Supplementary Materials

Develop machine learning based predictive models for engineering protein solubility

Prediction of protein solubility

1 SVM tuning

Different kernels and three parameters cost, gamma and epsilon in SVM were tuned. Cost is the regularization parameter that controls the trade-off between achieving a low training error and a low testing error. Cost can also be interpreted as the extent we penalize the SVM when data points lie within the dividing hyperplanes. Larger cost means fewer points are within the dividing hyperplanes and the space between two dividing hyperplanes is small, which also indicates that the unseen points are difficult to be separated because the data points belonging to two categories are too close to each other. On the contrary, lower cost gives larger margin but also higher error for training data. Epsilon is a margin of tolerance where no penalty is given to errors. Larger epsilon means less penalty to errors. Gamma is a parameter in the function of radial basis kernel, which indicates the threshold to determine if two points are considered to be similar. Gamma controls the standard deviation of the Gaussian function of radial basis kernel.

Table S1. Performance of SVM for different kernels

Kernel	Accuracy	R^2
linear	0.6684	0.2240
sigmoid	0.4528	0.0011
radial basis	0.7500	0.4064
polynomial	0.6339	0.0732

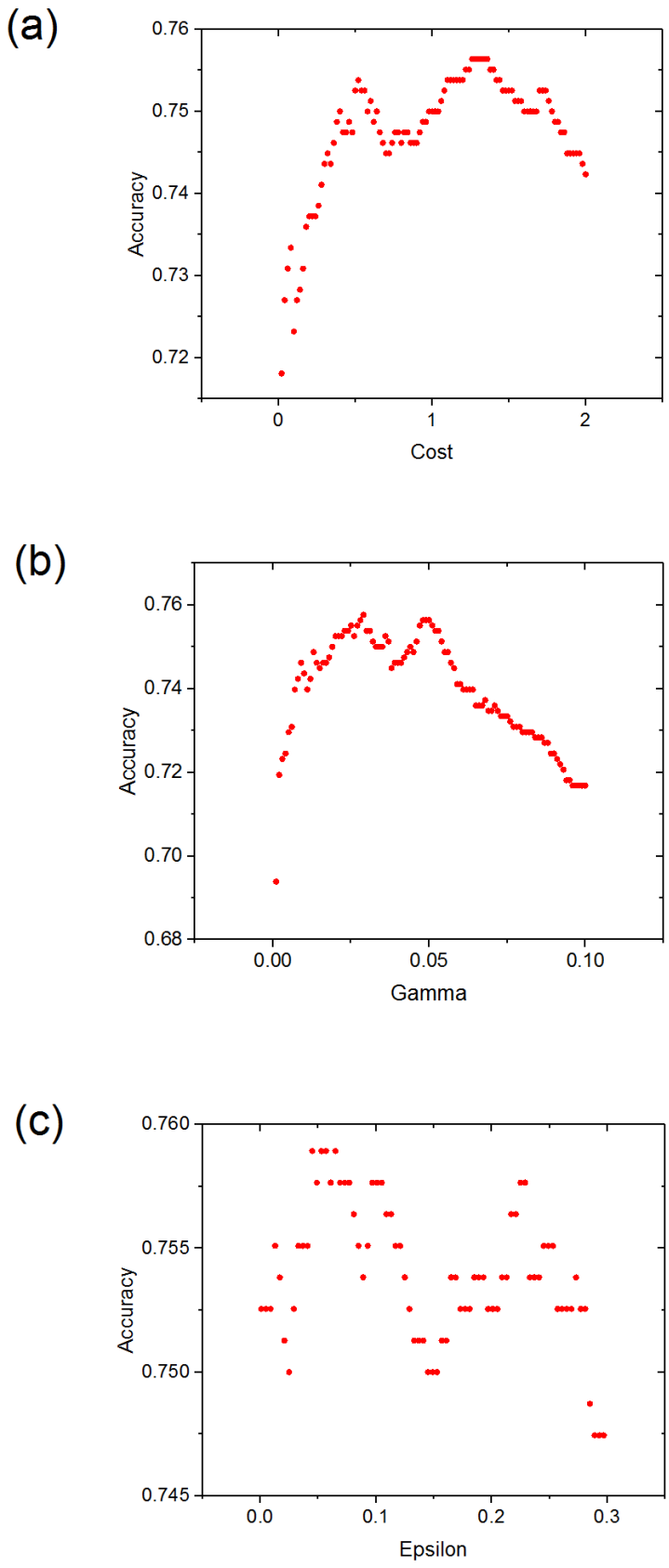


Fig. S1. Plot between accuracy of SVM and cost, gamma and epsilon respectively.

2 Data augmentation

We used 75% of the original data (2364 proteins) as training data to train GANs, which then generated the same size of artificial data (2364 proteins, including features used as model inputs solubility). We combined the artificial data with the 75% of the original data to train a SVM model, which was validated by using the 25% of original data that were not used.

In the first attempt, four versions of GANs were trained for 500 iterations to generate the artificial data, which were used together with the original data to train ML models by using SVM and R^2 was calculated by using the remaining validation data (Table S2). Figures of protein sequence after dimensionality reduction by Principal Component Analysis (PCA) and protein solubility are shown in Fig. S2 and Fig. S3. Different colours represent different values of solubility from 0-1. If we analyze quality of the generated data, GANs have better performance than CGAN according to the comparison between original data and generated data from GANs and CGAN (Fig. S2). Both CGAN and WCGAN are conditional versions of GANs, which may perform well for data grouped into classes, such as studies using binary values.

In the second attempt, we trained GANs for 5000 iterations. At the end of each 100 iterations, the generated data was used together with the original data to develop a SVM model. There are 50 values of R^2 corresponding to 50 sets of exported generated data when we run 5000 iterations. The highest R^2 among the 50 values was recorded for each training dataset (Table S3). In Table S3, suffix 1, 2 and 3 represent three randomly selected training datasets and for each training dataset, comparison of SVM model performance between original data and data including generated data was conducted.

Table S2. Performance of SVM based on data generated from different data augmentation algorithms

GAN version	R^2
No GAN	0.4092
GANs	0.4015
CGAN	0.4064
WGAN	0.3787
WCGAN	0.4003

Table S3. Performance of SVM based on GANs for 5000 iterations

GANs version	R^2
No GANs-1	0.4093
GANs-1	0.4044
No GANs-2	0.4200
GANs-2	0.4240
No GANs-3	0.4447
GANs-3	0.4462

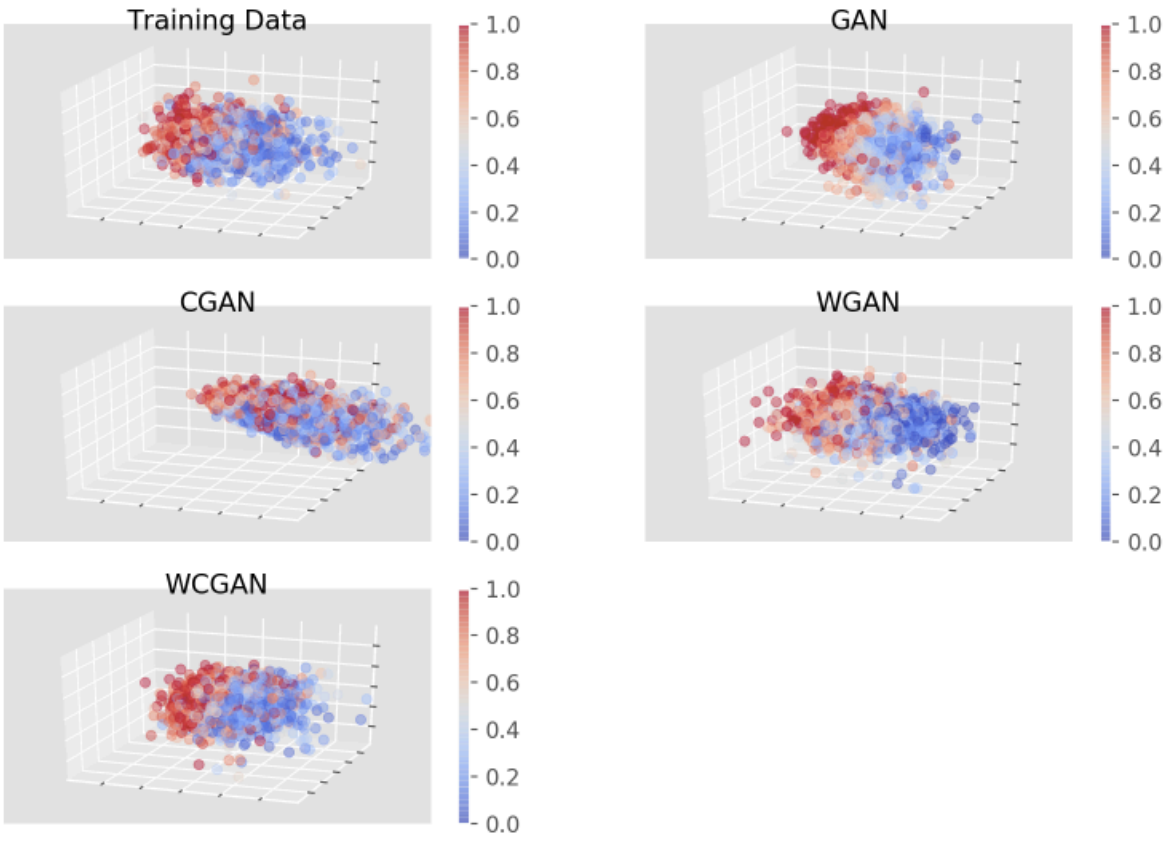


Fig. S2. Original data and generated data by different versions of GANs for 500 iterations.

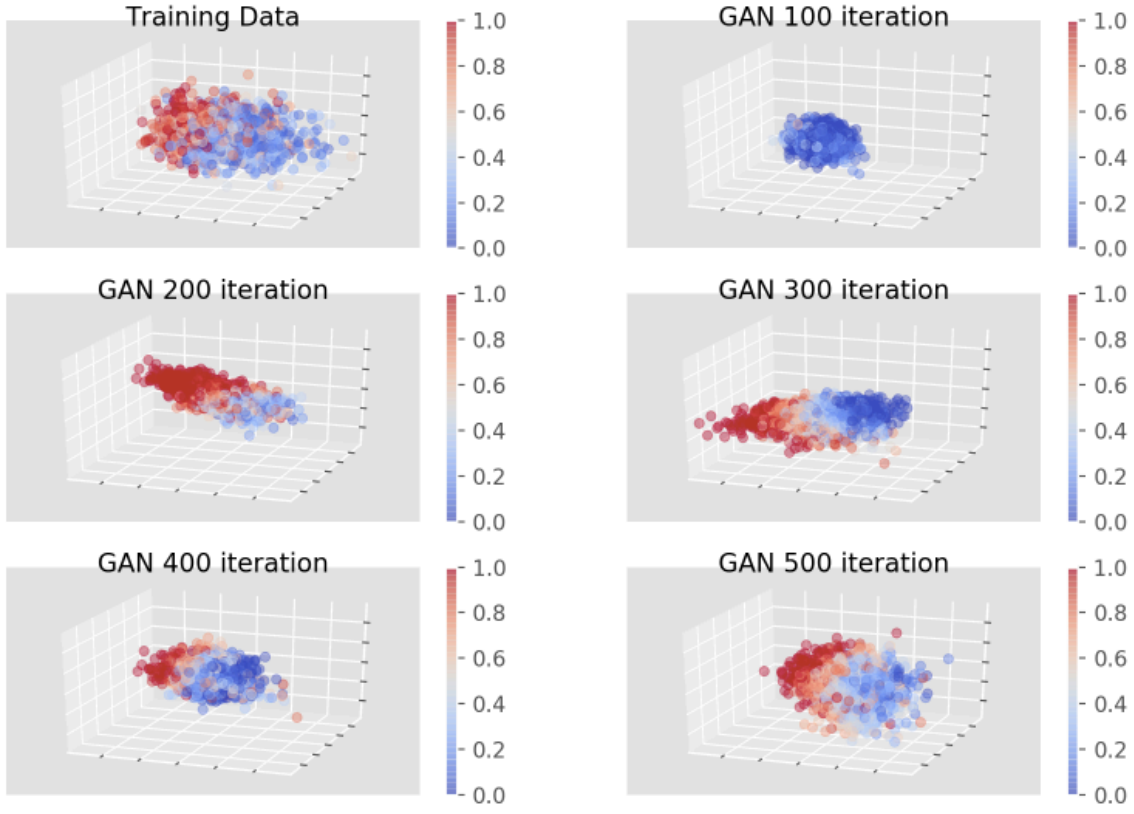


Fig. S3. Original data and generated data by GANs model in different steps for 500 iterations

Prediction of protein yield

1 Data preprocessing

We used the same dataset for protein solubility, eSol database, which also provides values of protein yield. Similar to the prediction of protein solubility, only proteins with reliable sequence information (3147 proteins) were selected from the 3173 items in the original database. After data pre-processing, 99 outliers were removed from 3147 proteins according to the 1.5 times IQR method. The training workflow, machine learning models and evaluation metrics used for prediction of protein yield are the same to the protein solubility study.
In yield dataset, two methods were applied to remove outliers. One widely used method is to remove any data points, which were more than 1.5 interquartile ranges (IQRs) below the first quartile or above the third quartile. Another approach is to remove any data points that are beyond 3 standard deviations from the mean. It can be observed that the IQR approach (R^2=0.0759) was better than the standard deviation approach (R^2=0.0622).

Table S4. Statistical description of yield of proteins

Data		Remove outliers by 1.5 times IQR	Remove outliers by 3 times standard deviation
Summary	Min.	0.0000	0.0000
	Median	0.1529	0.3108
	Mean	0.2706	0.3609
	Max.	1.0000	1.0000
R^2		0.0759	0.0622
Outliers removed		99	164

2 Performance of different ML models and different descriptors

We compared the performance of different ML models for predicting protein yield from amino acid sequence. And the tuning process of parameters was conducted for the best model SVM. Radial basis kernel (with 0.2, 0.027 and 0.241 as cost, gamma and epsilon respectively) was used to develop SVM model using extractAAC descriptors. In addition, after another descriptor extractAPAAC was used, the SVM model was tuned again.

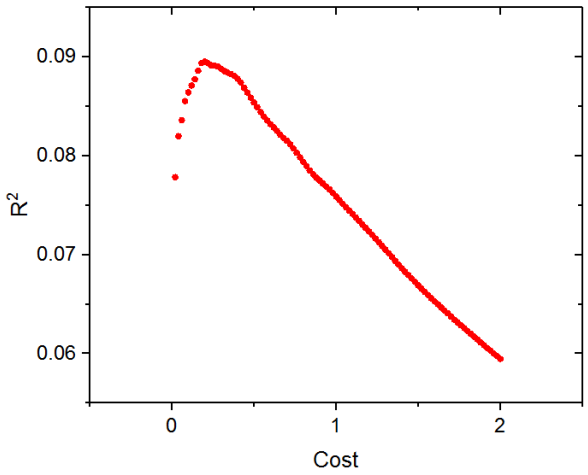
Table S5. Performance of SVM for yield of proteins after removing outliers using 1.5 times IQR

Model	R^2
SVM	0.1163
Logistic regression	0.0613
Decision tree	0.0357
Naïve Bayes	0.0609
cforest	0.0851
XGBoost	0.0723
ANNs	0.0140

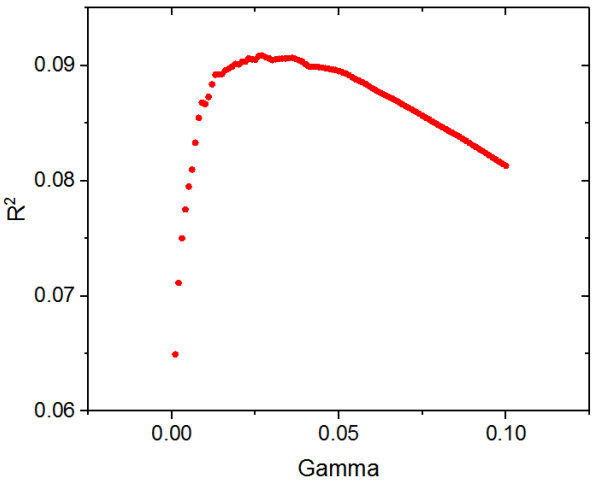
Table S6. Performance of different kernels in SVM using extractAAC descriptor

Kernel	R^2
polynomial	0.0156
radial basis	0.0759
linear	0.0624
sigmoid	0.0004

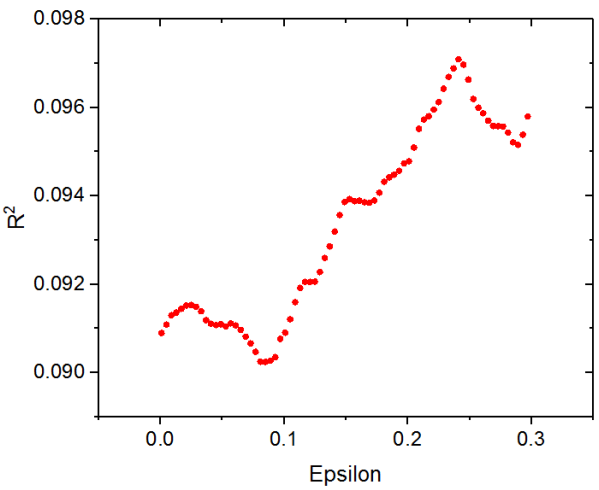
(a)


0.09
0.08
0.07
0.06
R2
0
1
2
Cost

(b)


0.09
0.08
0.07
0.06
R2
0.00
0.05
0.10
Gamma

(c)


0.098
0.096
0.094
0.092
0.090
R2
0.0
0.1
0.2
0.3
Epsilon

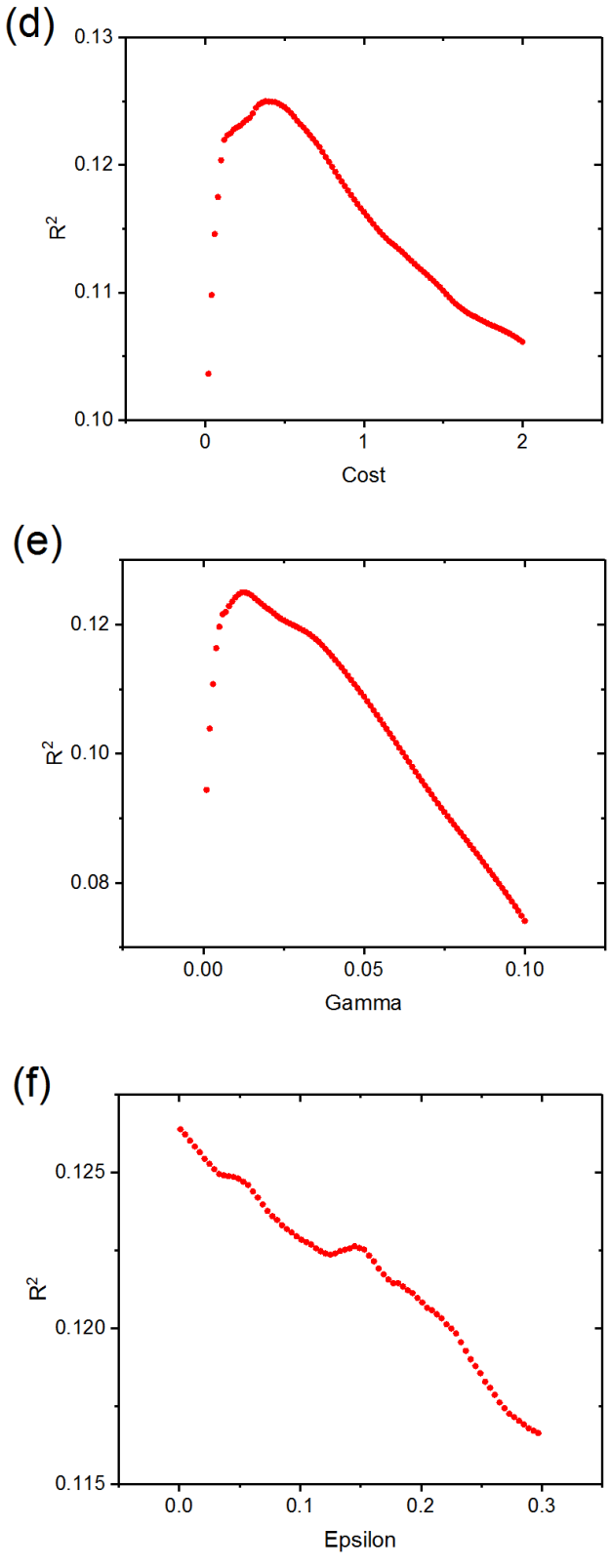


Fig. S4. Performance of different cost, gamma and epsilon in SVM respectively for extractAAC descriptor (a, b, c) and extractAPAAC (d, e, f)

Table S7. Performance of prediction of protein yield using different sequence descriptors by SVM

Descriptors	R^2	Extracted column
extractAAC	0.0971	20
extractPAAC	0.1157	50
extractAPAAC (after optimization)	0.1264	80
extractAPAAC	0.1163	80
extractMoreauBroto	0.0162	240
extractMoran	0.0030	240
extractGeary	0.0041	240
extractCTDC		21
extractCTDT	0.0832*	21
extractCTDD		105
extractCTriad	0.0190	343
extractSOCN	0.0798	80
extractQSO	0.0830	100

* The R^2 is the result combining extractCTDC, extractCTDT and extractCTDD