation at any site, compared with the two adjacent sequences along the path. Hence, we first draw a random site i, then we propose a new sequence $\hat{\mathbf{v}}_{t-1}$ amino-acids for that site $\hat{v}^i_t$ drawn from the distribution $\propto P^\beta_{model}(\cdot|\mathbf{v}^{\setminus i}_{t-1})$, where the amino acids are fixed on all the sites different from i. Then if the old sequence $\mathbf{v}_t$ already had a mutation with respect to $\mathbf{v}_{t-1}$ at given site j, we accept the new mutated sequence $\hat{\mathbf{v}}_t$ (which is equal to $\mathbf{v}_{t-1}$ apart from the amino acid at site i) with a probability
$$\begin{aligned} p_{acc}(\mathbf{v}_t \to \hat{\mathbf{v}}_t) &= \\ &= \min\left(1, \frac{\pi(\mathbf{v}_{t-1},\hat{\mathbf{v}}_t)^{2\beta}\sum_z P^\beta_{model}(M^z_i \mathbf{v}_{t-1})}{\pi(\mathbf{v}_{t-1},\mathbf{v}_t)^{2\beta}\sum_z P^\beta_{model}(M^z_j \mathbf{v}_{t-1})}\right), \end{aligned} \tag{1}$$
where M^z_i indicates the mutation z at site i. If $\mathbf{v}_t$ or $\hat{\mathbf{v}}_t$ are equal to $\mathbf{v}_{t-1}$, then the acceptance probability is $p_{acc}(\mathbf{v}_t \to \hat{\mathbf{v}}_t) = \min(1, \pi(\mathbf{v}_{t-1},\hat{\mathbf{v}}_t)^{2\beta}/\pi(\mathbf{v}_{t-1},\mathbf{v}_t)^{2\beta})$.
- $D_H(\mathbf{v}_{t-1},\mathbf{v}_{t+1}) = 1$. In this case the new sequence $\hat{\mathbf{v}}_t$ can have a single mutation only at the site i where $\mathbf{v}_{t-1}$ and $\mathbf{v}_{t+1}$ are different. At that site, we propose a new mutation from the distribution $\propto P^\beta_{model}(\cdot|\mathbf{v}^{\setminus i}_{t-1})$ and accept it with probability $p_{acc} = \exp[-\Lambda\beta(D_H(\hat{\mathbf{v}}_t,\mathbf{v}_{t-1}) + D_H(\hat{\mathbf{v}}_t,\mathbf{v}_{t+1}) - D_H(\mathbf{v}_t,\mathbf{v}_{t-1}) - D_H(\mathbf{v}_t,\mathbf{v}_{t+1}))]$.
- $D_H(\mathbf{v}_{t-1},\mathbf{v}_{t+1}) = 2$. In this case the previous and subsequent sequence present two mutations at site i and j. The new sequence $\hat{\mathbf{v}}_t$ can be of two forms: it can have the same mutation of $\mathbf{v}_{t+1}$ (with respect to $\mathbf{v}_{t-1}$) at site i or at site j. Hence, we extract one of the two possibilities with a probability weighted accordingly with $P^\beta_{model}(\hat{\mathbf{v}}_t)$.

1.2 Proof of detailed balance

To prove that dynamics given by our algorithm of path sampling converges to the target distribution we have to prove that it respects detailed balance, i.e. the reversibility of each Markov step. We consider the transition from a path $\{\mathbf{v}_t\}$ to a new path that differ only by one sequence $\mathbf{v}'_t$ at time t. we write the detailed balance condition as

$$\mathcal{P}_{path}(\{\mathbf{v}_t\})p_{trans}(\mathbf{v}_t \to \mathbf{v}_t') = \mathcal{P}_{path}(\{\mathbf{v}_t'\})p_{trans}(\mathbf{v}_t' \to \mathbf{v}_t)$$
$$\pi(\mathbf{v}_{t-1}, \mathbf{v}_t)\pi(\mathbf{v}_t, \mathbf{v}_{t+1})P_{model}(\mathbf{v}_t)p_{trans}(\mathbf{v}_t \to \mathbf{v}_t') = \pi(\mathbf{v}_{t-1}, \mathbf{v}_t')\pi(\mathbf{v}_t', \mathbf{v}_{t+1})P_{model}(\mathbf{v}_t')p_{trans}(\mathbf{v}_t' \to \mathbf{v}_t) \tag{2}$$

If $D_H(\mathbf{v}_{t-1}, \mathbf{v}_{t+1}) = 0$, the new sequence can have a mutation at any site i compared to its neighbour $\mathbf{v}_{t-1}$, while $\mathbf{v}_t$ will have the mutation at another site j (note that i and j can be equal. Hence the transition probability in this case will be

$$p_{trans}(\mathbf{v}_t \to \mathbf{v}_t') = \frac{1}{N}\frac{P_{model}(\mathbf{v}_t')}{\sum_{z=1}^{Q} P_{model}(M_i^z \mathbf{v}_{t-1})} p_{acc}(\mathbf{v}_t \to \mathbf{v}_t'),\ p_{trans}(\mathbf{v}_t' \to \mathbf{v}_t) = \frac{1}{N}\frac{P_{model}(\mathbf{v}_t)}{\sum_{z=1}^{Q} P_{model}(M_j^z \mathbf{v}_{t-1})} p_{acc}(\mathbf{v}_t' \to \mathbf{v}_t). \tag{3}$$

By substituting everything in the detailed balance condition we obtain the acceptance probability described in the main paper. This will hold similarly when $D_H(\mathbf{v}_{t-1}, \mathbf{v}_{t+1}) = 1$ (with $i = j$). For $D_H(\mathbf{v}_{t-1}, \mathbf{v}_{t+1}) = 2$, the sequences $\mathbf{v}_t$ and $\mathbf{v}_t'$ can either be equal to $\mathbf{v}_{t+1}$ or $\mathbf{v}_{t-1}$, from which the condition presented in the paper descends.

2 Lattice Proteins

To benchmark the performances of this MC procedure to find good transition path between two sequences, we test it on Lattice Proteins [LD89], a well known toy-model for protein structure. We consider a protein sequence of 27 amino acids folding into a 3D structure specified as a self-avoiding path over a 3x3x3 lattice where each amino acid occupies one node. The probability of a sequence $\mathbf{v}$ to fold into a specific structure $\mathcal{S}$ is given by the interaction energies between amino acids in contact in the structure (i.e. those who occupy neighbouring nodes of the lattice, but are not adjacent in the protein sequence). In particular, the total energy of a sequence with respect to a given structure is given by

$$\mathcal{E}_{LP}(\mathbf{v}|\mathcal{S}) = \sum_{i<j} c_{ij}^{\mathcal{S}} E_{MJ}(v_i, v_j) \tag{4}$$

where $c^{\mathcal{S}}$ is the contact map ($c_{ij}^{\mathcal{S}} = 1$ if sites are in contact and 0 otherwise), while the pairwise energy $E_{MJ}(v_i, v_j)$ represents the amino-acid physico-chemical interactions given by the the Miyazawa-Jernigan knowledge-based potential [MJ96]. The probability to fold into a specific structure is written as

$$p_{nat}(\mathcal{S}|\mathbf{v}) = \frac{e^{-\mathcal{E}_{LP}(\mathbf{v}|\mathcal{S})}}{\sum_{\mathcal{S}'} e^{-\mathcal{E}_{LP}(\mathbf{v}|\mathcal{S}')}}\,, \tag{5}$$

where the sum is over the entire set of self-avoiding path in the cubic lattice.
The function p_{nat} represents a suitable landscape that maps each sequence to a score measuring the quality of its folding. To study in more detail this landscape, we will consider an alignment of sequences folding into a specific structure (that we will call $\mathcal{S}$) sampled from a low temperature MC sampling using $-\beta \log p_{nat}(\cdot|\mathcal{S})$ (with $\beta = 10^3$) as effective energy [JGS+16].

3 Restricted Boltzmann Machines and training parameter

To study the problem of transition paths we first need a model to infer a landscape from our sequence data set. At this scope, we are going to use Restricted Boltzmann Machines, an unsupervised energy-based model able to learn representations of the data in a two-layer bipartite graph [FI12]. The first "visible" layer represents the protein sequence $\mathbf{v} = \{v_1, ..., v_N\}$ where each unit takes one out of 21 possible states (20 amino acids + 1 alignment gap). The second is the "hidden" layer which displays the real-valued representations $\mathbf{h} = \{h_1, ..., h_M\}$. The joint probability distribution for $\mathbf{v}$ and $\mathbf{h}$ is

$$P_{RBM}(\mathbf{v}, \mathbf{h}) \propto \exp\left[\sum_{i=1}^{N} g_i(v_i) + \sum_{i,\mu} w_{i\mu}(v_i)h_\mu - \sum_{\mu} \mathcal{U}(h_\mu)\right] \tag{6}$$

up to a normalization constant. visible and hidden units are coupled through the matrices $w_{i\mu}$ and the value of each unit is biased by the local fields g_i and $\mathcal{U}_\mu$. In [TCM19] it has been shown that this model is able to recover statistically relevant sequence motifs playing crucial roles in the structure and functionality of different protein families. Following their approach, we choose $\mathcal{U}_\mu$ to be double Rectified Linear Unit (dReLU) potentials of the form

$$\mathcal{U}_\mu(h) = \frac{1}{2}\gamma_{\mu,+}h_+^2 + \frac{1}{2}\gamma_{\mu,-}h_-^2 + \theta_{\mu,+}h_+ + \theta_{\mu,-}h_- \,, \text{ where } h_+ = \max(h,0),\; h_- = \min(h,0), \tag{7}$$

where we have defined the hyper-parameters $\gamma_{\mu,\pm}$, $\theta_{\mu,\pm}$.
At this point, we need a learning procedure to infer the hyper-parameters that best fit our data. We decided to use a Persistent Contrastive Divergence algorithm [Tie08] which has been shown to be sufficiently good and robust under cautious choice of the regularization hyper-parameters [Tub18a]. The code and the data used to train our RBMs for Lattice Proteins and WW domains can be found in [Tub18b]. The hyper-parameters used for learning are the following:

- For WW (N=31):
  - M = 50
  - Batch size = 100
  - Number of epochs = 500
  - Learning rate = 5×10^{-3} (which has a decay rate of 0.5 after 50% of iterations)
  - L_1b regularization = 0.25
  - Number of MC step between each update = 10
- For Lattice Protein (N=27):
  - $M = 100$
  - Batch size = 100
  - Number of epochs = 100
  - Learning rate = 5×10^{-3} (which has a decay rate of 0.5 after 50% of iterations)
  - L_1b regularization = 0.025
  - Number of MC step between each update = 5

For the local RBMs trained respectively on the three specificity classes of WW sequences predicted by the original RBM, we use $M = 30$ and keep all the other hyper-parameters unchanged.

4 Statistic of paths sampled in LP

ere we show relevant statistics of the paths sampled in the Lattice Protein model using the procedure described above. Looking at Figure S1 We see that many paths prefer to activate input 40, exploiting Cys-Cys interactions (see logos of the weights in the Appendix). Furthermore, statistically this input is more likely to be reduced before activating input 4.
Analysing the paths along each inputs we notice that those solutions maintain high scores in terms of p_{nat} by exploiting the interactions between amino-acids at sites 5,6,11 and 22 (see Figure 1 in the main paper). These paths are divided in two classes corresponding to the chemical nature of the interaction used to bind these amino-acids. One cluster (shown in maroon in Figure 1 of the main paper and in Figure S1) uses Cys-Cys bridges to establish this interactions. Instead the most populated cluster (shown in red in the same figures) exploits electrostatic interactions that are not initially present in the target sequences. Some of these interactions are shown in Figure S2.

To deeper characterize the behaviour of these solutions, we also plot in Figure S3 the average distance from the two edge sequences (also called distance from direct space) as a function of time obtained from the paths sampled using $P_{model} \propto P_{RBM}$, the likelihood from the trained RBM model.

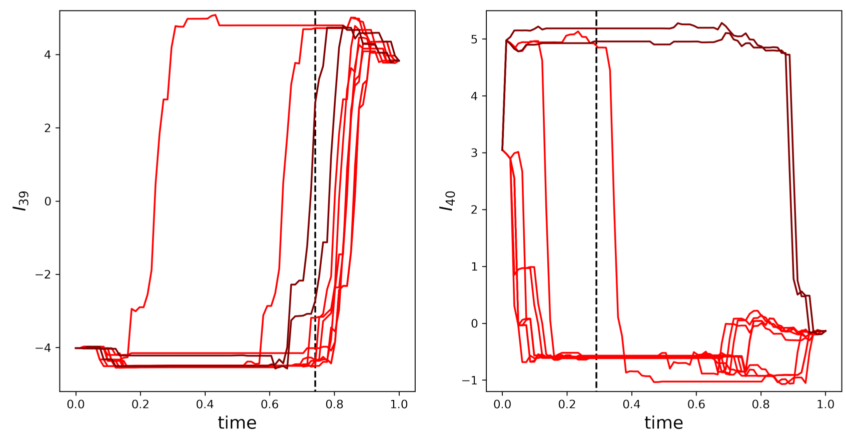


Figure S1: Plot of some relevant inputs as function of time sampled with our Monte-Carlo procedure ($\Lambda = 2$ and target $\beta = 3$). The colors respect those presented in Figure 1 in the main paper. Black lines correspond to the average time at which the input switch value (>0 for I_4 and <2 for I_{14}).

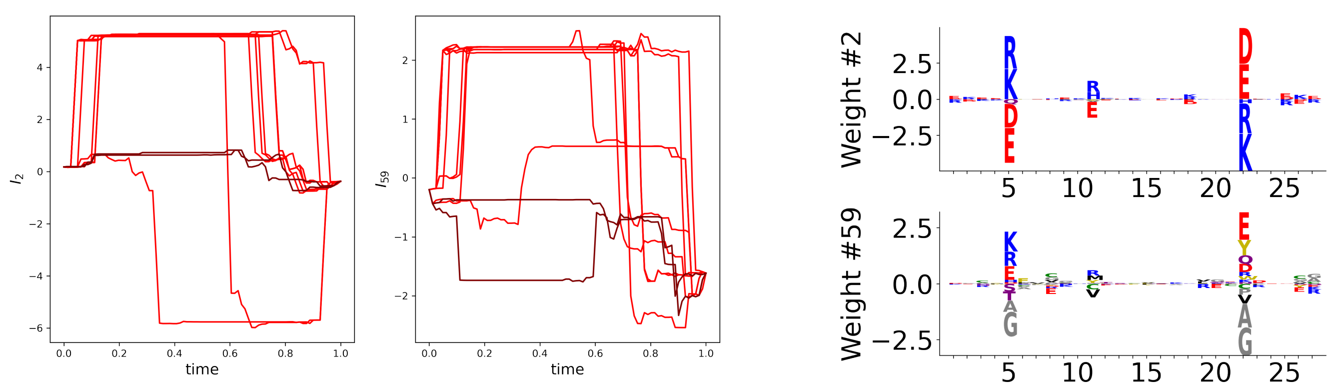


Figure S2: Plot of some relevant inputs (and their respective weights logos) as function of time sampled with our Monte-Carlo procedure ($\Lambda = 2$ and target $\beta = 3$) exploiting relevant electrostatic interactions not exploited in the edge sequences $\mathbf{v_{start}}$ and $\mathbf{v_{end}}$. The colors respect those presented in Figure 1 in the main paper.

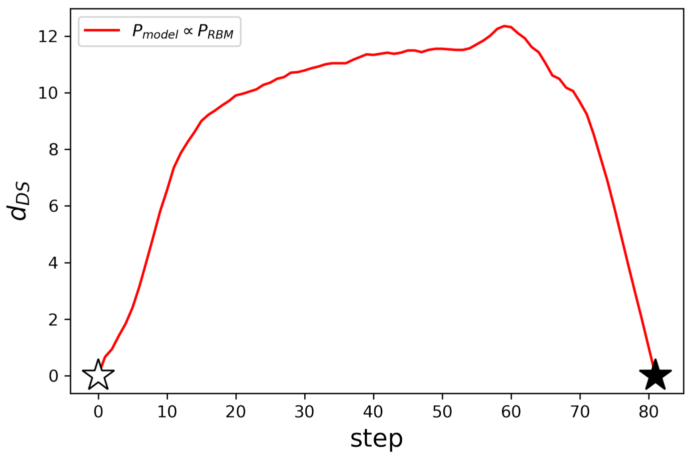


Figure S3: Average value of the distance from direct space (defined as $d_{DS}(\mathbf{v}) = \sum_i (1 - \delta_{v_i^{start}, v_i})(1 - \delta_{v_i^{end}, v_i})$) as a function of the step along the path connecting the two modes in the LP landscape. Black and White stars refer to the same sequences as in Figure 1 of the main text.

5 Additional information about mutational paths of the WW domain

5.1 Lists of the tested sequences

Here we present the reference sequences sampled with the MC algorithm and tested using AlphaFold (note that the first sequence of each list represent the YAP1 wild-type sequence from [ES99], while the last is the natural wild-type specific for each class) together with the predicted specificity using local RBMs:

- From YAP1 to wild-type of class II:
  - `LPAGWEMAKTSS-GQRYFLNHIDQTTTWQDP`
  - `LPAGWEMAKTSD-GERYFINHNTKTTTWQDP` predicted I
  - `LPPGWEEARTPD-GRVYFINHNTKTTTWQDP` predicted I
  - `LPPGWEEARAPD-GRTYYYNHNTKTTTWEKP` predicted II/III
  - `LPPGWTEHKAPD-GRTYYYNHNTKTSTWEKP` predicted II/III
  - `LPSGWTEHKAPD-GRTYYYNTETKQSTWEKP` predicted II/III
  - `AKSMWTEHKSPD-GRTYYYNTETKQSTWEKP`
- From YAP1 to wild-type of class IV:
  - `LPAGWEMAKTSS-GQRYFLNHIDQTTTWQDP`
  - `LPAGWEMRRTPS-GRVYFVNHITRTTQWEDP` predicted I
  - `LPPGWEERRDPS-GRVYYYVNHITRTTQWERP` predicted I
  - `LPPGWEERVSRS-GRVYYYVNHITRTTQWERP` predicted I
  - `LPPGWEKRMSRS-GRVYYYVNHITRTTQWERP` predicted I
  - `LPPGWEKRMSRSSGRVYYYVNHITRASQWERP` predicted IV
  - `LPPGWEKRMSRSSGRVYYYFNHITNASQWERP`
- From YAP1 to wild-type of class I:
  - `LPAGWEMAKTSS-GQRYFLNHIDQTTTWQDP`
  - `LPAGWEMAKTSE-GQRYFINHNTQTTTWQDP`

– LPPGWEMAYTPE-GERYFINHNTKTTTWLDP

– LPPGWEMGITRG-GRVFFINHETKSTTWLDP

– LPRSWTYGITRG-GRVFFINHEAKSTTWLHP

– LPRSWTYGITRG-GRVFFINEEAKSTTWLHP

5.2 details on the TM-score

To compare the inferred structures of the sampled sequences along the path with those of the target natural sequences we used the Template Modelling (TM) score developed in [ZS04] and represents a variation of the Levitt–Gerstein (LG) score [LG98]. Compared to other similarity score (like root-mean-square deviation (RMSD)) it gives a more accurate measure since it relies more on the global similarity of the full sequence rather than the local similarities.
Practically, we consider a target sequence of length L_{target} and a template one whose structure has to be compared with. First, we align the two sequences and we take the L_{common} pairs of residues that commonly appear aligned. Then the score is computed as

$$\text{TM-score} = \max_{\{d_i\}} \left[\frac{1}{L_{\text{target}}} \sum_{i=1}^{L_{\text{common}}} \frac{1}{1 + \left(\frac{d_i}{d_0(L_{\text{target}})} \right)^2} \right], \tag{8}$$

where d_i is the distance between the ith pair of residues between the template and the target structures after alignment, and $d_0(L_{\text{target}}) = 1.24(L_{\text{target}} - 15)^{1/3} - 1.8$ is a distance scale that normalizes distances. This formula gives a score between 0 and 1. if TM-score< 0.2, the two sequences are totally uncorrelated, while they can be considered to have the same structure if TM-score> 0.5. In Figure S4 we show the tables of the TM-scores associated with the sequences tested above.

	YAP1	wt class I		YAP1	wt class IV		YAP1	wt class II
YAP1	1	0.88	YAP1	1	0.76	YAP1	1	0.76
1	0.95	0.83	1	0.93	0.76	1	0.92	0.72
2	0.94	0.86	2	0.9	0.8	2	0.93	0.75
3	0.91	0.82	3	0.89	0.84	3	0.86	0.85
4	0.84	0.98	4	0.81	0.7	4	0.84	0.84
wt class I	0.88	1	5	0.83	0.93	5	0.86	0.83
			wt class IV	0.76	1	wt class II	0.76	1

Figure S4: Table of the TM-scores measured between the structures inferred from AlphaFold.

5.3 ProteinMPNN score

The ProteinMPNN model published in [DAB+22] takes a reference backbone structure (uploaded as a .pdb file) and gives as output a log-probability function over protein sequences $\log P_{MPNN}(\mathbf{v})$ measuring the affinity of the sequence $\mathbf{v}$ to fold into a specific structure and/or complex. In order to compare values from different models we re-normalised the log-score as

$$\text{norm. score}(\mathbf{v}) = \frac{\log P_{MPNN}(\mathbf{v}) - \log P_{MPNN}(\mathbf{v}_{wt1})}{\log P_{MPNN}(\mathbf{v}_{wt0}) - \log P_{MPNN}(\mathbf{v}_{wt1})}, \tag{9}$$

where $\mathbf{v}_{wt0}$ is one of the edge sequences of the path equal to the reference wild-type of the target structure, while $\mathbf{v}_{wt1}$ is the other edge sequence with different specificity.

Since the transition from class I to class IV requires an insertion along the path (see section 5.1), we decided to remove those insertion to test the sequences against the class I reference backbone structure (PDB ID: 2LTW) and to substitute the gap with Serine (S) to test them against class IV reference structure (PDB ID: 1I8G).

6 Derivation of mean-field equations for path sampling with a RBM model

To exploit the nature of the RBM as a mean-field model we rewrite the probability distribution of the model as

$$P_{RBM}(\mathbf{v}) = \frac{1}{Z_{RBM}} \int \prod_{\mu} \mathrm{d}h_\mu \exp\left(\sum_i g_i(v_i) + \sum_{i,\mu} w_{i\mu}(v_i) h_\mu - \sum_\mu \mathcal{U}_\mu(h_\mu)\right) \tag{10}$$

$$= \frac{1}{Z_{RBM}} \exp\left(\sum_i g_i(v_i) + N \sum_\mu \Gamma_\mu\left(\frac{1}{N} I_\mu\right)\right) , \tag{11}$$

where γ is defined in the main text.

By introducing the order parameters $m_t^\mu = I_{\mu,t}/N$, $q_t = \frac{1}{N}\sum_i \delta_{v_{i,t},v_{i,t+1}}$ as well as the overlap potential Φ, we can write the probability for a path in the order parameter space as (in the large N limit)

$$P_{path}(\{\mathbf{m}_t, q_t\}_{t=1}^T; \beta) \propto \sum_{\{\mathbf{v}_t\}} \prod_{t,\mu} \delta\left(\frac{1}{N}\sum_i w_{i\mu}(v_{i,t}) - m_t^\mu\right) \prod_t \delta\left(\frac{1}{N}\sum_i \delta_{v_{i,t},v_{i,t+1}} - q_t\right) P_{RBM}^\beta(\mathbf{v}_t)\pi^\beta(\mathbf{v}_t, \mathbf{v}_{t+1}) \tag{12}$$

$$= \exp\left[\beta N \left(\sum_{t,\mu} \Gamma(m_t^\mu) - \sum_t \Phi(q_t)\right)\right] \times \tag{13}$$

$$\times \int \left(\prod_{\mu,t} \mathrm{d}\hat{m}_t^\mu \prod_t \mathrm{d}\hat{q}_t\right) \sum_{\{\mathbf{v}_t\}} \exp\left[\sum_{i,t} g_i(v_{i,t}) + N\sum_{t,\mu} \hat{m}_t^\mu \left(\frac{1}{N}\sum_i w_{i\mu}(v_{i,t}) - m_t^\mu\right) + N \sum_t \hat{q}_t \left(\frac{1}{N}\sum_i \delta_{v_{i,t},v_{i,t+1}} - q_t\right)\right] \tag{14}$$

$$\approx \exp\left[\beta N \left(\sum_{t,\mu} \Gamma(m_t^\mu) - \sum_t \Phi(q_t)\right) + N\mathcal{S}(\{\mathbf{m}_t, q_t\}_{t=1}^T)\right] = \exp\left(-N\beta f_{path}(\{\mathbf{m}_t, q_t\})\right) , \tag{15}$$

where

$$\mathcal{S}(\{\mathbf{m}_t, q_t\}_{t=1}^T) = \min_{\{\hat{\mathbf{m}}_t, \hat{q}_t\}} \frac{1}{N}\sum_i \log Z_i(\{\hat{\mathbf{m}}_t, \hat{q}_t\}) - \sum_{t,\mu} m_t^\mu \hat{m}_t^\mu - \sum_t q_t \hat{q}_t \tag{16}$$

and

$$Z_i(\{\hat{\mathbf{m}}_t, \hat{q}_t\}) = \sum_{v_1,..,v_t,...,v_T} \exp\left[\sum_t g_i(v_t) + \sum_{t,\mu} \hat{m}_t^\mu w_{i\mu}(v_t) + \sum_t \hat{q}_t \delta_{v_t,v_{t+1}}\right] . \tag{17}$$

Under minimization we find the result shown in the main paper. To obtain numerically the set of magnetizations and overlap that minimize the free energy, we note that the saddle point equation for f_{path} leads to the following self-consistent equation:

$$m_t^\mu = \frac{1}{N}\sum_i \frac{1}{Z_i} \sum_{v_1,..,v_t,...,v_T} w_{i\mu}(v_t) \exp\left[\sum_t g_i(v_t) + \sum_{t,\mu} \hat{m}_t^\mu w_{i\mu}(v_t) + \sum_t \hat{q}_t \delta_{v_t,v_{t+1}}\right] \tag{18}$$

$$q_t = \frac{1}{N}\sum_i \frac{1}{Z_i} \sum_{v_1,..,v_t,...,v_T} \delta_{v_t,v_{t+1}} \exp\left[\sum_t g_i(v_t) + \sum_{t,\mu} \hat{m}_t^\mu w_{i\mu}(v_t) + \sum_t \hat{q}_t \delta_{v_t,v_{t+1}}\right] , \tag{19}$$

where $\hat{q}_t = -\beta\Phi'(q_t)$ and $\hat{m}_t^\mu = \beta\Gamma'_\mu(m_t^\mu)$. We solve this set of equations using gradient descent. To compute the LHS we first compute the partition functions Z_i using the transfer matrix method and then we take their gradient using automatic differentiation technique built in the Python library JAX [BFH+18].

6.1 Consensus sequence from MF solutions and the case for WW domain

To obtain the average distance from the direct space, we need to compute at each time at each site the probability of a specific state $a = 1, ..., A$. This can be computed as

$$f_{i,t}(a|\{\mathbf{m}_t, q_t\}) = \frac{\partial}{\partial g_{it}(a)} \log Z_{path} = \frac{\partial}{\partial g_{it}(a)} \sum_{\mathcal{V}} \exp\left[\sum_{i,t,a} g_{it}(a)\delta_{a,v_{it}} + N\sum_{\mu,t}\Gamma(m_\mu^t) - N\sum_t \Phi(q_t)\right] = \frac{\partial}{\partial g_{it}(a)} \log Z_i\,, \quad (20)$$

where we write $g_{it}(a) = g_i(a)$ for any t. Once $f_{i,t}$ is computed, we obtain the consensus sequence $\mathbf{v}_t = \{v_{i,t} = \text{argmax}_a f_{i,t}\}$. The consensus sequences for the MF path shown in Figure (3) of the main paper are:

- I→IV Cont scenario:
  - `LPAGWEMAKTSS-GQRYFLNHITRTTTWQDP`
  - `LPAGWEMRKTSS-GQVYFLNHITRTTTWEDP`
  - `LPPGWEMRKTSS-GRVYFLNHITRTTQWEDP`
  - `LPPGWEMRKSRS-GRVYFLNHITRTTQWEDP`
  - `LPPGWEKRKSRS-GRVYYLNHITRTTQWERP`
  - `LPPGWEKRMSRS-GRVYYLNHITRTTQWERP`
  - `LPPGWEKRMSRS-GRVYYFNHITRASQWERP`
- I→IV Evo scenario:
  - `LPPGWEKRKSRS-GRVYYLNHITKTTQWERP`
  - `LPPGWEKRKSRS-GRVYYLNHITKTTQWERP`
  - `LPPGWEKRKSRS-GRVYYLNHITKTTQWERP`
  - `LPPGWEKRKSRS-GRVYYLNHITKTTQWERP`
  - `LPPGWEKRKSRS-GRVYYLNHITKTTQWERP`
  - `LPPGWEKRMSRS-GRVYYLNHITKTTQWERP`
  - `LPPGWEKRMSRS-GRVYYLNHITKTTQWERP`
- I→II Cont scenario:
  - `LPAGWEMAKTSD-GQRYFLNHITQTTTWQDP`
  - `LPAGWEMAKTPD-GQRYFLNHITKTTTWEDP`
  - `LPAGWEEAKTPD-GRTYFYNHITKTTTWEDP`
  - `LPAGWEEAKTPD-GRTYYYNHITKTTTWEKP`
  - `LPAGWTEHKTPD-GRTYYYNHITKTTTWEKP`
  - `LPSGWTEHKTPD-GRTYYYNTITKQSTWEKP`
  - `LPSGWTEHKSPD-GRTYYYNTETKQSTWEKP`
- I→II Evo scenario:
  - `LPAGWEEAKTPD-GRRYFLNHITKTTTWEDP`
  - `LPAGWEEAKTPD-GRTYYYNHITKTTTWEDP`
  - `LPAGWEEAKTPD-GRTYYYNHITKTTTWEKP`
  - `LPAGWEEAKTPD-GRTYYYNHITKTTTWEKP`
  - `LPSGWTEHKTPD-GRTYYYNTITKQSTWEKP`
  - `LPSGWTEHKTPD-GRTYYYNTETKQSTWEKP`
  - `LPSGWTEHKSPD-GRTYYYNTETKQSTWEKP`

6.2 Free energy optimisation in mean-field theory

In order to obtain the minimum of the free energy in the mean-field approximation we first modify the energetic term of the RBM model by multiplying it with temperature factor β_0 (the interaction term $\sum_t \Phi(q_t)$ stays untouched). Starting from $\beta_0 = 0$, we minimize the free-energy using gradient descent. Here the landscape is convex and the gradient descent finds the global minima. Using this solution as a new initial configuration, we re-use gradient descent but after having increased β_0 by a small step $\delta\beta_0$. In such a way we can follow the global minimum (under the hypothesis that the system does not encounter zeroth-order phase transitions). Then we repeat this procedure until we reach $\beta_0 = 1$.

7 Neutral theory of evolution

Let's consider a sequence (with A number of states per site) evolving under mutations only. Given a site i along the sequence the probability of that site to be in a given state A at time t, $x_a^i(t)$, evolve through time under the following equation:

$$\frac{d}{dt}x_a^i(t) = -\mu x_a^i(t) + \frac{\mu}{A}\sum_{b\neq a} x_b^i(t) = \sum_b W_{a,b}x_b^i(t) \tag{21}$$

where μ is the mutation rate. Solving the linear differential equation, we can compute the probability that a site mutate into a specific new state in the time interval Δt as

$$p_{\neq} = \frac{1}{A}\left(1 - e^{\frac{A\mu\Delta t}{1-A}}\right) , \tag{22}$$

while the probability of not mutating is $p_= = 1 - (A-1)p_{\neq}$. Here we set $\Delta t = 1$. Hence the probability of evolving from a sequence $\mathbf{v}$ to $\mathbf{v}'$ is

$$\pi(\mathbf{v}, \mathbf{v}') = p_=^{Nq} p_{\neq}^{N(1-q)} = e^{-N\Phi(q)} , \tag{23}$$

where q is the overlap between the two sequence and $\Phi(q) = q \log \frac{p_{\neq}}{p_=} - \log p_{\neq}$. Hence, the probability to go from $\mathbf{v}_0$ to $\mathbf{v}_T$ in T steps is

$$P(\mathbf{v}_0 \to \mathbf{v}_T; T) = \sum_{\{\mathbf{v}\}_{t=1}^{T-1}} \pi(\mathbf{v}_0, \mathbf{v}_1)\pi(\mathbf{v}_1, \mathbf{v}_2)\ldots\pi(\mathbf{v}_{T-1}, \mathbf{v}_T) = \sum_{\{\mathbf{v}\}_{t=1}^{T-1}} e^{-N\sum_t \Phi(q_t)} , \tag{24}$$

which can be computed exactly as

$$P(\mathbf{v}_0 \to \mathbf{v}_T; T) = \frac{p_{\neq}^{TN}}{A^N}\left[\left(\frac{p_=}{p_{\neq}} + A - 1\right)^T - \left(\frac{p_=}{p_{\neq}} - 1\right)^T\right]^D \left[\left(\frac{p_=}{p_{\neq}} + A - 1\right)^T - (1-A)\left(\frac{p_=}{p_{\neq}} - 1\right)^T\right]^{N-D} , \tag{25}$$

where D is the Hamming distance between the two sequences. The last equation correspond to Kimura's theory of neutral evolution [Kim83] and the probability as a maximum for a certain optimal T^* and converges to $1/A^N$ for $T \to \infty$.

References

[BFH+18] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018. Available online at: `http://github.com/google/jax`.

[DAB+22] Justas Dauparas, Ivan Anishchenko, Nathaniel Bennett, Hua Bai, Robert J Ragotte, Lukas F Milles, Basile IM Wicky, Alexis Courbet, Rob J de Haas, Neville Bethel, et al. Robust deep learning–based protein sequence design using proteinmpnn. *Science*, page eadd2187, 2022.

[ES99] Xavier Espanel and Marius Sudol. A single point mutation in a group i ww domain shifts its specificity to that of group ii ww domains. *Journal of Biological Chemistry*, 274(24):17284–17289, 1999.

[FI12] Asja Fischer and Christian Igel. An Introduction to Restricted Boltzmann Machines. In *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, pages 14–36. Springer, Berlin, Germany, September 2012.

[JGS+16] Hugo Jacquin, Amy Gilson, Eugene Shakhnovich, Simona Cocco, and Rémi Monasson. Benchmarking inverse statistical approaches for protein structure and design with exactly solvable models. *PLOS Computational Biology*, 12(5):1–18, 05 2016.

[Kim83] Motoo Kimura. *The neutral theory of molecular evolution*. Cambridge University Press, 1983.

[LD89] Kit Fun Lau and Ken A. Dill. A lattice statistical mechanics model of the conformational and sequence spaces of proteins. *Macromolecules*, 22(10):3986–3997, October 1989.

[LG98] Michael Levitt and Mark Gerstein. A unified statistical framework for sequence comparison and structure comparison. *Proceedings of the National Academy of sciences*, 95(11):5913–5920, 1998.

[MJ96] Sanzo Miyazawa and Robert L. Jernigan. Residue – Residue Potentials with a Favorable Contact Pair Term and an Unfavorable High Packing Density Term, for Simulation and Threading. *J. Mol. Biol.*, 256(3):623–644, March 1996.

[TCM19] Jérôme Tubiana, Simona Cocco, and Rémi Monasson. Learning protein constitutive motifs from sequence data. *eLife*, 8:e39397, March 2019.

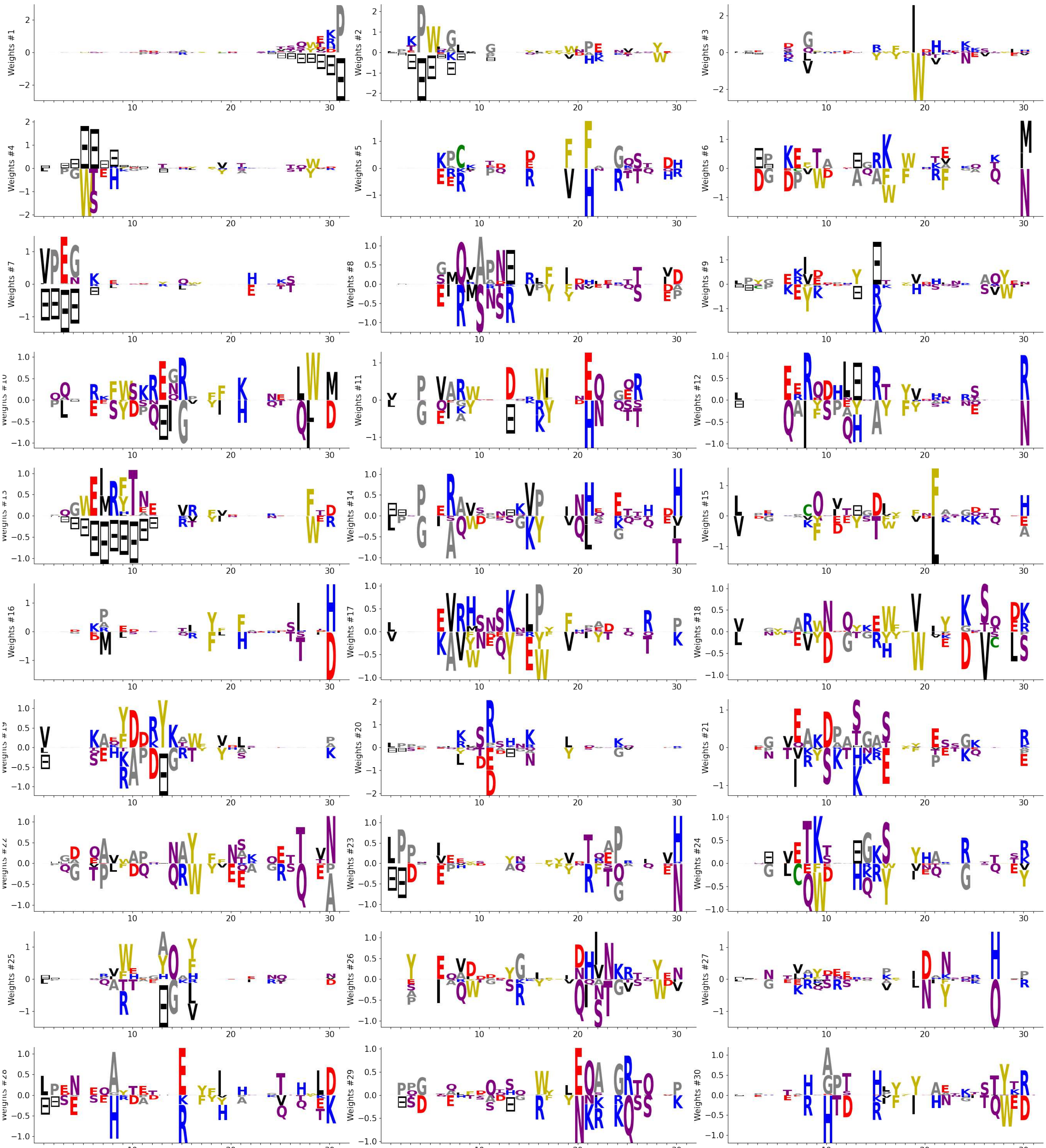
[Tie08] Tijmen Tieleman. Training restricted Boltzmann machines using approximations to the likelihood gradient. In *ICML '08: Proceedings of the 25th international conference on Machine learning*, pages 1064–1071. Association for Computing Machinery, New York, NY, USA, July 2008.

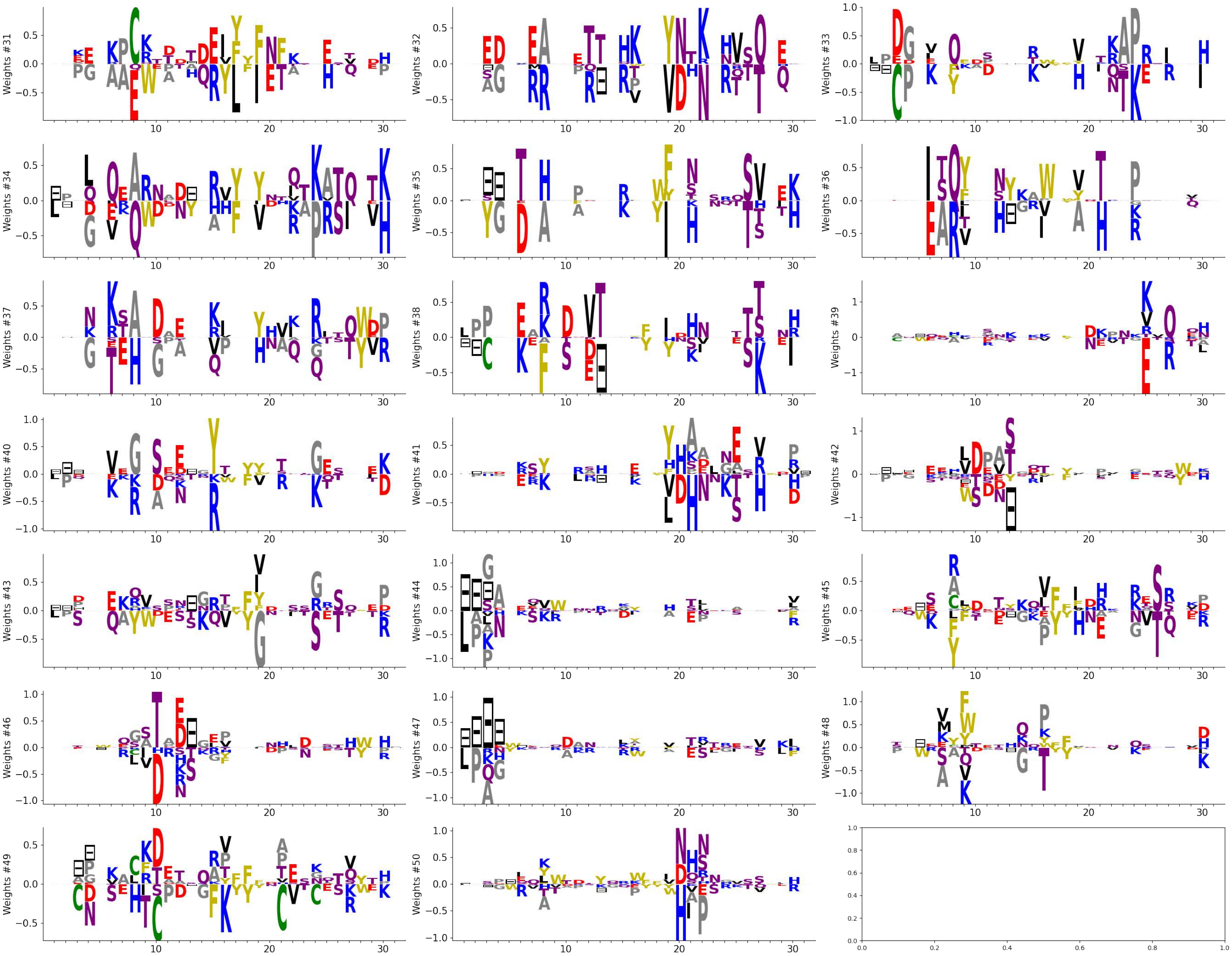
[Tub18a] Jérôme Tubiana. *Restricted Boltzmann machines : from compositional representations to protein sequence analysis*. PhD thesis, Université Paris sciences et lettres, Paris, France, November 2018.

[Tub18b] Jerome Tubiana. Probabilistic graphical models (pgm), 2018. Available online at: `https://github.com/jertubiana/PGM`.

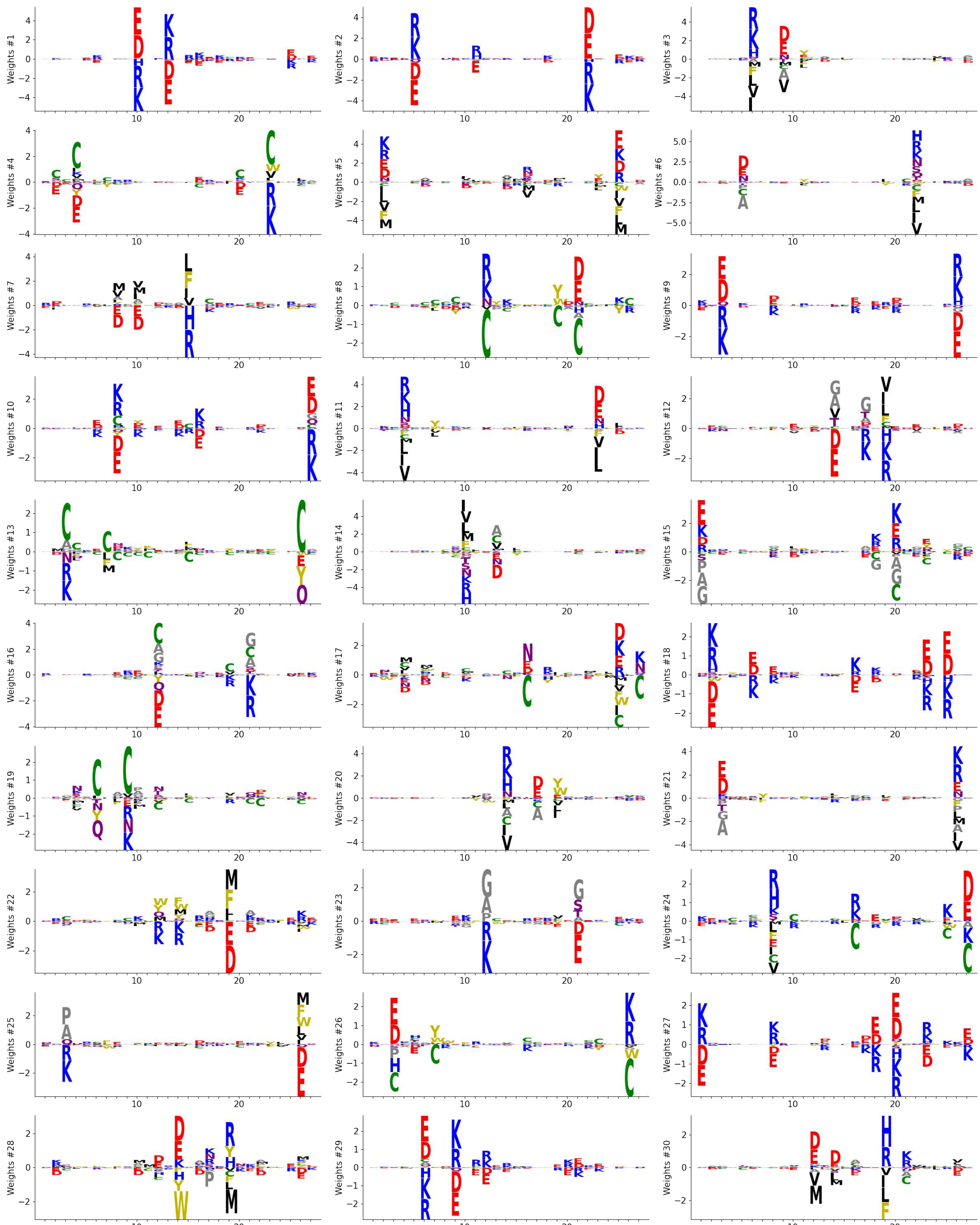
[ZS04] Yang Zhang and Jeffrey Skolnick. Scoring function for automated assessment of protein structure template quality. *Proteins: Structure, Function, and Bioinformatics*, 57(4):702–710, 2004.

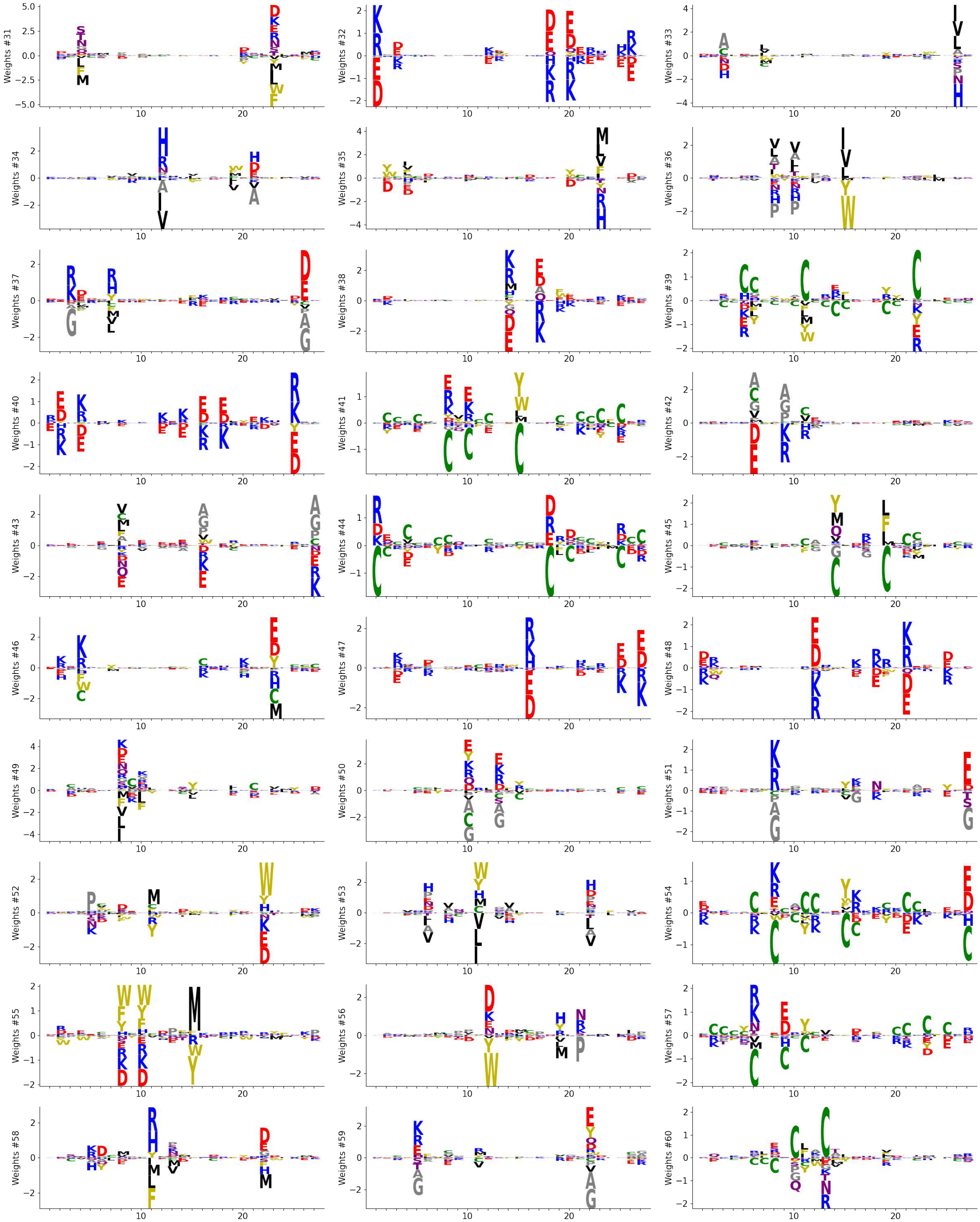
8 Appendix A: Weights Logo for the WW domain

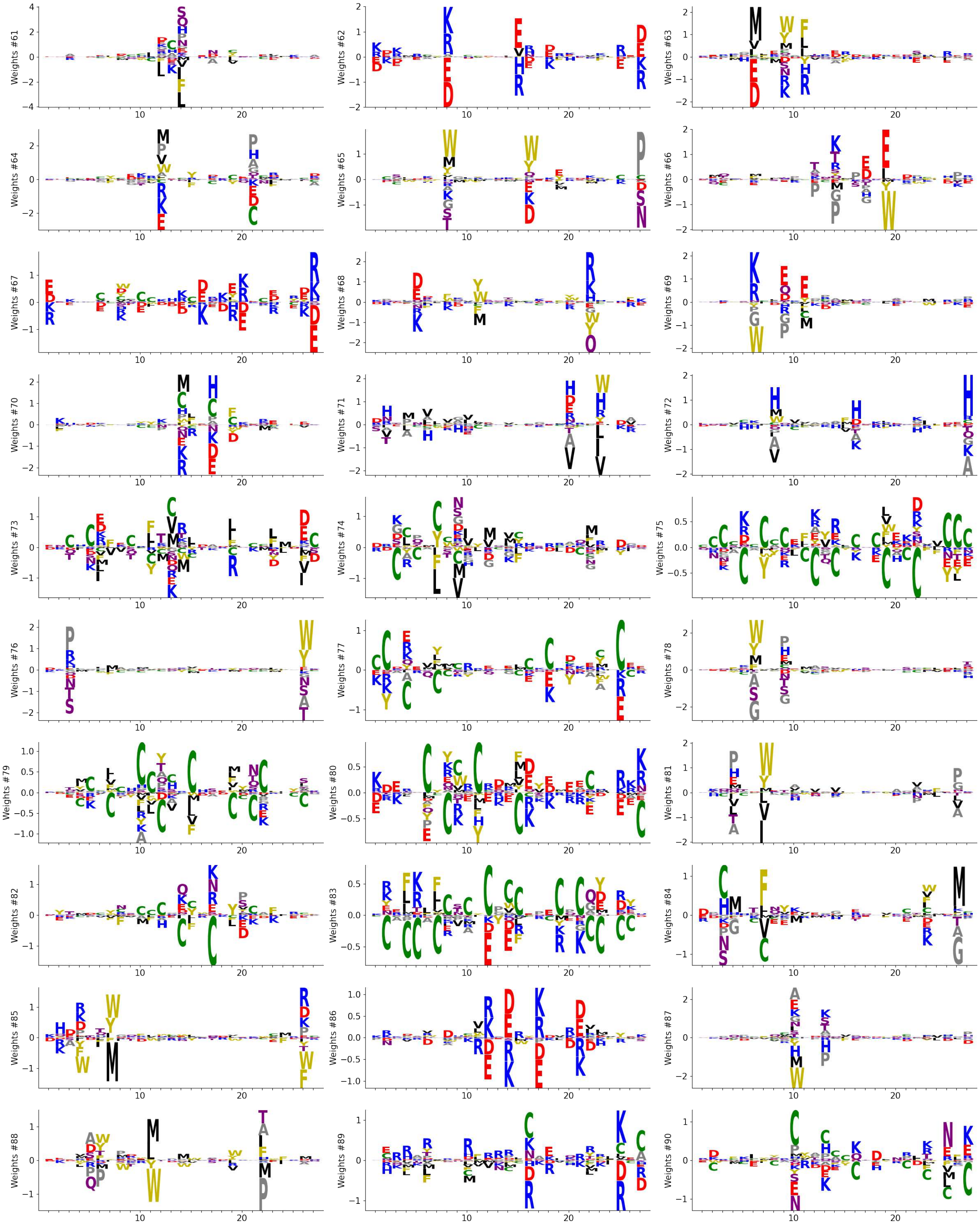


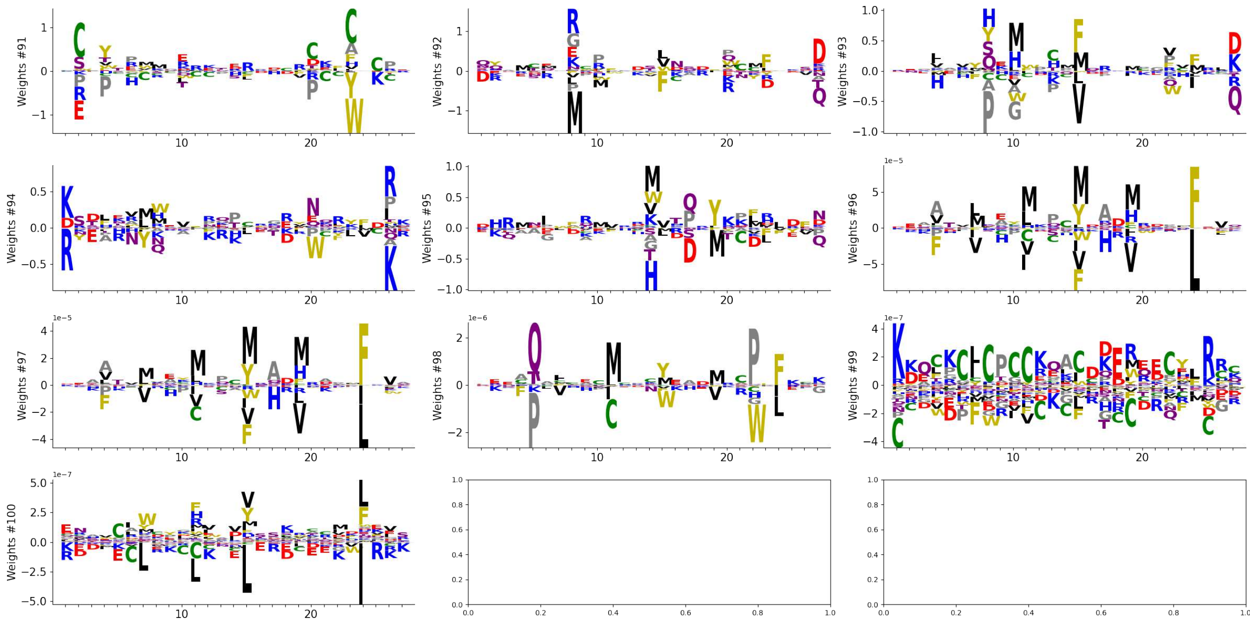

Weights #31
Weights #32
Weights #33
Weights #34
Weights #35
Weights #36
Weights #37
Weights #38
Weights #39
Weights #40
Weights #41
Weights #42
Weights #43
Weights #44
Weights #45
Weights #46
Weights #47
Weights #48
Weights #49
Weights #50

9 Appendix B: Weights Logo for the Lattice Proteins




Weights #31
Weights #32
Weights #33
Weights #34
Weights #35
Weights #36
Weights #37
Weights #38
Weights #39
Weights #40
Weights #41
Weights #42
Weights #43
Weights #44
Weights #45
Weights #46
Weights #47
Weights #48
Weights #49
Weights #50
Weights #51
Weights #52
Weights #53
Weights #54
Weights #55
Weights #56
Weights #57
Weights #58
Weights #59
Weights #60


Weights #61
Weights #62
Weights #63
Weights #64
Weights #65
Weights #66
Weights #67
Weights #68
Weights #69
Weights #70
Weights #71
Weights #72
Weights #73
Weights #74
Weights #75
Weights #76
Weights #77
Weights #78
Weights #79
Weights #80
Weights #81
Weights #82
Weights #83
Weights #84
Weights #85
Weights #86
Weights #87
Weights #88
Weights #89
Weights #90


Weights #91
Weights #92
Weights #93
Weights #94
Weights #95
Weights #96
Weights #97
Weights #98
Weights #99
Weights #100
1e−5
1e−5
1e−6
1e−7
1e−7