Smooth orientation-dependent scoring function for coarse-grained protein quality assessment

Mikhail Karasikov, Guillaume Pagès, Sergei Grudinin

Smooth orientation-dependent scoring function for coarse-grained protein quality assessment

Mikhail Karasikov[1,2,3,4], **Guillaume Pagès**[1], **and Sergei Grudinin**[1,✉]

[1]Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LJK, 38000 Grenoble, France.
[2]Center for Energy Systems, Skolkovo Institute of Science and Technology, Moscow, 143026, Russia.
[3]Moscow Institute of Physics and Technology, Moscow, 141701, Russia.
[4]Present affiliation: Department of Computer Science, ETH Zurich, Zurich, 8092, Switzerland.

Motivation: Protein quality assessment (QA) is a crucial element of protein structure prediction, a fundamental and yet open problem in structural bioinformatics. QA aims at ranking predicted protein models to select the best candidates. The assessment can be performed based either on a single model or on a consensus derived from an ensemble of models. The latter strategy can yield very high performance but substantially depends on the pool of available candidate models, which limits its applicability. Hence, single-model QA methods remain an important research target, also because they can assist the sampling of candidate models.

Results: We present a novel single-model QA method called SBROD. The SBROD (Smooth Backbone-Reliant Orientation-Dependent) method uses only the backbone protein conformation, and hence it can be applied to scoring coarse-grained protein models. The proposed method deduces its scoring function from a training set of protein models. The SBROD scoring function is composed of four terms related to different structural features: residue-residue orientations, contacts between backbone atoms, hydrogen bonding, and solvent-solute interactions. It is smooth with respect to atomic coordinates and thus is potentially applicable to continuous gradient-based optimization of protein conformations. Furthermore, it can also be used for coarse-grained protein modeling and computational protein design. SBROD proved to achieve similar performance to state-of-the-art single-model QA methods on diverse datasets (CASP11, CASP12, and MOULDER).

Availability and Implementation: The standalone application implemented in C++ and Python is freely available at https://gitlab.inria.fr/grudinin/sbrod and supported on Linux, MacOS, and Windows.

Contact: sergei.grudinin@inria.fr

Supplementary information: Supplementary data are available at *Bioinformatics* online.

Correspondence: ***sergei.grudinin@inria.fr***

1. Introduction

Proteins play an important role in fundamental biological processes such as biological transport, formation of new molecules, or cellular protection through binding to specific foreign particles such as viruses. This importance has triggered an extensive research of their function and mechanisms involved in these processes. In particular, investigation of protein folding, which plays an essential functional role in living cells, requires costly experiments that can be potentially replaced by cheaper and faster computational methods for modeling undiscovered protein structures (14).

A lot of progress has been recently made in protein structure prediction, a computational problem of determining the target protein structure given its amino acid sequence. Most of the methods proposed for protein structure prediction first generate a pool of plausible protein conformations (protein models) and then rank them using a certain QA method to select the top-ranked candidates. Therefore, being aimed at ranking protein models by their quality, QA methods constitute a crucial part of pipelines for protein structure prediction. Usually, these QA methods are based on scoring functions that predict similarity between protein models and the target structures in terms of such similarity measures as RMSD, GDT-TS, and TM-score (24). In particular, RMSD measures the average distance between the atoms of two superimposed protein conformations. GDT-TS and TM-score are designed to assess the quality of protein models being protein size independent and robust to local structural errors (24).

There are generally two types of QA methods. Consensus-model QA methods decide on the quality of individual protein models based on their statistics in the assessed model pool. In contrast, single-model QA methods consider only atoms of the assessed protein model with no additional information about other models in the pool and hence, these can be used for conformational sampling and structure refinement. Furthermore, the performance of consensus-model QA methods usually depends on single-model QA methods involved in the conformational sampling used for generating pools of assessed protein models. In addition, single-model QA methods are proved to achieve better performance compared to consensus-model QA methods on unbalanced protein model pools and in cases where protein models within assessed pool are very similar (21). In addition to these two main types of QA methods, techniques combining both ideas have also been proposed (11, 18), referred to as quasi-single model QA methods.

Among recently proposed single-model QA methods, there are generally two main approaches to design a scoring function: physics-based and knowledge-based (data-driven) approaches (7, 17). Physics-based scoring functions are constructed according to some physical knowledge of interactions in the system. This approach takes its roots from the Gibbs free energy minimization principle, which states that all target protein structures minimize the Gibbs free energy over the whole conformational space. However, precise esti-

mation of the Gibbs free energy requires exhaustive sampling of a huge number of conformational states (4, 23), which is computationally intractable in most practical cases. The physics-based approaches are aimed at constructing scoring functions (often called energy potentials or force-fields) that approximate the enthalpic part of the Gibbs free energy and can be estimated efficiently. Usually, these potentials decompose the total energy into a sum of additive terms (contributions) that represent stretching of bonds or angles, dihedral potentials, electrostatic and van der Waals interactions, etc. Alongside with the physics-based approaches, there are so-called knowledge-based approaches that deduce the essential energies of molecular interactions from the structural and sequence databases assuming a certain distribution of conformations or minimizing a certain loss function. The respective scoring functions are typically derived either by machine learning or by estimating the probabilities of certain conformations (statistical QA methods) using statistics of determined native protein structures from structural databases. Section A in Supplementary Information overviews several commonly used representative QA methods.

Although plenty of QA methods have been proposed, often they miss such meaningful contributions as solvation-related terms and terms related to hydrogen bonding interactions. However, these contributions are important and generally should be taken into account. For instance, hydrogen bonds provide structural organization of distinct protein folds (10). In addition, most of QA methods require all-atom protein models as input, and thus their performance critically depends on the accuracy of side-chain packing, that is, positions of the side-chain atoms. These can be modeled with the widely-used SCWRL4 tool (15), as in (3), or any other method (16). A possibility of working in a simplified coarse-grained representation of amino acids, as in (14), overcomes this issue and also reduces the overall computational complexity. Another drawback of many existing protein scoring functions is their discontinuity caused, e.g, by penalties introduced for mismatched inferred and predicted secondary structures. Because of that, these methods cannot be used for gradient-based structure optimization.

In this paper, we propose a novel method for protein quality assessment, the Smooth Backbone-Reliant Orientation-Dependent (SBROD) scoring function. SBROD is a single-model QA method that scores protein models using only geometric structural features along with the explicit representation of solvent generated on a regular grid around assessed proteins. It requires only coordinates of the protein backbone, and thus is insensitive to conformations of the side-chains. In addition, the SBROD scoring function is continuous with respect to coordinates of the protein atoms, which makes it also potentially applicable for being used in molecular mechanics applications.

2. Methods

The workflow of SBROD comprises two stages. First, the method extracts features from each protein model in the dataset. Then, the scoring function assigns a score to each processed protein model based on its features extracted on the first stage. Figure 1 schematically shows the workflow with four groups of geometric features, which are based on the four types of inter-atomic interactions described in detail below. Once these features are extracted and preprocessed, a Ridge Regression model (5) is trained to predict the GDT-TS of protein models. For the preprocessing, the features are either scaled individually, so that they lie in the range $[-1,1]$ for the whole training set (the Scaler boxes in Figure 1), or they are normalized, so that the ℓ^2 norm of each non-zero feature vector is equal to one (the Normalizer boxes in Figure 1).

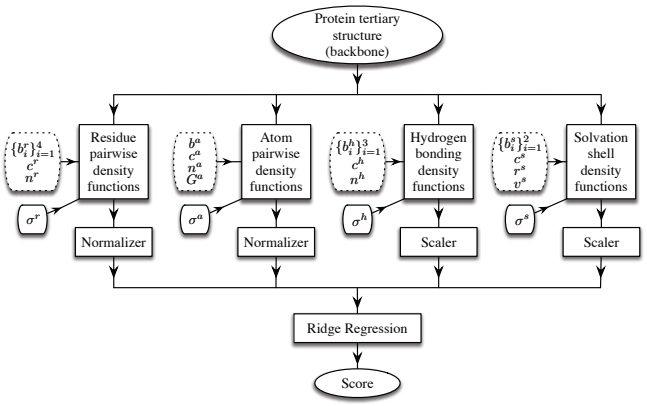

Fig. 1. Workflow of the SBROD QA method. The dotted blocks correspond to tunable structural parameters that are to be chosen at the training stage.

We should note that we also tried to use more sophisticated models including Lasso (22), Elastic Net (26), Bayesian Regression (20), Ranking SVM (12) in combination with PCA and Random Projections (1) for dimensionality reduction, as well as their different modifications and ensembles. However, these did not surpass Ridge Regression significantly regarding the prediction performance. In this section, we thoroughly describe the proposed method: from feature generation to training the scoring functions.

A. Feature extraction. We build a feature space that reflects four types of physically interpretable interactions: residue-residue pairwise interactions, backbone atom-atom pairwise interactions, hydrogen bonding interactions, and solvent-solute interactions. The four respective procedures for feature extraction are implemented in a unified manner. Namely, we iterate over predefined pairs of atomic groups and for each pair we compute feature descriptors that characterize configuration of atoms of one group in the pair with respect to atoms of another group in this pair. The atomic groups are defined by the aforementioned interactions and consist of either individual backbone atoms, atoms that encode orientation of side-chains, atoms specific to the backbone hydrogen bonds, or atoms specific to protein-solvent interactions. We should specifically emphasize that our initial protein model representation contains only heavy backbone atoms. The required positions of backbone amide hydrogens and missing C_β atoms are unambiguously reconstructed using geometry of the input backbone.

Figure 2 schematically shows descriptors that we use. Indices

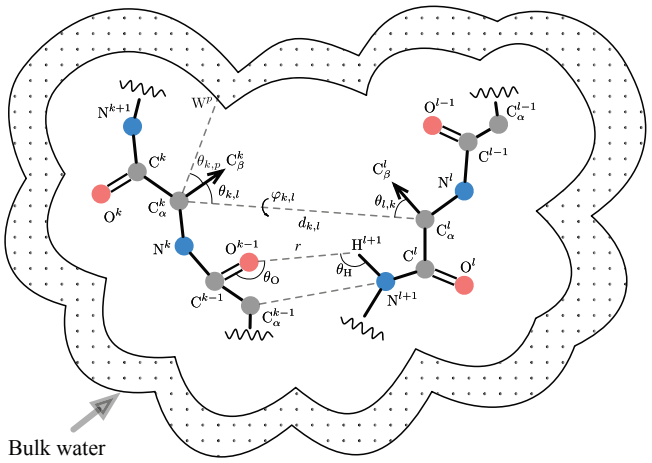

Fig. 2. Schematic representation of a protein tertiary structure with four types of structural features. First, the residue-residue pairwise features encode relative geometry of residues k and l. These features are the distance $d_{k,l}$ between C_α atoms, and three angular parameters $\phi_{k,l}$, $\theta_{k,l}$, and $\theta_{l,k}$. Second, each distance between a pair of heavy backbone atoms within a certain cutoff distance, e.g., C_α^{k-1} and N^{l+1}, contributes to the backbone atoms' features. Third, the hydrogen bonding features are based on the orientations of the donor-acceptor pairs of atoms relative to the respective residues, which are defined by the donor angles θ_H, the acceptor angles θ_O, and the bond lengths r. Finally, the solvation features are comprised of relative positions of the C_α atoms with respect to explicitly generated regular grid of water oxygens (bulk water).

k and l designate a pair of residues in a protein sequence. Symbols $d_{k,l}$ and r correspond to distances between atoms, $\theta_{k,l}$, θ_O, and θ_H denote angles between vector pairs, and $\phi_{k,l}$ is the dihedral angle between two planes passing trough carbon atoms $C_\alpha^k, C_\alpha^l, C_\beta^l$ and through carbon atoms $C_\alpha^l, C_\alpha^k, C_\beta^k$ from residues k and l. In a degenerate case when the dihedral angle is undefined, we choose its value randomly from the interval of possible values. While the intervals of possible values for the angle descriptors are bounded ($\theta \in [0,\pi]$ and $\phi \in [0,2\pi)$), the distance descriptors can generally fall into $[0,\infty)$. However, we introduce a cutoff distance $c < \infty$ and assume interactions between atoms beyond this distance negligible, thereby restricting this interval to the segment $[0,c]$.

For each descriptor, we partition the interval of its possible values into bins of equal width and compute the continuous number density functions (CNDF) for these bins. CNDF is a continuous function of descriptors that generalizes the notion of a standard histogram. This generalization makes the final scoring function smooth with respect to coordinates of the protein atoms. Let us demonstrate the computation of CNDF on example descriptors $\{(d_i^1, d_i^2)\}_{i=1}^n$. Let K be the truncated Gaussian kernel with the support width h:

$$K(x;\sigma,h) = \frac{\mathbb{1}[-h/2 \le x \le h/2] f_\sigma(x)}{\int_{-h/2}^{h/2} f_\sigma(\xi) d\xi},$$
$$f_\sigma(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}}, \quad \textbf{(1)}$$

where $\mathbb{1}[\cdot]$ designates the truth predicate, which converts any logical proposition into number 1 if the proposition is correct, and 0 otherwise. We define CNDF for a bin $[a^1,b^1)\times[a^2,b^2)$ as the sum of convolutions

$$\sum_{i=1}^n \int_{a^1}^{b^1} K\left(x - d_i^1; \sigma^1, h^1\right) dx \int_{a^2}^{b^2} K\left(x - d_i^2; \sigma^2, h^2\right) dx, \quad \textbf{(2)}$$

but not the number of hits into this bin as in the standard histogram,

$$\sum_{i=1}^n \mathbb{1}\left[a^1 \le d_i^1 < b^1\right] \mathbb{1}\left[a^2 \le d_i^2 < b^2\right]. \quad \textbf{(3)}$$

Below we specify four proposed feature extraction procedures, where each is parametrized by tunable parameters shown on the left side of each of the four feature blocks in Figure 1. These parameters change the estimated descriptors and the computation of CNDF.

A.1. Residue-residue pairwise features. The first type of structural features corresponds to interactions between protein residues. We treat amino acids of different types independently and compute CNDF for each pair of residues. Overall, we use 22 amino acid types that include the 20 standard types as well as selenocysteine and selenomethionine: $\mathcal{A} = \{\text{Ala}, \text{Arg}, \ldots, \text{Val}, \text{Sec}, \text{Mse}\}$. For a pair of residues, we compute four descriptors of their relative orientation as it is shown in Figure 2. For residues k and l, these are the distance $d_{k,l}$ between centers of the alpha carbon atoms, the dihedral angle $\phi_{k,l}$, and two angles, $\theta_{k,l} = C_\beta^k C_\alpha^k C_\alpha^l$ and $\theta_{l,k} = C_\beta^l C_\alpha^l C_\alpha^k$. Note, these descriptors depend only on positions of the C_α and C_β atoms. The CNDF are then computed as follows:

$$\begin{aligned} d_{a'a''}(i_1,i_2,i_3,i_4) = \\ = \sum_{(k,l)} \Bigg(& \int_{\frac{c^r}{b_1^r}(i_1-1)}^{\frac{c^r}{b_1^r} i_1} K\left(x - d_{k,l}; \sigma^r, \frac{c^r}{2b_1^r}\right) dx \\ & \times \int_{\frac{2\pi}{b_2^r}(i_2-1)}^{\frac{2\pi}{b_2^r} i_2} K\left(x - \phi_{k,l}; \sigma^r, \frac{\pi}{b_2^r}\right) dx \\ & \times \int_{\frac{\pi}{b_3^r}(i_3-1)}^{\frac{\pi}{b_3^r} i_3} K\left(x - \theta_{k,l}; \sigma^r, \frac{\pi}{2b_3^r}\right) dx \\ & \times \int_{\frac{\pi}{b_3^r}(i_4-1)}^{\frac{\pi}{b_3^r} i_4} K\left(x - \theta_{l,k}; \sigma^r, \frac{\pi}{2b_3^r}\right) dx \Bigg), \end{aligned} \quad \textbf{(4)}$$

where b_t^r is the number of bins for the t-th descriptor, $i_t \in \{1,\ldots,b_t^r\}$ are the indexes of bins into which the interval of possible values for the t-th descriptor was partitioned, and the sum is taken over all residue-residue pairs (k,l) of certain types $(a',a'') \in \mathcal{A}^2$ for which the distance between their alpha carbon atoms is less than $c^r + R_k + R_l$, where R_k and R_l are the effective side-chain sizes of the k-th and l-th residues correspondingly. These side-chain sizes vary from 0 Å for glycine, to 6.3 Å for arginine. We take $c^r = 5$ Å as the cutoff distance, $b_1^r = 10$ bins for the distance descriptor, and $b_{2,3}^r = 12$ bins for the angle descriptors. These descriptor parameters were chosen on the cross-validation step described

below. We have also conducted additional experiments where we excluded pairs of residues neighboring in the protein sequence ($n^r > 0$ in Figure 1), but the cross-validation revealed that counting all such pairs ($n^r = 0$) works best. To preserve sparsity of the features, the support width of each truncated Gaussian kernel was set to one half of the respective bin's width.

A.2. Backbone atom-atom pairwise features. The second type of structural features corresponds to interactions between the backbone atoms. We use residue-specific backbone atom types G^a. More precisely, we define types of heavy backbone atoms of each amino acid by their element symbols (C, N, O) and the amino acid type,

$$\begin{aligned} G^a &= \mathcal{A} \times \{\mathrm{C},\mathrm{N},\mathrm{O}\} \\ &= \{(\mathrm{Ala},\mathrm{C}),\dots,(\mathrm{Ala},\mathrm{O}),(\mathrm{Arg},\mathrm{C}),\dots,(\mathrm{Arg},\mathrm{O}),\dots\}. \end{aligned} \tag{5}$$

Overall, we use 22×3 backbone atom types. We iterate over each pair of atoms of certain types within the cutoff distance $c^a = 7\,\text{Å}$ and describe their relative configuration by the interatomic distance. To compute the CNDF, we use the following formula,

$$d_{g'g''}(i) = \sum_{(k,l)} \int_{\frac{c^a}{b^a}(i-1)}^{\frac{c^a}{b^a}i} K\left(x - d_{k,l}; \sigma^a, \frac{c^a}{2b^a}\right) dx, \tag{6}$$

where $b^a = 25$ is the number of bins, $i \in \{1,\dots,b^a\}$ are the indexes of bins into which the interval $[0, c^a]$ was partitioned, and the sum is taken over all atom-atom pairs (k,l) of all types $g', g'' \in G^a$ within the cutoff distance c^a specified above. Similarly to the case of the residue-residue pairwise descriptors, the conducted cross-validation revealed that counting all covalently bonded atoms in proteins ($n^a = 0$ in Figure 1) works best.

A.3. Hydrogen bonding features. The structural features of the third type represent the hydrogen bonding interactions. To compute CNDF for the hydrogen bonds, we iterate over all donor-acceptor pairs (N,O) in the backbone within the cutoff distance $c^h = 6\,\text{Å}$. To describe the directionality of these interactions, three descriptors shown in Figure 2 are used. These are the distance r between the hydrogen atom H and the oxygen atom O, the donor angle $\theta_\mathrm{H} = \mathrm{NHO}$, and the acceptor angle $\theta_\mathrm{O} = \mathrm{HOC}$. Then, we compute CNDF $d(i_1, i_2, i_3)$ with $b^h_{1,2,3} = 6$ bins as for the case of residue-residue pairwise descriptors. The CNDF accumulates all pairs (N,O) observed in amino acids that are spaced apart in at least $n^h = 2$ positions in the protein sequence, i.e. we skip all the (N,O) pairs where the atoms N and O occur in the same residue or residues topologically neighboring in the amino acid sequence.

A.4. Solvent-solute features. To take into account the solvent-solute interactions, which make up the fourth type of structural features, we explicitly construct a regular grid of water oxygen atoms around the protein with a period of $r^s = 3\,\text{Å}$, as explained in (2, 8). Each point of the grid is located further than $v^s = 2\,\text{Å}$ from any protein backbone atom but closer than $20\,\text{Å}$ to at least one backbone atom. Note that we use only coordinates of the protein backbone atoms to construct the grid. Then, for each pair of alpha carbon and generated water oxygen atoms $(\mathrm{C}^k_\alpha, \mathrm{W}^p)$ within the cutoff distance $c^s = 15\,\text{Å}$, we compute two descriptors. These are the distance $d_{k,p}$ between these two atoms, and the angle $\theta_{k,p}$ between vectors $\mathrm{C}^k_\alpha\mathrm{C}^k_\beta$ and $\mathrm{C}^k_\alpha\mathrm{W}^p$ pointing towards the side-chain and the water oxygen, respectively. The distance $c^s = 15\,\text{Å}$ is made somewhat large to implicitly include the interactions of solvent with the protein side-chains.

To eliminate the effect of abrupt appearing and disappearing of water oxygens interacting with the protein atoms at short distances, we count interactions between alpha carbons and water oxygens with weights that smoothly decay when the oxygen atom approaches the protein backbone. First, for each water oxygen atom W^p, we calculate the distance between W^p and its nearest imaginary side-chain defined as follows:

$$d_p := \min_k d(\mathrm{W}^p, \mathrm{C}^k_\alpha) - R_k, \tag{7}$$

where $d(\mathrm{W}^p, \mathrm{C}^k_\alpha)$ is the distance between atoms W^p and C^k_α, and R_k is the effective side-chain size of the k-th residue. Then, the weights for the water oxygens are calculated as follows. The weight w_p for the water oxygen atom W^p equals to 0 if the distance between W^p and its nearest side-chain, i.e. d_p in Eq. Eq. (7), is less than the minimum threshold distance v^s, and w_p grows linearly to 1 when increasing the distance d_p:

$$w_p = \begin{cases} 0, & d_p < v^s, \\ \frac{d_p - v^s}{\Delta}, & v^s \le d_p < v^s + \Delta, \\ 1, & \text{otherwise}, \end{cases} \tag{8}$$

where $\Delta = 1\,\text{Å}$ is the width of the penalized window. Then, for each amino acid type $a \in \mathcal{A}$, we compute weighted CNDF $d_a(i_1, i_2)$ as follows:

$$\begin{aligned} d_a(i_1,i_2) = \sum_{(k,p)} w_p \Bigg(&\int_{\kappa_{i_1-1}}^{\kappa_{i_1}} K\left(x - d_{k,p}; \sigma^s, \frac{c^s}{2b^s_1}\right) dx \\ &\times \int_{\frac{\pi}{b^s_2}(i_2-1)}^{\frac{\pi}{b^s_2}i_2} K\left(x - \theta_{k,p}; \sigma^s, \frac{\pi}{2b^s_2}\right) dx \Bigg), \end{aligned} \tag{9}$$

where numbers of bins for the distance and angle descriptors are set to $b^s_1 = 3$ and $b^s_2 = 2$, respectively; $\kappa_i = v^s + \frac{c^s - v^s}{b^s_1} i$, $i = 0,\dots,b^s_1$ are the bin edges for the distance descriptor, and the sum is taken over all alpha carbon atoms C^k_α and over all generated water oxygen atoms W^p in the grid. The specific values for parameters $v^s = 2\,\text{Å}$ and $\Delta = 1\,\text{Å}$ were chosen at the cross-validation stage along with the values of other tunable parameters.

B. Machine learning. To train the SBROD scoring function, we apply *Ridge Regression*, a classical machine learning technique to build a linear model, for which the scores are

the weighted sums of features extracted from the assessed instances. The problem of training a scoring function can be formulated as follows. Let us denote the space of all protein structures by $\mathcal{P}$, and let $\mathcal{D}_1,\dots,\mathcal{D}_n \subset \mathcal{P}$ be decoy sets, where each decoy set $\mathcal{D}_i$ is a set of protein models corresponding to the same target protein structure $P_0^{(i)}$:

$$\mathcal{D}_i = \{\underbrace{P_0^{(i)}}_{\text{native}}, \underbrace{P_1^{(i)},\dots,P_{t_i}^{(i)}}_{\text{protein models}}\} \subset \mathcal{P}, \quad i=1,\dots,n, \qquad \textbf{(10)}$$

and let $S^*(P_j^{(i)}, P_0^{(i)})$ denote the ground truth score of the model $P_j^{(i)}$ from the decoy set $\mathcal{D}_i$, which reflects the similarity between the protein model $P_j^{(i)}$ and the native protein conformation $P_0^{(i)}$. Let $\mathbf{f}: \mathcal{P} \to \mathbb{R}^k$ be the feature extractor described in section A. Our task is to train a scoring function $S_{\mathbf{w},\mathbf{f}}: \mathcal{P} \to \mathbb{R}$ by minimizing the regularized empirical loss

$$\min_{\mathbf{w},\mathbf{b}} \quad R(\mathbf{w},\mathbf{b}) + \sum_{i=1}^{n}\sum_{j=0}^{t_i} L\left(S_{\mathbf{w},\mathbf{f}}(P_j^{(i)}) + b_i,\ S^*(P_j^{(i)}, P_0^{(i)})\right) \qquad \textbf{(11)}$$

over parameters $\mathbf{w} \in \mathbb{R}^k$ and $\mathbf{b} \in \mathbb{R}^n$. Here we introduce additional bias parameters $b_i \in \mathbb{R}$, $i=1,\dots,n$, to make the loss function L independent of the score shifts equal for all protein models in the decoy set $\mathcal{D}_i$, since we are interested only in ranking capacity of the trained scoring function. That is, we are only interested in the capability of the scoring function $S_{\mathbf{w},\mathbf{f}}(P)$ to rank protein models in $\mathcal{D}_i$ but not to predict the exact ground truth scores S^*. In other words, scoring functions $S_{\mathbf{w},\mathbf{f}}(P)$ and $S_{\mathbf{w},\mathbf{f}}(P) + \sum_{i=1}^{n}\sum_{j=1}^{t_i} b_i \mathbb{1}[P = P_j^{(i)}]$ $\forall b_1,\dots,b_n \in \mathbb{R}$ have the same performance when ranking the protein models from the training decoy sets $\mathcal{D}_1,\dots,\mathcal{D}_n$.

B.1. Training set. We train the SBROD scoring function on protein models from various CASP (Critical Assessment of protein Structure Prediction) experiments. We used multidomain models, as training on models split into single domains did not provide any noticeable change in the performance of the trained scoring function. For the same reason, we did not filter out any abnormal structures or target structures with all models of poor quality. Server predictions participated in CASP were downloaded from the official CASP website at http://predictioncenter.org/download_area and were used in training as protein decoy models.

The total number of structural features extracted from the training protein models was 4,371,840 for the residue-residue features (with 99.92% of zeros on average, i.e. average sparsity), 239,775 for the backbone atom-atom features (96.29% sparsity), 216 for h-bonding (65.32% sparsity), and 138 for solvent-solute (27.32% sparsity). The average total number of nonzero elements in the features was 12,617.

Augmenting training sets with NMA-based decoy protein models. We propose a new approach for augmentation of protein decoy sets. For each target structure in the CASP training set, we generate random structure perturbations based on the Normal Mode Analysis. These decoy models are generated by the NOLB tool (9) combining deformations along 100 slowest normal modes with random amplitudes. We generate 300 decoy models for each target structure with RMSD in the range of 0.5–6 Å.

B.2. Model scores. Although there are multiple ways to measure the similarity between protein models and target structures, the most accepted one in the protein structure prediction community is the global distance test total score (GDT-TS). The GDT-TS of a protein model is an average percent of its residues that can be superimposed with the corresponding residues in the target structure under selected distance cutoffs of 1, 2, 4, and 8 Å. We use the TM-score utility developed by (25) to compute the GDT-TS of protein models. The computed GDT-TS of a protein model $P_j^{(i)}$ against its corresponding target structure $P_0^{(i)}$ (see section B for notations) is denoted by $S^*(P_j^{(i)}, P_0^{(i)})$ and treated as the ground truth score of the model $P_j^{(i)}$.

B.3. Ranking model. In our method Eq. (11), we use a linear ranking function $S_{\mathbf{w},\mathbf{f}}(P) = \mathbf{w}^{\mathsf{T}}\mathbf{f}(P)$ with quadratic loss function and ridge regularization,

$$L(x,y) = (x-y)^2, \quad R(\mathbf{w},\mathbf{b}) = \alpha\left(\|\mathbf{w}\|_2^2 + \frac{1}{\beta^2}\|\mathbf{b}\|_2^2\right). \qquad \textbf{(12)}$$

Thus, the empirical loss minimization Eq. (11) can be rewritten as follows,

$$\min_{\tilde{\mathbf{w}}} \quad \alpha\|\tilde{\mathbf{w}}\|_2^2 + \sum_{i=1}^{n}\sum_{j=0}^{t_i}\left(\tilde{\mathbf{w}}^{\mathsf{T}}\tilde{\mathbf{f}}(P_j^{(i)}) - S^*(P_j^{(i)}, P_0^{(i)})\right)^2, \qquad \textbf{(13)}$$

which allows to train the scoring function using standard solvers, where

$$\tilde{\mathbf{f}}(P_j^{(i)}) = \Big[\mathrm{f}_1(P_j^{(i)}),\dots,\mathrm{f}_k(P_j^{(i)}),\underbrace{0,\dots,0,\beta}_{i},0,\dots,0\Big]^{\mathsf{T}} \in \mathbb{R}^{k+n},$$
$$\tilde{\mathbf{w}} = [w_1,\dots,w_k,b_1,\dots,b_n]^{\mathsf{T}} \in \mathbb{R}^{k+n}. \qquad \textbf{(14)}$$

Optimization. The optimization problem Eq. (13) is reduced to a system of linear equations and is solved by the conjugate gradient iterative method implemented in the SciPy Python library (13), adapted particularly to sparse matrices of a huge dimension.

Cross-validation. To estimate the best values of the tunable parameters in the feature extraction procedure (b_i^r, c^r, n^r, b^a, c^a, etc., see Figure 1), and also to select the best regularization parameters α and β in Eq. (13) and Eq. (14), we use a 3-fold cross-validation on the CASP[5-10] datasets. This is a standard technique for tuning free parameters of a predictive model. More precisely, the original dataset is randomly partitioned into k (here $k=3$) even parts. Then, the predictive model is trained on $k-1$ parts and validated on the remaining

single part. This process is repeated k times with each of the k parts used exactly once as the validation data. The k results from the folds are then averaged to produce a single estimation serving as a criterion of picking the best free parameters of the predictive model. Thus, all the training CASP[5-10] data is used for both training and validation. However, the remaining CASP[11-12] datasets are not involved in this process and are left for the final evaluation. As a result of the described process, the regularization parameters were set to be $\alpha = 5$, $\beta = 50$. The optimal parameters of the feature extraction procedure are specified above in section A.

3. Results and Discussion

We measured the performance of SBROD on the very recent CASP11 and CASP12 Stage1 and Stage2 datasets (19). We downloaded these datasets from the official CASP website at http://predictioncenter.org/download_area and merged them with the published crystallographic target structures. As a result, we obtained 84 and 83 decoy sets of protein models with the corresponding target structures for the CASP11 Stage1 and Stage2 datasets, respectively. Similarly, we obtained 40 decoy sets for CASP12 Stage1 and 40 decoy sets for CASP12 Stage2. The ground truth GDT-TS values were computed using the TM-score utility (25). The rest of CASP11 and CASP12 data were filtered out either because their corresponding target protein structures had not been published on the official CASP website or the TM-score utility terminated for those structures with error. No other data were filtered out.

To estimate the performance of a scoring function $S : \mathcal{P} \to \mathbb{R}$ on a decoy set $\mathcal{D} = \{P_0, \dots, P_t\}$ with a target structure P_0 from the test set, we evaluate the predicted scores $S(P_j)$, $j = 0, \dots, t$ and then, estimate the following performance measures:

- score loss

$$\text{Loss}(S, \mathcal{D}) = \left| S^*(\hat{P}, P_0) - \max_{j=1,\dots,t} S^*(P_j, P_0) \right|, \quad \textbf{(15)}$$

  where $\hat{P} = \underset{P \in \{P_1,\dots,P_t\}}{\arg\max} S(P_j)$ is the top-ranked protein model;

- the Pearson correlation coefficient between predicted scores $S(P_j)$ and the ground truth $S^*(P_j, P_0)$ for decoy models $j = 1, \dots, t$;

- the Spearman rank correlation coefficient, i.e. the Pearson correlation coefficient between ranks of scores $\text{rg}S(P_j)$ and $\text{rg}S^*(P_j, P_0)$, where $\text{rg}X_j$ denotes the rank of the value X_j in a set of numbers $\{X_j\}_{j=1}^t$;

- the Kendall rank correlation coefficient.

Note that the target protein structures P_0 are excluded when estimating the performance measures and are used only to compute the ground truth scores of the decoy protein models. Finally, we compute the average of the estimated performance measures over all decoy sets in the test set.

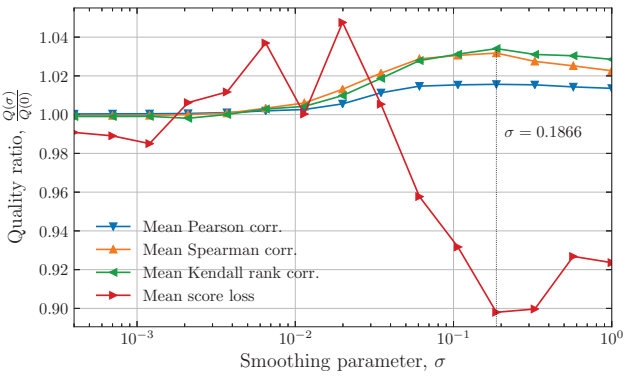

Fig. 3. The performance of SBROD on the CASP10 dataset (Stage1 and Stage2 combined) for different values of the smoothing parameters $\sigma^a = \sigma^r = \sigma^h = \sigma^s = \sigma$. The SBROD scoring function was trained on the CASP[5-9] datasets using features without smoothing ($\sigma = 0$).

A. Smoothness of CNDF. The parameters of calculated CNDF (see section A for definition) affect the extracted features and hence the performance of SBROD. Although the parameters of the feature extraction procedures were either optimized on the cross-validation stage or chosen manually, the smoothing parameters σ^r, σ^a, σ^h, σ^s were tuned independently. Moreover, these parameters were set to zero during all training stages (i.e. only degenerate CNDF with $\sigma \to 0$ in the truncated Gaussian kernel Eq. (1) were used in training) to increase sparsity of the features in training sets, which reduced the complexity and made the training tractable.

To optimize the smoothing parameters and thereby to improve the scoring capacity of SBROD, we first trained a distinct scoring function on the CASP[5-9] datasets without smoothing, i.e. $\sigma^a = \sigma^r = \sigma^h = \sigma^s = 0$. Then, we measured the dependence of the four performance measures described above (mean score loss, mean Pearson, Spearman, and Kendall rank correlation coefficients) on values of the smoothing parameters when testing on the CASP10 dataset (Stage1 and Stage2 combined) with different levels of smoothing by changing the support widths of the truncated Gaussian kernels σ^a, σ^r, σ^h, σ^s. Figure 3 shows the ratio of the prediction performance with and without the feature smoothing. One can see that the smoothing technique improves the performance of the scoring function. According to all the performance measures, the optimal smoothing parameter appeared to be $\sigma = 0.187$. Thus, we used this value in all other experiments.

B. Feature contributions. To calculate individual contributions for all the four types of structural features, we set to zero all trained weights w_i (see Eq. Eq. (14)) corresponding to three out of the four feature groups that are not under consideration (see Eq. Eq. (13)) and estimated the performance measures on the CASP11 Stage2 test set. Then, we repeated this procedure for each of the other three feature types. Table 1 lists the results. It can be observed that the features corresponding to residue-residue pairwise interactions contribute to the performance of SBROD the most. However, features representing backbone atom-atom pairwise interactions ensure the best GDT-TS loss performance. We should

Table 1. Contributions of different feature groups to the SBROD performance. This was measured on the CASP11 Stage2 dataset.

Feature groups	GDT-TS loss	Pearson	Spearman	Kendall
All features	0.057	0.441	0.426	0.298
Residue-residue	0.078	0.380	0.365	0.253
Backbone atom-atom	0.069	0.344	0.327	0.224
Solvation shell	0.107	0.267	0.271	0.189
Hydrogen bonds	0.112	0.142	0.126	0.089

also note that the protein-solvent interactions alone already give information sufficient to score protein models with a fair enough performance. Weights respective to the hydrogen bonds features provide the poorest predictive ability. This might be the case because the information about the hydrogen bonds is already included in other features and can be inferred from the relative orientation of protein residues, for example. Finally, one can see from Table 1 that usage of all the proposed features provides a significant gain in performance of SBROD compared to the individual contributions.

C. Amount of training data. An interesting question is whether we can improve the performance of our scoring function by training on more decoy sets or by artificially augmenting the training set. To study this, we conducted a computational experiment where we trained SBROD on different subsets of the CASP[5-10] datasets using both CASP server submissions and NMA-based decoy protein models (see section B.1). The trained scoring functions were validated on the CASP11 Stage2 dataset. Figure 4 shows the learning curves for estimated performance measures. One can observe that the performance of SBROD trained on the NMA-based decoy protein models becomes stable when the number of decoy sets used for training reaches 300, and no further extension of the training set improves this performance. In contrast, the performance of SBROD trained on the CASP protein models grows steadily when increasing size of the training set. Note that usage of both datasets combined together improves the correlation criteria for training sets with more than 150 decoy sets. Finally, Figure 4 makes reasonable the assumption that the performance of SBROD can be improved by extending the training set, e.g. by including the CASP12 protein models.

D. Comparison with the state-of-the-art. To compare the performance of SBROD against nine state-of-the-art QA methods, we first used the results obtained by (3). They assessed the performance of several QA methods against the ground truth GDT-TS computed with the LGA utility (24) for structures with side-chains repacked with SCWRL4 (15) on the CASP11 Stage1 and Stage2 datasets. Since the LGA utility (24) is not openly available, we used the TM-score utility (25) instead. Nonetheless, SBROD is not sensitive to the side-chains packing, and the difference between the GDT-TS computed by the TM-score and LGA utilities is negligible. Therefore, the measurements estimated by (3) are consistent with ours, measured as described above, and all of these can be fairly compared to each other.

Tables S1a and S1b in Supplementary Information list the

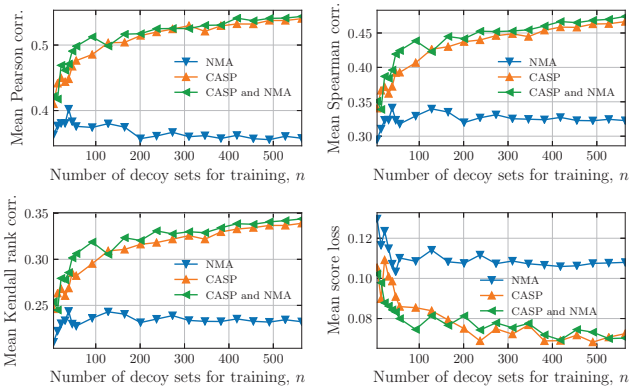

Fig. 4. Learning curves for the performance of SBROD on the validation set as a function of the number of training decoy sets. The training was performed on random subsamples of CASP[5-10]. The validation was done using the CASP11 Stage2 set.

performance measures computed for the SBROD scoring function (trained on the CASP[5-10] data augmented with the generated NMA-based decoy models, with the CNDF smoothing parameters of $\sigma^a = \sigma^r = \sigma^h = \sigma^s = 0.187$ on the testing stage) and for nine other state-of-the-art methods on the CASP11 Stage1 and Stage2 datasets, correspondingly. It can be seen that our method outperforms all other methods on both stages of the CASP11 experiment if assessed by the mean score loss, and it is highly competitive to the other methods if assessed by the other performance measures.

We also repeated a similar experiment using the CASP12 Stage1 and Stage2 data. For this experiment, the SBROD function was trained on CASP[5-11] data augmented with the generated NMA-based decoy models, and more recent methods were added for the comparison (Section B in Supplementary Information provides details on those). Tables 2 list the results on the original CASP12 server submissions, and Tables 3 list the results for the CASP12 data preprocessed with side-chains repacking. As in the previous experiment, we can see that SBROD is highly competitive to the other methods, especially on the Stage2 data.

Finally, we assessed the performance of SBROD together with several other QA methods on the MOULDER dataset (6). This is a conventional dataset for testing physics-based and statistical energy potentials. Table S2a in Supplementary Information lists the results and one can see that SBROD is among the best performers there as well.

4. Conclusion

In this paper, we presented SBROD, a novel method for the single-model protein quality assessment. SBROD was developed in a general supervised machine learning framework. First, features were extracted and then, a predictive model was trained to construct the SBROD scoring function. It utilizes only geometric structural features, which can be directly extracted from the conformation of the protein backbone. Thus, conformations of the protein side-chains are not taken into account when ranking the protein structures. The SBROD scoring function includes four contributions from

Table 2. Performance of the selected QA methods measured on the CASP12 dataset (Stage1 and Stage2). Native protein structures were filtered out from the dataset. The second column lists GDT-TD losses, the last column lists average Z-scores estimated over the dataset.

QA Method	Loss	Pearson	Spearman	Kendall	Z-score
ProQ2-refine	0.098	0.623	**0.651**	**0.503**	2.403
ProQ2	0.099	0.633	0.646	0.495	2.327
ProQ3-repack	0.078	0.634	0.638	0.487	2.512
ProQ3	**0.028**	**0.661**	0.630	0.475	**3.000**
SBROD (this study)	0.076	0.649	0.612	0.462	2.535
VoroMQA	0.085	0.611	0.554	0.414	2.460
RWplus	0.132	0.479	0.465	0.344	2.090

QA Method	Loss	Pearson	Spearman	Kendall	Z-score
SBROD (this study)	**0.069**	**0.614**	**0.559**	**0.406**	1.024
ProQ2-refine	0.096	0.590	0.538	0.388	0.731
ProQ3	0.089	0.572	0.535	0.386	0.898
ProQ2	0.091	0.578	0.529	0.381	0.809
ProQ3-repack	0.070	0.601	0.526	0.381	**1.078**
VoroMQA	0.106	0.559	0.501	0.362	0.692
RWplus	0.103	0.417	0.378	0.265	0.778

Table 3. Performance of the selected QA methods measured on the CASP12 dataset (Stage1 and Stage2) with side-chain repacking by scwrl4 (15). Native protein structures were filtered out from the dataset. The second column lists GDT-TD losses, the last column lists average Z-scores estimated over the dataset.

QA Method	Loss	Pearson	Spearman	Kendall	Z-score
ProQ2-refine	0.097	0.623	**0.653**	**0.501**	2.429
ProQ2	0.098	0.623	0.650	0.500	2.397
ProQ3-repack	0.095	0.630	0.640	0.490	2.223
ProQ3	**0.060**	0.631	0.617	0.470	**2.581**
SBROD (this study)	0.076	**0.649**	0.613	0.463	2.535
VoroMQA	0.081	0.602	0.546	0.409	2.515
RWplus	0.124	0.481	0.464	0.341	2.102

QA Method	Loss	Pearson	Spearman	Kendall	Z-score
SBROD (this study)	0.069	**0.614**	**0.559**	**0.406**	1.024
ProQ2	0.086	0.594	0.540	0.393	0.881
ProQ3	0.082	**0.614**	0.539	0.392	1.026
ProQ2-refine	0.083	0.591	0.538	0.390	0.861
ProQ3-repack	**0.060**	0.599	0.522	0.378	**1.177**
VoroMQA	0.100	0.574	0.504	0.366	0.924
RWplus	0.104	0.477	0.412	0.291	0.679

residue-residue, backbone atom-atom, hydrogen bonding, and solvent-solute pairwise interactions. Performed computational experiments on diverse structural datasets proved SBROD to achieve the state-of-the-art performance of single-model protein quality assessment. More precisely, on both Stage1 and Stage2 datasets from the CASP11 protein structure prediction exercise (see Tables

Acknowledgements

The authors thank Research Center for Molecular Mechanisms of Aging and Age-Related Diseases, Moscow Institute of Physics and Technology. The authors also thank Prof. Yury Maximov from Skolkovo Institute of Science and Technology for his valuable advice and Prof. Vadim Strijov from Moscow Institute of Physics and Technology for helpful discussions. The authors also thank Georgy Derevyanko for his valuable help with running ProQ2 and ProQ3 on the CASP12 datasets and Elodie Laine for help with the manuscript.

Funding

This work was partially supported by the Inria Internships Program, L'Agence Nationale de la Recherche (grant number ANR-15-CE11-0029-03), and by the Ministry of Education and Science of the Russian Federation (grant number RFMEFI58715X0011).

1. Ailon, N. and Chazelle, B. (2009). The Fast Johnson–Lindenstrauss Transform and Approximate Nearest Neighbors. *SIAM Journal on Computing*, **39**(1), 302–322.
2. Artemova, S., Grudinin, S., and Redon, S. (2011). A comparison of neighbor search algorithms for large rigid molecules. *Journal of Computational Chemistry*, **32**(13), 2865–2877.
3. Cao, R. and Cheng, J. (2016). Protein single-model quality assessment by feature-based probability density functions. *Scientific Reports*, **6**, 23990.
4. Cecchini, M., Krivov, S. V., Spichty, M., and Karplus, M. (2009). Calculation of Free-Energy Differences by Confinement Simulations. Application to Peptide Conformers. *The Journal of Physical Chemistry B*, **113**(29), 9728–9740.
5. Draper, N. R. and Smith, H. (2014). *Applied regression analysis*, volume 326. John Wiley & Sons.
6. Eramian, D., Shen, M.-y., Devos, D., Melo, F., Sali, A., and Marti-Renom, M. A. (2006). A composite score for predicting errors in protein structure models. *Protein Science : A Publication of the Protein Society*, **15**(7), 1653–1666.
7. Faraggi, E. and Kloczkowski, A. (2014). A global machine learning based scoring function for protein structure prediction. *Proteins: Structure, Function, and Bioinformatics*, **82**(5), 752–759.
8. Grudinin, S., Garkavenko, M., and Kazennov, A. (2017). Pepsi-saxs: an adaptive method for rapid and accurate computation of small-angle x-ray scattering profiles. *Acta Crystallographica Section D: Structural Biology*, **73**(5).
9. Hoffmann, A. and Grudinin, S. (2017). NOLB: Nonlinear Rigid Block Normal-Mode Analysis Method. *Journal of Chemical Theory and Computation*, **13**(5), 2123–2134.
10. Hubbard, R. E. and Kamran Haider, M. (2001). Hydrogen Bonds in Proteins: Role and Strength. In *eLS*. John Wiley & Sons, Ltd.
11. Jing, X. and Dong, Q. (2017). Mqaprank: improved global protein model quality assessment by learning-to-rank. *BMC Bioinformatics*, **18**(1), 275.
12. Joachims, T. (2002). Optimizing Search Engines Using Clickthrough Data. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '02, pages 133–142, New York, NY, USA. ACM.
13. Jones, E., Oliphant, T., Peterson, P., and Others (2001). {SciPy}: Open source scientific tools for {Python}.
14. Kmiecik, S., Gront, D., Kolinski, M., Wieteska, L., Dawid, A. E., and Kolinski, A. (2016). Coarse-Grained Protein Models and Their Applications. *Chemical Reviews*, **116**(14), 7898–7936.
15. Krivov, G. G., Shapovalov, M. V., and Dunbrack, R. L. (2009). Improved prediction of protein side-chain conformations with SCWRL4. *Proteins*, **77**(4), 778–795.
16. Liang, S., Zheng, D., Zhang, C., and Standley, D. M. (2011). Fast and accurate prediction of protein side-chain conformations. *Bioinformatics*, **27**(20), 2913–2914.
17. Liu, Y., Zeng, J., and Gong, H. (2014). Improving the orientation-dependent statistical potential using a reference state. *Proteins*, **82**(10), 2383–2393.
18. Maghrabi, A. H. A. and McGuffin, L. J. (2017). Modfold6: an accurate web server for the global and local quality estimation of 3d protein models. *Nucleic Acids Research*, **45**.
19. Moult, J., Fidelis, K., Kryshtafovych, A., Schwede, T., and Tramontano, A. (2016). Critical assessment of methods of protein structure prediction: Progress and new directions in round XI. *Proteins: Structure, Function, and Bioinformatics*, **84**, 4–14.
20. Neal, R. M. (1996). *Bayesian Learning for Neural Networks*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.
21. Ray, A., Lindahl, E., and Wallner, B. (2012). Improved model quality assessment using ProQ2. *BMC Bioinformatics*, **13**(1), 224.
22. Tibshirani, R. (1996). Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, **58**(1), 267–288.
23. Tyka, M. D., Clarke, A. R., and Sessions, R. B. (2006). An Efficient, Path-Independent Method for Free-Energy Calculations. *The Journal of Physical Chemistry B*, **110**(34), 17212–17220.
24. Zemla, A. (2003). LGA: A method for finding 3D similarities in protein structures. *Nucleic Acids Research*, **31**(13), 3370–3374.
25. Zhang, Y. and Skolnick, J. (2007). Scoring function for automated assessment of protein structure template quality. *Proteins: Structure, Function, and Bioinformatics*, **68**(4), 1020.
26. Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society. Series B: Statistical Methodology*, **67**(2), 301–320.