nature machine intelligence

ARTICLES

https://doi.org/10.1038/s42256-020-0222-1

Protein function prediction is improved by creating synthetic feature samples with generative adversarial networks

Cen Wan[1,2] and David T. Jones[1,2] ✉

Protein function prediction is a challenging but important task in bioinformatics. Many prediction methods have been developed, but are still limited by the bottleneck on training sample quantity. Therefore, it is valuable to develop a data augmentation method that can generate high-quality synthetic samples to further improve the accuracy of prediction methods. In this work, we propose a novel generative adversarial networks-based method, FFPred-GAN, to accurately learn the high-dimensional distributions of protein sequence-based biophysical features and also generate high-quality synthetic protein feature samples. The experimental results suggest that the synthetic protein feature samples are successful in improving the prediction accuracy for all three domains of Gene Ontology through augmentation of the original training protein feature samples.

Protein function prediction is an important but challenging task in bioinformatics. The challenge comes from the inherent high dimensionality of the input feature space and the cryptic relationship between sequence and function. The importance is clear from the fact that very few proteins in data banks have complete or reliable functional annotations. Up to the year 2015, fewer than 0.1% of proteins deposited in the UniProt database had received even one experiment-based functional annotation, and fewer than 20% of proteins had even been electronically annotated in all three domains of gene function as defined by Gene Ontology (GO)[1]. Although recent community-wide efforts have overall pushed forward the development of computational prediction methods, the accuracy of predicting the vast majority of protein functions remains very low[2–4]. This is not only because of the natural diversity of protein function, but also because of the limited number of existing functionally annotated protein samples. This issue has led to a bottleneck in the performance of prediction methods, especially for machine learning-based methods[5–7], on making accurate predictions based on such small reference or training datasets. Owing to the expense of obtaining protein function data experimentally, it is highly desirable to develop computational methods that can make better use of existing limited data. To that end, here we explore the possibility that high-quality synthetic samples can be created to augment the existing annotation data and further improve the predictive accuracy of our prediction models.

Generative adversarial networks (GANs)[8–13] are a new type of generative model and aim to generate high-quality synthetic samples by accurately learning the underlying distributions of target data samples. The novel aspect of GANs is that they adopt an adversarial training paradigm, where two neural networks 'fight' against each other to learn the distribution of samples. One network (the generator) attempts to generate synthetic data and the other network (the discriminator) attempts to decide whether a given sample is real or synthetic. Each network gets better and better at its task until an equilibrium is reached, where the generator cannot make better samples, and the discriminator cannot detect more synthetic samples. GANs have already shown outstanding performance on different machine learning tasks in the image processing field, such as image to image translation[14–16], image segmentation[17–19] and image reconstruction[20–22]. In addition to handling image data, GANs have also performed well with other types of data, such as gene expression data and raw gene sequence data. Wang et al. (2018)[23] and Dizaji et al. (2018)[24] proposed a conditional GAN-based framework for the task of gene expression profiles inference by modelling the conditional distribution of target genes given the corresponding landmark genes' profiles. Ghahramani et al. (2018)[25] also adopted the Wasserstein GAN–gradient penalty (WGAN-GP), a variant of GANs, to capture the diversity of cell types based on large and sparse scRNA-seq data. More recently, Gupta and Zou (2019)[26] and Wang et al. (2019)[27] successfully proposed GAN-based methods to generate synthetic genes and promoters, respectively.

The data augmentation task is also an area where GANs show a great potential to achieve good performance. Most of the existing works on GAN-based data augmentation methods also focus on image processing tasks such as image classification[28–30]. For example, Frid-Adar et al. (2018)[28] adopted the well-known DCGAN[9] method to generate synthetic liver lesion images, which successfully improved the accuracy of liver lesion classification. Most recently, Marouf et al. (2018)[31] adopted GANs to generate synthetic scRNA-seq profiles, which were used for downstream cell type classification tasks. In this work, we propose a new GAN-based data augmentation approach—FFPred-GAN—which successfully employs GANs to cope with protein sequence-based data distributions to tackle the protein function prediction problem. The novelties of this approach are threefold. First, FFPred-GAN successfully learns the distribution of protein amino acid sequence-based biophysical features and generates high-quality synthetic protein feature samples. Moreover, those high-quality synthetic protein feature samples successfully augment the original training samples and obtain significantly higher accuracy in predicting all three domains of GO terms. FFPred-GAN also shows good computational time efficiency, which is valuable when dealing with the large amount of sequence data in present data banks. These properties also encourage further extension of FFPred-GAN in exploiting other types of protein-related features.

[1]Biomedical Data Science Laboratory, The Francis Crick Institute, London, UK. [2]Department of Computer Science, University College London, London, UK. ✉e-mail: d.t.jones@ucl.ac.uk

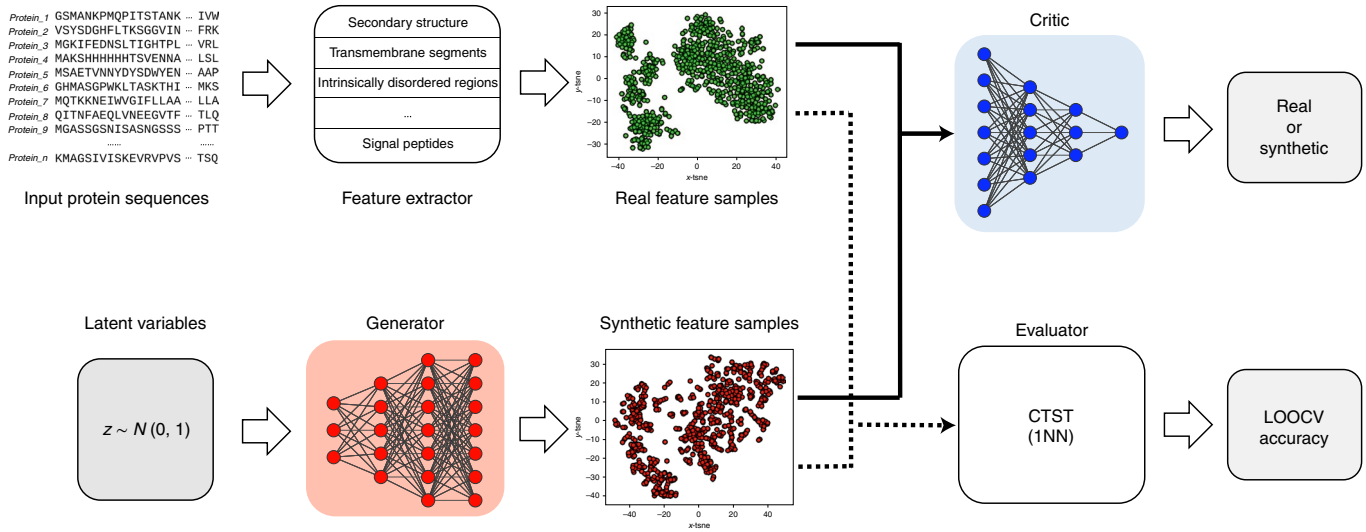


Fig. 1 | The flowchart for FFPred-GAN. The FFPred-GAN framework consists of three steps to generate high-quality synthetic training protein feature samples. The first step is extracting protein biophysical features (as shown in the three upperleft panels), followed by training a WGAN-GP model (as shown in the three bottom-left panels and the upper-right panel) to generate several synthetic feature samples. Those synthetic samples then are further selected by using the CTST (as shown in the bottom-right panel) for the data augmentation task.

Results

Overview of FFPred-GAN. In general, the FFPred-GAN framework consists of three steps to generate high-quality synthetic training protein feature samples, as shown in Fig. 1. First, FFPred-GAN adopts the widely used FFPred[32] feature extractor to derive protein biophysical information based on raw amino acid sequences. For each input protein sequence, 258 dimensional features are generated to describe 13 groups of protein biophysical information, such as secondary structure, amino acid composition and presence of motifs. FFPred-GAN then adopts the Wasserstein generative adversarial network with gradient penalty (WGAN-GP) approach[11] to learn the actual high-dimensional distributions of these training proteins' features. The generator of WGAN-GP is used to output the synthetic training protein feature samples during different training stages of FFPred-GAN. In the last step, FFPred-GAN uses the classifier two-sample test (CTST)[33] to select the optimal synthetic training protein feature samples, which are used to augment the original training samples. During the downstream machine learning classifier training stage, the optimal synthetic samples are expected to derive better classifiers, leading to higher predictive accuracy.

FFPred-GAN successfully generates high-quality synthetic protein biophysical feature samples. In general, FFPred-GAN successfully learns the distributions of the training protein biophysical feature samples and generates high-quality synthetic ones. We train two FFPred-GAN models for each GO term by using two different sets of protein samples with different class labels. The first FFPred-GAN model is trained by using the protein samples that are annotated with that GO term (hereafter we denote those proteins as positive samples). The other FFPred-GAN model is trained by using the protein samples that are not annotated by that GO term (hereafter we denote those proteins as negative samples). Therefore, in total, we train 602 FFPred-GAN models for all 301 GO terms in the FFPred-fly library.

We adopt the one-nearest-neighbour classification algorithm and leave-one-out cross-validation (LOOCV) to conduct the classifier two-sample tests, which are used for evaluating the quality of synthetic protein feature samples. The closer the value of LOOCV accuracy is to 0.500, the higher the quality of the synthetic samples. Figure 2a–c shows the LOOCV accuracies obtained for synthetic positive and negative protein feature samples (denoted as stars and circles, respectively) generated by individual GO term-based FFPred-GANs. The *x* axis denotes the index of each GO term, while the *y* axis denotes the LOOCV accuracy, which ranges from 0.000 to 1.000. In general, the synthetic positive protein feature samples generated by FFPred-GAN for nearly half of the biological process (BP), molecular function (MF) and cellular component (CC) terms obtained a LOOCV accuracy of 0.500. The average LOOCV accuracies for the BP, MF and CC domains of the GO terms are 0.573, 0.584 and 0.590, respectively. Figure 2d–h displays the *t*-SNE (*t*-distributed stochastic neighbour embedding) transformed two-dimensional (2D) visualization of real and synthetic positive protein feature samples that are generated during different training stages of FFPred-GAN for the BP term GO:0000375. In detail, at the beginning of FFPred-GAN training (that is, after the first iteration), the real positive protein feature samples (green dots) are distributed distantly from the synthetic ones (red dots), leading to a LOOCV accuracy of 1.000, which suggests obvious differences between the real and synthetic sets of protein feature samples. After 1,000 iterations of further training, FFPred-GAN shows that it has started to capture the distribution of the real protein feature samples and has generated synthetic ones that are beginning to be similar to the real ones; this is seen by the distributions of the red and green dots having overlapping areas around the diagonal, which gives a better LOOCV accuracy of 0.737. After even further training of FFPred-GAN, on the 10,001st iteration, the overlapping areas of the two sets of protein samples become broader, giving a LOOCV accuracy of 0.645, which also indicates the substantially improved training quality of FFPred-GAN. The training quality of FFPred-GAN continues to improve with more iterations of training, with the LOOCV accuracy reaching 0.515 after another 10,000 iterations. Finally, after 29,601 iterations of training, FFPred-GAN has been successfully trained, achieving the desired LOOCV accuracy of 0.500. Also, as shown in Fig. 2h, the two sets of protein feature samples project into almost exactly the same areas. This pattern is consistent when training FFPred-GAN for the positive protein feature samples for the MF and CC domains of the GO terms. As shown in Supplementary Fig. 1a–e and 1f–j, respectively, the quality of the GO:0000981 and GO:0000785 synthetic positive protein feature samples gradually improves with an increasing number of

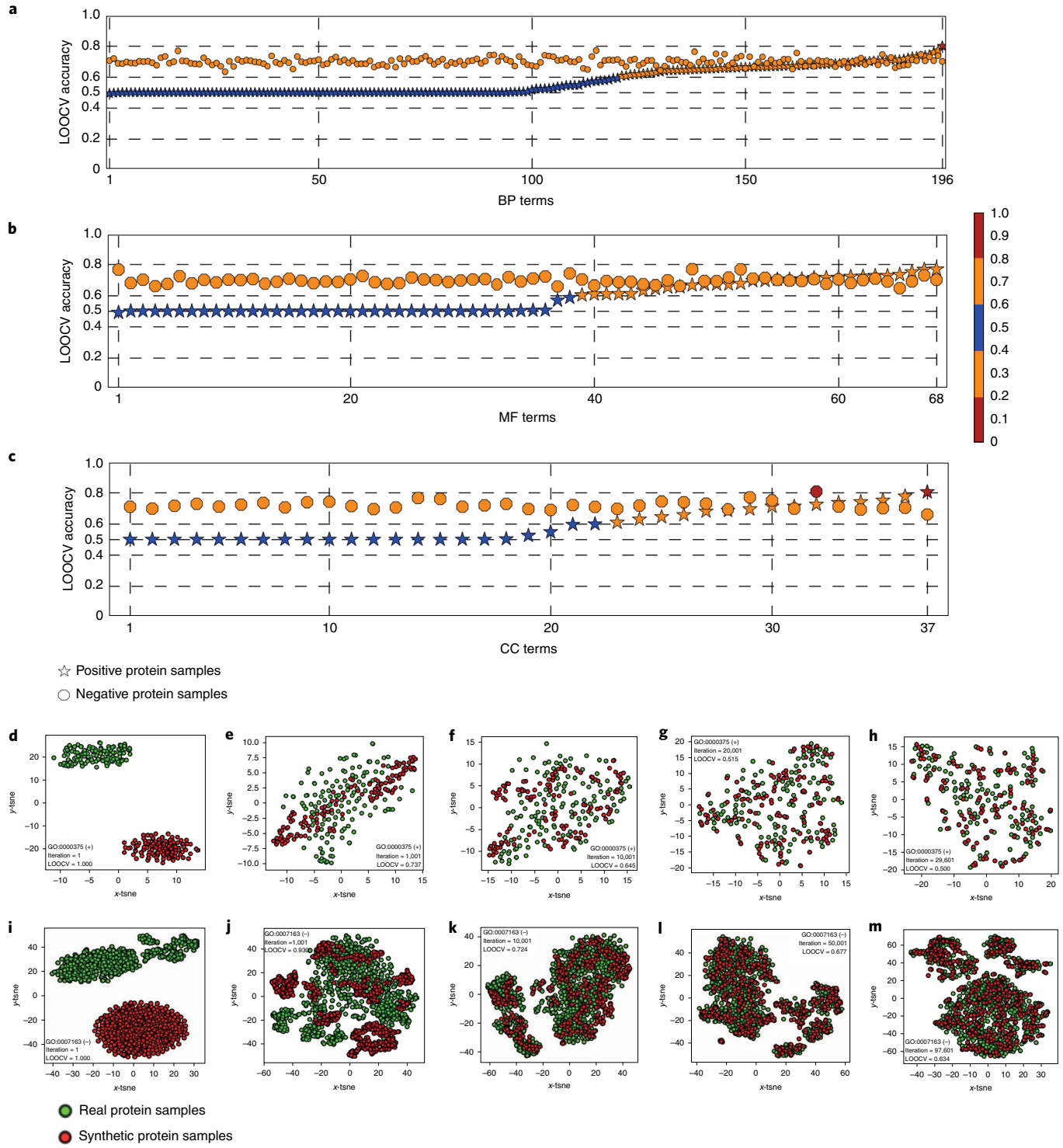


Fig. 2 | The CTST results and 2D visualization of real and synthetic protein feature samples. a–c, LOOCV accuracy of CTST obtained for real and synthetic protein samples for GO terms from the biological process (BP) (**a**), molecular function (MF) (**b**) and cellular component (CC) (**c**) domains. **d–m**, *t*-SNE-transformed 2D visualization of real (green dots) and synthetic (red dots) protein feature samples obtained during different training iterations of FFPred-GAN by using positive and negative protein feature samples for BP terms GO:0000375 (**d**, 1 iteration; **e**, 1,001 iterations; **f**, 10,001 iterations; **g**, 20,001 iterations; **h**, 29,601 iterations) and GO:0007163 (**i**, 1 iteration; **j**, 1,001 iterations; **k**, 10,001 iterations; **l**, 50,001 iterations; **m**, 97,601 iterations).

training iterations. The green and red dots for the synthetic positive protein feature samples of GO:0000981 are distributed similarly after 44,201 iterations of training. Analogously, the distributions of synthetic positive protein feature samples for GO:0000785 also become similar to the corresponding real ones after 31,801 iterations of training, because the LOOCV accuracy reaches 0.500.

Owing to the much higher diversity of negative feature samples, in that there are few ways of representing a positive case but many

ways of representing a negative case, the accuracy for negative cases is lower, as might be expected. The LOOCV accuracies obtained for the synthetic negative protein feature samples range between 0.600 and 0.800 for all three domains of GO terms. The average LOOCV accuracies are 0.700, 0.698 and 0.720, respectively, for BP, MF and CC domains. Analogously to the cases when training the synthetic positive protein feature samples, at the beginning of the FFPred-GAN training stage (that is, after the first iteration), the real and synthetic negative samples for term GO:0007163 are obviously different, because the two sets are distributed in different areas in Fig. 2i. After 1,001 iterations of training, the distributions for both sets begin to overlap, but the LOOCV accuracy of 0.930 is still far from optimal. After the 10,001st iteration, the overlapping areas of both sets' distributions become broader, with an improved LOOCV accuracy of 0.724. The training quality of FFPred-GAN continues to improve even after 50,001 iterations of training, and finally the optimal negative synthetic protein feature samples are obtained after 97,601 iterations of training, with an optimal LOOCV accuracy of 0.634. As shown in Fig. 2m, both green and red dots distribute in similar areas. This pattern is consistent when training FFPred-GAN for two other domains of GO terms, such as GO:0046872 and GO:0016020. As shown in Supplementary Fig. 1o and 1t, the real and synthetic negative protein feature samples for those two terms distribute in similar patterns after 52,201 and 49,001 iterations of training, leading to optimal LOOCV accuracies of 0.648 and 0.661, respectively.

Synthetic protein feature samples generated by FFPred-GAN successfully improve the predictive accuracy of *Drosophila* function annotation using FFPred-fly. We evaluated the predictive power of using synthetic protein feature samples on the task of protein function prediction applied to *Drosophila*. We integrated the synthetic and real protein feature samples as the augmented training protein feature samples in eight different ways, that is, synthetic positive + real positive + real negative, synthetic negative + real positive + real negative, synthetic positive + synthetic negative + real positive + real negative, synthetic positive + real negative, synthetic negative + real positive, synthetic positive + synthetic negative + real positive, synthetic positive + synthetic negative + real negative and synthetic positive + synthetic negative. Predictions using all these different combinations are compared with each other and with those from the original (benchmark) combination, real positive + real negative. Three well-known classification methods—support vector machine (SVM), *k*-nearest-neighbour (*k*NN) and random forests (RF) are used to train models for predicting the GO term annotations of test protein samples. Extended Data Figs. 1 and 2 present boxplots of distributions of ranks for the 27 individual modelling strategies, according to their corresponding Matthews correlation coefficient (MCC) and area under receiver operating characteristic curve (AUROC) values (Supplementary File 2). Within each box, the black line indicates the value of average rank, ranging from 1 to 27, where a lower rank denotes better predictive performance. Information about the average ranks of all 27 individual modelling strategies is included in Supplementary Tables 1 and 2.

In general, the synthetic protein feature samples successfully improve the predictive performance of the original combination of training protein feature samples and lead to the overall highest accuracy for predicting all three domains of GO terms with an SVM classification algorithm. To predict the BP domain of GO terms, the combination of synthetic positive + real positive + real negative gives the overall best average ranks of 5.88 and 4.84, respectively, according to the MCC and AUROC values, by using SVM as the classification algorithm. However, the benchmark combination of real positive + real negative with SVM only gives average ranks of 7.92 and 5.66. Figure 3a,b shows pairwise comparisons of MCC and AUROC values obtained by those two types of combination with an SVM classifier over each of the 196 BP terms. As shown by green dots, 106 and 103 BP terms, respectively, obtain higher MCC and AUROC values by using synthetic positive sample augmented training data. Wilcoxon signed-rank tests also confirm that the synthetic positive augmented training samples significantly outperform the standard benchmark combination, as evidence by the *P* values of 5.86×10^{-3} and 3.12×10^{-2}. In addition, the synthetic positive + synthetic negative + real positive + real negative training protein samples also obtain the second best average rank of 5.95 with SVMs according to the MCC values.

Analogously, for predicting MF GO terms, the combination of synthetic positive + real positive + real negative also obtains the overall best average ranks of 3.82 and 4.40, respectively, according to MCC and AUROC values obtained using an SVM classifier, whereas the corresponding benchmark combination of training protein samples with SVMs only obtains average ranks of 4.60 and 4.83. Figure 3c,d shows that 37 and 34 MF terms obtain higher MCC and AUROC values, respectively. The *P* value of 3.91×10^{-2} also suggests that the former obtains significantly higher MCC values than the latter. The second best-performing combination is also synthetic positive + synthetic negative + real positive + real negative, which obtains the average ranks of 4.13 and 4.81 using an SVM classification algorithm.

For predicting CC GO terms, the combination of synthetic positive + synthetic negative + real positive + real negative gives the overall best average ranks of 4.95 and 4.70, respectively, according to MCC and AUROC values. It obtains higher MCC values than the benchmark combination of training protein samples when working with an SVM, and also higher AUROC values when using RF as the classification algorithm. Figure 3e,f shows that 25 CC terms obtain higher MCC and AUROC values when using the combination of synthetic positive + synthetic negative + real positive + real negative as the training samples, respectively, when SVM and RF classification algorithms are used. The *P* value of 5.29×10^{-3} also confirms a significant difference between the MCC values obtained by the two combinations.

We further evaluated the performance of the FFPred-GAN augmented training samples using a new set of CAFA 3 (3rd Critical Assessment of protein Function Annotation)[4] targets that do not overlap with any protein samples used for training any of the GO term-based classifiers. In general, the FFPred-GAN augmented training samples lead to higher accuracy in predicting all three domains of GO terms. As shown in Fig. 4, to predict BP terms, the middle part of the black curve locates above the yellow curve, suggesting a higher F_{max} score of 0.325 obtained by the FFPred-GAN augmented training samples than a lower F_{max} score of 0.308 obtained by the original training samples. Analogously, the FFPred-GAN augmented training samples also obtain a higher F_{max} score of 0.385 for predicting MF terms, as shown by the middle part of the light-blue curve above the orange curve, which leads to a lower F_{max} score of 0.381 obtained by the original training samples. To predict CC terms, the FFPred-GAN augmented training samples obtain the highest F_{max} score of 0.629, as shown by the blue curve, which locates above the red curve obtained by the original training samples with a lower F_{max} score of 0.605. We also compare with another protein sequence-based prediction method—ProLanGO[34]—which uses recurrent neural networks. The experimental results confirm that both FFPred-GAN and FFPred outperform ProLanGO on predicting all three domains of GO terms.

FFPred-GAN augmented training samples obtain higher predictive accuracy than the training samples augmented by SMOTE. We also compared FFPred-GAN with a well-known data augmentation method—the 'synthetic minority over-sampling technique' (SMOTE)[35]—which performs over-sampling on the minority class by creating synthetic samples between each individual minority class sample and their corresponding randomly selected *k* nearest

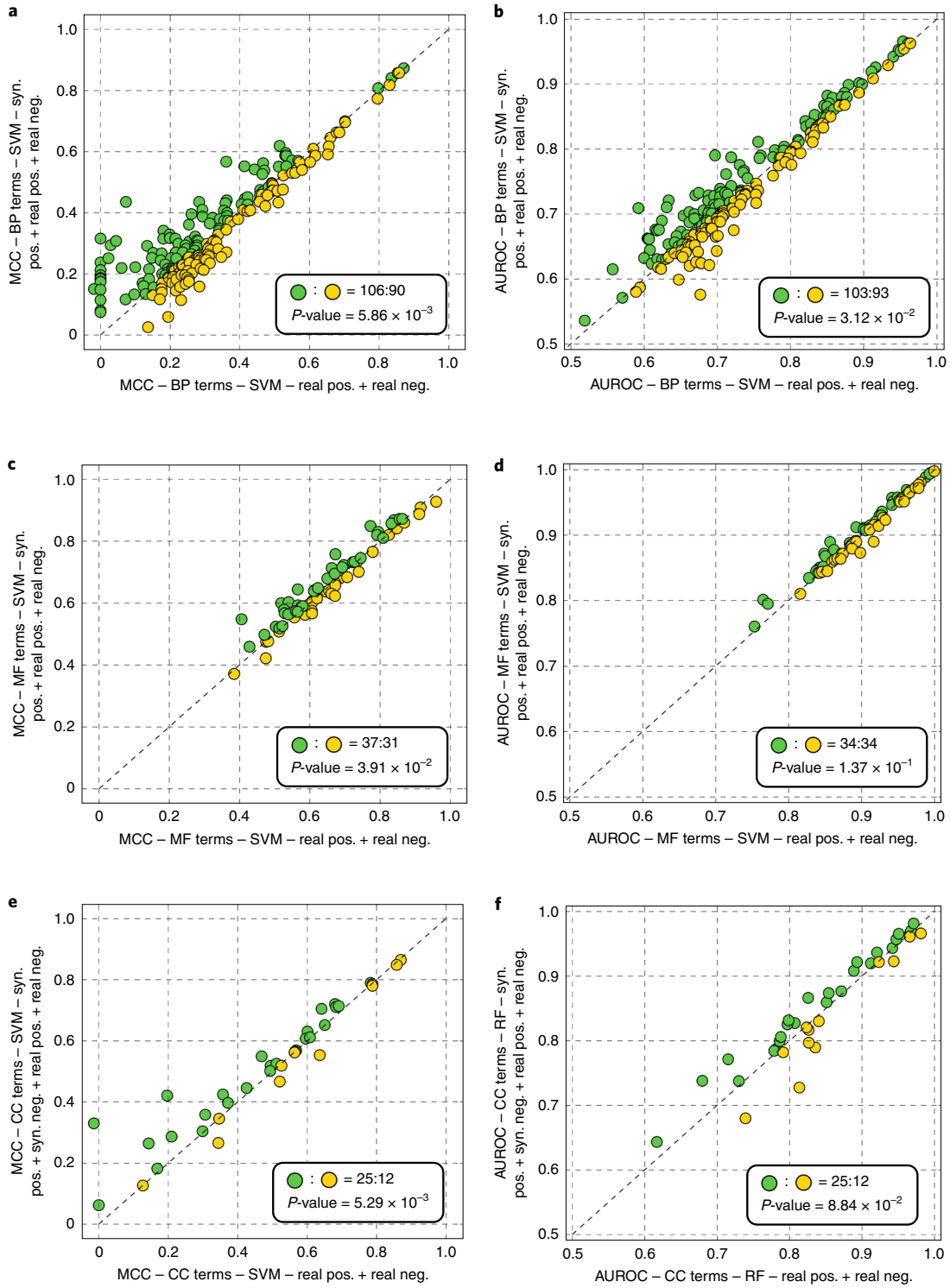


Fig. 3 | Comparison of predictive accuracy obtained by augmented training samples and the original training samples. a–**f**, Scatter plots of the MCC (**a**,**c**,**e**) and AUROC (**b**,**d**,**f**) values obtained by the optimal combination of real and synthetic protein samples and the benchmark combination of real protein samples (as indicated on the axes) for predicting three domains of GO terms using SVM (**a**–**e**) and RF (**f**) classification algorithms. The green dots indicate those GO terms whose MCC or AUROC values obtained by training samples A (as shown by the *y* axis) are higher than the ones obtained by training samples B (as shown by the *x* axis); vice versa, the yellow dots indicate those GO terms whose MCC or AUROC values obtained by training samples A are lower or equal to the ones obtained by training samples B.

neighbours. We set the number of over-sampled minority class samples for SMOTE to be the same as the number of synthetic samples generated by FFPred-GAN and use the default values of other hyper-parameters implemented in ref. [36]. The SVM classifiers are trained using the SMOTE augmented training samples to predict all three domains of GO terms. In addition, a set of RF classifiers are trained to evaluate the AUROC values obtained on predicting the cellular component terms.

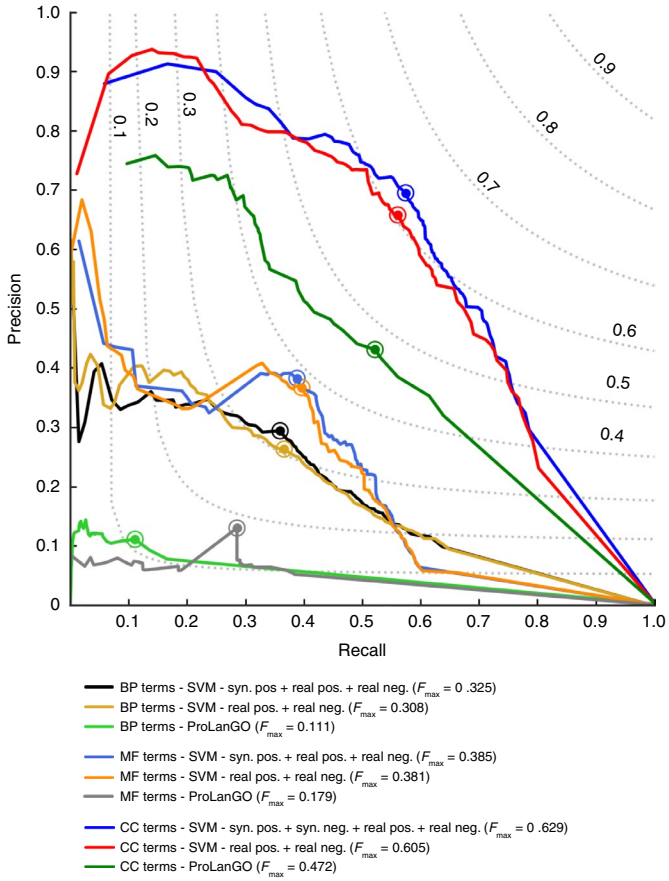


Fig. 4 | Precision-recall curves. The precision-recall curves were obtained from FFPred-GAN augmented training samples, the original training samples and ProLanGO in predicting three domains of GO terms for the CAFA 3 targets.

In general, the FFPred-GAN augmented training samples obtain higher predictive accuracy than those augmented by SMOTE for predicting all three domains of GO terms. As shown in Extended Data Fig. 3a,b, compared with SMOTE, FFPred-GAN obtains higher MCC and AUROC values, respectively, when predicting 104 and 102 BP terms. Wilcoxon signed-rank tests also confirm the significant difference between the two methods at a significance level of 5.00×10^{-2}, according to the P values of 2.88×10^{-3} and 2.75×10^{-2}. Analogously, for predicting MF terms, as shown in Extended Data Figs. 3c,d, FFPred-GAN augmented training samples obtain higher MCC values than the SMOTE augmented training samples on predicting 40 out of 68 terms, with a P value of 8.09×10^{-3}, while the former also show higher AUROC values on 34 out of 68 MF terms. To predict CC terms, as shown in Extended Data Fig. 3e,f, the training samples augmented by FFPred-GAN obtain higher MCC values than the SMOTE augmented training samples on predicting 24 out of 37 terms, leading to a P value of 3.47×10^{-2}, while the former also obtain higher AUROC values on predicting 20 out of 37 CC terms using the RF classifiers. The F_{max} scores obtained on the CAFA 3 targets also confirm that the FFPred-GAN augmented training samples lead to higher accuracy in predicting all three domains of GO terms, as shown in Supplementary Table 3.

We also compared the SMOTE augmented training samples with the original training samples. In general, the SMOTE augmented training samples obtain only slightly higher predictive accuracy on predicting a small proportion of GO terms. As shown in Supplementary Fig. 2a, both the green and yellow dots locate on the area close to the diagonal, suggesting that the differences in MCC values obtained by the two different groups of training samples are small. Only 54 out of 196 BP terms receive higher MCC values using the SMOTE augmented training samples, while 85 out of 196 BP terms receive the same MCC values using the two different groups of training samples (shown by the blue dots). In addition, the AUROC values obtained by the two different groups of training samples are almost the same, as shown in Supplementary Fig. 2b, where almost all dots locate on the diagonal. Those consistent patterns are also observed when predicting MF and CC terms. The significance test results further confirm that there is no significant difference in the MCC and AUROC values obtained by the two different groups of training samples on predicting all three domains of GO terms. The F_{max} scores obtained on the CAFA 3 targets also confirm that the SMOTE augmented training samples merely lead to slightly higher accuracy on predicting the MF and CC domains of GO terms, as shown in Supplementary Table 3.

FFPred-GAN augmented training samples successfully improve the predictive accuracy of a state-of-the-art protein function prediction method. We further evaluated the predictive performance of the FFPred-GAN augmented training samples (hereafter denoted FFPred-GAN) by integrating with the state-of-the-art protein function prediction method NetGO[37], which is an improved version of GOLabeler[38], the top-ranked method in the recent CAFA 3 competition. Analogous to GOLabeler, NetGO makes predictions of GO terms by using the learning-to-rank approach based on different component classifiers trained by multiple data sources, for example protein sequence, structure and protein–protein interaction network. In terms of the approach to integrate the predictions of FFPred-GAN and NetGO, we first back-propagate the predictions of all GO terms made by individual methods. For each target, we compare the predictive probability of individual GO terms with the predictive probabilities of their corresponding ancestors that are defined by the 'is-a' relationship retained in the GO hierarchy. If the predictive probability of that GO term is higher than any of its ancestor GO term's probability, the predictive probability of that ancestor GO term will be replaced. We then trained a library of logistic regression models for those GO terms receiving predictions from both methods simultaneously. For each GO term included in the back-propagated predictions, if one target receives predictive probabilities from both methods, we adopt the predictive probabilities as features to create a new instance. The value of that instance's label is defined according to the true GO term annotation label set, where the labels are also back-propagated according to the GO hierarchy. A grid search is conducted to optimize the hyper-parameters of logistic regression. During the testing stage, we use the library of logistic regression models to obtain the predictions for corresponding GO terms, and merge with NetGO's predictions on those GO terms that do not overlap with the GO terms predicted by FFPred-GAN. We also integrate the predictions made by the original training samples (hereafter denoted FFPred) and NetGO following the same approach. We conduct 10-fold cross-validation on the new set of CAFA 3 targets to evaluate this integration approach.

In general, the experimental results confirm that the predictions made by FFPred-GAN successfully improve the performance of NetGO, leading to state-of-the-art accuracy in predicting all three domains of GO terms. In Fig. 5a, the blue dot locates in the highest position, indicating that the highest F_{max} score of 0.567 is obtained by the integration of FFPred-GAN and NetGO on predicting BP terms. The second best-performing method is the integration of FFPred and NetGO, with an F_{max} score of 0.552, which is also higher than the one obtained by NetGO alone (0.536). Analogously, when predicting MF terms, the integration of FFPred-GAN and NetGO also obtains the highest F_{max} score of 0.750 among all methods, as shown in Fig. 5b, where the blue dot locates the highest position. The integration of FFPred and NetGO obtained the second-highest

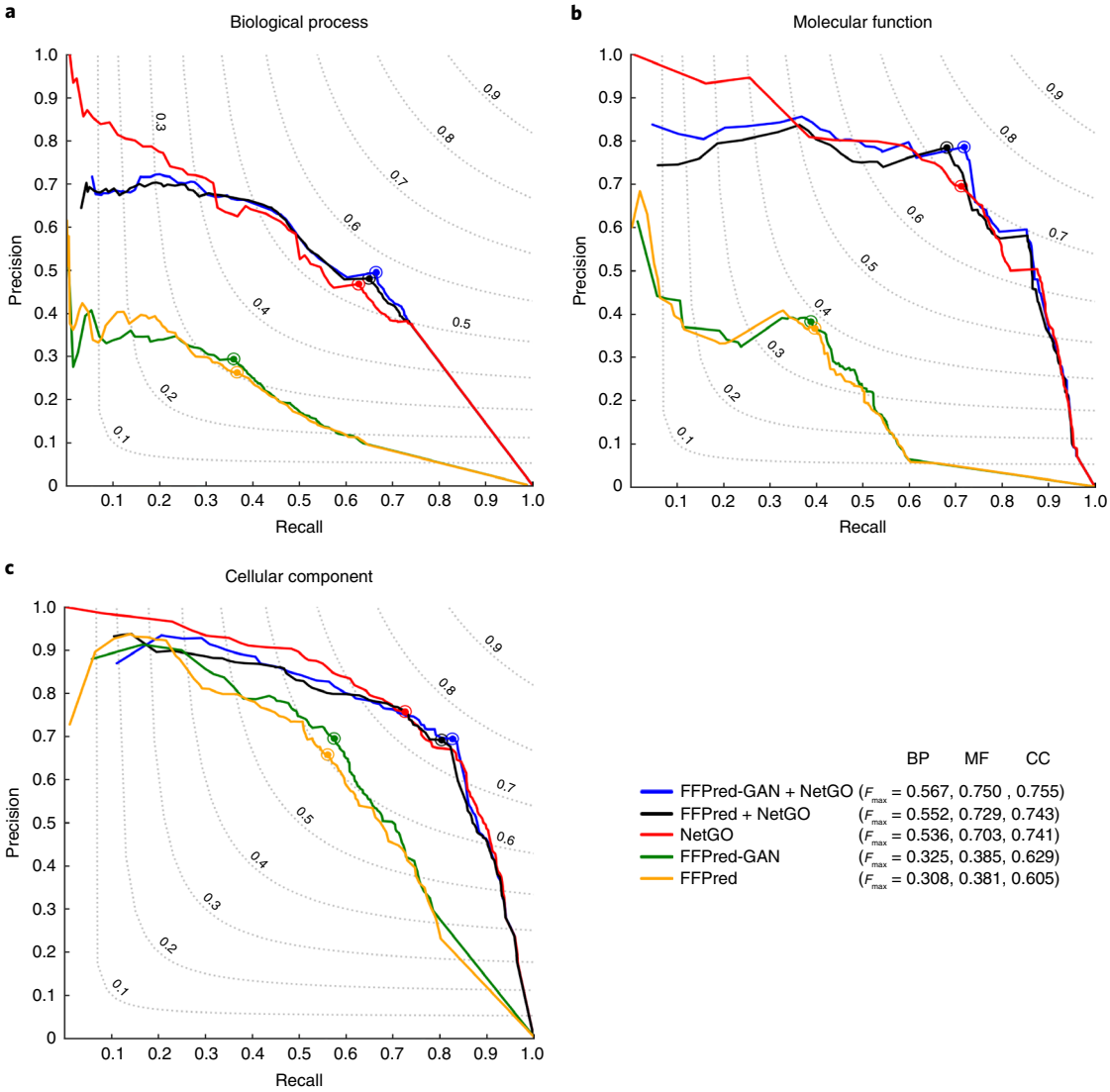


Fig. 5 | Comparison of five different methods' predictive performance. a–c, Precision-recall curves obtained with five different methods on predicting the BP (**a**), MF (**b**) and CC (**c**) domains of GO terms for the CAFA 3 targets.

F_{max} score of 0.729, which is higher than the F_{max} score of 0.703 obtained by NetGO alone. When predicting CC terms, as shown in Fig. 5c, the highest F_{max} score of 0.755 is obtained by the integration of FFPred-GAN and NetGO. The second best-performing method is the integration of FFPred and NetGO (F_{max} of 0.743), which is higher than NetGO alone.

Discussion

Overall, as discussed in previous sections, the synthetic samples generated by FFPred-GAN successfully improve the predictive accuracy from the original training samples. In this section, we further explore the reasons for the improvement and the computational cost of generating optimal synthetic training samples by FFPred-GAN.

Augmented training samples induce better SVM decision boundaries. The synthetic positive protein feature samples successfully improve the accuracy of predicting all three domains of GO terms using an SVM classification algorithm. This suggests that the augmented training protein feature samples successfully derive better SVM decision boundaries. We further analysed the changes on the SVM decision boundaries with an example case of predicting the term GO:0034613 by using the original training protein feature samples and the synthetic positive protein feature samples augmented training samples, respectively. The former leads to an MCC value of 0.073, whereas the latter leads to an MCC value of 0.436. We visualize the 2D distributions of both protein sets using their first two principal components, which are also used for training the SVM classifiers for visualizing the corresponding 2D decision boundaries. In Fig. 6a,b, blue dots denote the negative protein samples, while red dots denote the positive protein samples. The white areas in the background denote the decision boundaries separating the blue and red areas where the negative and positive protein samples are distributed. It is clear that the decision boundaries shown in Fig. 6a and Fig. 6b are different. The ones in Fig. 6a suggest that the SVM trained by the original protein samples successfully learned the boundaries that separate the protein samples with different labels in the centre of the figure. However, as shown in Fig. 6c, the boundaries learned by the original training protein sets fail to separate the majority of negative and positive testing protein samples distributing in the right corner of the figure, where the majority of dots are in red. On

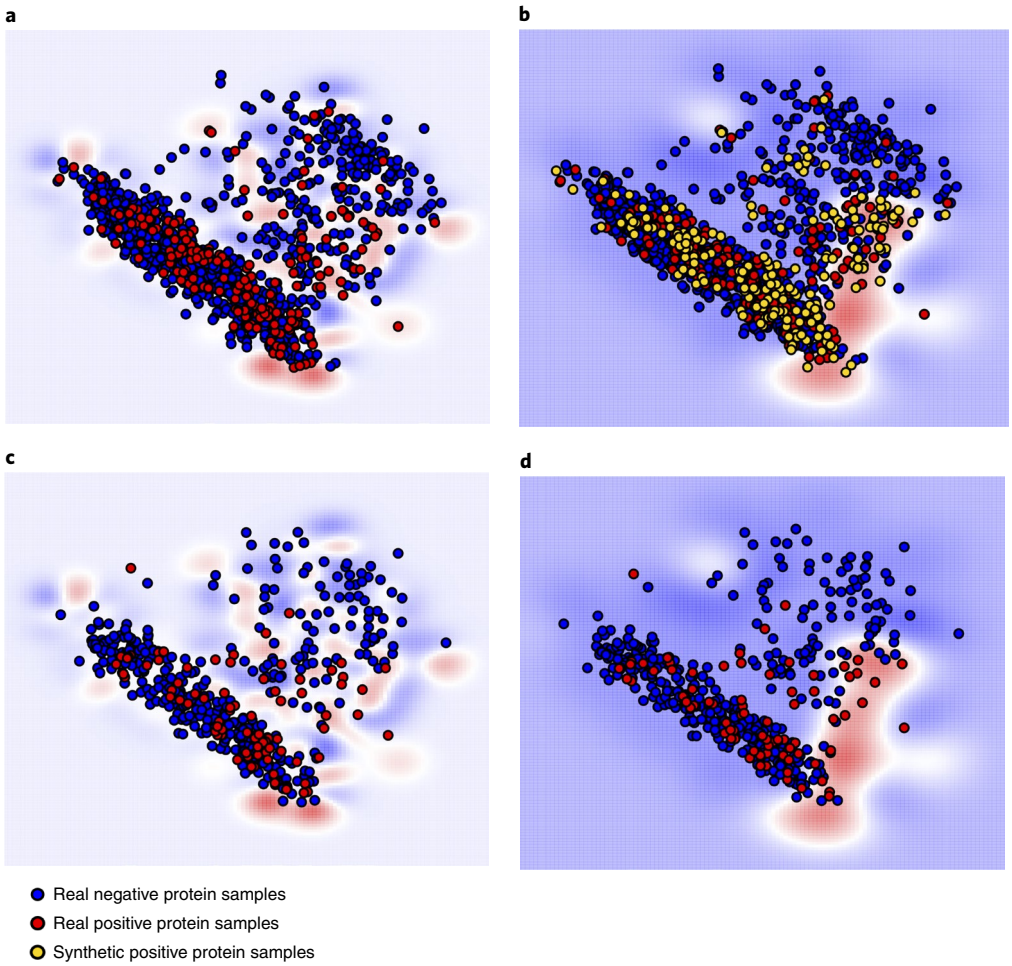


Fig. 6 | 2D visualizations of the learned SVM decision boundaries. a,**b**, The decision boundaries (white areas) learned by the original training protein feature samples (**a**) and the synthetic positive protein feature samples augmented training samples (**b**) for the term GO:0034613. **c**,**d**, The distributions of corresponding testing protein feature samples with those two types of decision boundaries.

the contrary, the SVM trained by the augmented training protein feature samples learned those decision boundaries that successfully separate the protein samples distributed in the right corner of the figure. As shown in Fig. 6d, when applying those decision boundaries on the testing protein feature samples, most of the red and blue dots in the right corner are successfully distinguished, leading to the increased MCC value.

We further explore the reason for the improved predictive accuracy obtained by the augmented training samples by investigating the relationship between the synthetic samples and the real testing samples. We use the corresponding GANs that derived the optimal synthetic samples to generate new sets of synthetic testing samples that are further compared with the corresponding real testing samples by conducting the CTST. In general, as shown in Supplementary Fig. 3, the vast majority of CTST results (that is, the LOOCV accuracy) for individual GO terms range between 0.600 and 0.800, suggesting that the augmented training samples successfully derive improved SVM classifiers that have better generalization ability on classification, rather than due to the issue that the exact distributions of testing samples were observed during the classifier training stage, because the encoded distributions of training samples by GANs are similar but not identical to the distributions of testing samples. This is consistent with the fact that both training and testing samples are randomly sampled with a proportion of 7:3 for individual GO terms.

FFPred-GAN can generate high-quality synthetic feature samples at reasonable computational cost. We now discuss the computational time cost (that is, the actual running time obtained using CPU-based PyTorch with a standard Linux computing cluster) and the training sample sizes (that is, the number of training protein feature samples) for running FFPred-GAN to generate the optimal synthetic protein feature samples for individual GO terms. Extended Data Fig. 4a,b presents boxplots of the distributions of computational time and training samples size, respectively (the complete information is provided in Supplementary Table 4). In general, the computational time for generating optimal synthetic positive protein samples for the majority GO terms from all three individual domains (shown by blue, golden and green boxes) is less than that for generating the optimal negative protein samples (shown by yellow, grey and orange boxes). The corresponding median values are 20,038.6 s (~5.6 h), 24,187.2 s (~6.7 h) and 20,973.8 s (~5.8 h) for generating the optimal positive synthetic protein samples for the BP, MF and CC domains of GO terms, while the median values for generating the optimal negative protein samples are 28,624.6 s (~8.0 h), 29,401.6 s (~8.2 h) and 113,777.8 s (~31.6 h), respectively, for those three domains of GO terms. This fact is relevant, with a pattern such that the training samples sizes of positive proteins for the majority of GO terms are smaller than for the negative ones, as shown in Extended Data Fig. 4b, where the blue, golden and green boxes are located in lower positions than the yellow, grey and orange

boxes. Analogously, the corresponding median values of the sample sizes for those positive protein samples are 226.0, 234.0 and 238.0, respectively, for the BP, MF and CC domains of GO terms, while the median values of samples sizes for those negative protein samples are 873.0, 875.0 and 1,680.0, respectively.

We then also calculate Pearson's correlation coefficient between the computational time and the training samples sizes, as shown by the scatter plots in Extended Data Fig. 4c–h, where the x axes denote sample size and the y axes denote computational time. The correlation coefficient values r for positive protein samples are 0.521, 0.379 and 0.900, respectively, for the BP, MF and CC domains of GO terms, while the negative protein samples have correlation coefficient values of 0.321, 0.349 and 0.140, respectively. The positive and negative protein samples from all three domains of GO terms all show positive correlation between the computational time and training samples size. This indicates that larger sample size leads to longer training time of FFPred-GAN to obtain the optimal synthetic protein samples.

In this work, we have presented a novel generative adversarial networks-based method that successfully generates high-quality synthetic feature samples, which significantly improve the accuracy in predicting all three domains of GO terms through augmenting the original training data. Based on this same framework, there is significant scope to employ new GANs-based architectures, but, more importantly, the same basic approach can be applied to other types of feature used in function prediction, such as proteomics or gene expression data, which are often difficult or expensive to produce in large quantities. Finally, perhaps the most useful benefit of using GANs to augment data is that they can offer a powerful means to balance training sets in the usual situation of having many examples of proteins with one GO term label and very few of others. We hope to explore these applications in the future.

Methods

Generating synthetic protein feature samples with WGAN-GP. WGAN-GPs[11] are a type of GAN[8] that are well known to be highly capable of learning high-dimensional distributions from data samples. In general, conventional GANs are composed of two neural networks—the generator G and the discriminator (a.k.a. critic) D. The former takes random Gaussian noise (a.k.a. the latent variables $z \approx \mathcal{N}(0, 1)$) as inputs to generate outputs that are considered as the synthetic samples. The latter takes the synthetic or real samples as the inputs to distinguish whether they are synthetic or not. To train the GANs, those two networks play a minimax two-player game; that is, the generator aims to generate the synthetic samples as well as possible so as to fool the discriminator, whereas the discriminator aims to distinguish the real and synthetic samples as well as possible, as shown by equation (1) (the minimax objective). Ideally, the GANs are successfully trained when those networks reach the Nash equilibrium; that is, the generator is trained to optimally encode the actual distribution of target samples, while the discriminator is trained to optimally distinguish the real and synthetic samples. Usually, the weights of the generator are updated after several iterations of discriminator training. In essence, this process is equivalent to minimizing the Jensen–Shannon (JS) or Kullback–Leibler (KL) divergences between the target distribution and the one encoded by the generator, given an optimal discriminator.

WGAN[10] is a well-known extension of conventional GANs. It adopts the earth-mover (Wasserstein) distance to replace the JS or KL divergences to avoid the vanishing gradient problem due to their natural limitation on handling non-overlapping distributions. In addition, WGAN adopts the weight clipping mechanism to enforce the 1–Lipschitz constraint for the critic w.r.t. the corresponding inputs. More recently, another extension of GANs has been proposed, namely WGAN-GP, which further improves the training stability of WGAN by adopting a penalty mechanism on the norm of the gradient of the critic. The objective is shown in equation (2), where the left two terms denote the loss of the critic and the right term denotes the gradient penalty term (that is, ensuring the L_2 norm penalty to be around 1.00).

$$\min_G \max_D \mathop{\mathbb{E}}_{x\sim\mathbb{P}_r}[\log(D(x))] + \mathop{\mathbb{E}}_{\tilde{x}\sim\mathbb{P}_g}[\log(1 - D(\tilde{x}))] \quad (1)$$

$$\mathop{\mathbb{E}}_{\tilde{x}\sim\mathbb{P}_g}[D(\tilde{x})] - \mathop{\mathbb{E}}_{x\sim\mathbb{P}_r}[D(x)] + \lambda \mathop{\mathbb{E}}_{\hat{x}\sim\mathbb{P}_{\hat{x}}}[(||\nabla_{\hat{x}} D(\hat{x})||_2 - 1)^2] \quad (2)$$

In this work, we use the generator of well-trained WGAN-GP models to generate synthetic samples. Each WGAN-GP model consists of a three-hidden-layer generator and a three-hidden-layer critic. The generator takes 258 dimensions of random Gaussian noise inputs and outputs 258 dimensions of synthetic samples. The ReLU activation function is adopted for all three hidden layers (of 512 units each) followed by the output layer, which adopts the tanh activation function. The critic network takes 258 dimensions of inputs (that is, the real and synthetic protein feature samples) and uses the leaky ReLU activation function for all layers including three hidden layers (of 86 units each). The Adam optimizer is used for training both generator and critic networks, with a learning rate of 1.00×10^{-4}. The total number of iterations for training the WGAN-GP is 100,000, and the weights of the generator networks are updated after every five iterations of the critic training. The generated synthetic protein feature samples are saved after finishing every 200 iterations of WGAN-GP training for the purpose of downstream quality assessment by using the classifier two-sample tests approach[33].

Selecting optimal synthetic training protein feature samples with the CTST. FFPred-GAN evaluates and selects the optimal synthetic protein feature samples by using the CTST[33] approach. The optimal synthetic protein feature samples are considered as those following the same distribution of the real (training) protein feature samples while not being identical to the real (training) ones. The CTST approach is an extension of conventional single-variable-based statistical significance test methods (for example, the Wilcoxon signed-rank test) to high-dimensional cases. More specifically, given two equal-sized sets of samples respectively following two distributions P and Q, the CTST considers accepting or rejecting a null hypothesis of P being equal to Q. If the null hypothesis is accepted, the classification accuracy on predicting the binary labels of held-out samples will be near the chance level (that is 50.0%). Therefore, in terms of a metric evaluating the quality of generated synthetic samples, a classification accuracy of 100.0% means that the synthetic samples are of poor quality due to the fact that the synthetic samples are significantly different to the real ones.

In this work, we conduct the CTST by using the one-nearest-neighbour classification algorithm due to its simplicity on hyper-parameter tuning. The real and generated synthetic protein feature samples are merged as a union set of protein feature samples assigned binary labels; for example, label 1 for the real samples and label 0 for the synthetic samples. The LOOCV is used to obtain the classification accuracy of the CTST by using different synthetic protein feature samples during 200 iterations of FFPred-GAN training. Finally, the synthetic protein feature samples that obtain the best LOOCV accuracy (that is, closest to 50.0%) are selected as the optimal synthetic feature samples.

Evaluating the predictive power of synthetic protein feature samples generated by FFPred-GAN for augmenting the original training samples. We use the same protein sets as discussed in ref. [5], that is, 10,519 *Drosophila* proteins with 301 GO terms. The protein set for each GO term was further split into training and testing protein sets in the ratio 7:3. A total of 258 dimensions of protein sequence-derived biophysical features (in fact, a mixture of distributions of different feature groups ranging from 11 to 50 dimensions) are used to describe the proteins, including information about the protein secondary structure, intrinsic disorder regions, signal peptides and so on. Full information about these feature groups is provided in Supplementary Table 5. The predictive power of the synthetic protein feature samples is evaluated by three different classification algorithms: SVM, kNN and RF. A fivefold cross-validation-based grid search is used for conducting the hyper-parameter optimization for different classification algorithms. Detailed information about the hyper-parameter searching space is provided in Supplementary Table 6. The classification algorithms and grid search procedure are implemented using Scikit-learn[39]. We also integrate the predictions made by FFPred-GAN/FFPred and NetGO (version 1.0) by using logistic regression with the optimal hyper-parameters obtained by the grid search according to the average precision score. Detailed information about the hyper-parameter searching space is provided in Supplementary Table 7. MCC and AUROC are used to evaluate the predictive performance of FFPred-GAN. As shown in equation (3), the MCC value is calculated by considering the true positive (TP), true negative (TN), false positive (FP) and false negative (FN) rates, and ranges from −1 to 1, with a value of 0 meaning a random prediction and a value of 1 denoting perfect predictive accuracy. The AUROC value is another well-known metric for evaluating the accuracy of a binary classification task. It is calculated by considering the TP and FP rates obtained using different decision thresholds. The AUROC value ranges from 0 to 1, with a value of 0.5 indicating a random prediction and a value of 1.00 denoting perfect predictive accuracy. The MCC value is given by

$$\mathrm{MCC} = \frac{\mathrm{TP} \times \mathrm{TN} - \mathrm{FP} \times \mathrm{FN}}{\sqrt{(\mathrm{TP} + \mathrm{FP})(\mathrm{TP} + \mathrm{FN})(\mathrm{TN} + \mathrm{FP})(\mathrm{TN} + \mathrm{FN})}} \quad (3)$$

We further generate a new set of CAFA 3 targets, which also do not overlap with any protein samples used for training all individual GO term-based classifiers. We first retrieve the latest experimental annotations (identified with evidence codes EXP, IDA, IPI, IMP, IGI and IEP) for the CAFA 3 *Drosophila melanogaster* targets from UniProtKB/SwissProt (release 2019_12) and the experimentally validated annotations produced by the recent CAFA 3 competition. Those annotated targets are then further filtered if they were included in the protein samples used for training the GO term-based classifiers, leading to a new set of CAFA 3 targets

consisting of 360 proteins in total. To evaluate the performance of the integration approach based on the predictions made by FFPred-GAN/FFPred and NetGO, a 10-fold cross-validation is conducted by using the new set of CAFA 3 targets, that is, 260, 191 and 231 targets for the BP, MF and CC domains of GO terms, respectively.

The F_{max} score is used to evaluate the performance of the prediction methods on this new set of CAFA 3 targets. As shown in equations (4) to (6), F_{max} is calculated using Pr_τ and Rc_τ. The former is calculated as the total amount of precision values obtained by predicting all S protein sequence GO term annotations according to the decision threshold τ, divided by m protein sequences with at least one GO term annotation's predictive posterior probability being greater than or equal to the τ. Analogously, the Rc_τ value is calculated as the total amount of recall values obtained by predicting all S protein sequence GO term annotations divided by the total n protein sequences:

$$F_{max} = \max_\tau\left\{\frac{2Pr_\tau \times Rc_\tau}{Pr_\tau + Rc_\tau}\right\} \quad (4)$$

$$Pr_\tau = \frac{1}{m_\tau}\sum_{s=1}^{m_\tau}\frac{TP_{s,\tau}}{TP_{s,\tau} + FP_{s,\tau}} \quad (5)$$

$$Rc_\tau = \frac{1}{n}\sum_{s=1}^{n}\frac{TP_{s,\tau}}{TP_{s,\tau} + FN_{s,\tau}} \quad (6)$$

Data availability

All data can be downloaded via http://bioinfadmin.cs.ucl.ac.uk/downloads/FFPredGAN.

Code availability

The source code can be accessed via https://github.com/psipred/FFPredGAN.

Received: 22 October 2019; Accepted: 23 July 2020;
Published online: 31 August 2020

References

1. Cozzetto, D. & Jones, D. T. Computational methods for annotation transfers from sequence. *Gene Ontol. Handb.* **1446**, 55–67 (2017).
2. Radivojac, P. et al. A large-scale evaluation of computational protein function prediction. *Nat. Methods* **10**, 221–227 (2013).
3. Jiang, Y. et al. An expanded evaluation of protein function prediction methods shows an improvement in accuracy. *Genome Biol.* **17**, 184 (2016).
4. Zhou, N. et al. The CAFA challenge reports improved protein function prediction and new functional annotations for hundreds of genes through experimental screens. *Genome Biol.* **20**, 244 (2019).
5. Wan, C., Lees, J. G., Minneci, F., Orengo, C. A. & Jones, D. T. Analysis of temporal transcription expression profiles reveal links between protein function and developmental stages of *Drosophila melanogaster*. *PLoS Comput. Biol.* **13**, e1005791 (2017).
6. Fa, R., Cozzetto, D., Wan, C. & Jones, D. T. Predicting human protein function with multi-task deep neural networks. *PLoS ONE* **13**, e0198216 (2018).
7. Wan, C., Cozzetto, D., Fa, R. & Jones, D. T. Using deep maxout neural networks to improve the accuracy of function prediction from protein interaction networks. *PLoS ONE* **14**, e0209958 (2019).
8. Goodfellow, I. J. et al. Generative Adversarial Nets. In *Advances in Neural Information Processing Systems* (eds Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D. & Weinberger, K. Q.) Vol. 27, 2672–2680 (Curran Associates, 2014).
9. Radford, A., Metz, L. & Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. Preprint at https://arxiv.org/abs/1511.06434 (2015).
10. Arjovsky, M., Chintala, S. & Bottou, L. Wasserstein GAN. In *Proceedings of the 34th International Conference on Machine Learning* (PMLR, 2017).
11. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V. & Courville, A. Improved Training of Wasserstein GANs. In *Advances in Neural Information Processing Systems* (eds Guyon, I. et al.) Vol. 30, 5767–5777 (Curran Associates, 2017).
12. Mao, X. et al. Least squares generative adversarial networks. In *2017 IEEE International Conference on Computer Vision* (*ICCV*) 2813–2821 (IEEE, 2017).
13. Chen, X. et al. InfoGAN: interpretable representation learning by information maximizing generative adversarial nets. In *Advances in Neural Information Processing Systems* (eds Lee, D. D., Sugiyama, M., Luxburg, U. V., Guyon, I. & Garnett, R.) Vol. 29, 2172–2180 (Curran Associates, 2016).
14. Zhu, J.-Y., Park, T., Isola, P. & Efros, A. A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *2017 IEEE International Conference on Computer Vision* (*ICCV*) 2223–2232 (IEEE, 2017).
15. Isola, P., Zhu, J.-Y., Zhou, T. & Efros, A. A. Image-to-image translation with conditional adversarial networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*) 1125–1134 (IEEE, 2017).
16. Choi, Y. et al. StarGAN: unified generative adversarial networks for multi-domain image-to-image translation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*) 8789–8797 (IEEE, 2018).
17. Souly, N., Spampinato, C. & Shah, M. Semi supervised semantic segmentation using generative adversarial network. In *2017 IEEE International Conference on Computer Vision* (*ICCV*) 5688–5696 (IEEE, 2017).
18. Zhang, Z., Yang, L. & Zheng, Y. Translating and segmenting multimodal medical volumes with cycle- and shape-consistency Generative Adversarial Network. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*) 9242–9251 (IEEE, 2018).
19. Zhu, W., Xiang, X., Tran, T. D., Hager, G. D. & Xie, X. Adversarial deep structured nets for mass segmentation from mammograms. In *2018 IEEE 15th International Symposium on Biomedical Imaging* 847–850 (IEEE, 2018).
20. Ledig, C. et al. Photo-realistic single image super-resolution using a generative adversarial network. In *2017 IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*) 4681–4690 (IEEE, 2017).
21. Yang, G. et al. DAGAN: Deep de-aliasing generative adversarial networks for fast compressed sensing MRI reconstruction. *IEEE Trans. Med. Imaging* **37**, 1310–1321 (2017).
22. Seeliger, K., Güçlü, U., Ambrogioni, L., Güçlütürk, Y. & van Gerven, M. Generative adversarial networks for reconstructing natural images from brain activity. *NeuroImage* **181**, 775–785 (2018).
23. Wang, X., Dizaji, K. G. & Huang, H. Conditional generative adversarial network for gene expression inference. *Bioinformatics* **34**, i603–i611 (2018).
24. Dizaji, K. G., Wang, X. & Huang, H. Semi-supervised generative adversarial network for gene expression inference. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 1435–1444 (ACM, 2018).
25. Ghahramani, A., Watt, F. M. & Luscombe, N. M. Generative adversarial networks simulate gene expression and predict perturbations in single cells. Preprint at *BioRxiv* https://www.biorxiv.org/content/10.1101/262501v2 (2018).
26. Gupta, A. & Zou, J. Feedback GAN for DNA optimizes protein functions. *Nat. Mach. Intell.* **1**, 105–111 (2019).
27. Wang, Y. et al. Synthetic promoter design in *Escherichia coli* based on a deep generative network. *Nucl. Acids Res.* **48**, 6403–6412 (2020).
28. Frid-Adar, M. et al. GAN-based synthetic medical image augmentation for increased CNN performance in liver lesion classification. *Neurocomputing* **321**, 321–331 (2018).
29. Zhu, X., Liu, Y., Li, J., Wan, T. & Qin, Z. Emotion classification with data augmentation using generative adversarial networks. In *Proceedings of the 22nd Pacific-Asia Conference on Knowledge Discovery and Data Mining* (*PAKDD 2018*) (eds Phung, D. et al.) 349–360 (Springer, 2018).
30. Volpi, R., Morerio, P., Savarese, S. & Murino, V. Adversarial feature augmentation for unsupervised domain adaptation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*) 5495–5504 (IEEE, 2018).
31. Marouf, M. et al. Realistic in silico generation and augmentation of single-cell RNA-seq data using generative adversarial networks. *Nat. Commun.* **11**, 166 (2020).
32. Minneci, F., Piovesan, D., Cozzetto, D. & Jones, D. T. FFPred 2.0: improved homology-independent prediction of gene ontology terms for eukaryotic protein sequences. *PLoS ONE* **8**, e63754 (2013).
33. Lopez-Paz, D. & Oquab, M. Revisiting classifier two-sample tests. In *Proceedings of the International Conference on Learning Representations* (ICLR, 2017).
34. Cao, R. et al. ProLanGO: protein function prediction using neural machine translation based on a recurrent neural network. *Molecules* **22**, E1732 (2017).
35. Chawla, N. V., Bowyer, K. W., Hall, L. O. & Kegelmeyer, W. P. SMOTE: synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **16**, 321–357 (2002).
36. Lemaître, G., Nogueira, F. & Aridas, C. K. Imbalanced-learn: a Python toolbox to tackle the curse of imbalanced datasets in machine learning. *J. Mach. Learn. Res.* **18**, 1–5 (2017).
37. You, R. et al. NetGO: improving large-scale protein function prediction with massive network information. *Nucleic Acids Res.* **47**, W379–W387 (2019).
38. You, R. et al. GOLabeler: improving sequence-based large-scale protein function prediction by learning to rank. *Bioinformatics* **34**, 2465–2473 (2018).
39. Pedregosa, F. et al. Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).

Acknowledgements

We thank the members of the UCL Bioinformatics Group for valuable discussions. We also acknowledge the support of the high-performance computing facility of the Department of Computer Science at University College London. C.W. and D.T.J. acknowledge funding from the Biotechnology and Biological Sciences Research Council (BB/L002817/1) and the European Research Council Advanced

Grant 'ProCovar' (Project ID 695558). This work was supported by the Francis Crick Institute, which receives its core funding from Cancer Research UK (FC001002), the UK Medical Research Council (FC001002) and the Wellcome Trust (FC001002).

Author contributions

C.W. and D.T.J. conceived the idea. C.W. implemented the FFPred-GAN system and carried out the experiments. C.W. and D.T.J. analysed the experimental results and wrote the manuscript.

Competing interests

The authors declare no competing interests.

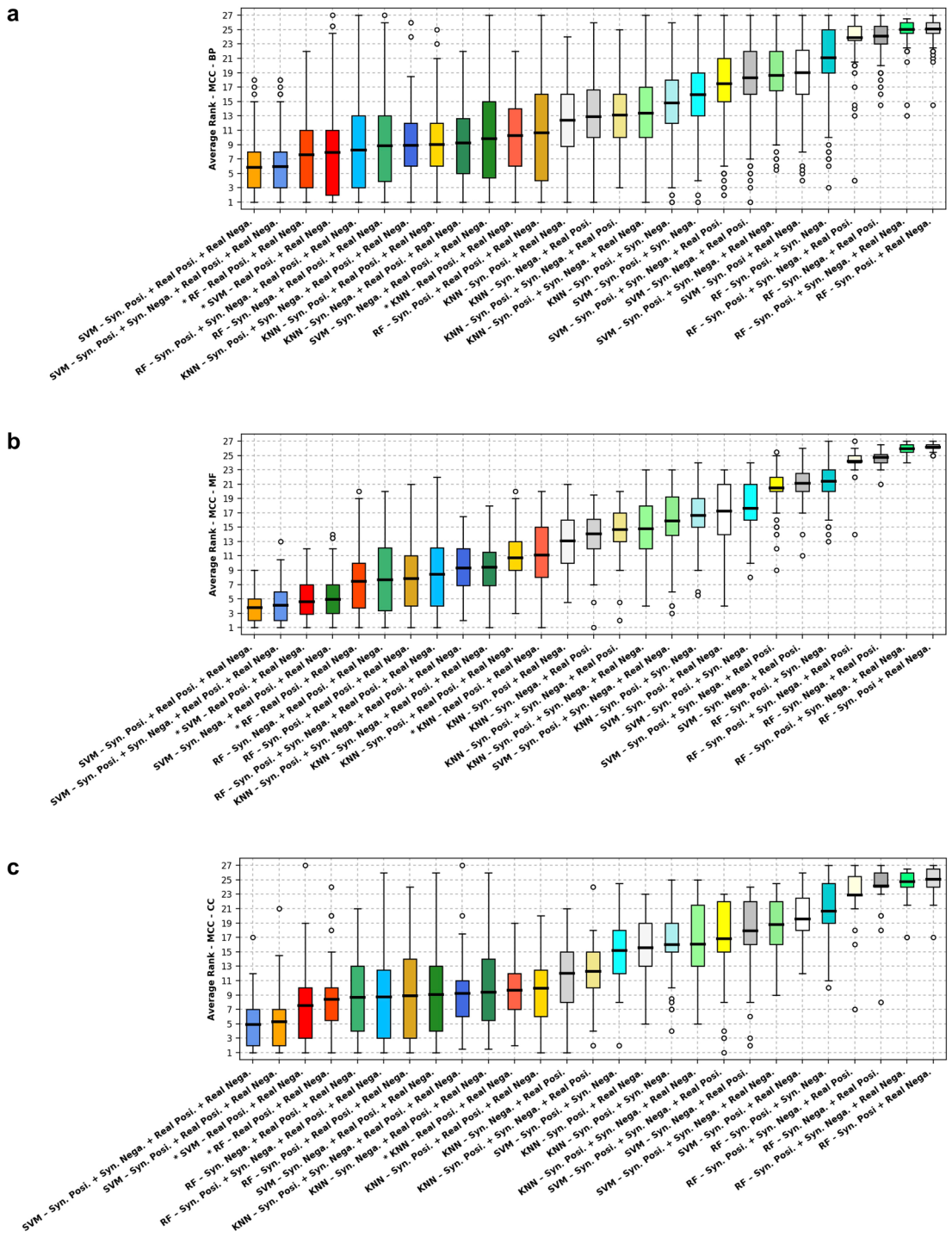
Additional information

Supplementary information is available for this paper at https://doi.org/10.1038/s42256-020-0222-1.

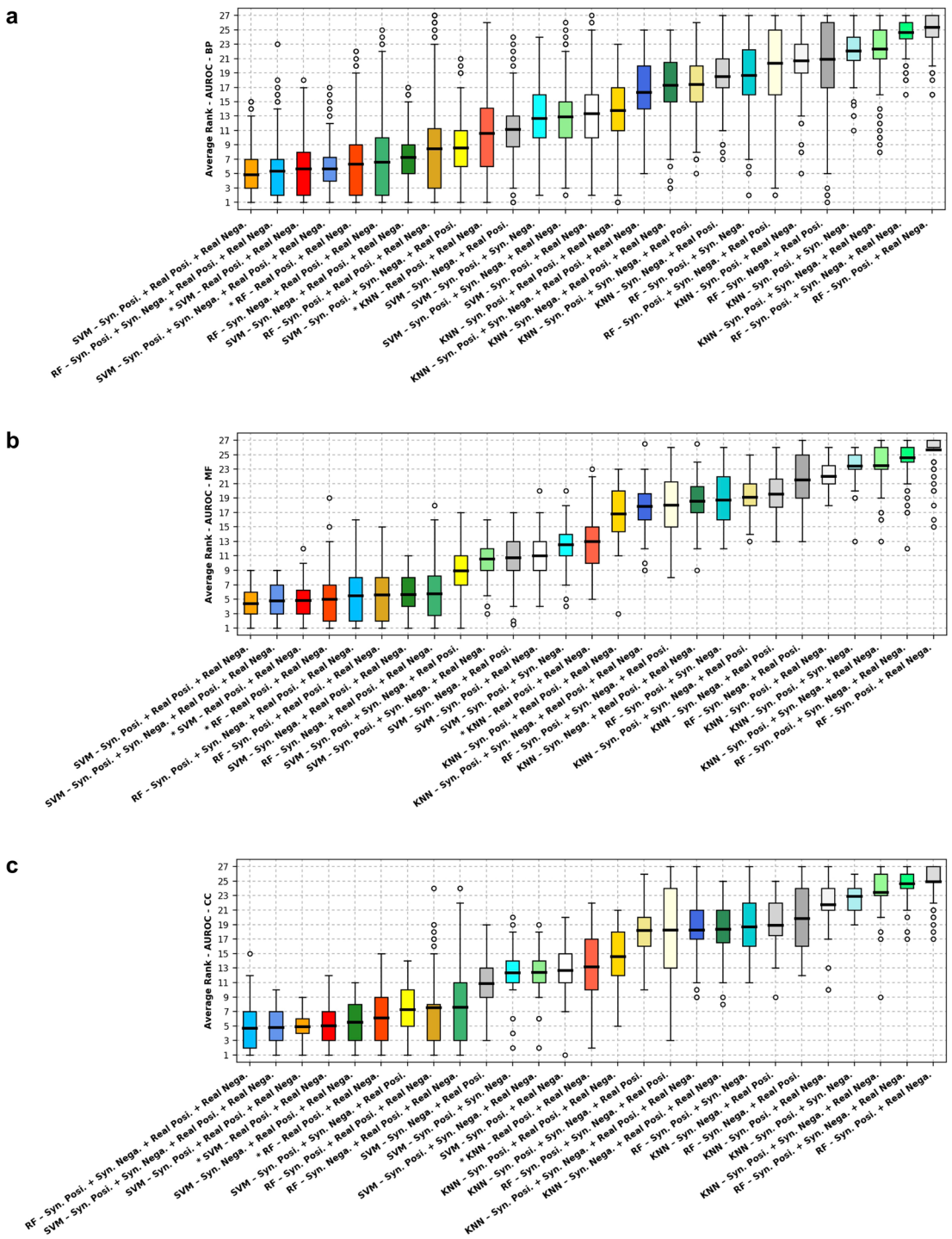
Correspondence and requests for materials should be addressed to D.T.J.

Reprints and permissions information is available at www.nature.com/reprints.

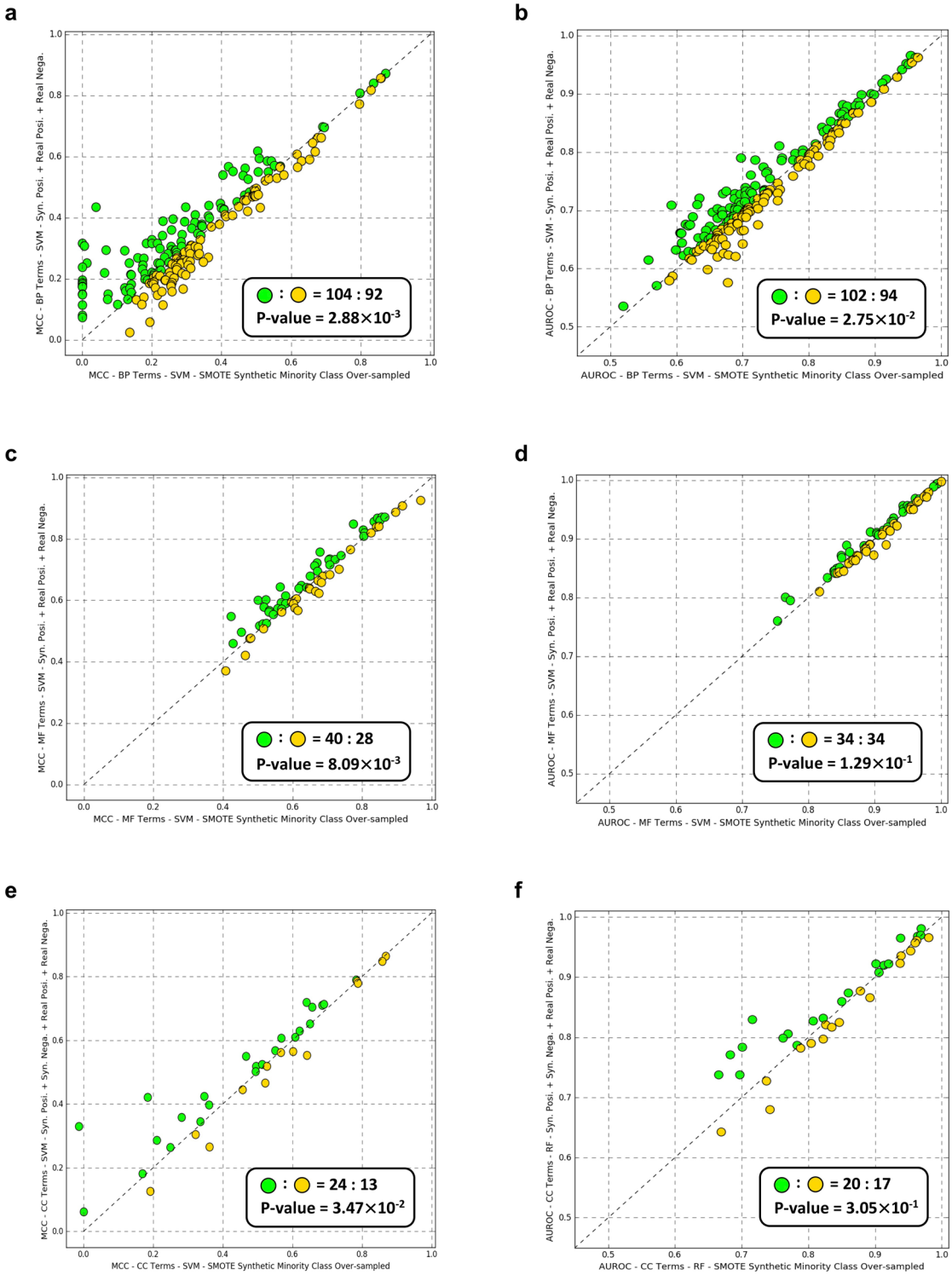
Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



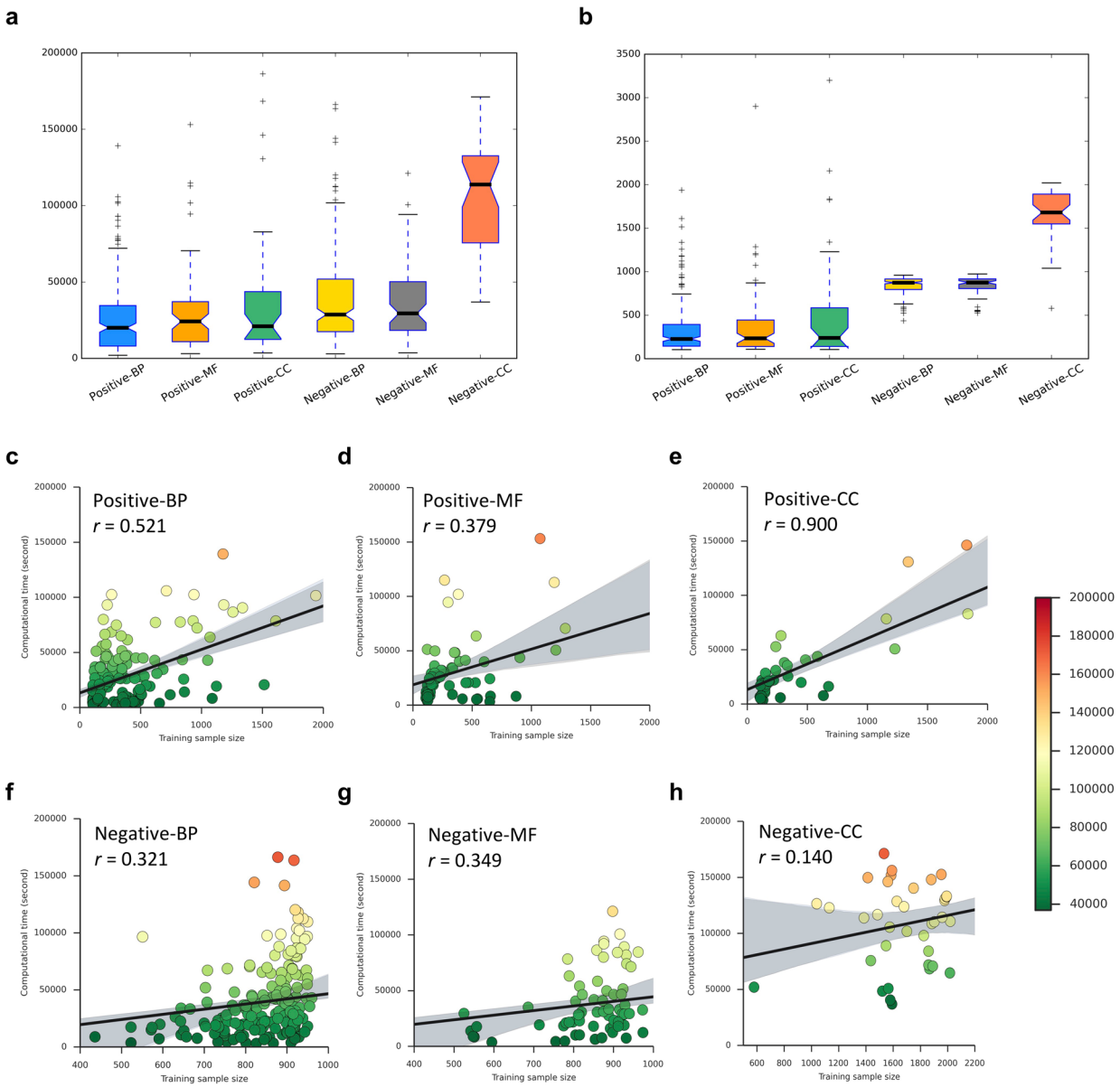
Extended Data Fig. 1 | The boxplot about the rankings of MCC values. a-c, The rankings of MCC values obtained by different combinations of synthetic and real protein samples and three different classification algorithms for predicting biological process (**a**), molecular function (**b**) and cellular component (**c**) domains of protein functions.



Extended Data Fig. 2 | The boxplot about the rankings of AUROC values. a-c, The rankings of AUROC values obtained by different combinations of synthetic and real protein samples and three different classification algorithms for predicting biological process (**a**), molecular function (**b**) and cellular component (**c**) domains of protein functions.



Extended Data Fig. 3 | The comparison of predictive accuracy obtained by the FFPred-GAN augmented training samples and the SMOTE augmented training samples. a-**f**, The scatter-plots about the MCC and AUROC values obtained by the FFPred-GAN augmented training samples and the SMOTE augmented training samples for predicting three domains of GO terms by using SVM (**a**-**e**) and RF (**f**) classification algorithms.



Extended Data Fig. 4 | Characteristics about the computational time and sample size. a, The boxplot about the distributions of computational time on obtaining the optimal synthetic protein samples for different GO terms; **b**, The boxplot about the distributions of sample sizes for different GO terms; c-h, The scatter-plots of correlation coefficient values between the computational time and sample sizes for positive and negative protein samples of different domains of GO terms.