Supervised Learning Assignment

Chenkang Wang
CS7641 -- Assignment 1
Cwang811@gatech.edu

Abstract

In this assignment, I will be implement five learning algorithms, which are Decision Trees, Neural Networks , Boosted Trees , Support Vector Machines and K-nearest neighbors, to different datasets and tuning the parameters and optimizing on both classification datasets . After the implementation, an analysis and comparison would be given.

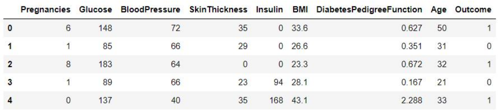
1. Datasets Introduction

1.1 Diabetes Datasets

Datasets info -- This datasets is originally from the National Institute of Diabetes and Digestive and Kidney Diseases. The objective is to predict based on diagnostic measurements whether a patient has diabetes. The datasets has 8 attributes which includes Pregnancies, glucose , blood pressure, skin thickness, insulin , BMI , diabetes pedigree function and age, the outcome is whether the patient has Diabetes or not.

Why The datasets are interesting? -- This dataset has been considered as top 10 datasets for machine learning, it contains around 600 rows and less than 10 attributes. In this dataset, all the data has been cleaned so there is not a lot data clean-up need to be performed. The associate task is classification which also meet the requirement based on Office Hour suggestion.

As an society perspective, the Diabetes is a normal disease that many people have, using machine learning as a tool to predict whether a person has Diabetes can help people understand which group of people is more likely to get it , thus an early preparation could be taken to save more people from it. Figure 1-1 is an example of the top 5 rows of the datasets.



	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1

Figure 1-1

1.2 Customer relationship marketing Datasets

Datasets info -- This datasets is originally from Kaggle best 10 datasets for machine learning. The objective is to predict what type of people is anticipated to make a purchase who is a new customer, for the coming year, from its first purchase. This datasets has 9000 rows and 20 attributes including the income, debts, education level and Age,etc.

Why The datasets are interesting? -- This datasets has also been considered as one of the most classic machine learning datasets , it contains around 9000 rows of data , with regression task it distinguished itself from the last datasets. And from the data standpoint, it is clean and the prediction of the data could be well used for market development for company, thus we implemented machine learning in both medical and business filed, which makes our assignment more interesting and meaningful, since we are not focus only on computer science area.

From the later analysis we found these two datasets has completely different performance over different learning algorithms, this is the most important reason to make he combination of these two datasets being interesting.

2. Datasets cleanup & Pre-Processing

The datasets for diabetes are well-organized and not much of work need to be done to clean it up. Noticed that it is a classification task we use the final column “Outcome” as the score for our diabetes prediction, while 1 means this person have diabetes and 0 means he/she doesn’t have it. For several columns like “Blood Pressure”, “BMI” etc we replace NaN with 0. Below is a figure for all the features Normal Distribution.

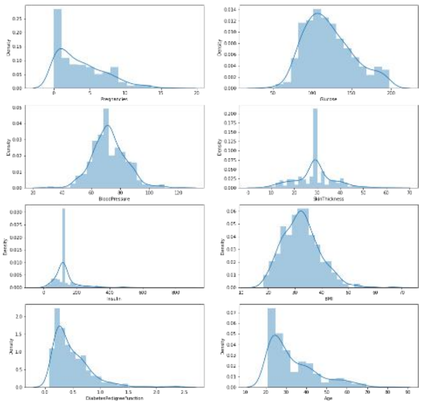
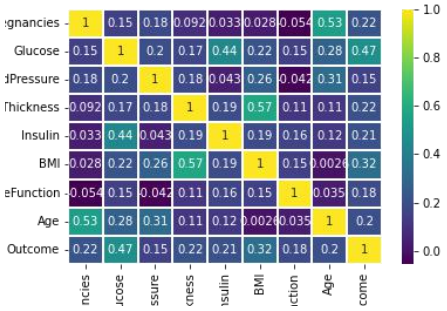


Figure 2-1 Normal Distribution

After that, we check if the data has any correlated values and the result shown as figure 2-2.



	ncies	ucose	ssure	kness	nsulin	BMI	iction	Age	come
gnancies	1	0.15	0.18	0.092	0.033	0.028	0.054	0.53	0.22
Glucose	0.15	1	0.2	0.17	0.44	0.22	0.15	0.28	0.47
dPressure	0.18	0.2	1	0.18	0.043	0.26	0.042	0.31	0.15
Thickness	0.092	0.17	0.18	1	0.19	0.57	0.11	0.11	0.22
Insulin	0.033	0.44	0.043	0.19	1	0.19	0.16	0.12	0.21
BMI	0.028	0.22	0.26	0.57	0.19	1	0.15	0.0026	0.32
eFunction	0.054	0.15	0.042	0.11	0.16	0.15	1	0.035	0.18
Age	0.53	0.28	0.31	0.11	0.12	0.0026	0.035	1	0.2
Outcome	0.22	0.47	0.15	0.22	0.21	0.32	0.18	0.2	1

Figure 2-2 Attributes Correlation

The second datasets for customer segmentation needs some pre-processing and clean up work. Firstly I check the distribution of each variable, after that, I deal with the missing values and categorical values , noticed that there are many categorical variables includes “state”, "Coverage", "Education" and so on, I use one-hot encoding here to make these categorical variables converted into a form that could be provided to ML algorithms to do a better job in prediction. To normalizing data, we convert the numerical data to a scale where the maximum value is 1 and the minimum value is 0. I cleaned up the data and partition it. The outcome of this dataset is “Response” while customer purchase is either Yes or No.

Neither datasets contain negative value so I consider using ***sklearn.preprocessing.scale*** for datasets, since it standardize a datasets along any axis.

In order to analyze the outcome relationship among each features I plot an outcome diagram to virtually understand it as figure below.

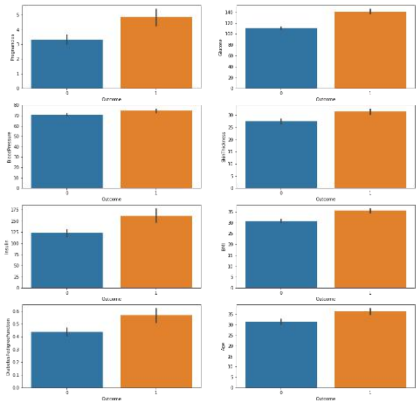


Figure 2-3

70% of the data is used for training and 30% of the data is used for testing. I used cross validation on datasets since the goal is to predict and we want to estimate how accuracy the predictive model will perform in this test.

3. Decision Trees

Advantages: Speed is fast and it's easy to understand classification rules. It can handle continuous and category features and do not need any domain knowledge and parameter assumptions. It's suitable for high-dimensional data.

Disadvantages: For data with imbalanced sample size in each category, the information gain is biased towards those features with larger sample size. It is easy to overfit and ignores correlations between features.

3.1 Diabetes Datasets

The first learning algorithm we implement on diabetes datasets is the decision tree. I implemented ***sklearn*** as a library , it has embedded coding for different algorithms and is useful for machine learning starters.

In order to control the prediction and prevent overfitting from happening, we need to constrain our max depth for searching since it is one of the critical hyper-parameters. Cross validation is a good method to use to help us determine what is the best max depth of the tree.

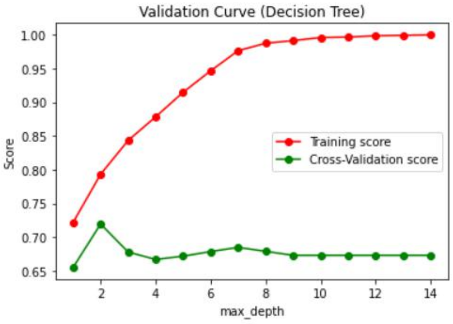


Figure 3-1

From the Figure 3-1 we can tell that when max depth is 8 , the train score has a solids performance after 8 ,besides after that the data starts overfitting. I also dig into the answer for why the train score has such better performance than the cross validation score and I believe that the validation score is less than the training score, because model fits on training data, and validation data is unseen by the model. When the noise from the training set is took and it will cause overfitting when apply test set , which is main reason the train score higher than test set data.

Implementing Cost-Complexity Pruning (CCP), we have our second hyper parameter , which is critical for prevent overfitting. In CCP, the regulation parameter alpha can be used for pruning after trained datasets . The higher of the alpha, means the more nodes being pruned from the original tree.

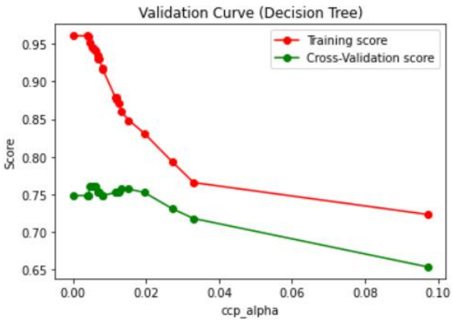


Figure 3-2

As we found from Figure 3-2, with the increase of alpha the training score decreased, the reason is more of the nodes being pruned out from the original tree. After calculation I figured that the optimal ccp alpha is 0.00461.

In order to find the best parameters and calculate the decision tree accuracy, I use Grid Search in python library, which is a function that is a member of sclera's model selection package. After implementing max depth of 8 and ccp alpha of 0.046, we have the learning curve as below.

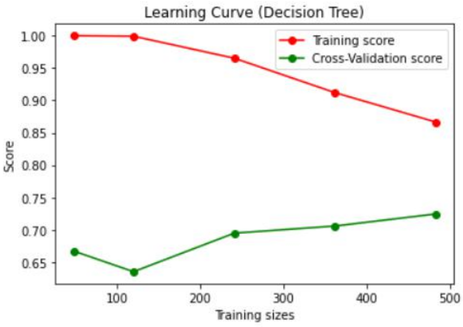


Figure 3-3

For Decision Tree, the cross-validation score can reach about 72.4% accuracy. From above learning curves, we can see that with the increase of training size, the cross-validation score generally show an upward trend, while the training had a downward trend. This indicates that when the training size is small, the model has the problem of overfitting. In general, the effect of the decision tree model on the current data set is mediocre, perhaps because of the sample imbalance.

3.2 CRM Datasets

For CRM dataset, we also apply decision tree on it and check out its performance. First step we repeat the hyper-parameter searching and plot validation curve based off of it.

For CRM dataset , we also search for the best max depth and ccp. Figure 3-4 and Figure 3-5 represents the validation curve for max depth and ccp.

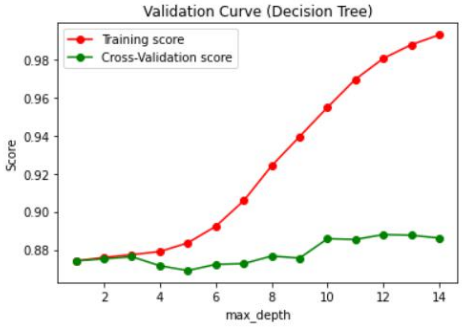


Figure 3-4

The best max depth of this dataset is 12. We noticed that the max depth for this dataset is way higher than the previous diabetes max depth, the reason is this dataset has more features and more samples so the max depth for searching also tend to be higher.

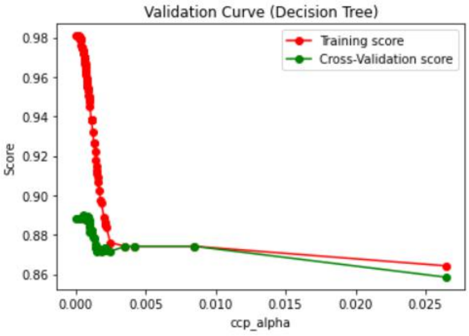


Figure 3-5

The optimal ccp alpha here is 0.00054.

After applying the best max of depth and ccp alpha we have the learning curve plot as below:

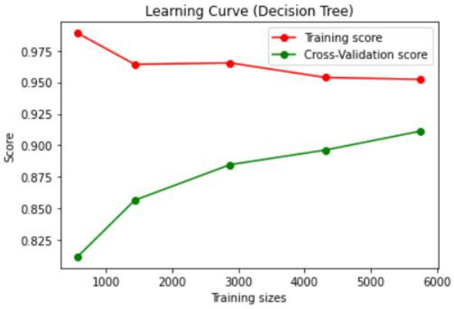


Figure 3-6

For this decision tree, the cross-validation score can reach about 91.1%. It's much higher than the first dataset. Although this dataset is also sample imbalanced, the larger sample size gives the decision tree more opportunities to fully mine the decision rules.

4. Neural Network

Advantages: The main advantage of neural networks is that they can outperform almost all other machine learning algorithms.

Disadvantages: the famous "black box" problem, time consuming, energy consuming.

4.1 Diabetes Datasets

For Neural Network algorithm , hyperparameters includes hidden layers, learning rate, momentum , minibatch size , L2 penalty (regularization term) parameter , etc. For this assignment, I pick learning rate and L2 penalty (regularization term) parameter to do validation curve, and I select solid hidden layer size. Figure 4-1 is the validation curve for learning rate and Figure 4-2 is the validation curve for L2 penalty (regularization term) parameter.

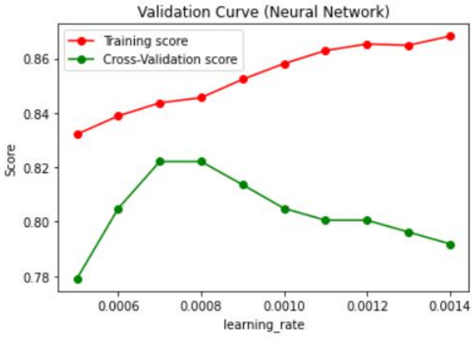


Figure 4-1

From the testing I found the best learning rate is 0.0007. Before that learning rate the cross validation score is climbing and after the learning rate get above 0.0007 ~ 0.0008 the validation score started to drop.

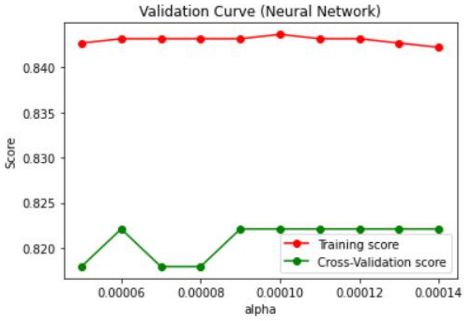


Figure 4-2

From the testing we can find that the best alpha is 0.00006 and anything above 0.00009.

With the best two hyper parameters being find, we choose 0.0007 as our learning rate and 0.0001 as our optimal alpha to plot the learning curve, below Figure 4-3 is our final result.

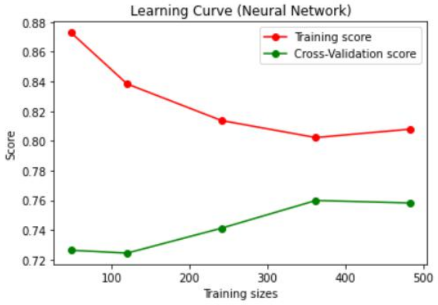


Figure 4-3

For this two hidden layers neural network, the cross-validation score can reach about 0.76. From above learning curves, cross-validation curve had an overall upward trend with the increase of training size, and finally become stable. In general, the neural network is effective on the current dataset.

It can be seen that more complex model didn't bring higher score. Instead, more complex models increase the risk of overfitting.

4.2 CRM Datasets

For Neural Network algorithm we use on CRM dataset, we also choose hyperparameters of learning rate and regularization L2 penalty alpha.

Figure 4-4 is the validation curve for learning rate and Figure 4-5 is the validation rate for regularization L2 penalty alpha.

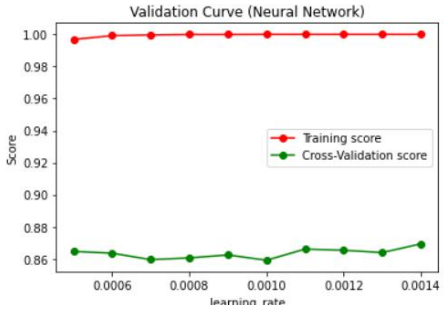


Figure 4-4

From the test we learned that the best learning rate is 0.0014.

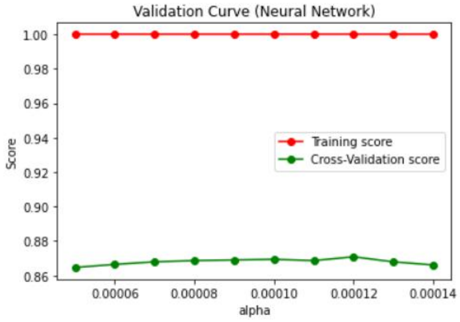


Figure 4-5

From the test we learned that the best alpha is 0.00012.

With the best learning rate and best alpha picked, I implemented them into the final learning curve plotting, with the two hidden layers, including 64 and 32 neural cells , the final learning curve is shown as below:

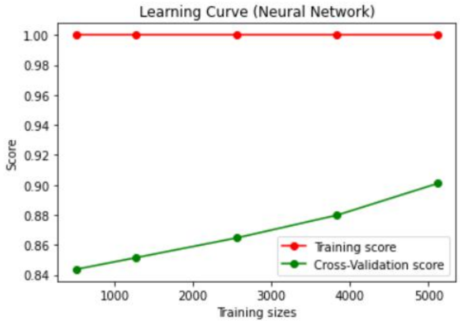


Figure 4-6

For neural network, the cross-validation score on this dataset is also high, about 0.901. It works better.

4.3 Analyze – why NN beat Decision tree?

Neural Network has more hyper-parameters than decision tree , and with the fact that the real data outcome and it's input might not be linear relationship, the more data size is , the more complexity for decision tree to make predictions, that is also the reason why the second dataset, which has more features, when applying NN it has so big of improvement on the score. Neural network has better filtering ability. However, I'd like to say that there are no absolutes, decision trees might also perform better on some other dataset.

5. Boosting

Advantages: Boosting samples based on error rates, so Boosting generally has a better classification accuracy than Bagging. In particular, for classifiers like XGBoost, model performance has been greatly improved through the use of a variety of techniques, including regularization, allowing it to perform well in a wide range of classification tasks.

Disadvantages: Sensitive to outliers, outliers will get higher weight.

5.1 Diabetes Datasets

For Boosting algorithm, we also need to pick hyper-parameters to draw validation curve, for Boosting algorithm, the hyper-parameter includes max depth , learning rate, n estimators , Col sample bytree. For this assignment, I choose to test what is the best learning rate . Figure 5-1 is the validation curve for learning rate & Accuracy.

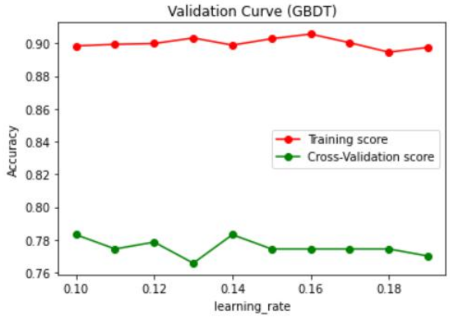


Figure 5-1

We find the best learning rate is 0.14, after 0.14 the cross validation curve starts to decrease and that is overfitting.

With applying 0.14 learning rate into the module, selecting max depth 8, ccp alpha of 3, and below Figure 5-2 is the learning curve based on it.

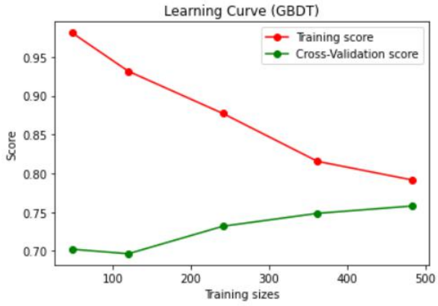


Figure 5-2

The best score we got from the Boosting algorithm is 0.758. Noticed that it is close to neural network, and better than decision tress.

In general, the trend of learning curves are similar to that of decision tree, and there is also the problem of over-fitting when the training size is small. Because GBDT is an ensemble method, it builds an additive model in a forward stage-wise fashion. In each stage trees are fit on the negative gradient of the binomial or multinomial deviance loss function. So, it works better than a decision tree.

5.2 CRM Datasets

For Boosting algorithm for CRM dataset, we also need to pick hyper-parameters learning rate in terms of consistency.

Figure 5-3 shows the validation curve for CRM dataset.

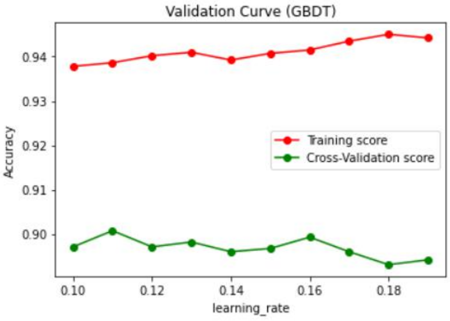


Figure 5-3

We find the optimal learning rate is 0.11.

After we apply 0.11 to the module, with max depth of 12 we have the final learning curve as below Figure 5-4.

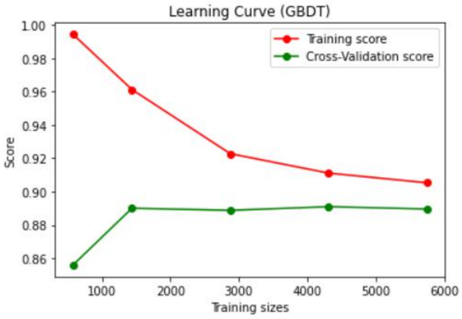


Figure 5-4

The best score for this dataset turned out to be 0.891, it is a little bit lower than NN but higher than decision tree. The reason it is not very high, in my opinion is that it is sensitive to outliers, I think it was influenced by these samples. But with more data samples and more features, it surely has a higher score compared to the Diabetes dataset.

6. Support Vector Machine

Advantages:

- It is supported by strict mathematical theory and does not rely on statistical methods, thus simplifying classification and regression problems.
- Ability to identify key samples that are critical to the task (i.e., support vectors).
- With the kernel technique, nonlinear classification/regression tasks can be handled.
- The final decision function is only determined by a few support vectors, and the complexity of calculation depends on the number of support vectors rather than the dimension of the sample space, which avoids "dimension disaster" in a sense.

Disadvantages:

- Long training time.
- When the kernel technique is used, the space complexity is high if the kernel matrix needs to be stored.
- In model prediction, the prediction time is proportional to the number of support vectors. When the number of support vectors is large, the computational complexity of prediction is high.

6.1 Diabetes Datasets

For Support Vector Machine algorithm, we also need to pick hyper-parameters to draw validation curve. The main hyperparameter of the SVM is the kernel. It maps the observations into some feature space. For this assignment, in order not to over complicated it, I use the linear kernel and RBF kernel testing the regularization parameter. Figure 6-1 is Regularization parameter based on Linear kernel and Figure 6-2 is Regularization parameter based on RBF kernel.

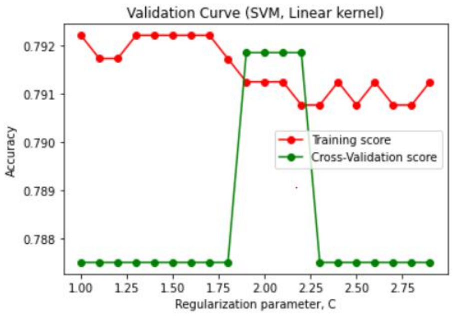


Figure 6-1

The optimal C: any value in the interval [1.7, 2.2].

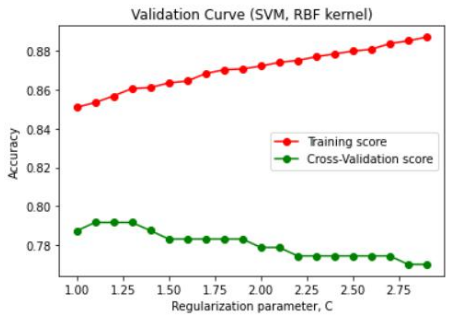


Figure 6-2

The optimal C: 1.1 or 1.3.

I plot the learning curve based on Linear kernel with its optimal C and it is shown as Figure 6-3. The best score is 0.756.

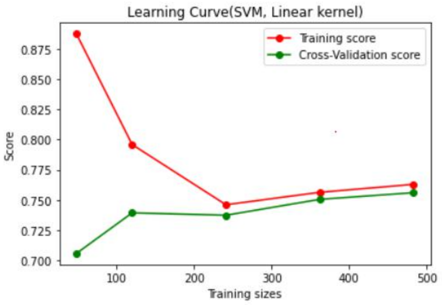


Figure 6-3

I plot the learning curve based on RBF kernel with its optimal C and it is shown as Figure 6-4. The best score is 0.748.

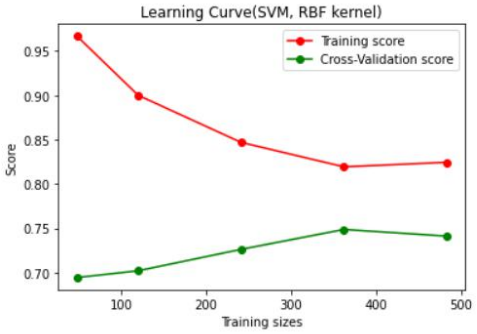


Figure 6-4

For SVM, the cross-validation score of linear kernel and RBF kernel reached about 0.756 and 0.749 respectively. Linear kernel works better for this dataset. When the training data is linearly separable, the linear kernel function is generally used.

When the training data is linearly indivisible, it is necessary to use the kernel technique to map the training data to another high-dimensional space, in which the data is linearly separable. However, it should be noted that if the number of samples N and the number of features M are large and M>>N, a linear kernel is needed. Considering that the space dimension after Gaussian kernel RBF mapping is higher, more complex and easier to overfit, the disadvantages of using Gaussian kernel RBF are greater than the advantages, so it is better to use linear kernel. If the number of samples N is not particularly large and the number of features M is small, then Gaussian kernel RBF can not only achieve linear separability in high-dimensional space, but also has no great consumption in calculation. Therefore, the advantages outweigh the disadvantages and Gaussian kernel RBF is suitable. If N is large but M is small, it is also difficult to avoid complex problems in calculation, so linear kernel will be considered more.

6.2 CRM Datasets

For Support Vector Machine algorithm. Figure 6-5 is Regularization parameter based on Linear kernel and Figure 6-6 is Regularization parameter based on RBF kernel.

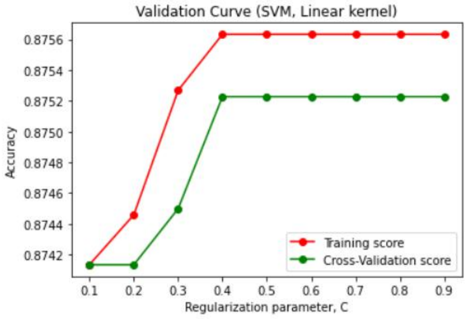


Figure 6-5

The optimal C: any value in the interval [0.4, 1).

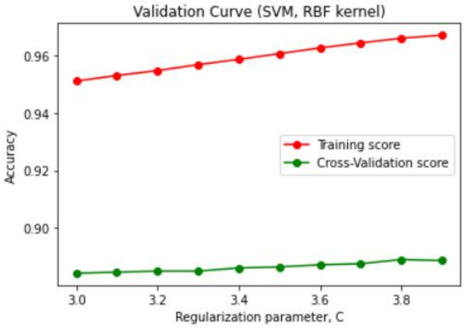


Figure 6-6

The optimal C: 3.8.

I plot the learning curve based on Linear kernel with its optimal C and it is shown as Figure 6-7.

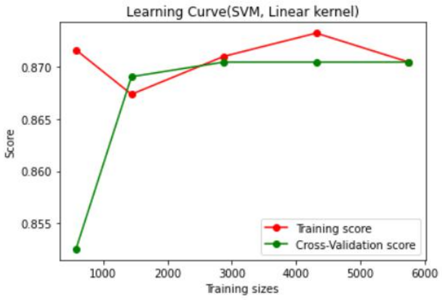


Figure 6-7

The best score is 0.87.

I plot the learning curve based on RBF kernel with its optimal C and it is shown as Figure 6-8.

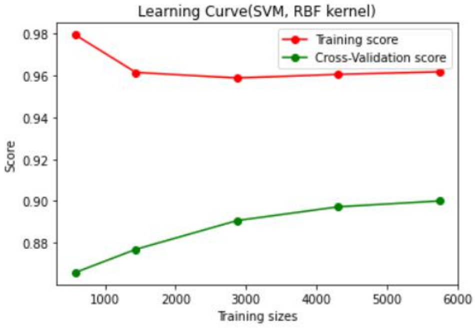


Figure 6-8

For this dataset, the cross-validation score of linear kernel and RBF kernel reached about 0.87 and 0.90 respectively. This time, RBF kernel works better than linear kernel, but it takes more time to compute. Which is due to the larger number of datasets and features.

7. k-Nearest Neighbors

Advantages: easy, insensitive to outliers.

Disadvantages: Lazy algorithm with high time complexity. Depending on sample balance, when there is sample imbalance, there will be deviation in classification. The higher the dimension of the vector, the weaker the distinguishing ability of Euclidean distance. Not suitable for data with too large sample space.

7.1 Diabetes Datasets

The hyper-parameters for KNN includes n_neighbors, weights, metric. For this assignment, I choose to test based on n_neighbors. Figure 7-1 is the validation curve.

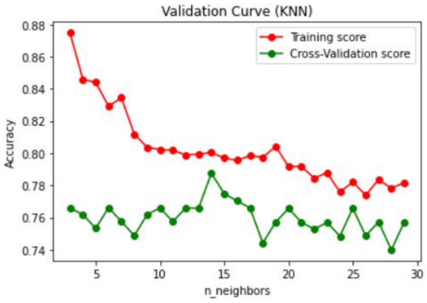


Figure 7-1

The optimal n_neighbors: 14.

After applying n_neighbors of 14 to the KNN module , we got learning curve as below:

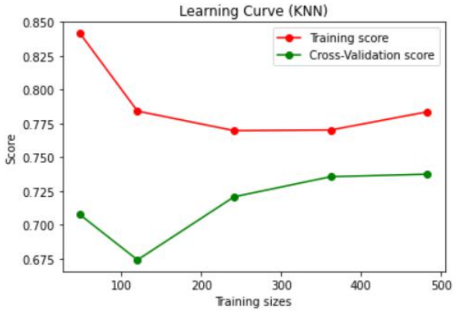


Figure 7-2

For KNN, the cross-validation score can reach about 0.737, higher than the decision tree, but lower than the other three models, the effect is not particularly ideal. It may also be affected by the sample imbalance.

7.1 CRM Datasets

The hyper-parameters for KNN includes n_neighbors, weights, metric. For for CRM dataset, I choose to test based on n_neighbors. Figure 7-3 is the validation curve.

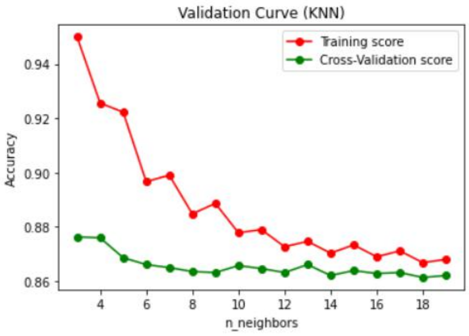


Figure 7-3

The optimal n_neighbors: 4.

After applying n_neighbors of 14 to the KNN module, we got learning curve as below:

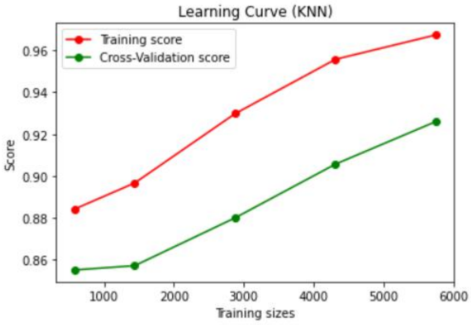


Figure 7-4

For this dataset, cross-validation score of KNN is the highest, reaching 0.926. The larger sample size of the current dataset and clearer classification boundaries give the KNN model a chance to play its full role.

For this dataset, all five models performed better than the first dataset. The reason may be that although the current dataset still has the problem of sample imbalance, the large sample size gives the model more opportunities to mine rules. In addition, the current dataset may be more linearly separable.

8. Summary

In order to better compare the difference among the five learning algorithm , I draw a chart to do compare the data virtually. Figure 8-1 is for Diabetes dataset and Figure 8-2 is for CRM dataset.

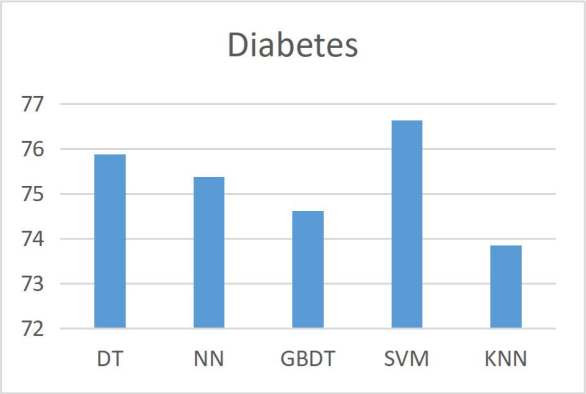


Figure 8-1

From the chart we can clearly see that for Diabetes dataset, SVM has the highest score of while KNN has lowest. One of the reasons that KNN got lowest is that every characteristic of the method has the same result on calculating distance. SVM is a good algorithm for doing classification problem because SVM works by mapping data to a high-dimensional feature space so that data points can be categorized, even when the data are not otherwise linearly separable.

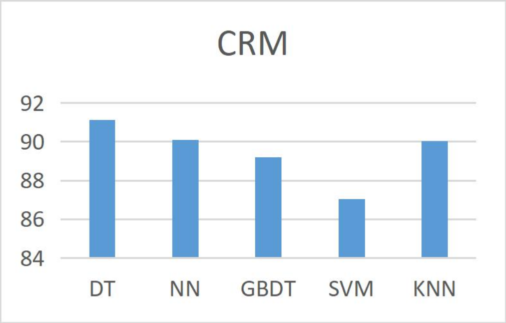


Figure 8-2

For CRM dataset, Decision Tree has highest score while SVM has lowest , Decision trees are better for categorical data and it deals colinearity better than SVM. This is also a good example that no algorithm works best for all datasets.