Supplementary Note

MetaRNN: Differentiating Rare Pathogenic and Rare Benign Missense SNVs and InDels Using Deep Learning

Chang Li, Degui Zhi, Kai Wang, Xiaoming Liu

Method

Training Datasets

ClinVar[1] dataset files clinvar_20190102.vcf.gz and clinvar_20200609.vcf.gz were downloaded from https://www.ncbi.nlm.nih.gov/clinvar/. Mutations in the older file were used in the training phase of model development. Next, we prepared separate datasets for point mutations and insertion/deletions (InDels). For SNVs, non-synonymous SNVs (nsSNVs) labeled as 'Pathogenic' or 'Likely pathogenic' were used as true positives (TPs), and nsSNVs labeled as 'Benign' or 'Likely Benign' were used as true negatives (TNs). Rare variants that are absent from at least one of the three datasets (gnomAD[2], ExAC[3], and the 1000 Genomes Project[4]) were retained. A further filter removed any nsSNVs that are absent in all three datasets. In the end, 26,517 rare nsSNVs with 9,009 TPs and 17,508 TNs **(SupplementaryTable 1**) were used for training. For InDels, the same criteria were applied to obtain true positives and true negatives. Additionally, only InDels annotated as non-frameshift (nfINDELs) and have length >1 and < 50 base-pairs were included. A total of 2,057 rare nfINDELs with 1,348 TPs and 709 TNs passed the filtering criteria **(SupplementaryTable 2**).

Test Datasets

We constructed 6 test sets to evaluate the performance of our SNV based model, namely MetaRNN (Summary in **SupplementaryTable 3**). The first test dataset (rare nsSNV test set, RNTS) was constructed from rare pathogenic nsSNVs (absent from at least one of the three datasets, namely gnomAD, ExAC and the 1000 Genomes Project, but not absent in all three datasets) added to the ClinVar database after 20190102 and rare nsSNVs absent from at least one of the three datasets but not absent in all three datasets while not reported in ClinVar and matching on genomic location (randomly selected non-pathogenic nsSNVs within 200kb from the pathogenic ones), resulting in 12,406 variants with 6,203 TPs and 6,203 TNs **(SupplementaryTable 4**). The second test dataset (rare clinvar-only test set, RCTS) was constructed from recently-added (after 20190102) ClinVar pathogenic nsSNVs (n=1,528) and benign nsSNVs (n=2,389) that were absent from at least one of the three datasets but not absent in all three datasets **(SupplementaryTable 5**). The third test dataset (allele-frequency-filtered RNTS, AF-RNTS) was constructed from RNTS replacing the allele count filter with an AF filter that only variants with AF⩽0.01 in all three datasets were kept which resulted in a data set with 5,770 TPs and 5,770 TNs (**SupplementaryTable** 6). The fourth test dataset (allele-frequency-filtered RCTS, AF-RCTS) was constructed from RCTS replacing the allele count filter with an AF filter that only variants with AF⩽0.001 in all populations were kept which resulted in 6065 TPs and 3,220 TNs (**SupplementaryTable** 7). The fifth test dataset (all-allele-frequency set, AAFS) was constructed from all pathogenic and benign nsSNVs added to the ClinVar database after 20190102, resulting in 29,924 variants with 6,205 TPs and 22,808 TNs

Supplementary Note

(**SupplementaryTable 8**). The sixth test set, TP53 test set (TP53TS), was constructed from the TP53 mutation website (https://p53.fr/index.php).). Variants with median activity <50 were considered pathogenic, while variants with median activity >=100 are considered benign. After removing variants used in the training set, 824 variants remained with 385 TPs and 439 TNs (**SupplementaryTable 9**). For our InDel-based model, namely MetaRNN-indel, the test dataset was constructed from InDels added to the ClinVar database after 20190102, which resulted in 989 InDels with 491 TPs and 498 TNs (**SupplementaryTable 10**).

Flanking region mutation

After obtaining all target variants, we retrieved their flanking sequences using dbNSFP4.1a. Specifically, the variant's genomic location and the affected amino acid position as to the protein and the affected codon were first identified in dbNSFP. Then, a window size of +/ – 1 codon around the affected codon was identified, and all nsSNVs inside this window were retrieved with maximum length of 9 base-pairs (bp). For a given target mutation, the maximum number of nucleotides on either side is 5 bp (3 bp from one flanking codon and 2 bp from the target codon). To center the input window on target mutation and have uniform shape for all inputs, the input window is padded for an additional 2 bp to reach an 11 bp window for each target mutation so that there are 5 bp around the target mutation. This window, including the target variants, was used as one input for our model (**S.Figure 1**). For each position, multiple alternative alleles may exist. Thus, to reduce the dimension of our input data, annotations were averaged across all alleles at the same locus, except for the target mutation where the actual mutation is used. After these steps, the input dimension for the MetaRNN model becomes 11 (bp) by 28 (features, see below).

The same rule to adopting flanking region applies to InDels with one difference: instead of affecting only one codon, target InDels may affect multiple codons simultaneously. Thus, the +/– 1 codon window was defined as the window beyond all the directly affected codons. Since we focus on InDels with length < 50, the input dimension for the MetaRNN-indel model is 58 (bp) by 28 (features, see below).

Feature selection

For each variant, including target variant and flanking sequence variants, 28 features were calculated or retrieved from the dbNSFP database, including 16 functional prediction scores including SIFT[5], Polyphen2_HDIV[6], Polyphen2_HVAR, MutationAssessor[7], PROVEAN[8], VEST4[9], M-CAP[10], REVEL[11], MutPred[12], MVP[13], PrimateAI[14], DEOGEN2[15], CADD[16], fathmm-XF[17], Eigen[18] and GenoCanyon[19], eight conservation scores including GERP[20], phyloP100way_vertebrate[21], phyloP30way_mammalian, phyloP17way_primate, phastCons100way_vertebrate, phastCons30way_mammalian, phastCons17way_primate and SiPhy[22], and four calculated allele frequency (AF) related scores. The highest AF values among subpopulations were used as the AF score for each of the four data sets from three studies, namely the 1000 Genomes Project (1000GP), ExAC, gnomAD exome, and gnomAD genome. All missing scores in the dbNSFP database were first imputed using BPCAfill[23], and all scores were standardized before feeding to the model for training. Some more recently developed scores were removed to minimize type I circularity in training our ensemble model, such as ClinPred, which used ClinVar variants in their training process.

Supplementary Note

Model development

We applied a recurrent neural network with Gated Recurrent Units[24](GRU) to extract and learn the sequence information around target variants (**S.Figure 2**). To select the best performing model structure, Bayesian Hyperparameter Optimization[25] was used, and a wide range of model structures was tested. This process used 70% training data for model training and 30% of training data for performance evaluation, so no test datasets were used in this step. Python packages sci-kit-learn[26] and TensorFlow 2.0 (https://www.tensorflow.org/) were used to develop the models, and KerasTuner (https://keras-team.github.io/keras-tuner/) was adopted to apply Bayesian Hyperparameter Optimization. The search space for all the hyperparameters was shown in **SupplementaryTable 8.** The models with the smallest validation log loss were used as our final models for nsSNV (MetaRNN) and nfINDEL (MetaRNN-indel).

Compare the Performance of Individual Predictors

As a model diagnosis step, for each model, a permutation feature importance was calculated for each of the individual features. The permutation importance (PI) was calculated as:

$$PI = \frac{Loss^{Perm}}{Loss^{orig}}$$

A PI value of 1 means the feature is useless, with higher values indicating more important features.

To quantitatively evaluate model performance, we retrieved 28 annotations from dbNSFP to compare with our MetaRNN model. For the MetaRNN-indel model, four popular methods were compared, including DDIG-in[27], CADD[16], PROVEAN[8], and VEST4[28]. We plotted receiver operating characteristic (ROC) curves and calculated the area under the ROC curve (AUC) for each method being compared using our test datasets. Python package matplotlib (https://matplotlib.org/) was used to plot ROC curves, and Python package sci-kit-learn was used to calculate AUC log loss.

RESULTS

Comparison of other model structures

To show that our MetaRNN model, which adopted flanking sequence information, can provide additional predictive power, we trained another feed-forward neural network model using only dense layers (MetaRNN.feedforwad). The same KerasTuner parameters were used, such as the maximum number of trials and the number of hidden layers. Additionally, we checked if limiting training data to only those rare nsSNVs (MAF⩽0.01 in all populations; n=18,070) could improve the model's performance in separating rare pathogenic from rare benign nsSNVs (MetaRNN.rare). The comparison showed that our MetaRNN model outperforms all these model structures (**S. Figure 3**). Models which adopted flanking information (MetaRNN and MetaRNN.rare) outperformed the model structure that used only target site information (MetaRNN.feedforward). Additionally, limiting training data to only rare variants did not help with model performance. This observation could have two implications. First, our initial training data include more data points (26,517 vs. 18,070). These additional data points can provide more information that may be missed in the new training set. Second, limiting training data to have

Supplementary Note

MAF⩽0.01 may lead to a loss of useful information otherwise provided by AF features. Our initial training data, which removed only those "easy" benign nsSNVs observed in all populations, seem to be a good trade-off between posing a difficult enough training set for the model to learn useful information and preserving valuable information from AF features.

Characterization of MetaRNN Features

MetaRNN ensemble score combined 24 individual prediction scores and four allele frequency features from the 1000 Genomes Project, ExAC, and gnomAD. As shown in **S. Figure 4A**, most of the conservation scores and ensemble scores showed moderate to strong correlations (correlation coefficient between 0.4 and 1). However, MutationTaster[29] and GenoCanyon[19] showed a weak correlation with all other features. Since most SNVs are not observed in multiple populations, correlations between different AF features are also strong (>0.8). However, AF features showed a weak correlation with all other individual predictors implying that previous annotation scores have not fully exploited such information. This observation is also supported by the permutation feature importance analysis (**S. Figure 4B**). The most important features are two exome AF features, followed by two whole-genome AF features. Some functional predictors showed higher importance compared with others, such as VEST4, MutationTaster, and MutPred. This observation is in concordance with a recent observation[30] highlighting the importance of population AF data in inferring the functional importance of nsSNVs.

Ensemble score outperforms competitors regardless of test set allele frequency

We composed two main test sets to compare the performance of MetaRNN with 24 other methods including MutationTaster[29], FATHMM[31], FATHMM-XF[17], VEST4[9], MetaSVM[32], MetaLR[32], M-CAP[10], REVEL[11], MutPred[12], MVP[13], PrimateAI[14], DEOGEN2[15], BayesDel_addAF[33], ClinPred[30], LIST-S2[34], CADD[16], Eigen[18], GERP[35], phyloP100way_vertebrate[21], phyloP30way_mammalian, phyloP17way_primate, phastCons100way_vertebrate, phastCons30way_mammalian, and phastCons17way_primate : 1) Rare nsSNV test set (RNTS) is comprised of rare ClinVar pathogenic SNVs and rare control SNVs from gnomAD matching on genomic location; 2) All-allele-frequency set (AAFS) is comprised of all available ClinVar pathogenic SNVs and benign SNVs (rare+common) that are not used for model development. As shown in **S. Figure 7**, using AAFS as benchmarking dataset, MetaRNN outperforms all competitors with an AUC equals to 0.9862. The second-best model is ClinPred (AUC=0.9841) and followed by BayesDel (AUC=0.9759). In general, ensemble methods and functional predictors outperform conservation-based methods in this comparison which agrees with previous findings.

Performance Evaluation in an Independent Test Set

Next, we compared MetaRNN's ability to identify the most functionally important nsSNVs in the TP53 gene. As shown in **S. Figure 8**, MetaRNN topped the list with an AUC equals to 0.8074. The PrimateAI, BayesDel_addAF, REVEL, and VEST4 showed slightly inferior performances. Methods based purely on conservation again had the worst performance. This result indicates that MetaRNN can effectively predict the pathogenicity of nsSNVs and the functional impacts, while other top methods are only good at some test datasets.

MetaRNN-indel significantly outperformed all competitors on nfINDELs

Supplementary Note

Using the hold-out set from the ClinVar database as benchmarking dataset, we compared MetaRNN-indel with four other currently available scores. As shown in **Figure 4**, MetaRNN-indel performed the best with an AUC equals 0.9433, followed by PROVEAN (AUC=0.9271) and DDIG-in (AUC=0.9152). These results indicate that our training framework for MetaRNN and MetaRNN-indel consistently outperforms other methods concerning both nsSNVs and nfINDELs.

Supplementary Note

SupplementaryTable 3. Summary statistics for test datasets used to evaluate MetaRNN

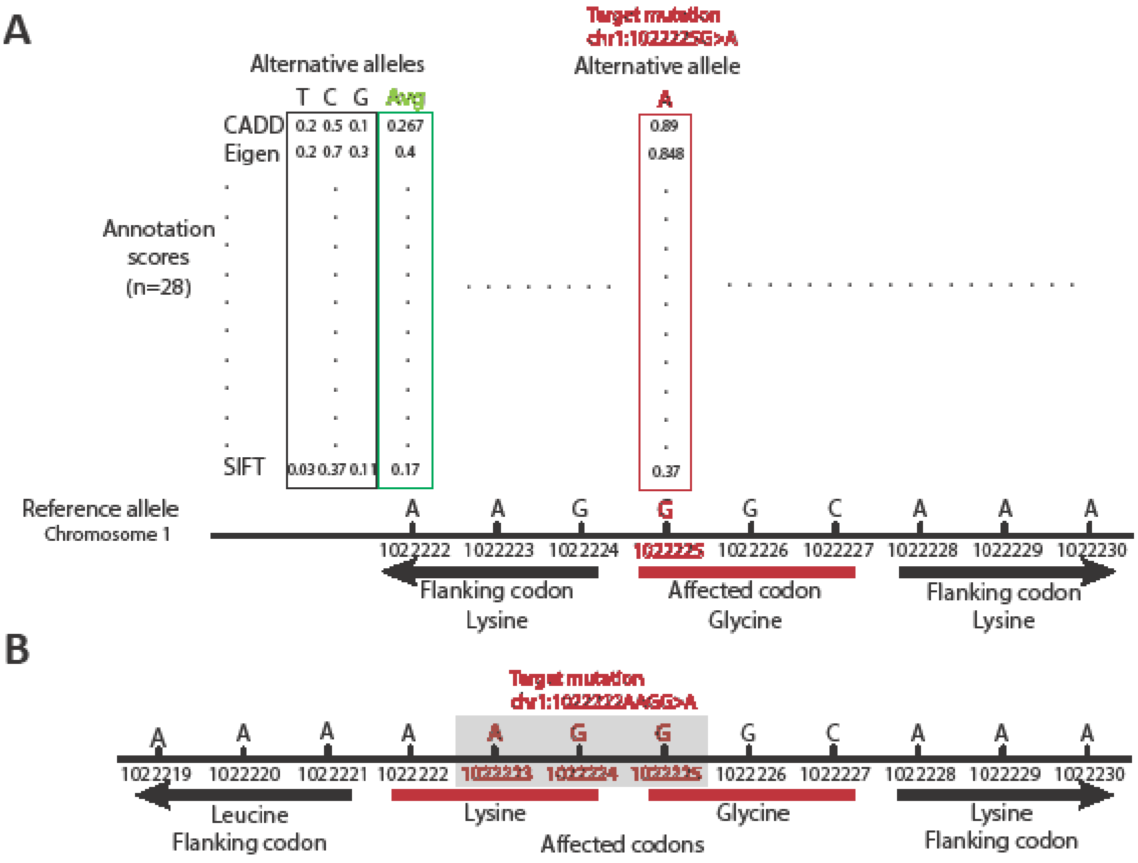
Name	Allele frequency filter	Allele count filter	No. TP	No. TN	Total No.
RNTS	NA	Observed in at least 1 dataset, removed if observed in all 3	6203	6203	12406
RCTS	NA	Observed in at least 1 dataset, removed if observed in all 3	1528	2389	3917
AF-RNTS	MAF<=0.01 in all 3 datasets	NA	5770	5770	11540
AF-RCTS	MAF<=0.001 in all 3 datasets	NA	6065	3220	9285
AAFS	NA	NA	6295	22808	29103
TP53TS	NA	NA	385	439	824

Supplementary Note

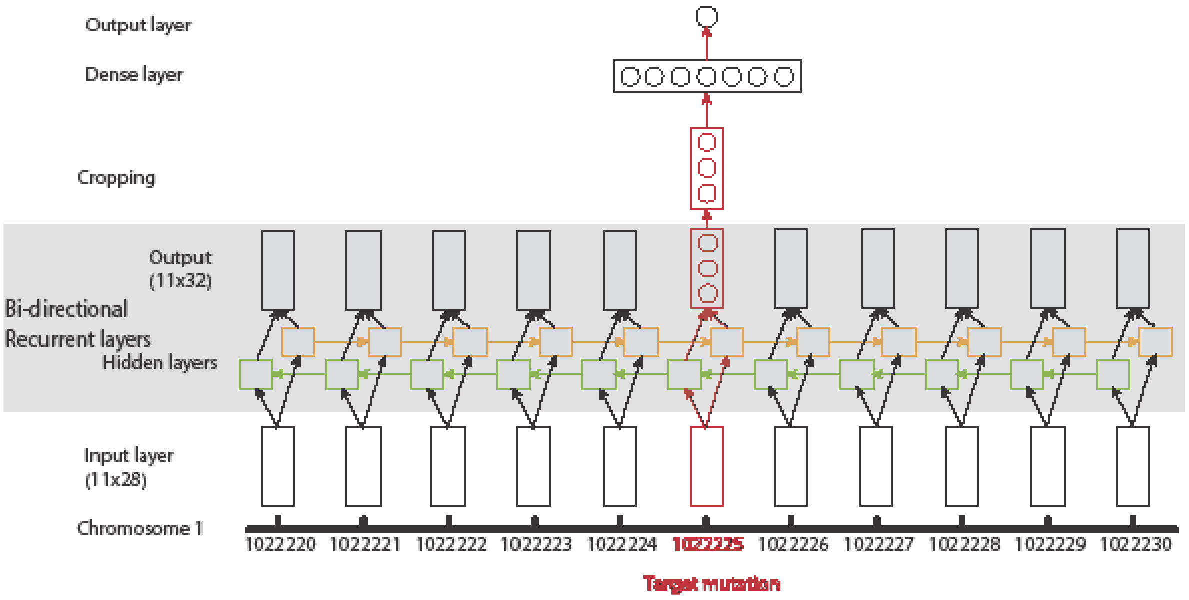
SupplementaryTable 8. Search space for all hyperparameters using Bayesian Optimization.

Hyperparameters	Minimum	Maximum	Step
GRU layers	1	3	1
GRU units	16	128	16
Dropout rate	0.1	0.5	0.1
Dense layers	1	4	1
Dense units	16	128	16
Learning rate	$1x10^{-5}$	$1x10^{-2}$	Log sampling

Supplementary Note

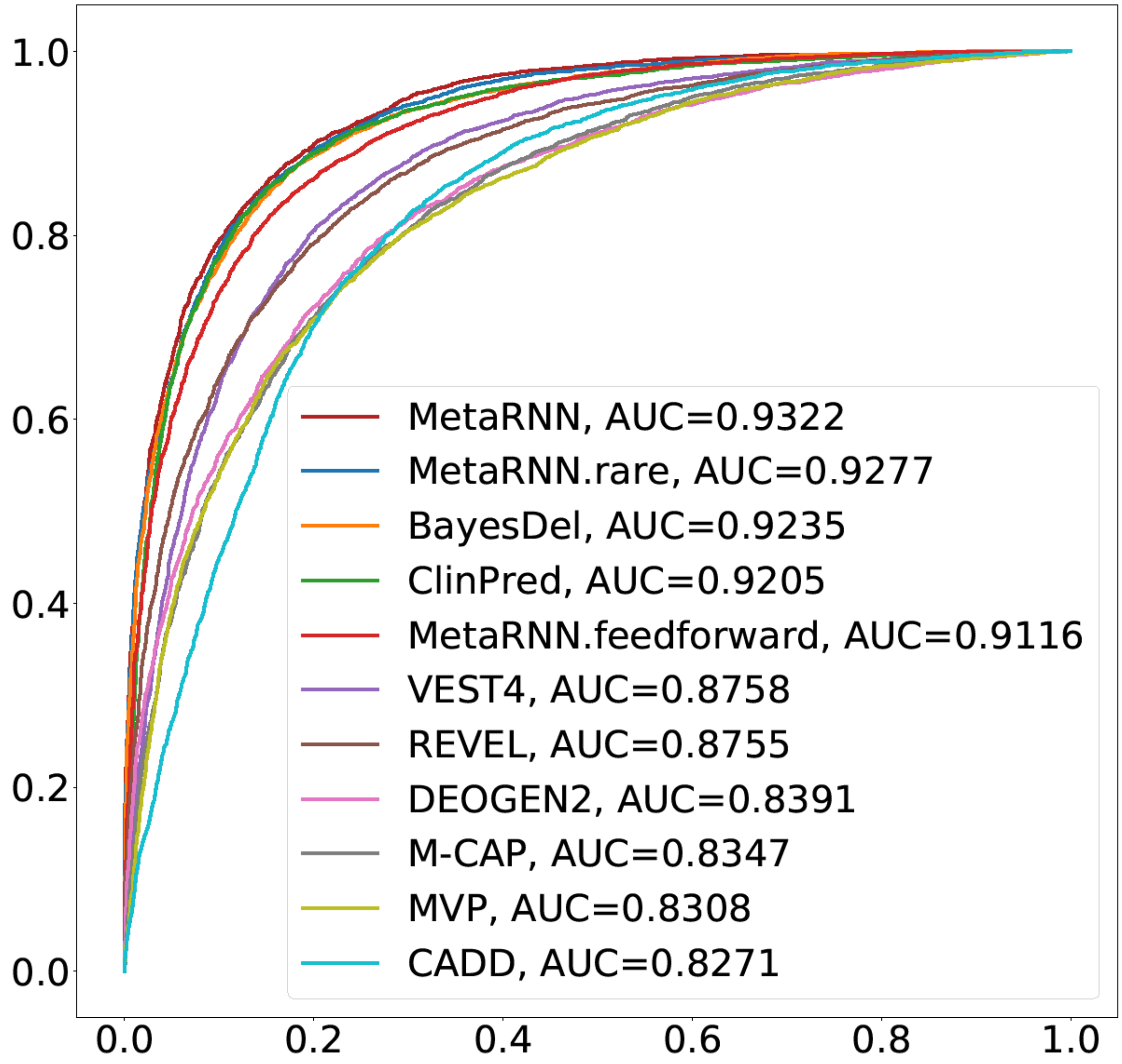


S. Figure 1. Steps to prepare data for model input.



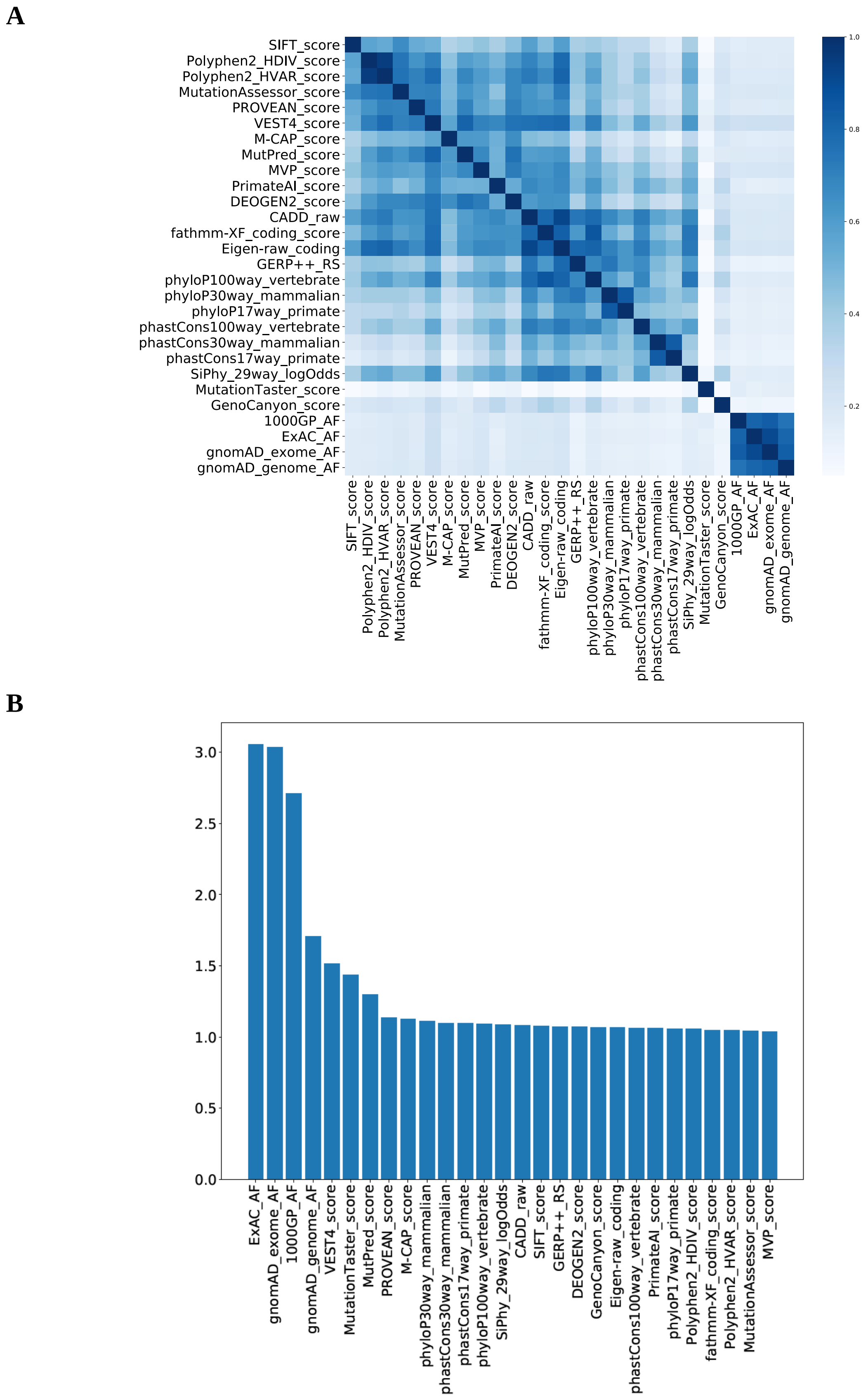
S. Figure 2. Illustration of MetaRNN models.

Supplementary Note



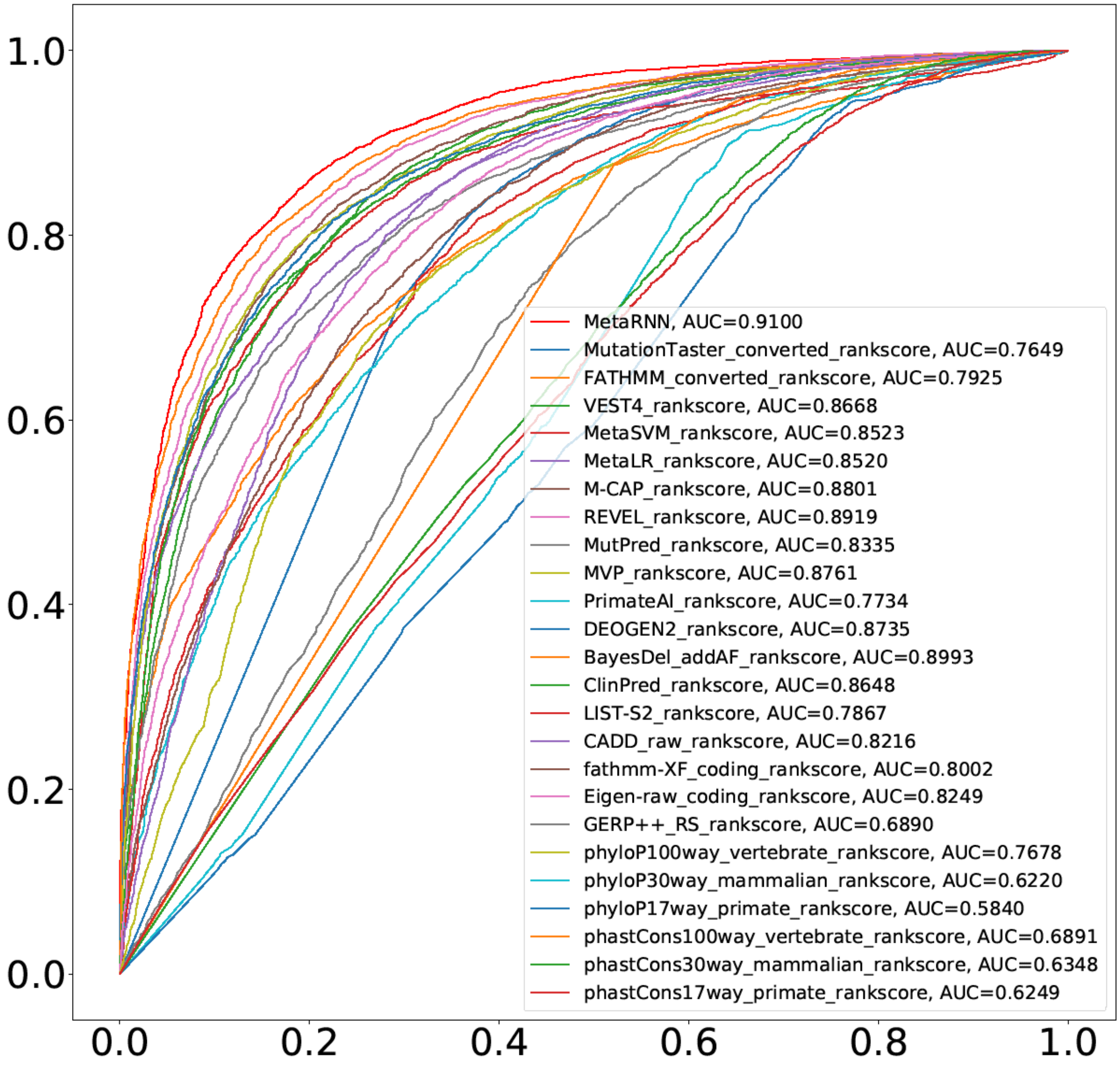
S. Figure 3. Performance of different model structures and selected predictors benchmarked using the allele-frequency-filtered rare nsSNV test set (AF-RNTS).

Supplementary Note



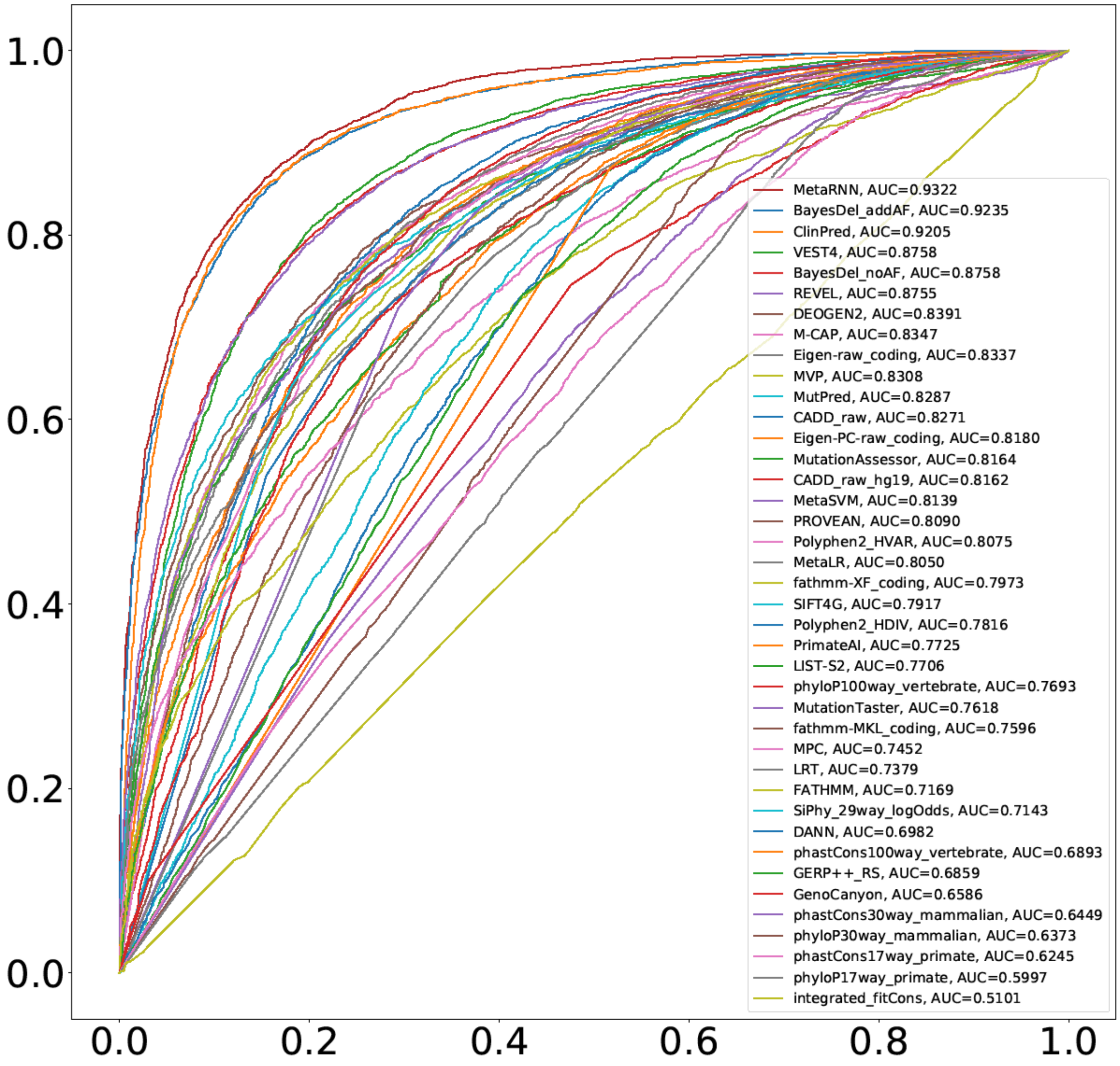
S. Figure 4. A. Correlation between features used to train MetaRNN. B. Feature importance for all features used in the MetaRNN model.

Supplementary Note



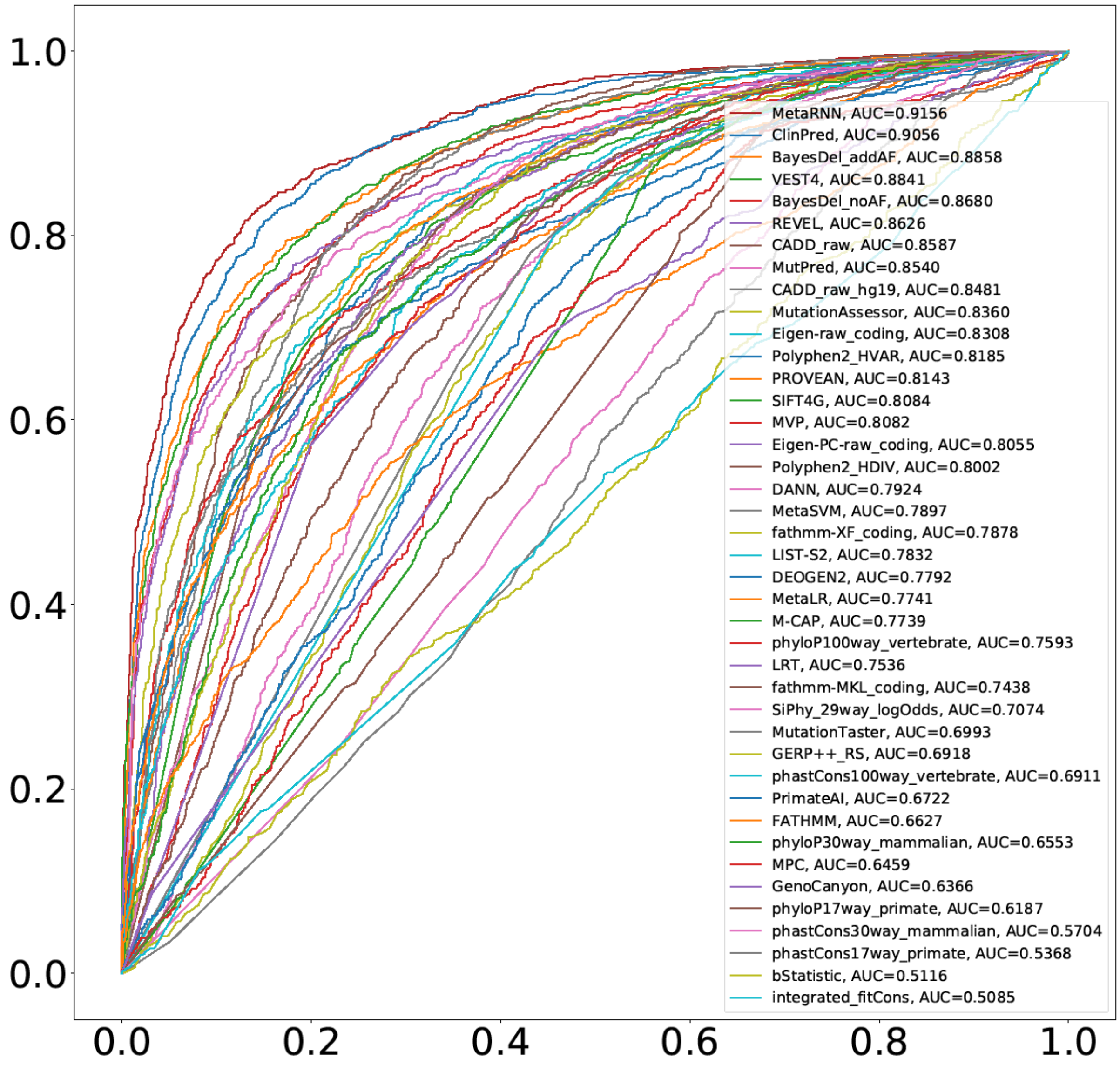
S. Figure 5. Performance of different methods benchmarked using the rare nsSNV test set (RNTS).

Supplementary Note



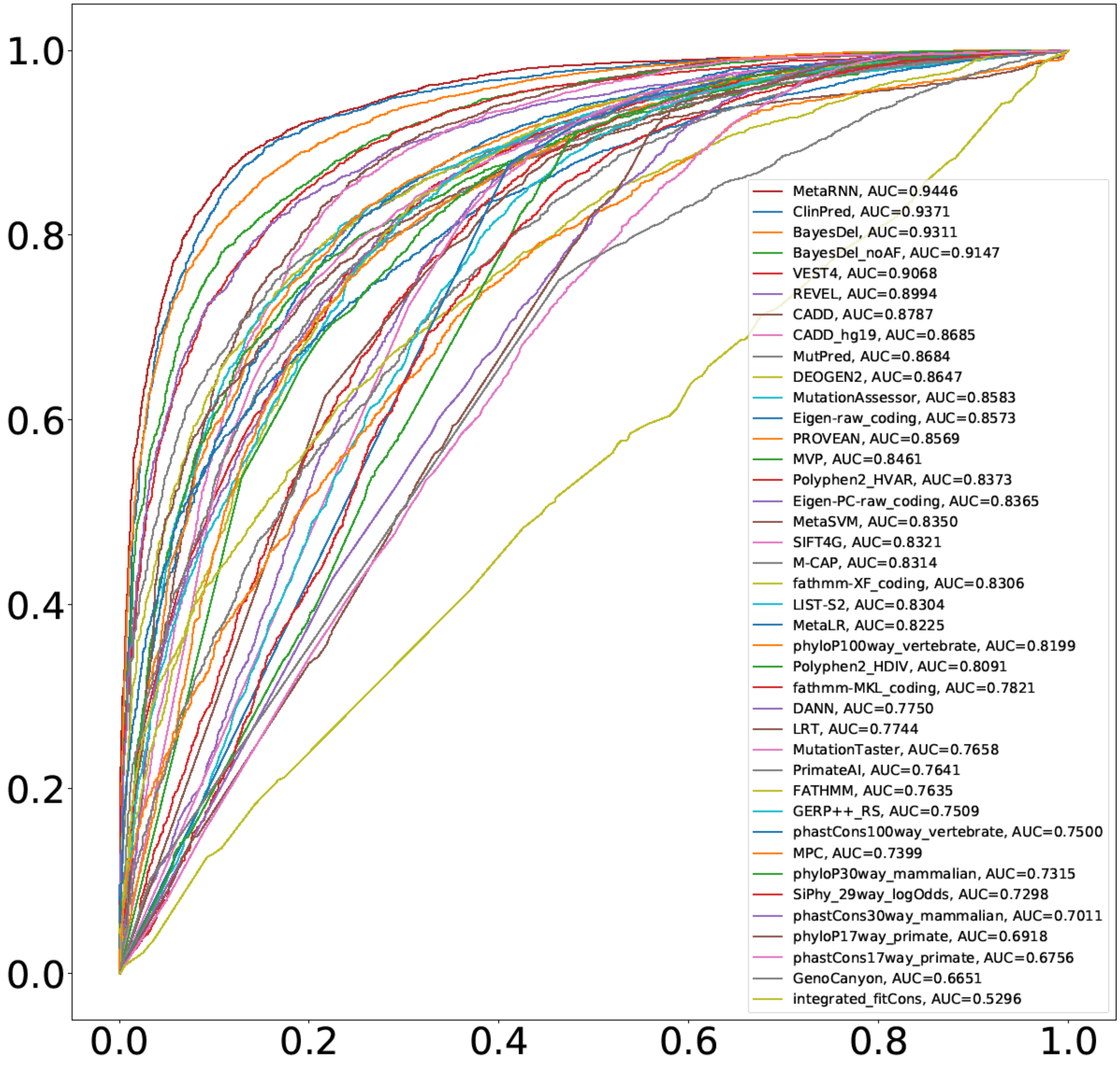
S. Figure 6. Performance of different methods benchmarked using the allele-frequency-filtered rare nsSNV test set (AF-RNTS).

Supplementary Note



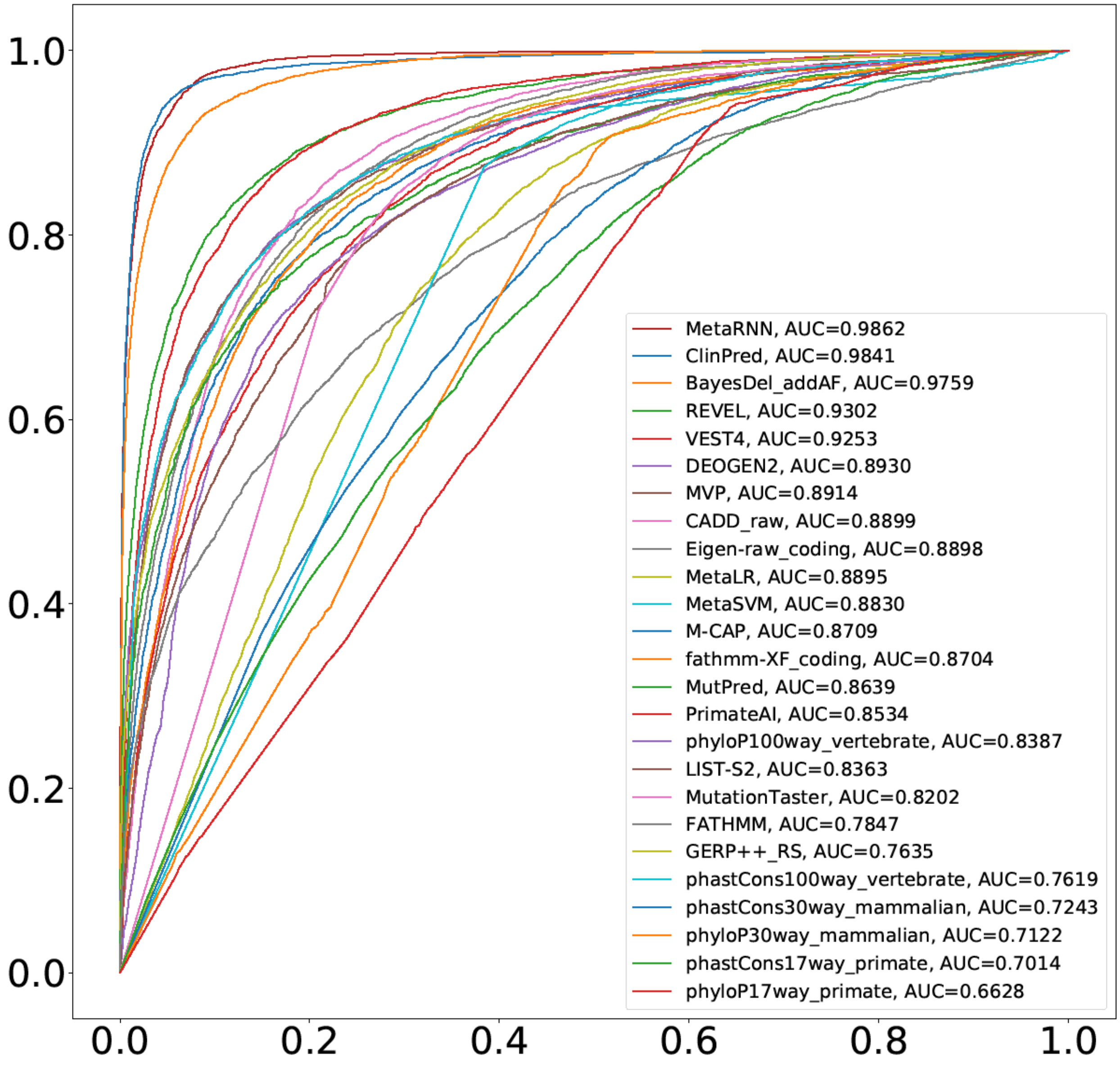
S. Figure 7. Performance of different methods benchmarked using rare ClinVar-only test set (RCTS).

Supplementary Note



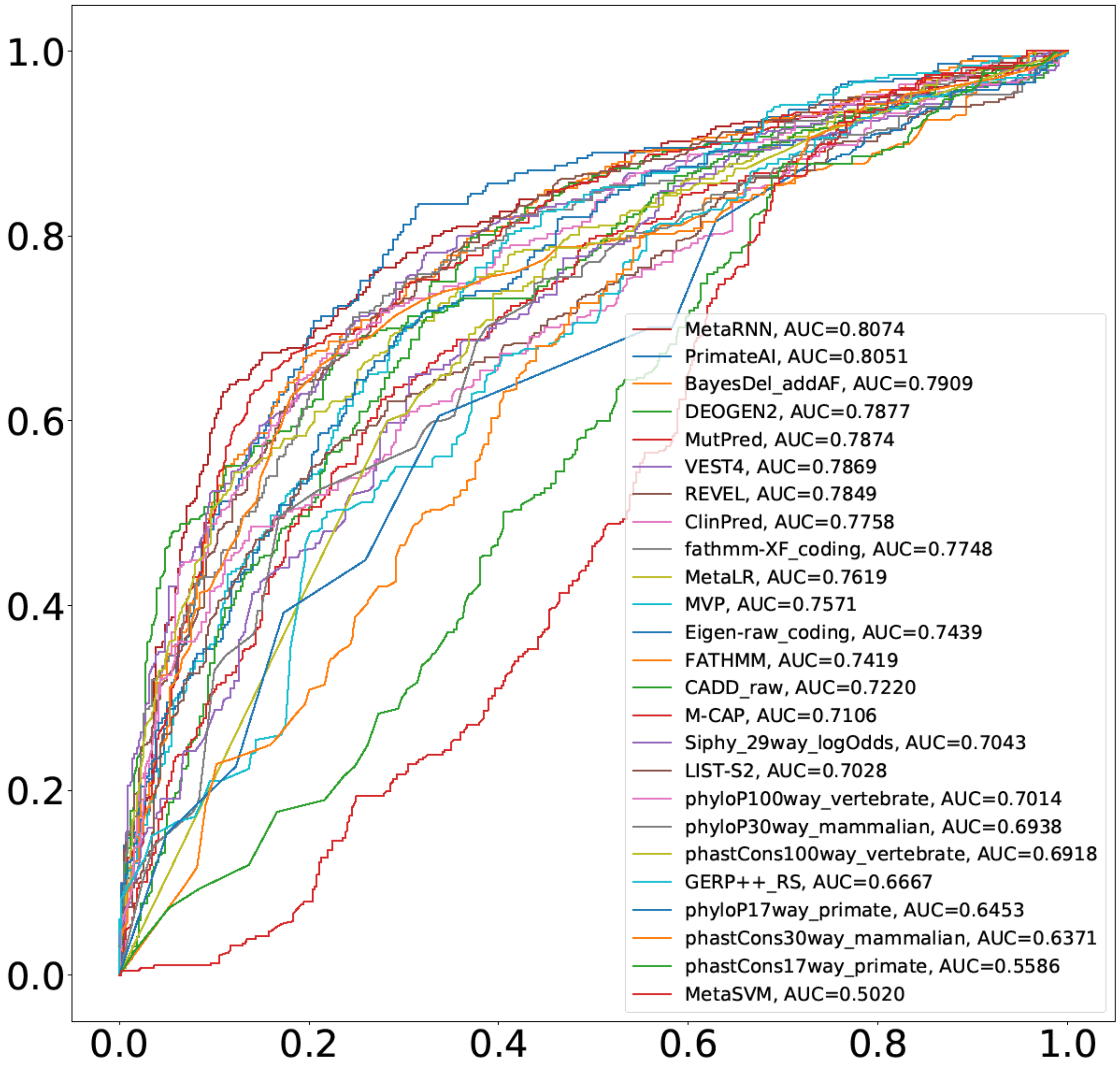
S. Figure 8. Performance of different methods benchmarked using allele-frequency-filtered rare ClinVar-only test set (AF-RCTS).

Supplementary Note



S. Figure 9. Performance of different methods benchmarked using the all-allele-frequency set (AAFS).

Supplementary Note



S. Figure 10. Performance of different methods benchmarked using TP53 test set (TP53TS).

Supplementary Note

References

1 Landrum, M. J. *et al.* ClinVar: public archive of interpretations of clinically relevant variants. *Nucleic Acids Res* **44**, D862-868, doi:10.1093/nar/gkv1222 (2016).

2 Karczewski, K. J. *et al.* The mutational constraint spectrum quantified from variation in 141,456 humans. *Nature* **581**, 434-443, doi:10.1038/s41586-020-2308-7 (2020).

3 Lek, M. *et al.* Analysis of protein-coding genetic variation in 60,706 humans. *Nature* **536**, 285-291, doi:10.1038/nature19057 (2016).

4 Genomes Project, C. *et al.* A global reference for human genetic variation. *Nature* **526**, 68-74, doi:10.1038/nature15393 (2015).

5 Kumar, P., Henikoff, S. & Ng, P. C. Predicting the effects of coding non-synonymous variants on protein function using the SIFT algorithm. *Nature protocols* **4**, 1073 (2009).

6 Adzhubei, I., Jordan, D. M. & Sunyaev, S. R. Predicting functional effect of human missense mutations using PolyPhen‐2. *Current protocols in human genetics* **76**, 7.20. 21-27.20. 41 (2013).

7 Reva, B., Antipin, Y. & Sander, C. Predicting the functional impact of protein mutations: application to cancer genomics. *Nucleic acids research* **39**, e118-e118 (2011).

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

9 Carter, H., Douville, C., Stenson, P. D., Cooper, D. N. & Karchin, R. Identifying Mendelian disease genes with the variant effect scoring tool. *BMC genomics* **14**, 1-16 (2013).

10 Jagadeesh, K. A. *et al.* M-CAP eliminates a majority of variants of uncertain significance in clinical exomes at high sensitivity. *Nature genetics* **48**, 1581 (2016).

11 Ioannidis, N. M. *et al.* REVEL: an ensemble method for predicting the pathogenicity of rare missense variants. *The American Journal of Human Genetics* **99**, 877-885 (2016).

12 Pejaver, V. *et al.* Inferring the molecular and phenotypic impact of amino acid variants with MutPred2. *Nature communications* **11**, 5918 (2020).

13 Qi, H. *et al.* MVP predicts the pathogenicity of missense variants by deep learning. *Nature Communications* **12**, 510 (2021).

14 Sundaram, L. *et al.* Predicting the clinical impact of human mutation with deep neural networks. *Nature genetics* **50**, 1161-1170 (2018).

15 Raimondi, D. *et al.* DEOGEN2: prediction and interactive visualization of single amino acid variant deleteriousness in human proteins. *Nucleic acids research* **45**, W201-W206 (2017).

16 Rentzsch, P., Witten, D., Cooper, G. M., Shendure, J. & Kircher, M. CADD: predicting the deleteriousness of variants throughout the human genome. *Nucleic Acids Res* **47**, D886-D894, doi:10.1093/nar/gky1016 (2019).

17 Rogers, M. F. *et al.* FATHMM-XF: accurate prediction of pathogenic point mutations via extended features. *Bioinformatics* **34**, 511-513 (2018).

18 Ionita-Laza, I., McCallum, K., Xu, B. & Buxbaum, J. D. A spectral approach integrating functional genomic annotations for coding and noncoding variants. *Nature genetics* **advance on**, 214-220, doi:10.1038/ng.3477 (2016).

19 Lu, Q. *et al.* A statistical framework to predict functional non-coding regions in the human genome through integrated analysis of annotation data. *Scientific reports* **5**, 10576, doi:10.1038/srep10576 (2015).

20 Cooper, G. M. *et al.* Distribution and intensity of constraint in mammalian genomic sequence. *Genome research* **15**, 901-913 (2005).

21 Pollard, K. S., Hubisz, M. J., Rosenbloom, K. R. & Siepel, A. Detection of nonneutral substitution rates on mammalian phylogenies. *Genome Research* **20**, 110-121, doi:10.1101/gr.097857.109 (2010).

22 Garber, M. *et al.* Identifying novel constrained elements by exploiting biased substitution patterns. *Bioinformatics* **25**, 54-62, doi:10.1093/bioinformatics/btp190 (2009).

23 Oba, S. *et al.* A Bayesian missing value estimation method for gene expression profile data. *Bioinformatics* **19**, 2088-2096, doi:10.1093/bioinformatics/btg287 (2003).

24 Cho, K., Bart, Bahdanau, D. & Bengio, Y. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches. *arXiv pre-print server*, doi:arxiv:1409.1259 (2014).

25 Snoek, J., Larochelle, H. & Ryan. Practical Bayesian Optimization of Machine Learning Algorithms. *arXiv pre-print server*, doi:None
arxiv:1206.2944 (2012).

26 Pedregosa, F. *et al.* Scikit-learn: Machine learning in Python. *the Journal of machine Learning research* **12**, 2825-2830 (2011).

27 Zhao, H. *et al.* DDIG-in: discriminating between disease-associated and neutral non-frameshifting micro-indels. *Genome biology* **14**, 1-13 (2013).

28 Douville, C. *et al.* Assessing the Pathogenicity of Insertion and Deletion Variants with the Variant Effect Scoring Tool (VEST-Indel). *Hum Mutat* **37**, 28-35, doi:10.1002/humu.22911 (2016).

29 Schwarz, J. M., Cooper, D. N., Schuelke, M. & Seelow, D. MutationTaster2: mutation prediction for the deep-sequencing age. *Nature methods* **11**, 361-362 (2014).

30 Alirezaie, N., Kernohan, K. D., Hartley, T., Majewski, J. & Hocking, T. D. ClinPred: prediction tool to identify disease-relevant nonsynonymous single-nucleotide variants. *The American Journal of Human Genetics* **103**, 474-483 (2018).

31 Shihab, H. a. *et al.* An integrative approach to predicting the functional effects of non-coding and coding sequence variation. *Bioinformatics (Oxford, England)* **31**, 1536-1543, doi:10.1093/bioinformatics/btv009 (2015).

32 Dong, C. *et al.* Comparison and integration of deleteriousness prediction methods for nonsynonymous SNVs in whole exome sequencing studies. *Human molecular genetics* **24**, 2125-2137 (2015).

33 Feng, B. J. PERCH: a unified framework for disease gene prioritization. *Human mutation* **38**, 243-251 (2017).

34 Malhis, N., Jacobson, M., Jones, S. J. & Gsponer, J. LIST-S2: taxonomy based sorting of deleterious missense mutations across species. *Nucleic acids research* **48**, W154-W161 (2020).

35 Davydov, E. V. *et al.* Identifying a high fraction of the human genome to be under selective constraint using GERP++. *PLoS Computational Biology* **6**, doi:10.1371/journal.pcbi.1001025 (2010).