ELASPIC2 (EL2): Combining contextualized language models and graph neural networks to predict effects of mutations

Alexey Strokach[1], Tian Yu Lu[1] and Philip M. Kim[1,2,3,4,*]

1. *Department of Computer Science. University of Toronto, Toronto, ON M5S 3E1, Canada.*
2. *Donnelly Centre for Cellular and Biomolecular Research. University of Toronto, Toronto, ON M5S 3E1, Canada.*
3. *Department of Molecular Genetics. University of Toronto, Toronto, ON M5S 3E1, Canada.*
4. *Lead Contact*

*Correspondence: pi@kimlab.org

Highlights

- The ELASPIC web server (http://elaspic.kimlab.org), first described in 2016, provides users with an intuitive interface for evaluating the effect of mutations on protein folding and protein-protein interaction.

- Since the publication of the ELASPIC web server, advances in deep learning have led to the development of graph neural networks as well as contextualized language models offering accurate representations of protein sequence and structure. Moreover, a wealth of additional training data has been released.

- We introduce two machine learning models which leverage the recent advances to more accurately predict the effect of mutations on protein folding and protein-protein interaction. The models incorporate features generated using pre-trained transformer- and graph convolution-based neural networks, and are trained to optimize a ranking objective function, which permits the use of heterogeneous training data.

- The ELASPIC web server has been updated to incorporate outputs from the new models. A REST API, interactive notebooks, and a standalone package are provided in order to facilitate the use of the new models in a variety of domains.

Additional information

Advance since any previous publication

- New and improved machine learning models for predicting the effect of mutations on protein folding and protein-protein interaction have been added. Those models incorporate features generated using pre-trained neural networks with attention and graph convolutional layers [1–3] and achieve state-of-the-art results on several validation tasks.
- A REST API has been added to the web server and exposed to the end user, allowing for programmatic evaluation of mutations in protein structures.
- The backend to the web server has been updated to run user-submitted jobs on a local SLURM cluster, allowing for greater scaling in response to user demand.

Usage statistics

Since the ELASPIC web server was published in 2016 [4], it has processed more than 8,300 user-submitted jobs and has evaluated more than 108,000 mutations. The monthly usage of the web server has been growing. The paper describing the ELASPIC web server has been cited 30 times [4], while the paper describing the original ELASPIC methodology has been cited 57 times [5].

Abstract

The ELASPIC web server allows users to evaluate the effect of mutations on protein folding and protein-protein interaction on a proteome-wide scale. It uses homology models of proteins and protein-protein interactions, which have been precalculated for several proteomes, and machine learning models, which integrate structural information with sequence conservation scores, in order to make its predictions. Since the original publication of the ELASPIC web server, several advances have motivated a revisiting of the problem of mutation effect prediction. First, progress in neural network architectures and self-supervised pre-trained has resulted in models which provide more informative embeddings of protein sequence and structure than those used by the original version of ELASPIC. Second, the amount of training data has increased several-fold, largely driven by advances in deep mutation scanning and other multiplexed assays of variant effect. Here, we describe two machine learning models which leverage the recent advances in order to achieve superior accuracy in predicting the effect of mutation on protein folding and protein-protein interaction. The models incorporate features generated using pre-trained transformer- and graph convolution-based neural networks, and are trained to optimize a ranking objective function, which permits the use of heterogeneous training data. The outputs from the new models have been incorporated into the ELASPIC web server, available at http://elaspic.kimlab.org.

Keywords

Variant effect prediction, stability prediction, affinity prediction, machine learning, graph convolutional neural network.

Introduction

Advances in DNA sequencing technology have drastically lowered the cost and improved the accuracy of high-throughput sequencing [6]. Interpreting the vast amount of genomic data that are generated to produce meaningful and actionable results remains a challenge.

In vivo and *in vitro* experimental techniques remain the gold standard for elucidating the effect of coding DNA sequence variants at the protein level. However, evaluating the effect of all discovered variants experimentally is not feasible, both in terms of time and resources that would be required. Accordingly, many computational tools have been developed to predict the effect of different variants and to prioritize them for experimental validation. The majority of tools rely on some form of a conservation score, describing the frequency with which a particular nucleotide or amino acid is found at the given position in domain-, protein- or genome-level alignments, in order to make their prediction [11–17]. However, while those sequence-based tools can be successful in predicting the deleteriousness of mutations, they provide little insight into how a given mutation produces its effects. Structure-based tools exist for predicting the effect of mutations on protein stability and protein interaction affinity [18–23], and those tools can aid in the construction of a mechanistic model explaining the effects of a given mutation. However, the majority of structure-based tools take as input the structure of the protein or protein complex, shifting the onus of finding the protein structure and, potentially, the structures of protein-protein interactions that are affected by the mutation, onto the user. ELASPIC was developed to facilitate the structural analysis and interpretation of mutations in the context of a protein-protein interaction network, allowing for high-throughput evaluation of thousands of mutations [4,5]. It uses sequence profiles and homology models, which have been precalculated for all proteins and known protein-protein interactions in the human proteome, and are calculated on the fly for non-human proteins, to predict changes in the Gibbs free energy of protein folding and binding associated with mutations. Predictions made using ELASPIC have compared favorably to other methods in a number of independent assessments [24–26].

Here, we introduce two new machine learning models, ELASPIC2 core (EL2core) and ELASPIC2 interface (EL2interface), which are better able to predict mutation-induced changes in protein stability and protein binding affinity, respectively, while being more than an order of magnitude faster than existing approaches and lacking dependencies on external databases or proprietary software. The new models use two pre-trained neural networks, ProteinSolver [27] and ProtBert [3], to featurise individual mutations, and they employ the gradient boosting decision tree (GBDT) [28] machine learning algorithm with a ranking objective function, allowing them to integrate data obtained using different *in vivo* and *in vitro* techniques. Predictions made using the new models, both in cross-validation and on an independent test set, show consistently high correlations with experimental measurements when compared to other methods. The new models have been integrated into the ELASPIC web server. We additionally provide a REST API, interactive notebooks, and a standalone package to facilitate the use of the EL2 models in a wide range of applications.

Results and Discussion

Data preparation

We collected a diverse set of datasets listing effects of mutations on metrics that are correlated with protein stability (Table 1) or protein binding affinity (Table 2). In addition to ProTherm [29,30] and SKEMPI [31,32], which were used to train the original ELASPIC models, the collected datasets included an extended set of ΔΔG and Tm measurements [24,33,34], results of 20 *in vivo* and *in vitro* deep mutation scanning experiments collected by Dunham et al. [35], which report the effects of mutations on protein folding [36–46] or binding [46–51], and curated sets of deleterious and benign mutations mapped to protein structures or protein-protein interaction interfaces [52–54]. While not all deleterious mutations decrease the stability of a protein or a protein-protein interaction, we expect that a large fraction of deleterious mutations do have this effect. We removed proteins with less than three mutations with unique experimental measurements in a given dataset, and we removed redundant sequences in mutation deleteriousness datasets. This process produced a training dataset with more than 250,000 mutations with associated effects on protein stability (Table 1) and more than 50,000 mutations with associated effects on protein binding affinity (Table 2). We used MMSeqs2 [55] to divide sequences in our training datasets into clusters having at most 30% sequence identity, and we created six train-test splits to be used for cross-validation, where sequences in the train and test subsets belonged to different clusters. Finally, we constructed an independent test dataset listing the effects of 3,229 mutations on the stability of the SARS-CoV-2 spike protein and the effect of 3,669 mutations on its affinity to the human ACE2 receptor [56].

We used ProteinSolver [27] and ProtBert [3] to generate features describing every mutation in our training dataset (see Methods). We also evaluated every mutation using Rosetta [57] and, except for mutations in the Rocklin (2017), SKEMPI 2.0, and Dunham (2020) datasets, using ELASPIC [5,58] and FoldX [59]. For our test datasets, in addition to ELASPIC, FoldX, and Rosetta, we also generated predictions using the mCSM [19] and PoPMuSiC [22] web servers.

Model training and evaluation

We grouped all mutations in our training dataset by the protein that they affect and the type of experimental measurement that is available, and we trained the gradient boosting decision tree (GBDT) algorithm, implemented in LightGBM [60], to correctly rank mutations in each group according to their effect. We used hyperparameter tuning and feature elimination to select hyperparameters and features that produce the best-performing models, as measured by the average Spearman's correlation coefficient obtained through six-fold cross validation (see Supp. Figure S1). When evaluating the accuracy of the models on the test datasets, we took the average of the predictions made using the six EL2core models and six EL2interface models that were trained during cross validation.

Spearman's correlation coefficients for predictions obtained by cross-validation using the EL2core and EL2interface models are shown in Figures 1A and 2A, respectively. For comparison, we also show Spearman's correlation coefficients for predictions made using ELASPIC, FoldX, Rosetta, ProteinSolver, and ProtBert. In the case of ProteinSolver and ProtBert, predictions are calculated as the difference in probabilities assigned by the neural networks to the wildtype and mutant residues. On all but the ProTherm and SKEMPI datasets, the EL2 models show stronger correlations with experimental measurements than other methods (Figures 1A and 2A).

In the case of ProTherm, the Spearman's correlation coefficient obtained by EL2core is significantly lower than the Spearman's correlation coefficients obtained by ELASPIC, FoldX, and Rosetta, while in the case of SKEMPI, the Spearman's correlation coefficient obtained by EL2interface is marginally lower than the Spearman's correlation coefficient obtained by ELASPIC while being comparable to the correlation coefficients obtained by other methods. We see two possible explanations for the relatively poor performance of the EL2 models on those datasets. First, since ProTherm and SKEMPI are the two primary datasets that are used to train or optimize models to predict the effect of mutations on protein stability and protein binding affinity, respectively, existing models are likely to show overly-optimistic results on those datasets. This explanation is supported by the observation that the EL2core model shows a weaker correlation than Rosetta on the ProTherm dataset but a stronger correlation than Rosetta on the Rocklin (2017) dataset, which uses high-throughput measurements of stability. Second, since one of the goals of EL2 models is to be robust to different experimental systems and conditions, it is possible that we sacrifice some accuracy compared to methods that are optimized for a narrow use-case, such as predicting *in vitro* ΔΔG values. We view this as a suitable tradeoff for making predictions that generalize well to a wide range of experimental setups and measurement techniques.

After selecting EL2 models that perform best in cross-validation, we evaluated those models on an independent test dataset reporting the effect of 3,229 mutations on the stability of the SARS-CoV-2 spike protein and the effect of 3,669 mutations on the affinity between the SARS-CoV-2 spike protein and the ACE2 receptor [56]. For comparison, we include predictions made using ELASPIC, FoldX, Rosetta, ProteinSolver, and ProtBert, as well as predictions made using the mCSM [19] and PoPMuSiC [22] web servers. EL2 models achieve the highest Spearman's correlation coefficients on both datasets, with the EL2core model showing a Spearman's correlation coefficient of 0.49 (Figure 1B) and the EL2interface model showing a Spearman's correlation coefficient of 0.62 (Figure 2B) between predicted values of protein stability and affinity, respectively, and experimental measurements. On the stability dataset, FoldX and PoPMuSiC achieve marginally lower correlation coefficients of 0.48 and 0.43, Rosetta and mCSM achieve significantly lower correlation coefficients of 0.38 and 0.33, while predictions made using ELASPIC show a negative correlation of -0.09 (Figure 1B). One reason for the low concordance between the values predicted by ELASPIC and experimental measurements is that the large, heteromeric structure of the SARS-CoV-2 spike protein is poorly represented in the ELASPIC training dataset; it is possible that we would obtain more accurate predictions if we used ELASPIC to evaluate the effect of mutations only on the receptor binding domain (RBD) of the spike protein. On the affinity dataset, ELASPIC, FoldX, and Rosetta show significantly weaker correlations than EL2interface, with Spearman's correlation coefficients of 0.51, 0.53, and 0.54, respectively, while mCSM affinity shows the weakest correlation, with a correlation coefficient of 0.37 (Figure 2B).

One notable finding is that correlations produced using ProteinSolver are competitive, and often better, than those obtained using other methods, especially on the test dataset (Figures 1B and 2B). This is surprising, since ProteinSolver is trained to reconstruct the identity of masked amino acids given the general topology of the protein with no information on protein-protein interactions. ProtBert shows higher correlations on datasets capturing mutation deleteriousness, such as Humsavar, ClinVar, and COSMIC (Figures 1A and 1B), which is consistent with ProtBert being a sequence model trained to reconstruct masked amino acids in protein sequences with no extraneous information regarding their structures.

It is worth emphasizing that the EL2 models evaluate the impact of mutations one to two orders of magnitude faster than other methods, taking several seconds to evaluate a single mutation, rather than minutes to hours. Furthermore, while we obtained a crystal structure or created a homology model for every protein in our training dataset, neither ProteinSolver nor ProtBert, which we use to prepare our input features, require the high-grained structural information of the protein or protein interaction. ProteinSolver and ProtBert take as input the amino acid sequence of the protein and, in the case of affinity prediction, its interacting partner, and ProteinSolver in addition takes as input a coarse-grained distance matrix listing shortest distances between all pairs of residues within 12 Å of each other. The distance matrix can be mapped from a structural template by following the alignment without the need for a homology model (this is how the training data for ProteinSolver is generated), which side-steps the most computationally-intensive aspect of predicting the effect of mutations on a proteome-wide scale.

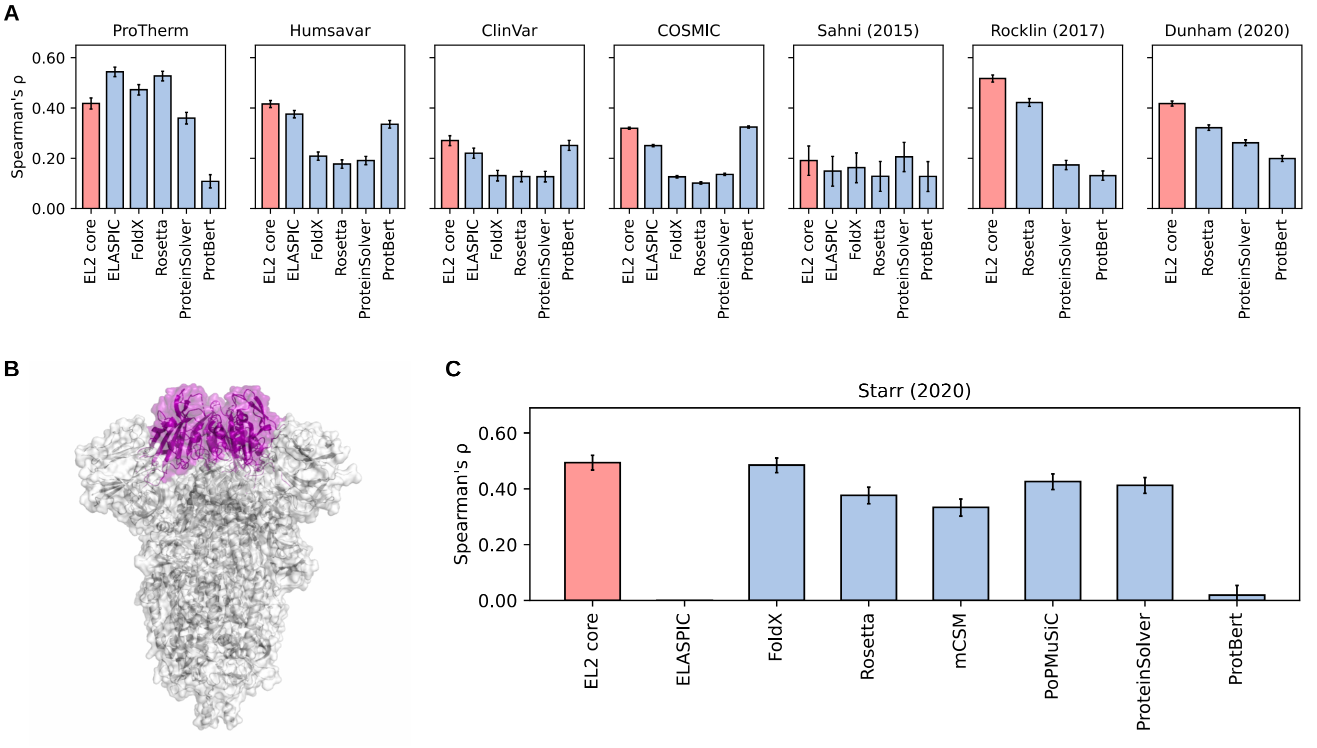

Figure 1. Performance of the EL2core model trained to rank the effect of mutations on protein folding. **(A)** Spearman's correlation coefficients obtained by EL2core, ELASPIC, FoldX, Rosetta, ProteinSolver and ProtBert on different datasets that were used to train the EL2core model (see Table 1). Correlations reported for EL2core and ELASPIC were obtained using six-fold cross validation. **(B)** Structure of the SARS-CoV-2 spike protein, with the region for which the stability of mutants was evaluated highlighted in purple. **(C)** Spearman's correlation coefficients obtained by EL2core, ELASPIC, FoldX, Rosetta, mCSM, PoPMuSiC, ProteinSolver and ProtBert for 3,229 mutations in the SARS-CoV-2 spike protein, which were not included in the EL2core training dataset. ELASPIC achieved a negative correlation of -0.09.

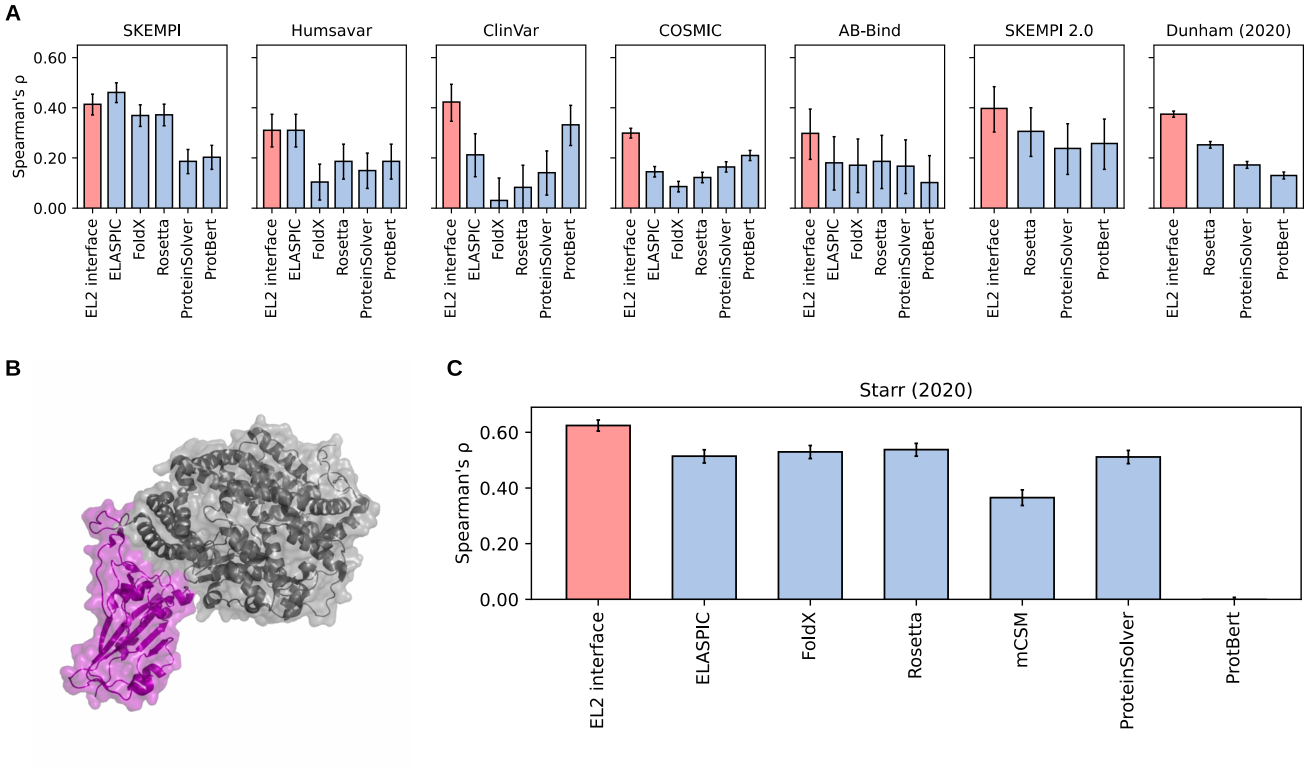

Figure 2. Performance of the EL2interface model trained to rank the effect of mutations on protein binding affinity. **(A)** Spearman's correlation coefficients obtained by EL2interface, ELASPIC, FoldX, Rosetta, ProteinSolver and ProtBert on different datasets that were used to train the EL2interface model (see Table 2). Correlations reported for EL2interface and ELASPIC were obtained using six-fold cross validation. **(B)** Structure of the SARS-CoV-2 spike protein receptor binding domain (RBD) (purple) in complex with the ACE2 protein (grey). The effect of mutating all residues in the RBD on the affinity of the interaction was evaluated. **(C)** Spearman's correlation coefficients obtained by EL2interface, ELASPIC, FoldX, Rosetta, mCSM, ProteinSolver and ProtBert for 3,669 mutations in the SARS-CoV-2 spike protein RBD.

Web server and other interfaces

We use several different interfaces to make EL2 models most widely accessible and useful. First, we have integrated the new EL2 models into the ELASPIC web server, available at http://elaspic.kimlab.org. Users of the web server are now provided EL2core and EL2interface predictions, in addition to the $\Delta\Delta G_{core}$ and $\Delta\Delta G_{interface}$ predictions made by the original version of ELASPIC. Second, we have developed a REST API which allows users to programmatically evaluate the impact of mutations and, if necessary, construct homology models of proteins or protein-protein interactions of interest, without downloading and installing the EL2 package locally. While tools such as the Ensembl Variant Effect Predictor (VEP) already provide a REST API for accessing predictions of mutation deleteriousness [61], we believe that EL2 is the first tool to provide a REST API for accessing predictions of mutation effects on protein stability and protein binding affinity. Documentation for the EL2 REST API, including several usage examples, is available at https://elaspic.uc.r.appspot.com/docs. Third, we have made available myBinder and Google Colab notebooks, which can be evaluated in order to reproduce EL2 predictions displayed in Figures 1B and 2B. Links to those notebooks are provided on the EL2 documentation page (https://elaspic.gitlab.io/elaspic-v2) and the notebooks can be easily modified to evaluate the impact of mutations on new proteins. Finally, the source code for EL2 has been made freely available at https://gitlab.com/elaspic/elaspic-v2.

Conclusion

In this work, we present EL2core and EL2interface, two machine learning models trained to predict the effect of mutations on protein folding and protein-protein interactions. We have updated the ELASPIC web server to incorporate the output of the two EL2 models, and we also provide a REST API, interactive notebooks, and a standalone package in order to facilitate their use in a variety of settings. The two EL2 models evaluate the effect of mutations orders of magnitude faster than existing approaches while rivaling or surpassing their accuracy when predicting the results of deep mutation scanning experiments. The ultimate test of accuracy for the new models would be an independent assessment, such as the Critical Assessment of Genome Interpretation community experiment [62], and we plan to participate in such assessments in the future.

Acknowledgments

AS acknowledges support from an NSERC PGS-D graduate scholarship and the Google Cloud Research Credits program. TYL acknowledges support from an NSERC Undergraduate Student Research Award. PMK acknowledges support from an NSERC Discovery grant (RGPIN-2017-064) and a CIHR Project grant (PJT-166008). We also acknowledge HPC support from a Compute Canada Resource Allocation and the NVIDIA academic GPU grant program.

Materials and Methods

Data preparation

Datasets that were used to train, validate, and test EL2core and EL2interface models are listed in Table 1 and 2, respectively. For each dataset, we obtained protein sequences, protein structures, lists of mutations, and the experimentally-measured effects of those mutations. In cases where a PDB structure of the protein was not available, we created homology models corresponding to the protein sequences. We removed proteins that have less than 3 mutations with unique experimental measurements, since the *lambdarank* objective function gives more robust results with a larger number of training examples per group. Furthermore, we excluded proteins from the ClinVar dataset if those proteins also appeared in the Humsavar dataset, and we excluded proteins from the COSMIC dataset if those proteins also appeared in either the Humsavar or the ClinVar datasets.

Table 1. Datasets used to train, validate, and test the EL2core model.

Dataset	Number of proteins	Number of mutations	Description	Cite
ProTherm	212	5,448	Mutation-induced changes in the Gibbs free energy of protein folding (ΔΔG) compiled from the Protherm database and from the datasets curated by Kellogg et al..	[29, 30]
Humsavar	1,129	13,236	Disease-causing mutations and polymorphisms obtained from the UniProt *humsavar.txt* file. Mutations annotated with at least one disease are assigned a value of 1. Mutations annotated as “polymorphisms” are assigned a value of 0.	[52]
ClinVar	1,293	8,664	Disease-causing mutations and polymorphisms obtained from ClinVar. Mutations with known phenotypic consequences are assigned a value of 1. Mutations with no known medical impact are assigned a value of 0.	[53]
COSMIC	10,524	184,323	Mutations commonly found in cancer. Mutations classified by FATHMM [16] as cancer drivers are assigned a value of 1. Mutations classified by FATHMM as cancer passengers are assigned a value of 0.	[54]
Sahni (2015)	376	1,064	Stability of wildtype and mutant proteins quantified by measuring their interaction with chaperones using the LUMIER assay.	[63]
Rocklin (2017)	14	10,674	Stability of existing and de novo designed proteins evaluated using thermal and chemical denaturation.	[34]
Dunham	12	26,049	Compilation of 12 deep mutation scanning	[36–

(2020)			experiments evaluating the stability and function of protein variants, curated by Dunham et al. [35].	46]
Starr (2020)	1	3,229	Deep mutation scanning experiment evaluating the effect of mutations in the receptor binding domain of the SARS-CoV-2 spike protein on protein expression, a close correlate of protein folding stability.	[56]

Table 2. Datasets used to train, validate, and test the EL2interface model.

Dataset	**Number of proteins**	**Number of mutations**	**Description**	**Cite**
SKEMPI	97	1,545	Mutation-induced changes in the Gibbs free energy of protein-protein interaction ($\Delta\Delta G$) compiled from the SKEMPI database and the dataset curated by Kortemme and Baker.	[31, 32]
Humsavar	173	739	Disease-causing mutations and polymorphisms obtained from the UniProt *humsavar.txt* file. Mutations annotated with at least one disease are assigned a value of 1. Mutations annotated as “polymorphisms” are assigned a value of 0. *Only those mutations that fall inside the protein binding interface of a known protein interaction are included in this dataset.*	[52]
ClinVar	146	545	Disease-causing mutations and polymorphisms obtained from ClinVar. Mutations with known phenotypic consequences are assigned a value of 1. Mutations with no known medical impact are assigned a value of 0. *Only those mutations that fall inside the protein binding interface of a known protein interaction are included in this dataset.*	[53]
COSMIC	1,880	8,594	Mutations commonly found in cancer. Mutations classified by FATHMM [16] as cancer drivers are assigned a value of 1. Mutations classified by FATHMM as cancer passengers are assigned a value of 0. *Only those mutations that fall inside the protein binding interface of a known protein interaction are included in this dataset.*	[54]
AB-Bind	40	319	Mutations explored in antibody-antigen affinity maturation experiments.	[64]
SKEMPI 2.0	210	3,146	Updated version of the SKEMPI database, listing more than twice the number of mutations and their associated effects on thermodynamic parameters governing specific protein-protein interactions.	[33]

Dunham (2020)	8	33,440	Compilation of 8 deep mutation scanning experiments evaluating the affinity between protein variants and their target, curated by Dunham et al. [35].	[46–51]
Starr (2020)	1	3,669	Deep mutation scanning experiment evaluating the effect of mutations in the receptor binding domain of the SARS-CoV-2 spike protein on its binding affinity to the ACE2 receptor.	[56]

Feature engineering

We used two pre-trained neural networks, ProteinSolver [27] and ProtBert [3], to generate features describing each protein or protein-protein interaction and their mutations. When predicting the effect of mutations on protein stability, we used ProteinSolver and ProtBert to obtain probabilities of wildtype and mutant residues, and we used ProtBert to obtain feature embeddings of wildtype and mutant residues and entire proteins, corresponding to the element-wise sum of the feature embeddings of all residues. We used the same features when predicting the effect of mutations on protein binding affinity, but we calculated those features both for the protein alone and the protein bound to its interaction partner. We extended the set of features, calculated as described above, by adding the differences in values obtained for wildtype and mutant proteins and, where applicable, the difference in values obtained for the protein alone and for the protein bound to its interaction partner. Finally, in the case of residue and protein embeddings, and the differences therein, we used principal component analysis (PCA) to reduce the 1024-valued vectors to 10 features, corresponding to the first 10 principal components.

For every mutation in our training dataset, we generated "reverse" mutations, converting mutant residues into wildtype residue, and we used those mutations to balance our training dataset, curtailing the over-representation of destabilizing mutations. The effect of such "reverse" mutations was assumed to be equal in magnitude and opposite in direction to the effect of the reference mutations.

Machine learning

We used the gradient boosting decision tree (GBDT) algorithm, implemented in LightGBM [60], to predict the effect of mutations on protein stability and protein binding affinity from the input features. While the original ELASPIC models were trained to minimize the root-mean-square deviation between predicted and actual ΔΔG values, the EL2 models were trained to optimize the *lambdarank* objective function [65,66], which encourages scores assigned to all pairs of mutations in a given group to be consistent with the relative rank of those mutations according to an experimental metric. Equation 1 details the essence of the *lambdarank* objective function, where *C* is the cost, s_i is the score assigned to mutation *i*, s_j is the score assigned to mutation *j*, and S_{ij} is 1 if mutation *i* is ranked higher than mutation *j* and 0 otherwise. The cost is lowest when $s_i >> s_j$ for all pairs *ij* where *Sij* is 1 and when *si* << *sj* for all pairs *ij* where *Sij* is 0.

$$C=\frac{1}{2}\left(1-S_{ij}\right)\sigma\left(s_i-s_j\right)+\log\left(1+\exp\left(-\sigma\left(s_i-s_j\right)\right)\right) \tag{1}$$

We used hyperparameter optimization and feature elimination to find sets of hyperparameters and features that produce models showing the highest Spearman's correlations in cross-validation. Hyperparameter optimization was performed by using the optuna framework [67] to select and evaluate 100 sets of hyperparameters. In each round of feature elimination, we selected the feature that should be removed by temporarily excluding each of the features from the training set, training the models, evaluating model performance using cross-validation, and finding the feature whose exclusion produced a model with the highest performance (see Supp. Figure S1). The final EL2core model uses 17 features (Supp. Table S2), and the final EL2interface model uses 16 features (Supp. Table S3).

Web server implementation

The EL2 REST API was implemented in Python using the FastAPI framework and is deployed to Google Cloud App Engine. We use GitLab CI to evaluate user-submitted mutations, which offers visibility into the status of every job and allows us to scale the number of machines in proportion to the number of user requests. The ELASPIC web server was implemented in Python using the Django framework.

Comparison with other methods

ELASPIC, FoldX, and Rosetta scores were computed by downloading and running locally ELASPIC v0.1.42 [58], FoldX v3.0b6 [59], and Rosetta v2020.37 [57]. System commands used to run the Rosetta protocols are provided in Supp. Table S1. For ProteinSolver and ProtBert, the score for each mutation was calculated as the difference in probabilities assigned by the networks to wildtype and mutant residues (Equations 2 and 3), taking into consideration the interacting partner, in the case of affinity prediction.

$$Score^{ProteinSolver} = {p^{ProteinSolver}}_{wt} - {p^{ProteinSolver}}_{mut} \qquad (2)$$

$$Score^{ProtBert} = {p^{ProtBert}}_{wt} - {p^{ProtBert}}_{mut} \qquad (3)$$

PoPMuSiC and mCSM scores were computed by using their respective web servers [19,22]. We also attempted to obtain predictions using the SSIPe, but the corresponding web servers did not return results within one week. ELASPIC and FoldX were not evaluated on the Rocklin (2017), SKEMPI 2.0, or Dunham (2020) datasets due to lack of computational resources.

Data availability

Data used to train and evaluate EL2 models are available at https://elaspic-v2.data.proteinsolver.org.

Software availability

Source code for EL2 is available at https://gitlab.com/elaspic/elaspic-v2.

References

[1] R. Rao, N. Bhattacharya, N. Thomas, Y. Duan, X. Chen, J. Canny, P. Abbeel, Y.S. Song, Evaluating Protein Transfer Learning with TAPE, ArXiv190608230 Cs Q-Bio Stat. (2019). http://arxiv.org/abs/1906.08230 (accessed August 10, 2020).

[2] J. Ingraham, V. Garg, R. Barzilay, T. Jaakkola, Generative Models for Graph-Based Protein Design, in: H. Wallach, H. Larochelle, A. Beygelzimer, F. d\textquotesingle Alché-Buc, E. Fox, R. Garnett (Eds.), Adv. Neural Inf. Process. Syst. 32, Curran Associates, Inc., 2019: pp. 15820–15831. http://papers.nips.cc/paper/9711-generative-models-for-graph-based-protein-design.pdf (accessed February 21, 2020).

[3] A. Elnaggar, M. Heinzinger, C. Dallago, G. Rehawi, Y. Wang, L. Jones, T. Gibbs, T. Feher, C. Angerer, M. Steinegger, D. Bhowmik, B. Rost, ProtTrans: Towards Cracking the Language of Life's Code Through Self-Supervised Deep Learning and High Performance Computing, BioRxiv. (2020) 2020.07.12.199554. https://doi.org/10.1101/2020.07.12.199554.

[4] D.K. Witvliet, A. Strokach, A.F. Giraldo-Forero, J. Teyra, R. Colak, P.M. Kim, ELASPIC web-server: proteome-wide structure-based prediction of mutation effects on protein stability and binding affinity, Bioinformatics. 32 (2016) 1589–1591. https://doi.org/10.1093/bioinformatics/btw031.

[5] N. Berliner, J. Teyra, R. Colak, S. Garcia Lopez, P.M. Kim, Combining Structural Modeling with Ensemble Machine Learning to Accurately Predict Protein Fold Stability and Binding Affinity Effects upon Mutation, PLoS ONE. 9 (2014) e107353. https://doi.org/10.1371/journal.pone.0107353.

[6] KA. Wetterstrand, DNA Sequencing Costs: Data from the NHGRI Genome Sequencing Program (GSP)., (2016). www.genome.gov/sequencingcostsdata (accessed September 14, 2016).

[7] C.-S. Ku, M. Wu, D.N. Cooper, N. Naidoo, Y. Pawitan, B. Pang, B. Iacopetta, R. Soong, Exome versus transcriptome sequencing in identifying coding region variants, Expert Rev. Mol. Diagn. 12 (2012) 241–251. https://doi.org/10.1586/erm.12.10.

[8] J. Eberwine, J.-Y. Sul, T. Bartfai, J. Kim, The promise of single-cell sequencing, Nat. Methods. 11 (2014) 25–27.

[9] C.C. Chrystoja, E.P. Diamandis, Whole genome sequencing as a diagnostic test: challenges and opportunities, Clin. Chem. 60 (2014) 724–733. https://doi.org/10.1373/clinchem.2013.209213.

[10] S. Nik-Zainal, H. Davies, J. Staaf, M. Ramakrishna, D. Glodzik, X. Zou, I. Martincorena, L.B. Alexandrov, S. Martin, D.C. Wedge, P. Van Loo, Y.S. Ju, M. Smid, A.B. Brinkman, S. Morganella, M.R. Aure, O.C. Lingjærde, A. Langerød, M. Ringnér, S.-M. Ahn, S. Boyault, J.E. Brock, A. Broeks, A. Butler, C. Desmedt, L. Dirix, S. Dronov, A. Fatima, J.A. Foekens, M. Gerstung, G.K.J. Hooijer, S.J. Jang, D.R. Jones, H.-Y. Kim, T.A. King, S. Krishnamurthy, H.J. Lee, J.-Y. Lee, Y. Li, S. McLaren, A. Menzies, V. Mustonen, S. O'Meara, I. Pauporté, X. Pivot, C.A. Purdie, K. Raine, K. Ramakrishnan, F.G. Rodríguez-González, G. Romieu, A.M. Sieuwerts, P.T. Simpson, R. Shepherd, L. Stebbings, O.A. Stefansson, J. Teague, S. Tommasi, I. Treilleux, G.G. Van den Eynden, P. Vermeulen, A. Vincent-Salomon, L. Yates, C. Caldas, L. van't Veer, A. Tutt, S. Knappskog, B.K.T. Tan, J. Jonkers, Å. Borg, N.T. Ueno, C. Sotiriou, A. Viari, P.A. Futreal, P.J. Campbell, P.N. Span, S. Van Laere, S.R. Lakhani, J.E. Eyfjord, A.M. Thompson, E. Birney, H.G. Stunnenberg, M.J. van de Vijver, J.W.M. Martens, A.-L. Børresen-Dale, A.L. Richardson, G. Kong, G. Thomas, M.R. Stratton, Landscape of somatic mutations in 560 breast cancer whole-genome sequences, Nature. 534 (2016) 47–54. https://doi.org/10.1038/nature17676.

[11] P.C. Ng, S. Henikoff, SIFT: predicting amino acid changes that affect protein function, Nucleic Acids Res. 31 (2003) 3812–3814.

[12] I. Adzhubei, D.M. Jordan, S.R. Sunyaev, Predicting Functional Effect of Human Missense Mutations Using PolyPhen-2, in: Curr. Protoc. Hum. Genet., John Wiley & Sons, Inc., 2001. http://onlinelibrary.wiley.com/doi/10.1002/0471142905.hg0720s76/abstract (accessed November 24, 2013).

[13] B. Li, V.G. Krishnan, M.E. Mort, F. Xin, K.K. Kamati, D.N. Cooper, S.D. Mooney, P. Radivojac,

Automated inference of molecular mechanisms of disease from amino acid substitutions, Bioinformatics. 25 (2009) 2744–2750. https://doi.org/10.1093/bioinformatics/btp528.

[14] T.C.G.A.R. Network, Integrated genomic analyses of ovarian carcinoma, Nature. 474 (2011) 609–615. https://doi.org/10.1038/nature10166.

[15] M. Kircher, D.M. Witten, P. Jain, B.J. O'Roak, G.M. Cooper, J. Shendure, A general framework for estimating the relative pathogenicity of human genetic variants, Nat. Genet. 46 (2014) 310–315. https://doi.org/10.1038/ng.2892.

[16] H.A. Shihab, J. Gough, M. Mort, D.N. Cooper, I.N. Day, T.R. Gaunt, Ranking non-synonymous single nucleotide polymorphisms based on disease concepts, Hum. Genomics. 8 (2014) 11. https://doi.org/10.1186/1479-7364-8-11.

[17] Y. Choi, G.E. Sims, S. Murphy, J.R. Miller, A.P. Chan, Predicting the Functional Effect of Amino Acid Substitutions and Indels, PLoS ONE. 7 (2012) e46688. https://doi.org/10.1371/journal.pone.0046688.

[18] A. Benedix, C.M. Becker, B.L. de Groot, A. Caflisch, R.A. Böckmann, Predicting free energy changes using structural ensembles, Nat. Methods. 6 (2009) 3–4. https://doi.org/10.1038/nmeth0109-3.

[19] D.E.V. Pires, D.B. Ascher, T.L. Blundell, mCSM: predicting the effects of mutations in proteins using graph-based signatures, Bioinformatics. 30 (2014) 335–342. https://doi.org/10.1093/bioinformatics/btt691.

[20] J. Laimer, H. Hofer, M. Fritz, S. Wegenkittl, P. Lackner, MAESTRO - multi agent stability prediction upon point mutations, BMC Bioinformatics. 16 (2015) 116. https://doi.org/10.1186/s12859-015-0548-6.

[21] M. Petukh, M. Li, E. Alexov, Predicting Binding Free Energy Change Caused by Point Mutations with Knowledge-Modified MM/PBSA Method, PLOS Comput. Biol. 11 (2015) e1004276. https://doi.org/10.1371/journal.pcbi.1004276.

[22] Y. Dehouck, J. Kwasigroch, D. Gilis, M. Rooman, PoPMuSiC 2.1: a web server for the estimation of protein stability changes upon mutation and sequence optimality, BMC Bioinformatics. 12 (2011) 151. https://doi.org/10.1186/1471-2105-12-151.

[23] M. Li, F.L. Simonetti, A. Goncearenco, A.R. Panchenko, MutaBind estimates and interprets the effects of sequence variants on protein–protein interactions, Nucleic Acids Res. 44 (2016) W494–W501. https://doi.org/10.1093/nar/gkw374.

[24] C. Savojardo, M. Petrosino, G. Babbi, S. Bovo, C. Corbi-Verge, R. Casadio, P. Fariselli, L. Folkman, A. Garg, M. Karimi, P. Katsonis, P.M. Kim, O. Lichtarge, P.L. Martelli, A. Pasquo, D. Pal, Y. Shen, A.V. Strokach, P. Turina, Y. Zhou, G. Andreoletti, S. Brenner, R. Chiaraluce, V. Consalvi, E. Capriotti, Evaluating the predictions of the protein stability change upon single amino acid substitutions for the FXN CAGI5 challenge, Hum. Mutat. 0 (n.d.). https://doi.org/10.1002/humu.23843.

[25] A. Strokach, C. Corbi-Verge, P.M. Kim, Predicting changes in protein stability caused by mutation using sequence-and structure-based methods in a CAGI5 blind challenge, Hum. Mutat. 40 (2019) 1414–1423. https://doi.org/10.1002/humu.23852.

[26] P. Huang, S.K.S. Chu, H.N. Frizzo, M.P. Connolly, R.W. Caster, J.B. Siegel, Evaluating Protein Engineering Thermostability Prediction Tools Using an Independently Generated Dataset, ACS Omega. 5 (2020) 6487–6493. https://doi.org/10.1021/acsomega.9b04105.

[27] A. Strokach, D. Becerra, C. Corbi-Verge, A. Perez-Riba, P.M. Kim, Fast and Flexible Protein Design Using Deep Graph Neural Networks, Cell Syst. (2020). https://doi.org/10.1016/j.cels.2020.08.016.

[28] J.H. Friedman, Stochastic gradient boosting, Comput. Stat. Data Anal. 38 (2002) 367–378. https://doi.org/10.1016/S0167-9473(01)00065-2.

[29] M.D.S. Kumar, K.A. Bava, M.M. Gromiha, P. Prabakaran, K. Kitajima, H. Uedaira, A. Sarai, ProTherm and ProNIT: thermodynamic databases for proteins and protein–nucleic acid interactions, Nucleic Acids Res. 34 (2006) D204–D206. https://doi.org/10.1093/nar/gkj103.

[30] E.H. Kellogg, A. Leaver-Fay, D. Baker, Role of conformational sampling in computing mutation-induced changes in protein structure and stability, Proteins. 79 (2011) 830–838. https://doi.org/10.1002/prot.22921.

[31] I.H. Moal, J. Fernández-Recio, SKEMPI: a Structural Kinetic and Energetic database of Mutant Protein Interactions and its use in empirical models, Bioinformatics. 28 (2012) 2600–2607. https://doi.org/10.1093/bioinformatics/bts489.

[32] T. Kortemme, D. Baker, A simple physical model for binding energy hot spots in protein–protein complexes, Proc. Natl. Acad. Sci. U. S. A. 99 (2002) 14116–14121. https://doi.org/10.1073/pnas.202485799.

[33] J. Jankauskaite, B. Jimenez-Garcia, J. Dapkunas, J. Fernandez-Recio, I.H. Moal, SKEMPI 2.0: An updated benchmark of changes in protein-protein binding energy, kinetics and thermodynamics upon mutation, BioRxiv. (2018) 341735. https://doi.org/10.1101/341735.

[34] G.J. Rocklin, T.M. Chidyausiku, I. Goreshnik, A. Ford, S. Houliston, A. Lemak, L. Carter, R. Ravichandran, V.K. Mulligan, A. Chevalier, C.H. Arrowsmith, D. Baker, Global analysis of protein folding using massively parallel design, synthesis, and testing, Science. 357 (2017) 168–175. https://doi.org/10.1126/science.aan0693.

[35] A. Dunham, P. Beltrao, Exploring amino acid functions in a deep mutational landscape, BioRxiv. (2020) 2020.05.26.116756. https://doi.org/10.1101/2020.05.26.116756.

[36] E. Ahler, A.C. Register, S. Chakraborty, L. Fang, E.M. Dieter, K.A. Sitko, R.S.R. Vidadala, B.M. Trevillian, M. Golkowski, H. Gelman, J.J. Stephany, A.F. Rubin, E.A. Merritt, D.M. Fowler, D.J. Maly, A Combined Approach Reveals a Regulatory Mechanism Coupling Src's Kinase Activity, Localization, and Phosphotransferase-Independent Functions, Mol. Cell. 74 (2019) 393-408.e20. https://doi.org/10.1016/j.molcel.2019.02.003.

[37] C.L. Araya, D.M. Fowler, W. Chen, I. Muniez, J.W. Kelly, S. Fields, A fundamental protein property, thermodynamic stability, revealed solely from large-scale measurements of protein function, Proc. Natl. Acad. Sci. 109 (2012) 16858–16863. https://doi.org/10.1073/pnas.1209751109.

[38] E.M. Jones, N.B. Lubock, A.J. Venkatakrishnan, J. Wang, A.M. Tseng, J.M. Paggi, N.R. Latorraca, D. Cancilla, M. Satyadi, J.E. Davis, M.M. Babu, R.O. Dror, S. Kosuri, Structural and functional characterization of G protein-coupled receptors with deep mutational scanning, ELife. 9 (2020) e54895. https://doi.org/10.7554/eLife.54895.

[39] K.A. Matreyek, L.M. Starita, J.J. Stephany, B. Martin, M.A. Chiasson, V.E. Gray, M. Kircher, A. Khechaduri, J.N. Dines, R.J. Hause, S. Bhatia, W.E. Evans, M.V. Relling, W. Yang, J. Shendure, D.M. Fowler, Multiplex assessment of protein variant abundance by massively parallel sequencing, Nat. Genet. 50 (2018) 874–882. https://doi.org/10.1038/s41588-018-0122-z.

[40] C.A. Olson, N.C. Wu, R. Sun, A Comprehensive Biophysical Description of Pairwise Epistasis throughout an Entire Protein Domain, Curr. Biol. 24 (2014) 2643–2651. https://doi.org/10.1016/j.cub.2014.09.072.

[41] B.P. Roscoe, D.N.A. Bolon, Systematic Exploration of Ubiquitin Sequence, E1 Activation Efficiency, and Experimental Fitness in Yeast, J. Mol. Biol. 426 (2014) 2854–2870. https://doi.org/10.1016/j.jmb.2014.05.019.

[42] B.P. Roscoe, K.M. Thayer, K.B. Zeldovich, D. Fushman, D.N.A. Bolon, Analyses of the Effects of All Ubiquitin Point Mutants on Yeast Growth Rate, J. Mol. Biol. 425 (2013) 1363–1377. https://doi.org/10.1016/j.jmb.2013.01.032.

[43] L.M. Starita, J.N. Pruneda, R.S. Lo, D.M. Fowler, H.J. Kim, J.B. Hiatt, J. Shendure, P.S. Brzovic, S. Fields, R.E. Klevit, Activity-enhancing mutations in an E3 ubiquitin ligase identified by high-throughput mutagenesis, Proc. Natl. Acad. Sci. 110 (2013) E1263–E1272. https://doi.org/10.1073/pnas.1303309110.

[44] B. Steinberg, M. Ostermeier, Shifting Fitness and Epistatic Landscapes Reflect Trade-offs along an Evolutionary Pathway, J. Mol. Biol. 428 (2016) 2730–2743. https://doi.org/10.1016/j.jmb.2016.04.033.

[45] G.M. Findlay, E.A. Boyle, R.J. Hause, J.C. Klein, J. Shendure, Saturation editing of genomic regions by multiplex homology-directed repair, Nature. 513 (2014) 120–123. https://doi.org/10.1038/nature13695.

[46] J. Weile, S. Sun, A.G. Cote, J. Knapp, M. Verby, J.C. Mellor, Y. Wu, C. Pons, C. Wong, N. van Lieshout, F. Yang, M. Tasan, G. Tan, S. Yang, D.M. Fowler, R. Nussbaum, J.D. Bloom, M. Vidal, D.E. Hill, P. Aloy, F.P. Roth, A framework for exhaustively mapping functional missense variants, Mol. Syst. Biol. 13 (2017) 957. https://doi.org/10.15252/msb.20177908.

[47] E.C. Hartman, C.M. Jakobson, A.H. Favor, M.J. Lobba, E. Álvarez-Benedicto, M.B. Francis, D. Tullman-Ercek, Quantitative characterization of all single amino acid variants of a viral capsid-based drug delivery vehicle, Nat. Commun. 9 (2018) 1385. https://doi.org/10.1038/s41467-018-03783-y.

[48] J.D. Heredia, J. Park, R.J. Brubaker, S.K. Szymanski, K.S. Gill, E. Procko, Mapping Interaction Sites on Human Chemokine Receptors by Deep Mutational Scanning, J. Immunol. 200 (2018) 3825–3839. https://doi.org/10.4049/jimmunol.1800343.

[49] A. Melnikov, P. Rogov, L. Wang, A. Gnirke, T.S. Mikkelsen, Comprehensive mutational scanning of a kinase in vivo reveals substrate-dependent fitness landscapes, Nucleic Acids Res. 42 (2014) e112–e112. https://doi.org/10.1093/nar/gku511.

[50] L.M. Starita, D.L. Young, M. Islam, J.O. Kitzman, J. Gullingsrud, R.J. Hause, D.M. Fowler, J.D. Parvin, J. Shendure, S. Fields, Massively Parallel Functional Analysis of BRCA1 RING Domain Variants, Genetics. 200 (2015) 413–422. https://doi.org/10.1534/genetics.115.175802.

[51] S. Sun, J. Weile, M. Verby, Y. Wu, Y. Wang, A.G. Cote, I. Fotiadou, J. Kitaygorodsky, M. Vidal, J. Rine, P. Ješina, V. Kožich, F.P. Roth, A proactive genotype-to-patient-phenotype map for cystathionine beta-synthase, Genome Med. 12 (2020) 13. https://doi.org/10.1186/s13073-020-0711-1.

[52] T.U. Consortium, UniProt: a hub for protein information, Nucleic Acids Res. 43 (2015) D204–D212. https://doi.org/10.1093/nar/gku989.

[53] M.J. Landrum, J.M. Lee, M. Benson, G. Brown, C. Chao, S. Chitipiralla, B. Gu, J. Hart, D. Hoffman, J. Hoover, W. Jang, K. Katz, M. Ovetsky, G. Riley, A. Sethi, R. Tully, R. Villamarin-Salomon, W. Rubinstein, D.R. Maglott, ClinVar: public archive of interpretations of clinically relevant variants, Nucleic Acids Res. 44 (2016) D862–D868. https://doi.org/10.1093/nar/gkv1222.

[54] S.A. Forbes, D. Beare, P. Gunasekaran, K. Leung, N. Bindal, H. Boutselakis, M. Ding, S. Bamford, C. Cole, S. Ward, C.Y. Kok, M. Jia, T. De, J.W. Teague, M.R. Stratton, U. McDermott, P.J. Campbell, COSMIC: exploring the world's knowledge of somatic mutations in human cancer, Nucleic Acids Res. 43 (2015) D805–D811. https://doi.org/10.1093/nar/gku1075.

[55] M. Steinegger, J. Söding, MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets, Nat. Biotechnol. 35 (2017) 1026–1028. https://doi.org/10.1038/nbt.3988.

[56] T.N. Starr, A.J. Greaney, S.K. Hilton, D. Ellis, K.H.D. Crawford, A.S. Dingens, M.J. Navarro, J.E. Bowen, M.A. Tortorici, A.C. Walls, N.P. King, D. Veesler, J.D. Bloom, Deep Mutational Scanning of SARS-CoV-2 Receptor Binding Domain Reveals Constraints on Folding and ACE2 Binding, Cell. 182 (2020) 1295-1310.e20. https://doi.org/10.1016/j.cell.2020.08.012.

[57] J.K. Leman, B.D. Weitzner, S.M. Lewis, J. Adolf-Bryfogle, N. Alam, R.F. Alford, M. Aprahamian, D. Baker, K.A. Barlow, P. Barth, B. Basanta, B.J. Bender, K. Blacklock, J. Bonet, S.E. Boyken, P. Bradley, C. Bystroff, P. Conway, S. Cooper, B.E. Correia, B. Coventry, R. Das, R.M. De Jong, F. DiMaio, L. Dsilva, R. Dunbrack, A.S. Ford, B. Frenz, D.Y. Fu, C. Geniesse, L. Goldschmidt, R. Gowthaman, J.J. Gray, D. Gront, S. Guffy, S. Horowitz, P.-S. Huang, T. Huber, T.M. Jacobs, J.R. Jeliazkov, D.K. Johnson, K. Kappel, J. Karanicolas, H. Khakzad, K.R. Khar, S.D. Khare, F. Khatib, A. Khramushin, I.C. King, R. Kleffner, B. Koepnick, T. Kortemme, G. Kuenze, B. Kuhlman, D. Kuroda, J.W. Labonte, J.K. Lai, G. Lapidoth, A. Leaver-Fay, S. Lindert, T. Linsky, N. London, J.H. Lubin, S. Lyskov, J. Maguire, L. Malmström, E. Marcos, O. Marcu, N.A. Marze, J. Meiler, R. Moretti, V.K. Mulligan, S. Nerli, C. Norn, S. Ó'Conchúir, N. Ollikainen, S. Ovchinnikov, M.S. Pacella, X. Pan, H. Park, R.E. Pavlovicz, M. Pethe, B.G. Pierce, K.B. Pilla, B. Raveh, P.D. Renfrew, S.S.R. Burman, A. Rubenstein, M.F. Sauer, A. Scheck, W. Schief, O. Schueler-Furman, Y. Sedan, A.M. Sevy, N.G. Sgourakis, L. Shi, J.B. Siegel, D.-A. Silva, S. Smith, Y. Song, A. Stein, M. Szegedy, F.D. Teets, S.B. Thyme, R.Y.-R. Wang, A. Watkins, L. Zimmerman, R. Bonneau, Macromolecular modeling and design in Rosetta: recent methods and frameworks, Nat. Methods. 17 (2020) 665–680. https://doi.org/10.1038/s41592-020-0848-2.

[58] A. Strokach, C. Corbi-Verge, J. Teyra, P.M. Kim, Predicting the Effect of Mutations on Protein Folding and Protein-Protein Interactions, in: T. Sikosek (Ed.), Comput. Methods Protein Evol., Springer New York, New York, NY, 2019: pp. 1–17. https://doi.org/10.1007/978-1-4939-8736-8_1.

[59] O. Buß, J. Rudat, K. Ochsenreither, FoldX as Protein Engineering Tool: Better Than Random Based Approaches?, Comput. Struct. Biotechnol. J. 16 (2018) 25–33.

https://doi.org/10.1016/j.csbj.2018.01.002.

[60] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, T.-Y. Liu, LightGBM: A Highly Efficient Gradient Boosting Decision Tree, in: I. Guyon, U.V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), Adv. Neural Inf. Process. Syst. 30, Curran Associates, Inc., 2017: pp. 3146–3154. http://papers.nips.cc/paper/6907-lightgbm-a-highly-efficient-gradient-boosting-decision-tree.pdf (accessed November 4, 2019).

[61] W. McLaren, L. Gil, S.E. Hunt, H.S. Riat, G.R.S. Ritchie, A. Thormann, P. Flicek, F. Cunningham, The Ensembl Variant Effect Predictor, Genome Biol. 17 (2016) 122. https://doi.org/10.1186/s13059-016-0974-4.

[62] G. Andreoletti, L.R. Pal, J. Moult, S.E. Brenner, Reports from the fifth edition of CAGI: The Critical Assessment of Genome Interpretation, Hum. Mutat. 40 (2019) 1197–1201. https://doi.org/10.1002/humu.23876.

[63] N. Sahni, S. Yi, M. Taipale, J.I. Fuxman Bass, J. Coulombe-Huntington, F. Yang, J. Peng, J. Weile, G.I. Karras, Y. Wang, I.A. Kovács, A. Kamburov, I. Krykbaeva, M.H. Lam, G. Tucker, V. Khurana, A. Sharma, Y.-Y. Liu, N. Yachie, Q. Zhong, Y. Shen, A. Palagi, A. San-Miguel, C. Fan, D. Balcha, A. Dricot, D.M. Jordan, J.M. Walsh, A.A. Shah, X. Yang, A.K. Stoyanova, A. Leighton, M.A. Calderwood, Y. Jacob, M.E. Cusick, K. Salehi-Ashtiani, L.J. Whitesell, S. Sunyaev, B. Berger, A.-L. Barabási, B. Charloteaux, D.E. Hill, T. Hao, F.P. Roth, Y. Xia, A.J.M. Walhout, S. Lindquist, M. Vidal, Widespread Macromolecular Interaction Perturbations in Human Genetic Disorders, Cell. 161 (2015) 647–660. https://doi.org/10.1016/j.cell.2015.04.013.

[64] S. Sirin, J.R. Apgar, E.M. Bennett, A.E. Keating, AB-Bind: Antibody binding mutational database for computational affinity predictions, Protein Sci. 25 (2016) 393–409. https://doi.org/10.1002/pro.2829.

[65] C.J.C. Burges, From RankNet to LambdaRank to LambdaMART: An overview, 2010. https://www.microsoft.com/en-us/research/publication/from-ranknet-to-lambdarank-to-lambdamart-an-overview/.

[66] C.J. Burges, R. Ragno, Q.V. Le, Learning to Rank with Nonsmooth Cost Functions, in: B. Schölkopf, J.C. Platt, T. Hoffman (Eds.), Adv. Neural Inf. Process. Syst. 19, MIT Press, 2007: pp. 193–200. http://papers.nips.cc/paper/2971-learning-to-rank-with-nonsmooth-cost-functions.pdf (accessed October 21, 2020).

[67] T. Akiba, S. Sano, T. Yanase, T. Ohta, M. Koyama, Optuna: A Next-generation Hyperparameter Optimization Framework, in: Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min., Association for Computing Machinery, New York, NY, USA, 2019: pp. 2623–2631. https://doi.org/10.1145/3292500.3330701.

Supporting Information

Supporting tables

Table S1. System commands used to run Rosetta's *cartesian_ddg* protocol to predict the effect of mutation on protein stability (left) or protein binding affinity (right).

Calculating protein stability	Calculating protein binding affinity

`cartesian_ddg.static.linuxgccrelease \`
`    -ddg::cartesian \`
`    -ddg::bbnbrs 1 \`
`    -in:file:s '{structure_file}' \`
`    -in::file::fullatom \`
`    -database '{rosetta_db}' \`
`    -ignore_unrecognized_res true \`
`    -ignore_zero_occupancy false \`
`    -fa_max_dis 9.0 \`
`    -ddg::mut_file '{mutation_file}' \`
`    -ddg::iterations 3 \`
`    -ddg::dump_pdbs true \`
`    -ddg::suppress_checkpointing true \`
`    -ddg::mean true \`
`    -ddg::min true \`
`    -ddg::output_silent true \`
`    -beta_nov16_cart \`
`    -score:weights beta_nov16_cart`
	`cartesian_ddg.static.linuxgccrelease \`
	`    -ddg::cartesian \`
	`    -ddg::bbnbrs 1 \`
	`    -in:file:s '{structure_file}' \`
	`    -in::file::fullatom \`
	`    -database '{rosetta_db}' \`
	`    -ignore_unrecognized_res true \`
	`    -ignore_zero_occupancy false \`
	`    -fa_max_dis 9.0 \`
	`    -ddg::mut_file '{mutation_file}' \`
	`    -ddg::iterations 3 \`
	`    -ddg::dump_pdbs true \`
	`    -ddg::suppress_checkpointing true \`
	`    -ddg::mean true \`
	`    -ddg::min true \`
	`    -ddg::output_silent true \`
	`    -beta_nov16_cart \`
	`    -score:weights beta_nov16_cart \`
	`    -interface_ddg 1`

Table S2. Features used by the final EL2core model trained to predict the effect of mutations on protein stability.

Feature name	**Feature description**
proteinsolver_core_score_mut	Probability assigned by ProteinSolver to the mutant residue.
proteinsolver_core_score_change	Difference between probabilities assigned by ProteinSolver to the mutant and wildtype residues.
protbert_core_score_wt	Probability assigned by ProtBert to the wildtype residue.
protbert_core_score_mut	Probability assigned by ProtBert to the mutant residue.
protbert_core_features_protein_wt_0_pc	Principal component #0 of the embedding generated by ProtBert for the wildtype protein.
protbert_core_features_protein_wt_5_pc	Principal component #5 of the embedding generated by ProtBert for the wildtype protein.
protbert_core_features_protein_wt_8_pc	Principal component #8 of the embedding generated by ProtBert for the wildtype protein.
protbert_core_features_protein_change_2_pc	Principal component #2 of the difference between embeddings generated by ProtBert for the wildtype and mutant proteins.
protbert_core_features_protein_change_4_pc	Principal component #4 of the difference between embeddings generated by ProtBert for the wildtype and mutant proteins.
protbert_core_features_protein_change_5_pc	Principal component #5 of the difference between embeddings generated by ProtBert for the wildtype and mutant proteins.
protbert_core_features_protein_change_6_pc	Principal component #6 of the difference between embeddings generated by ProtBert for the wildtype and mutant proteins.
protbert_core_features_protein_change_8_pc	Principal component #8 of the difference between embeddings generated by ProtBert for the wildtype and mutant proteins.
protbert_core_features_residue_wt_0_pc	Principal component #0 of the embedding generated by ProtBert for the wildtype residue.
protbert_core_features_residue_change_0_pc	Principal component #0 of the difference between embeddings generated by ProtBert for the wildtype and mutant residues.
protbert_core_features_residue_change_1_pc	Principal component #1 of the difference between embeddings generated by ProtBert for the wildtype and mutant residues.
protbert_core_features_residue_change_5_pc	Principal component #5 of the difference between embeddings generated by ProtBert for the wildtype and mutant residues.
protbert_core_features_residue_change_9_pc	Principal component #9 of the difference between embeddings generated by ProtBert for the wildtype and mutant residues.

Table S3. Features used by the final EL2interface model trained to predict the effect of mutations on protein binding affinity.

Feature name	**Feature description**
proteinsolver_core_score_mut	Probability assigned by ProteinSolver to the mutant residue of the protein without its interacting partner.
proteinsolver_interface_score_mut	Probability assigned by ProteinSolver to the mutant residue of the protein together with its interacting partner.
proteinsolver_core2interface_score_wt	Difference between probabilities assigned by ProteinSolver to the wildtype residue of the protein with and without its interacting partner.
protbert_core_score_mut	Probability assigned by ProtBert to the mutant residue of the protein without its interacting partner.
protbert_core_score_change	Difference between the probabilities assigned by ProtBert to the mutant and wildtype residues of the protein without its interacting partner.
protbert_core_features_protein_wt_7_pc	Principal component #7 of the wildtype protein embedding generated by ProtBert for the protein without its interacting partner.
protbert_core_features_residue_wt_4_pc	Principal component #4 of the wildtype residue embedding generated by ProtBert for the protein without its interacting partner.
protbert_core_features_residue_change_0_pc	Principal component #0 of the difference between the wildtype and mutant residue embeddings generated by ProtBert for the protein without its interacting partner.
protbert_core_features_residue_change_1_pc	Principal component #1 of the difference between the wildtype and mutant residue embeddings generated by ProtBert for the protein without its interacting partner.
protbert_interface_features_protein_wt_2_pc	Principal component #2 of the wildtype protein embedding generated by ProtBert for the protein together with its interacting partner.
protbert_interface_features_protein_wt_3_pc	Principal component #3 of the wildtype protein embedding generated by ProtBert for the protein together with its interacting partner.
protbert_interface_features_protein_change_1_pc	Principal component #1 of the differences between the wildtype and mutant protein embeddings generated by ProtBert for the protein together with its interacting partner.
protbert_interface_features_protein_change_6_pc	Principal component #6 of the differences between the wildtype and mutant protein embeddings generated by ProtBert for the protein together with its interacting partner.

protbert_interface_features_protein_change_7_pc	Principal component #7 of the differences between the wildtype and mutant protein embeddings generated by ProtBert for the protein together with its interacting partner.
protbert_core2interface_features_residue_wt_7_pc	Principal component #7 of the differences between wildtype residue embeddings generated by ProtBert for the protein with and without its interacting partner.
protbert_core2interface_features_residue_wt_9_pc	Principal component #9 of the differences between wildtype residue embeddings generated by ProtBert for the protein with and without its interacting partner.

Supporting figures

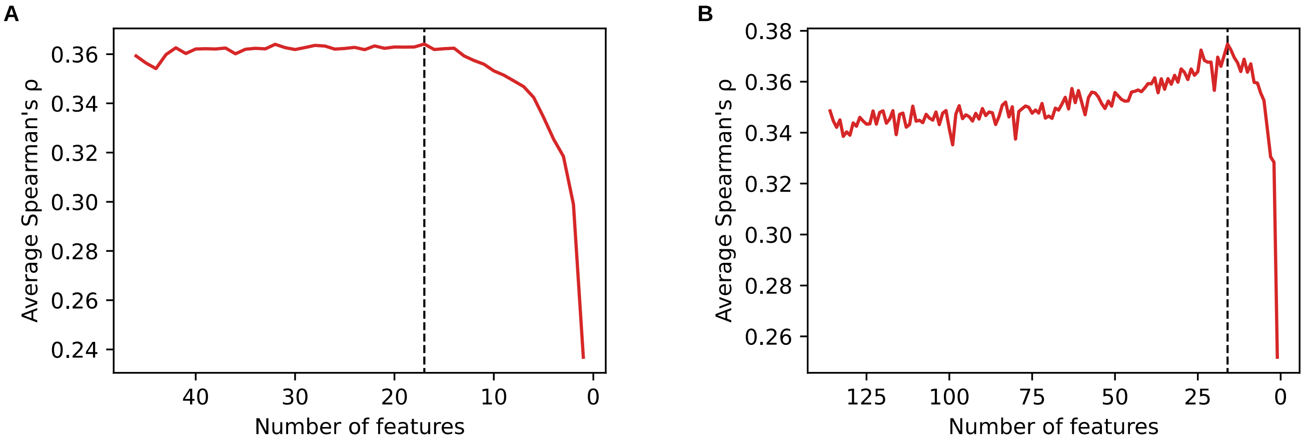

Figure S1. Average Spearman's correlation coefficients between predictions made using EL2core (left) and EL2interface (right) models and experimental measurements, as the number of features available to those models is successively reduced. Predictions were made using 6-fold cross-validation, and the feature elimination strategy is described in the methods section. Dashed vertical lines indicate the number of features that produced the best-performing models.