2021 年臺灣國際科學展覽會
優勝作品專輯

作品編號	190006
參展科別	電腦科學與資訊工程
作品名稱	Predicting the Binding Affinity between Medicine and Estrogen Receptor Beta

就讀學校　宜蘭縣私立慧燈高級中學

指導教師　林振慶、楊岱霖

作者姓名　蔡孟學

關鍵詞　Machine learning、Breast cancer、Estrogen Receptor Beta

作者簡介

I am a 10th grade student in Huey Deng High School, I have been interested in programming at an early age. I am thankful to my teachers, family members, and friends who supported me down this path, they are the ones who gave me the strength to move on. This is my first personal research project, I completed it stumbling down a narrow path littered with sharp stones. I may not be the one who had the luck to taste the sweetness of victory at the end, but I think the most valuable part of a research project isn’ t the final outcome, but the happiness of learning new knowledge in the process.

摘要

在 104 年衛福部發布的研究分析中指出，台灣女性罹患乳癌的機率在過去 30 年大幅攀升。我們將面臨的是更多、更年輕的乳癌患者。讓狀況更加嚴峻的是，患者在接受癌症治療時不但要忍受化療的副作用，還有一定的機率喪失生育能力。我們希望能開發一個程式來協助研究人員及醫生測試的乳癌治療藥物。藥物治療癌症的方式是透過與蛋白質進行連接，我們希望透過蒐集其 SMILES 化學式資料，轉換為 MACCS 形式，進而構成深度學習程式的訓練資料。因為資料筆數較少，所以採用了 ensemble learning 的方法，製作多個 weak model，然後再透過 ensemble voting 將其合併並投票，產出準確的預測結果。我們為了尋找 positive training data 與 negative training data 之間的比例，嘗試了四十種不同比例，在分析結果後發現兩者比為 1:3000 時，weak model 的成效最好。我們在分析模型及測試結果時使用了 precision、recall 及 AUC 來進行分析，結果顯示我們的模型有足夠的穩定性並能進行準確預測。

Abstract

Recent studies showed that the probability of Taiwanese females developing breast cancer has risen dramatically over the past 30 years. We are now facing younger and more breast cancer patients in Taiwan. What makes the matter even more severe, is the fact that patients that take cancer treating medicine will suffer from its serious side effects, some may even lose the ability to reproduce. We hope to develop a new system that can help doctors and researchers develop new medicine for treating breast cancer, the way medicine cures cancer tumors are by attaching onto the infected cells' receptors. After collecting MACCS data (converted from SMILES), the dataset will be used for training the machine learning program. Due to the problem of insufficient training data, we used an ensemble method to generate our machine learning model. Among the three basic ensemble techniques, Max Voting, Averaging, and Weighted Averaging. we selected the max voting technique to perform the prediction for this research. We created two separate datasets, positive and negative, the two datasets will later be used as training data for the program. We weren't sure of the ratio of positive and negative in the training data, therefore we compare 40 different ratios and evaluate the results. By comparing the accuracy of the models, we found out that when the ratio between positive data and negative data is 1:3000, the machine learning program will have the highest precision. After we created the final model through voting among the 1000 models generated, we evaluate the precision of the model through the following methods, AUC, precision, recall. The ultimate goal of this research is to assist doctors and researchers shorten the process of developing and testing new medicines.

I、Motivation

Since the year 2006, breast cancer has become the most commonly developed type of cancer among Taiwanese females. In the year 2015 alone, there were more than 12000 newly diagnosed breast cancer cases. The probability of developing breast cancer in 1980 was 11.72 people in 100000 people, and the probability of developing breast cancer in 2015 is 104.92 in every 100000 people. The data implies that the chance of developing breast cancer has risen significantly in the past 30 years. In addition to this, the average age of Taiwanese females diagnosed with breast cancer is 10 years younger than females in European countries and the USA. There is also the consideration of the serious side effects of the drugs used in treating breast cancer, therefore, Taiwan is now in a hurry to find a solution to this mounting problem. Cancer tumors have certain receptors, the receptors can be used to connect with medicines, however, most medicines cannot fully match the receptor of the tumor, which causes other healthy cells and tissues to be affected. We hope to find a way to help doctors calculate the binding affinity between receptors and medicine. Thus, finding the best medicine for treating a particular type of cancer. Ultimately helping patients stay healthy and survive the challenge of fighting cancer more easily.

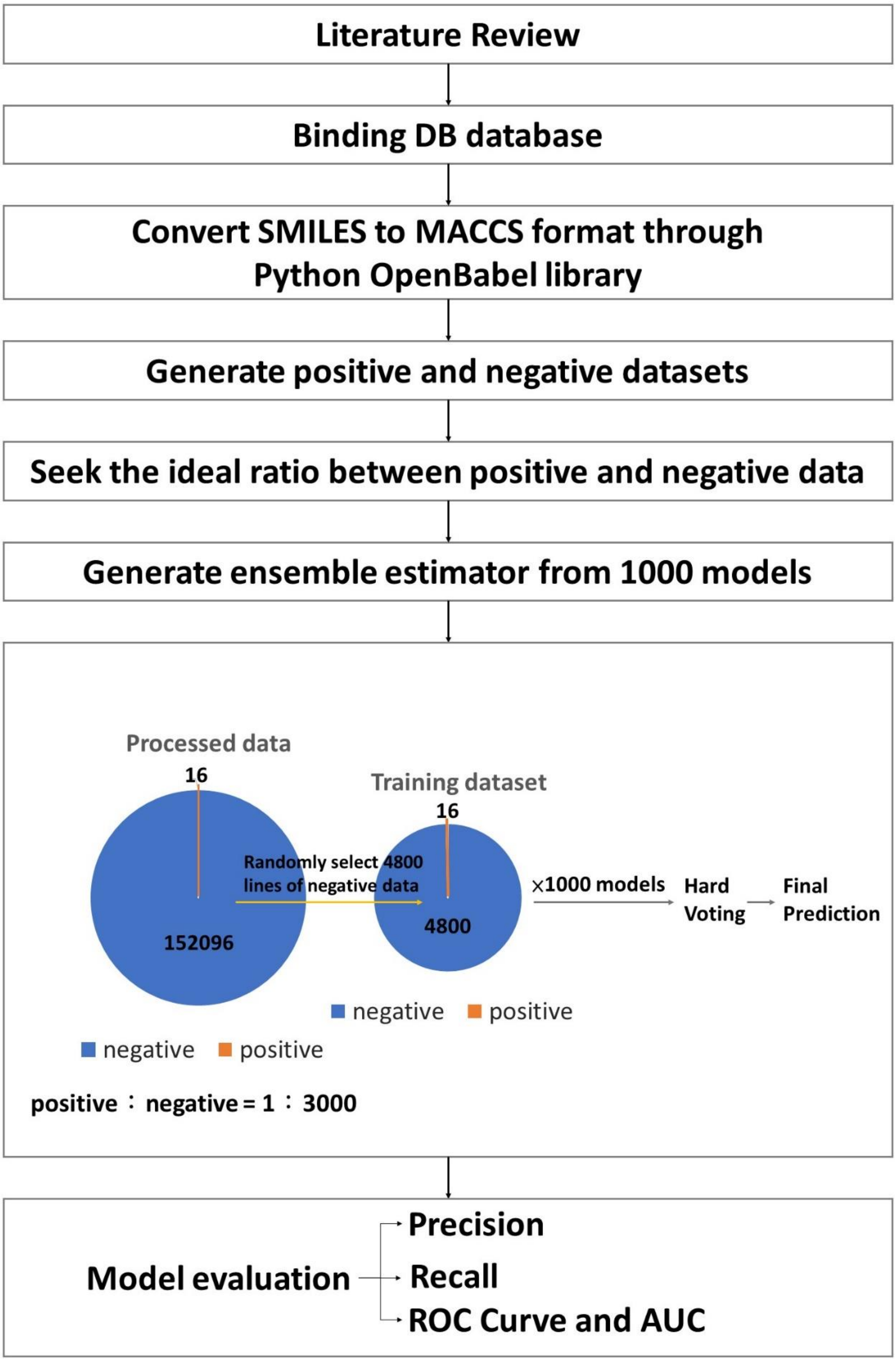
II、Research Purposes

- Predict the binding affinity of medicine and breast cancer tumors.
- Discover medicines with the potential of curing breast cancer
- Generate an effective model from a limited amount of training data
- Shorten research process on breast cancer related medicine

III、Research Equipment

1. Laptop (CPU: R9 4900H with Radeon Graphics, memory: 16GB, disk: 1TB)
2. Desktop (CPU: AMD Ryzen Threadripper 3970X 32-Core Processor, memory: 256GB, disk: 17TB)
3. Programming language: Python

IV、Research Process



Literature Review
Binding DB database
Convert SMILES to MACCS format through Python OpenBabel library
Generate positive and negative datasets
Seek the ideal ratio between positive and negative data
Generate ensemble estimator from 1000 models
Processed data
16
152096
Randomly select 4800 lines of negative data
Training dataset
16
4800
×1000 models
Hard Voting
Final Prediction
🟦 negative 🟧 positive
🟦 negative 🟧 positive
positive：negative = 1：3000
Model evaluation
Precision
Recall
ROC Curve and AUC

Step 1: Literature Review

There are several approaches toward predicting compound-protein affinity prediction, including CNN, LSTM, and regression. In 2018, research team from Bouazizi University proposed the DeepDTA [1] method. The DeepDTA method uses two CNN [2] blocks as machine learning models, then concatenate them into a final DeepDTA model, thus making the final prediction. Ensemble learning [3] is a popular model that comes in handy when encountering cases with little training data, according to scikit-learn documentation, there are two families of ensemble methods, averaging and boosting. Averaging methods include bagging methods, forests of randomized tree, … The way averaging methods work is by building several estimators independently and then averaging their predictions. Boosting methods include AdaBoost, Gradient Tree Boosting… Boosting methods work by combining several weak models to produce a powerful ensemble.

Step 2: Processing Data

We gathered data from the Binding DB database, the raw data was in the format of SMILES, we converted the SMILES data into the MACCS format in order to use it as the training data in our machine learning program. We used the openbabel library in Python to convert the raw data, after we finished converting the data, we generated the positive and negative datasets through Python code. There are 16 lines of data about medicines that were proven to be bondable with ER-Beta, the 16 lines of data was used as the positive training data for the next step. The rest of the data processed was stored in the negative dataset file, which was where we randomly selected negative training data from in the next step.

Step 3: Create the machine learning program

(1) Load training data:

We loaded the data collected in the previous step and assigned them to training variables. There were three datasets that we used in the machine program, positive data, negative data, and testing data. First, we assigned data to the positive dataset, the positive dataset was consisted of medicine data proven to be related to the protein ER-beta. Next, we created the negative dataset, similar to the previous positive dataset, the negative data was loaded from the data we acquired in previous step. However, in order to train multiple models with different training data, we created a function that selects the data from a file consisted of 152096 lines of negative data. Also, to avoid selecting medicine similar in nature or chemical structure, we used the random module in Python to select the negative training data. Finally, we loaded the testing dataset, the testing dataset contained all negative and positive data, which summed up to a total of 152112 lines of testing data that the model will predict after completing the training process.

(2) Data preprocessing

The data loaded in the step (1) were data in the form of arrays, in order to use them as training and testing data for machine learning, we converted the data type from integer to float, then assigned the MACCS and properties to X, and the labels (0 and 1) to y.

(3) Create the machine learning model

Medicine related to the protein ER-beta is very limited, we only found 16 types of medicine that were proven related to the protein, which isn't sufficient of training a machine learning model by itself. Therefore, we used the ensemble learning estimator as the solution to this problem. During the research process, we took two separate paths to create our estimator.

Our first attempt used the traditional ensemble method. There are two categories of ensemble methods, Averaging and Boosting. The Averaging method averages the predictions predicted by several estimators, thus generating a combined estimator that usually outperform all the single base ones; the Boosting method also create several base estimators, but instead of averaging their results, the weak models will be combined to form a single accurate estimator. In this attempt, we used the boosting method. Boosting uses majority voting (either soft voting or hard voting) to form the final estimator. There are two types of voting classifier in boosting, soft voting and hard voting. Hard voting is also called majority voting, the predicted class label will be the class that makes up the majority of the predictions of the individual classifiers. On the other hand, soft voting returns the class label as the argmax of the sum of predicted probabilities. Specific weights are assigned to each classifier, each classifier will output the probability of each class, then multiplying the prediction by the classifier's weight, and average it. The final class label is the one with the highest average. In this research, we used the hard voting technique to classify the testing data. We created a voting classifier consisted of 3 separate model trained beforehand. The three models we used were logistic regression, neural network, and SVM. We tried several different ratios between positive and negative training data, from 1:1 to 1:100. However, as we tested the model's performance by calculating the precision, we got a 1.0 output. Originally, we thought that the 1.0 output implied a great model performance, however, as we tested the model on the complete testing dataset, we received poor performance on it. Thus, we came to conclusion that the ensemble learning model was overfitted. Overfitting occurs when the model takes in too many details thus affecting the performance negatively. Facing this problem, we tried a different approach, instead of using three different types of models for ensemble, we focused on a single type of logistic regression model, changed the penalty to l1, then sought the ideal ratio between training and testing data. The default value of penalty in logistic regression models is l2, l2 limits the model’s coefficients to floats larger than 0. This causes the model to take in too many details, resulting to a lower performance. Therefore, we changed the value of the penalty to l1, allowing some less important coefficients to be set to 0, thus effectively solving the overfitting problem.

Step 4: Finding out the ideal ratio between positive and negative data

Though we used the ensemble learning technique to solve the problem of insufficient training data, we cannot determine the ideal ratio between positive and negative training data. Therefore, we tested the ensemble program created in the previous step with different training data. We test the program on many different ratios, including 1:1, 1:10, 1:20, 1:30, 1:40, 1:50, 1:60, 1:70, 1:80, 1:90, 1:100, 1:200, 1:300, 1:400, 1:500, 1:600, 1:700, 1:800, 1:900, 1:1000, 1:1100, 1:1200, 1:1300, 1:1400, 1:1500, 1:1600, 1:1700, 1:1800, 1:1900, 1:2000, 1:2100, 1:2200, 1:2300, 1:2400, 1:2500, 1:2600, 1:2700, 1:2800, 1:2900, 1:3000. As the goal of our program was to generate several models and combining them to create an accurate estimator, there would be a sweet spot where the individual classifiers met our needs. Also, considering the fact that different training data may affect the prediction and performance of the model itself, we created 1000 separate models for each ratio, each of which consisted of different negative training data, randomly selected from the dataset mentioned in (1). After calculating the individual performance of each model, the precision of the 1000 models will be averaged, forming an overall precision for the specific dataset ratio. In this case, we found out that when the ratio between positive and negative data is 1:3000, we would produce weak individual classifiers that are suitable for our ensemble learning program. After finding out the ideal ratio for the program, we then used the 1000 models to form a single estimator that use the hard voting method to make the final prediction.

Step 5: Test the model performance and stability.

We evaluated our model with the following three method.

(1) Precision

With the sklearn library, we used the precision_score function to analyze the model's performance. Precision took two values into consideration, true positive, and false positive. The two values will produce the precision of the model predicting a specific dataset. The precision value is calculated through this equation: $precision = \frac{TP}{TP+FP}$. Precision is the model's accuracy; therefore, a high precision implies that the model has a high accuracy. We calculated two separate precision for each model in this research. The first precision value of the model is calculated from its performance on the training dataset. The second precision value of the model is calculated from its performance on the complete dataset (all data at hand). Our goal is to produce models with similar precision value on both datasets, thus producing weak models that are suitable for hard voting.

(2) Recall (Sensitivity)

We used the recall_score function to test the model's performance by finding out the proportion of actual positives (TP+FN) were identified correctly. The recall value is calculated through the following equation: $recall = \frac{TP}{TP+FN}$. Similar to precision, we calculated two recall values for every model.

(3) ROC curve and AUC

We calculated the false positive rate, true positive rate, and AUC (area under curve) of the ensemble model by comparing the predictions with the actual answer, then we used the matplotlib library to display the ROC graph. The ROC curve showed the trade-off between TPR (true positive rate) and FPR (false positive rate). A curve close to the upper left corner suggest a high model performance. A random classifier generated a 45 degree curve, stretching from the lower left corner of the graph to the upper right corner. AUC (area under ROC curve) was a common value used to view the performance of the model, the closer the value is to 1.0, the higher the accuracy of the model was.

V、Discussion：

Through the ensemble machine learning technique, we successfully solved the problem of insufficient amount of training data. When seeking the ideal ratio between positive and negative data, we found out that when the ratio is 1:3000, the base estimators will be most fitted for their roles as weak models, thus improving the performance of the final max voting classifier. This implied that there is a point at which the models will reach the level of “weak” models. Then, we analyzed the precision and recall value we received from the models.

(1) Precision

Figure 1: model’s precision values

	1:1	1:10	1:20	1:30	1:40	1:50
training_dataset	0.947576855600549	0.948397115296123	0.950470589344756	0.953878009311967	0.951927423172943	0.957483672994069
complete_dataset	0.500271650528267	0.502809216947553	0.506009971126055	0.509062086452151	0.512533295144287	0.515705933559885

1:60	1:70	1:80	1:90	1:100	1:200	1:300
0.959297116337687	0.960241090761984	0.961655000748934	0.964381717264176	0.957698299601439	0.947894824714229	0.936760890388105
0.520145098823887	0.523585391484494	0.527309527591889	0.530540886403132	0.534574495588829	0.550881854963525	0.560248069030176

1:400	1:500	1:600	1:700	1:800	1:900	1:1000
0.922031739770196	0.912087147378543	0.906323953760067	0.889784858465003	0.878630386465627	0.877752419884079	0.871641564576326
0.568240821281621	0.573799222750292	0.577242978530340	0.581118407812626	0.584554248683826	0.586440035390486	0.588365499328657

1:1100	1:1200	1:1300	1:1400	1:1500	1:1600	1:1700
0.869157077969898	0.864442921277452	0.863392516781141	0.850783994287289	0.842851051428582	0.837563689404888	0.840384675537254
0.589233447404733	0.590113136846672	0.591637047089410	0.590173180655657	0.590580948065224	0.589725705114177	0.590821705130941

1:1800	1:1900	1:2000	1:2100	1:2200	1:2300	1:2400
0.820504354666027	0.796889546193515	0.784005325192348	0.785913310773215	0.773727126397535	0.750893737885821	0.743122394697206
0.586174913165581	0.581683364981786	0.578056501423613	0.577136074305802	0.573177498531559	0.567650582807707	0.565109382594604

1:2500	1:2600	1:2700	1:2800	1:2900	1:3000
0.720340514767219	0.708186116543339	0.684004345424718	0.672535133521100	0.667281623239231	0.637654000220546
0.559444892527933	0.556589705589028	0.549294510886517	0.546287919345994	0.544830843256725	0.537223844168633

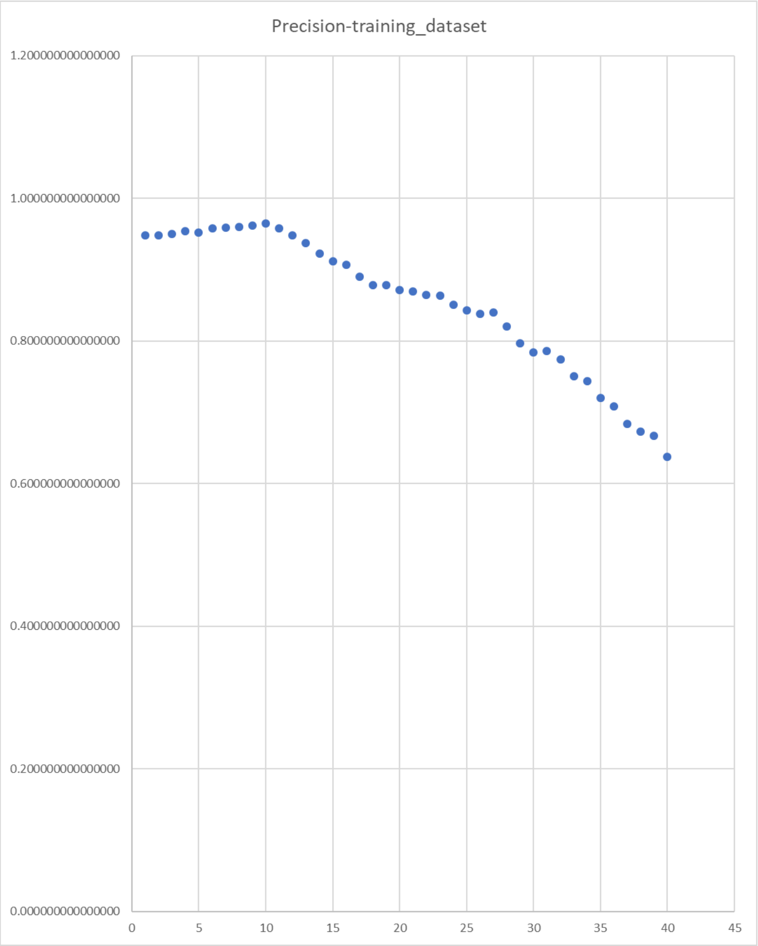


Figure 2: model’s precision when predicting training dataset

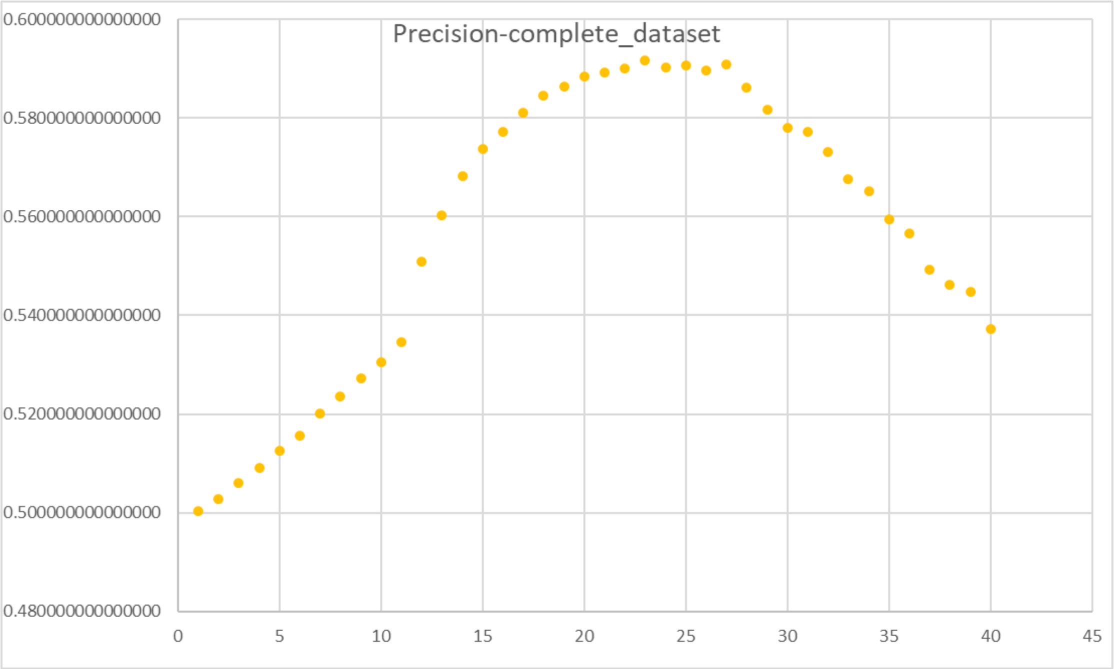


Figure 3: model’s precision when predicting the complete dataset (152112 lines of data)

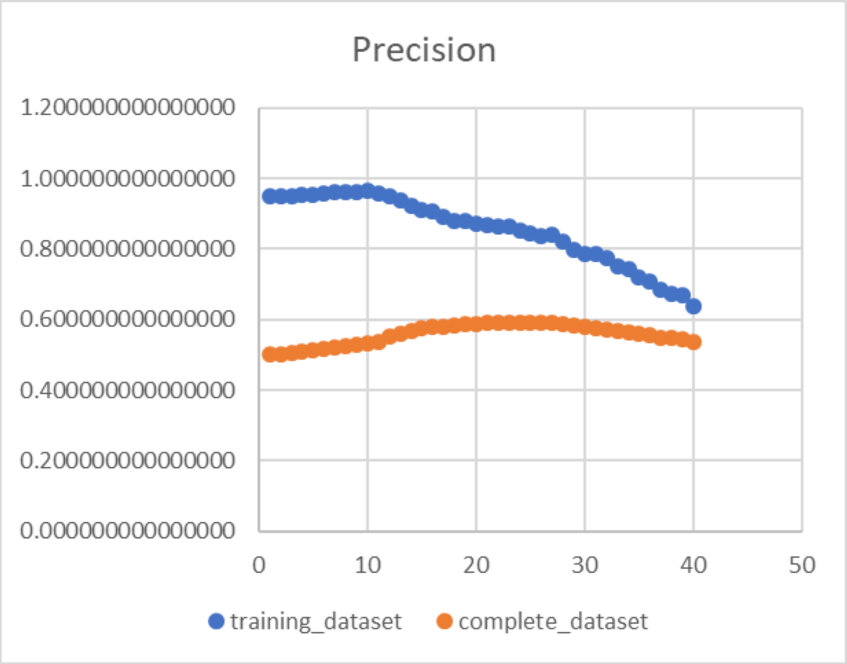


Figure 4: both precision values displayed in comparison

(2) Recall

Figure 5: model’s recall values

x	1:1	1:10	1:20	1:30	1:40	1:50
training_datatset	0.942562500000000	0.886696874999994	0.858542187500002	0.837301041666665	0.815329687500005	0.793899999999993
complete_dataset	0.875877731827267	0.881947717231222	0.856652653587208	0.836239315958341	0.814565642751946	0.793278383389438

x	1:60	1:70	1:80	1:90	1:100	1:200	1:300
training_datatset	0.777850520833336	0.767493750000003	0.758969140624997	0.754191666666669	0.750689999999996	0.736736250000004	0.723733958333329
complete_dataset	0.777370690221965	0.767105555044182	0.758680093493583	0.753967852540501	0.750491482353252	0.736648984851671	0.723665352803491

x	1:400	1:500	1:600	1:700	1:800	1:900	1:1000
training_datatset	0.707779218750001	0.690013374999993	0.677088489583328	0.660812098214287	0.640789218750002	0.627987847222227	0.613273937499994
complete_dataset	0.707730459709656	0.689972717888701	0.677048919761203	0.660783271749421	0.640766962970755	0.627963085814221	0.613253958026508

x	1:1100	1:1200	1:1300	1:1400	1:1500	1:1600	1:1700
training_datatset	0.603498778409087	0.594315364583329	0.581164038461536	0.569542656250005	0.561731979166664	0.554984414062499	0.548705533088231
complete_dataset	0.603478526719965	0.594296529823268	0.581147804675993	0.569528902798232	0.561720492320639	0.554974095308226	0.548696228040184

x	1:1800	1:1900	1:2000	1:2100	1:2200	1:2300	1:2400
training_datatset	0.543675659722219	0.537708256578951	0.534241281249998	0.531492247023806	0.529118181818183	0.525400394021737	0.523557369791669
complete_dataset	0.543667437013464	0.537701688407321	0.534234674153166	0.531486041710498	0.529112129839048	0.525395161608457	0.523552276193982

x	1:2500	1:2600	1:2700	1:2800	1:2900	1:3000
training_datatset	0.520745412499997	0.519558221153844	0.516559085648149	0.514872098214287	0.514997155172414	0.511841541666666
complete_dataset	0.520741064853776	0.519554097412160	0.516559085648149	0.514868711208710	0.514993596149799	0.511838749868503

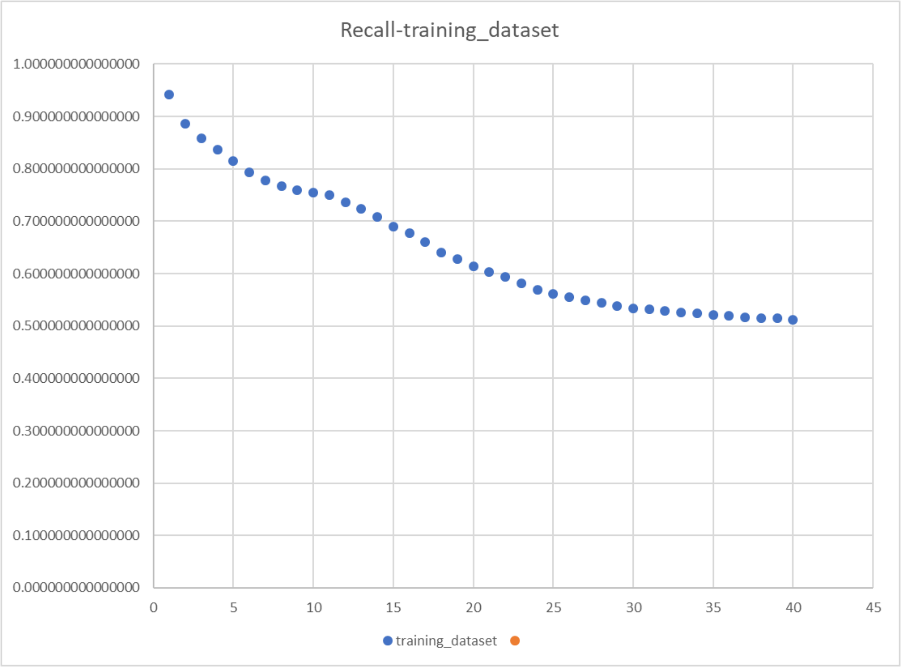


Figure 6: model’s recall when predicting training dataset

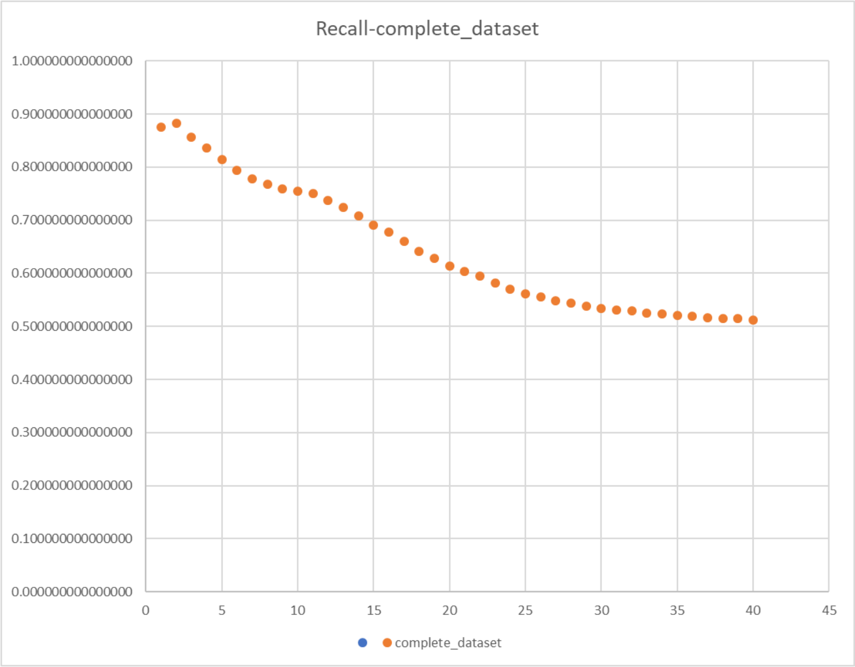


Figure 7: model’s recall when predicting complete dataset (152112 lines of data)

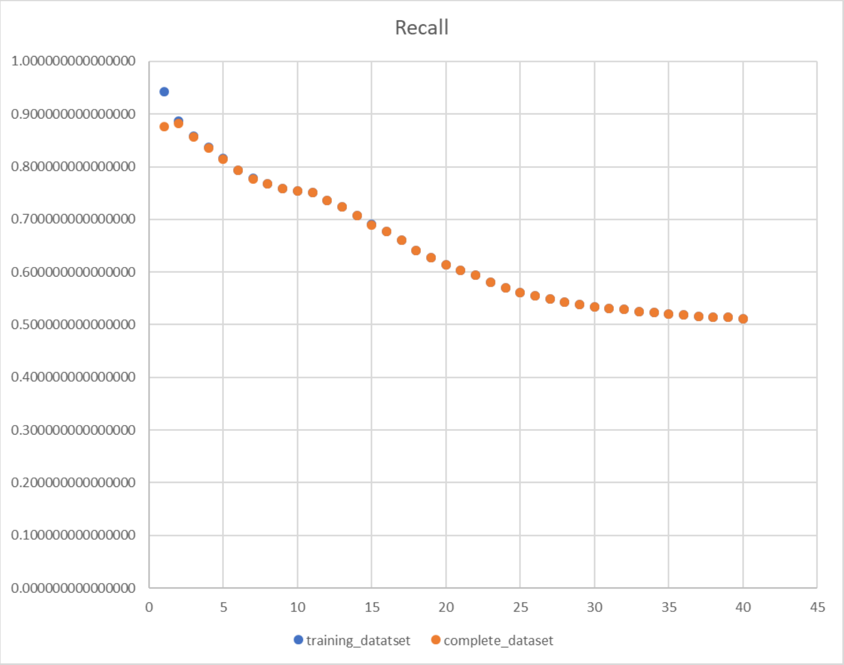


Figure 8: both recall values displayed in comparison

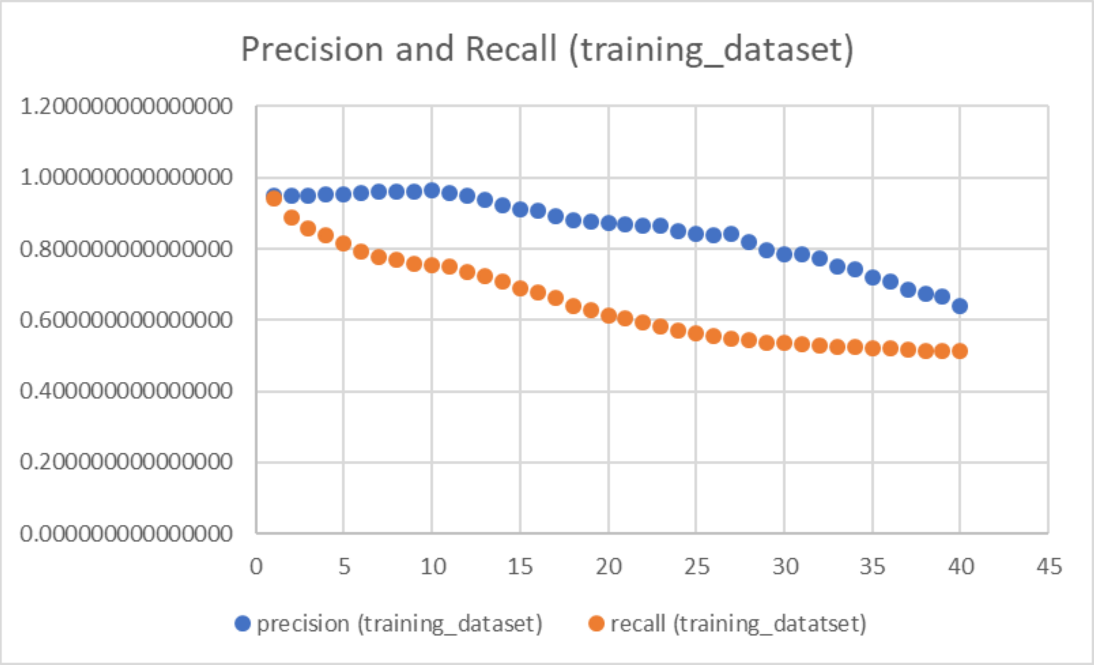


Figure 9: precision and recall values (training dataset)

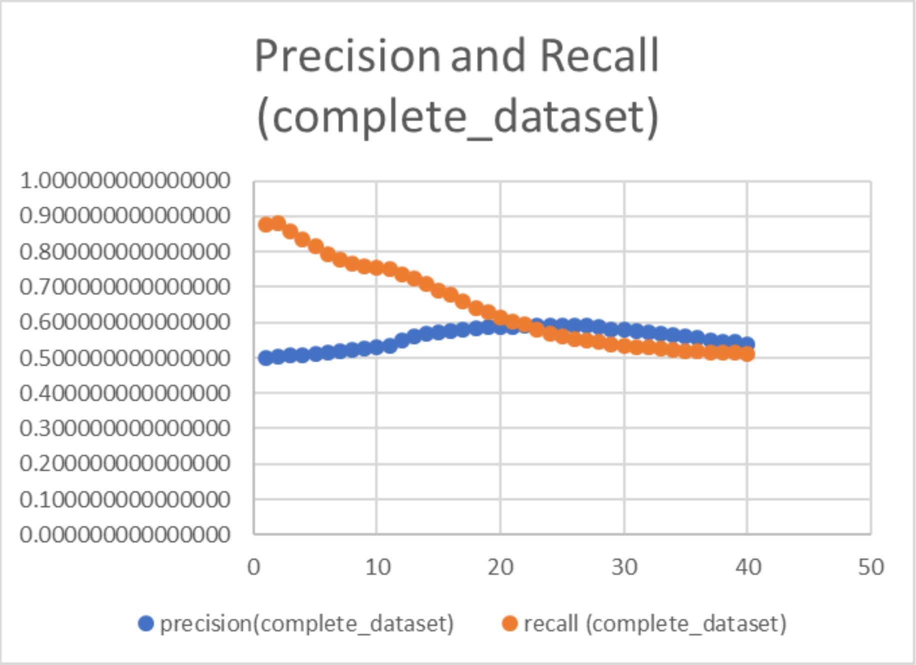


Figure 10: precision and recall values (complete dataset)

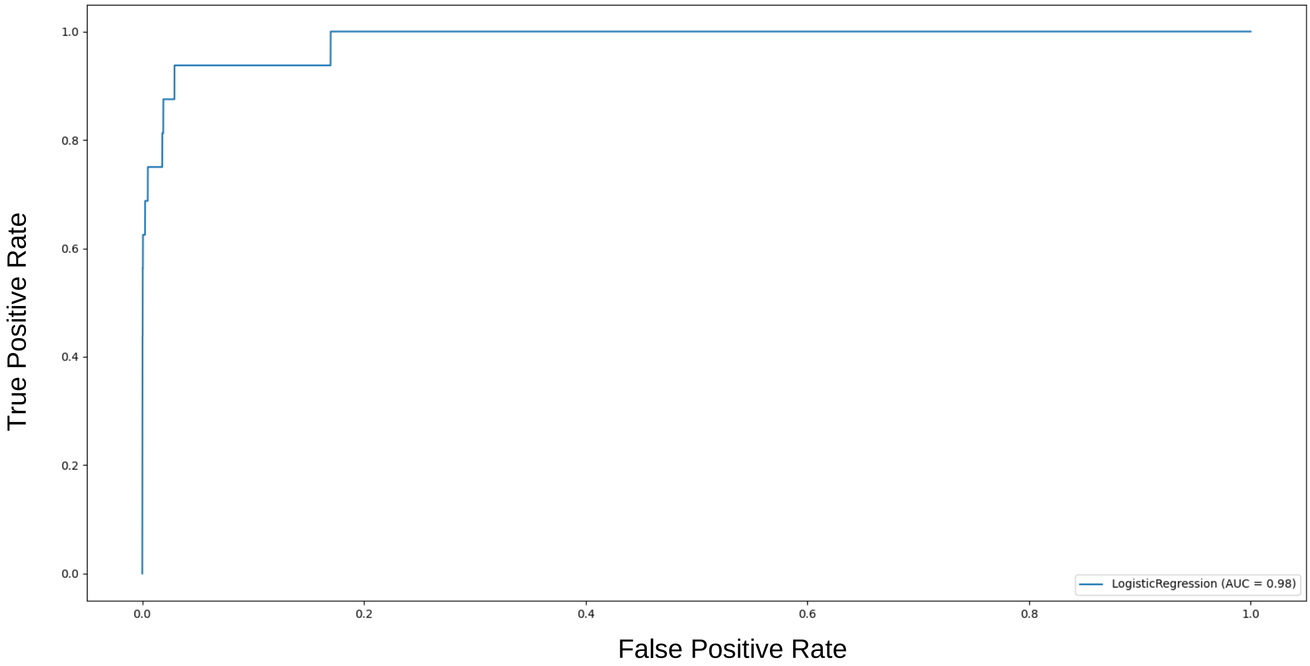
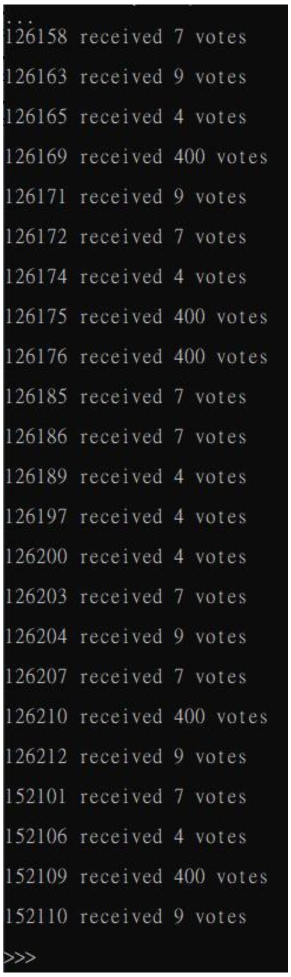


Figure 11: ROC curve



```
...
126158 received 7 votes
126163 received 9 votes
126165 received 4 votes
126169 received 400 votes
126171 received 9 votes
126172 received 7 votes
126174 received 4 votes
126175 received 400 votes
126176 received 400 votes
126185 received 7 votes
126186 received 7 votes
126189 received 4 votes
126197 received 4 votes
126200 received 4 votes
126203 received 7 votes
126204 received 9 votes
126207 received 7 votes
126210 received 400 votes
126212 received 9 votes
152101 received 7 votes
152106 received 4 votes
152109 received 400 votes
152110 received 9 votes
>>>
```

Figure 12: Medicine predicted by ensemble that may be potential to curing breast cancer

44396331	GCR_HUMAN	IMTGEDOKMRCXLE-DYIFYJHRSA-N	0	[0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1,
44396251	GCR_HUMAN	AOIBSIQVRJDUOX-RPBDTDFQSA-N	0	[0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, 0, 1, 1, 1,
44396330	GCR_HUMAN	QIWQUMBYZAMEPN-NFCXHWEESA-N	0	[0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, 0, 1, 1, 1,
44396250	PRGR_HUMAN	JDCXFBDZIJWEFO-PLCYUVIZSA-N	0	[0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1,

Figure 13: Medicine that received 400 votes (the last one was from the positive dataset so wasn't included in this list)

(3) Summary

(a) Precision

From figure 2, we saw that the model's precision on the training dataset decreased dramatically as the gap between positive and negative data amount became larger. In data ratio 1:1, 1:10, 1:20, 1:30, 1:40, 1:50, 1:60, 1:70, 1:80, 1:90, the decrease in precision wasn't clear. However, after 1:90, we saw the significant drop in precision as the negative training data amount increased.

From figure 3, we saw that the model's precision on the full dataset (included all positive and negative) rose as the negative training data amount increased, however, as the ratio between positive and negative data reached 1:1700, the precision dropped steeply, coming to around 0.5 at 1:3000.

From figure 4, we saw that when the two precision values are put together in a chart, discrepancy between the two values decreased, as the negative training data amount increased. The discrepancy came to around 0.1 at 1:3000.

(b) Recall

From figure 6, we saw that as the amount of negative training data increased, the model's precision on the training dataset decreased, it came to around 0.5 at 1:3000.

From figure 7, we saw that as the amount of negative training data increased, the model's precision on the complete dataset (included all positive and negative) decreased, it came to around 0.5 at 1:3000

From figure 8, we saw that the precision values under the same positive negative training data ratio is about the same, except for the first two data points: 1:1 and 1:10.

(c) Relationship between Precision and Recall

From figure 9, we saw that both the precision and recall value decreased as the amount of negative training data increased, also, the discrepancy between the two values first increased then decreased, it came closed to 0.1 at 1:3000

From figure 10, we saw that as the negative training data increased, the precision value increased, while the recall value decreased. The two values overlapped between 1:1100 and 1:1200.

(d) ROC curve and AUC score

From figure 10, we saw that the ROC curve has a low false positive rate and high true positive rate. Also, we received an AUC score of 0.98.

(e) Ensemble estimator results

From figure 11, we saw that most medicines received less than 10 votes as positive, only 5 medicines received 400 votes from the 1000 weak models, however, one of the medicines that received 400 votes was from the positive dataset (proven to be related be breast cancer related medicine beforehand). Therefore, we ended up with 4 types of medicine listed in figure 12 that are potential for curing breast cancer.

VI、Conclusion

We discovered that the model’s precision on the training dataset rose as the negative training data amount increased, implying that the both show negative correlation. As for the model’s precision on the complete dataset, the precision steadily rose until the ratio between positive and negative data came to 1:1300, therefore, 1:1300 will be the ideal positive negative data ratio when training a single model for classification. As we compared the two values (model’s precision on training dataset and complete dataset), the discrepancy decreased as the negative training data amount increased, therefore, we guessed that increasing the negative training data amount had a positive influence on the model performance.

As for the recall values, we discovered that the two recall values were very close when the model was given the same amount of training data, this implied that our model had a stable performance on different datasets.

When comparing precision and recall values, we discovered that when predicting the complete dataset, as the precision value increased, the recall will decrease. Thus, we concluded that the both are negatively correlated.

In the process of seeking the ideal ratio between the positive and negative datasets, we found out that when the data amount between the two datasets is 1:3000, we would receive a similar precision score on both training and full dataset, thus producing 1000 weak models that could be used for max voting.

Upon making the final prediction on the full dataset with the ensemble learning estimator, we found out that 3 of the 4 types of medicine that were predicted as possible medicine for curing breast cancer had the same target protein: GCR_HUMAN (glucocorticoid receptor). Upon finding this, we suspected that breast cancer related protein ESR_HUMAN (Estrogen receptor beta) may have something in common with GCR_HUMAN, thus implying that their medicines may be effective upon each other. As for the fourth medicine predicted, its target protein is Progesterone. Progesterone prepares the endometrium for the potential of pregnancy after ovulation, research studies conducted by scientists and doctors had proved that Progesterone was related to the development of breast cancer cells. Therefore, this could further support the accuracy of our max voting estimator.

Traditionally, to find out whether a protein and medicine can bind properly, scientist and doctors needed to use protein crystallization to determine the result. However, though protein crystallization is effective when it came to such tasks, the time and money required was often higher than most can afford. Our research can serve as the preliminary work for protein crystallization, we hope to shorten the time and money needed for testing new medicines, through our program, medicine compounds can be tested whether they can possibility bind with their target proteins without having to actually stepping into laboratories. In the future, we hope that we can train the model with a larger amount of training data, hopefully, with more data at hand, we will be able to make more accurate predictions with our estimator.

References

[1]衛生福利部乳癌相關資料

https://www.mohw.gov.tw/cp-2641-21095-1.html

[2]使用深度學習策略與圖形演算法改進疾病潛在藥物預測之準確度，取自 https://www.grb.gov.tw/search/planDetail?id=12484695

[3]利用巨量資料及深度學習預防用藥錯誤，取自 https://www.grb.gov.tw/search/planDetail?id=12486219

[4]Deep DTA: deep drug-target binding affinity prediction by Hakime Ozturk, Arzucan Ozgur, and Elif Ozkirmli

[5]A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way

https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53

[6]Ensemble methods

https://scikit-learn.org/stable/modules/ensemble.html

[7]GRB 政府研究資訊系統，取自 https://www.grb.gov.tw/search;keyword=%E4%B9%8B%E5%8F%B0;type=GRB05;scope=1

[8]weka SMYCEFORGE download，取自 https://smyceforge.net/projects/weka/

[9]Coprescription of Chinese herbal medicine and Western medication among female patients with breast cancer in Taiwan: analysis of national insurance claims. 取自 https://europepmc.org/article/med/24855343

[10]Types of Breast Cancer，取自 https://www.cancer.org/cancer/breast-cancer/understanding-a-breast-cancer-diagnosis/types-of-breast-cancer.html

[11]Ductal Carcinoma in Situ (DCIS)，取自 https://www.breastcancer.org/symptoms/types/dcis

[12]A Novel Feature Selection Technique for Improved Survivability Diagnosis of Breast Cancer (research paper)

[13]Breast cancer data in Taiwan，取自 https://wd.vghtpe.gov.tw/cbhc/Fpage.action?muid=12738#:~:text=%E8%87%AA%E6%B0%91%E

5%9C%8B95%E5%B9%B4%E8%B5%B7,%E7%99%8C%E7%97%87%E9%98%B2%E6%B2%BB%E7%9A%84%E9%87%8D%E8%A6%81%E8%AA%B2%E9%A1%8C%E3%80%82

[14]Hormone Receptor Status，取自 https://www.breastcancer.org/symptoms/diagnosis/hormone_status#:~:text=A%20cancer%20is%20called%20estrogen,if%20it%20has%20progesterone%20receptors.

[15]Invasive Breast Cancer (IDC/ILC)，取自 https://www.cancer.org/cancer/breast-cancer/understanding-a-breast-cancer-diagnosis/types-of-breast-cancer/invasive-breast-cancer.html

[16]Types of breast cancer

https://www.bcna.org.au/understanding-breast-cancer/what-is-breast-cancer/types-of-breast-cancer/

[17]How to Develop Voting Ensembles with Python

https://machinelearningmastery.com/voting-ensembles-with-python/

[18]Hard Voting and Soft Voting

https://www.cnblogs.com/emanlee/p/13466950.html

[19]Ensemble methods

https://scikit-learn.org/stable/modules/ensemble.html

[20]How to Calculate Nonparametric Rank Correlation in Python

https://machinelearningmastery.com/how-to-calculate-nonparametric-rank-correlation-in-python/#:~:text=Spearman's%20rank%20correlation%20can%20be,the%20significance%20of%20the%20coefficient.

[21] What Is Regularization In Machine Learning?

https://afteracademy.com/blog/what-is-regularization-in-machine-learning#:~:text=The%20penalty%20is%20the%20sum,want%20to%20penalize%20the%20model.

[22]DrugBank online medicine database

Wishart DS, Feunang YD, Guo AC, Lo EJ, Marcu A, Grant JR, Sajed T, Johnson D, Li C, Sayeeda Z, Assempour N, Iynkkaran I, Liu Y, Maciejewski A, Gale N, Wilson A, Chin L, Cummings R, Le D, Pon A, Knox C, Wilson M. DrugBank 5.0: a major update to the DrugBank database for 2018. Nucleic Acids Res. 2017 Nov 8. doi: 10.1093/nar/gkx1037.

https://go.drugbank.com/

【評語】190006

本作品採用電腦科學中的機器學習理論，解決醫學領域上的實際問題，是極佳的跨領域作品。本作品在機器學習的過程中，在模型的取用與參數的選取上，皆展現良好的科學實驗方法與步驟，若能更進一步探討機器學習的模型優化，同時對於實驗結果提供更深入的探討，將能更強化本作品整體的科學精神展現。