CS7641 A1: Supervised Learning

Trung Pham
tpham328@gatech.edu

I. Introduction

In this report, we will explore three distinct classification algorithms used on supervised learning environment: K-Nearest Neighbors (KNN), Neural Networks (NN), and Support Vector Machine (SVM). Each of these models has unique strength that make them suitable for various type of classification problems.

In KNN, Similar data points that are close in distance in a feature space are group together. For a new given data point, KNN identifies the k nearest data points of it, and assigns the most common class among those neighbors to the new point.It is simple and fast to train but computationally expensive with large data points or large 'k'.

Neural Networks are interconnected nodes organized in layers settings. Each connection has a weights that are updated based on error or loss function. NN can represent complex relationships in data and make precise prediction.

Support Vector Machine seeks to find a hyperplane that can separate data points of different labels in multi-dimension space. The objective of the hyperplane is maximizing margin or distance from data points to the hyperplane that separate them. SVM used kernel function tricks to calculate this margin without having to transform the data to higher dimension. It is very efficient and effective in solving classification problem.

Using the three supervised learning models, we will explore and analyze the following two datasets. The first one is the Diabetes dataset from the National Institute of Diabetes and Digestive Kidney Diseases (dataset downloaded from Kaggle). The goal is to predict whether a patient has diabetes based on diagnostic measurements. All patients data here are females at least 21 years old of Pima Indian heritage. This dataset caught our attention because it can provide insights into genetic and lifestyle factors influencing diabetes in this particular group. Diabetes is a major health concern and early detection is important for effective early treatment and prevention. In addition, the binary outcome (have or not have diabetes) of this dataset make it suitable for our classification machine learning exploration.

The second dataset we analyze is white wine quality from Vinho Verde wine sample from north of Portugal. The objective is to predict the wine quality score based on various wine physicochemical test results. This dataset is interesting to our exploration because wine quality assessment is important topic for the wine industry. The quality affect both consumer satisfaction and market value. This dataset is also large and not balanced, producing some challenge that we would like to explore to examine the robustness of our interested supervised learning model.

II. Data and Methodology

DATA - Diabetes dataset have 768 instances with 8 feature that we want to study. The target variable is binary of either diabetes or no diabetes. All the feature values are numerical and being scaled so that three models can perform well. They are: number of times pregnant, glucose concentration, blood pressure, triceps skin fold thickness, insulin, BMI, Diabetes pedigree function, and age.

There is no significant correlation between features in the data (no multicollinearity). The data consists of 35% with diabetes and 65% without diabetes. It is not perfectly balance but acceptable. Due to small size of data and testing show poor performance in minority group, the data is applied with Synthetic Minority Over-sampling technique (SMOTE) to address class imbalance.

Wine dataset have both white and red wine but our focus is on the white wine due to higher sample size as we want to study the learning rate on high sample size. Our target variable is white wine quality score, which is a score from 0 to 9. It is normally distributed but highly imbalanced due to small number of low score. Majority of the score is 6. Due to this, we attempt to transform the data into 3 different quality. Score equal to 5 or lower is considered 'low' quality. Score of 6 is 'medium' quality. Score equal 7 and higher is 'high' quality. After transformation, 45% of data is medium quality, 33% is low quality and 22% is high quality. It is not perfectly balanced but acceptable. In fact, the score metrics for minority class is not bad and wine dataset does not required oversampling.

The input variables for white wine are based on physicochemical tests and are numerical or already converted to numerical values. They are fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates, and alcohol content. The data is scaled so our models can perform better.

METHODOLOGY - We will explore and apply each of the three learning machine models to each dataset in Python and provide our analysis in the order of the model. The process is the same for each algorithm. Before we begins, 20

We explored several combination of hyperparameters and create several validation curve for each hyperperameter test. The validation curve is based on the metric score after training the model and apply cross-validation technique. We used 5-folded cross validation, which mean we divided the data into 5 sets, and use 4 sets to train the model and 1 set to validate. The validate set is rotated 5 times and we recorded the average score metric. The metric we used to score is recall, f1 and accuracy. We focus mainly on f1 score for diabetes dataset due to imbalance and requirement for oversampling. For white wine dataset, since data is more balance after transformation, we used accuracy score metric.

The value of the validation curve is the score metric calculated on the validation dataset while the training curve is the score metric calculated on the training dataset during cross validation process (k fold equal 5). The original training dataset is divided into 5 sets, each set is rotated to be come the validation set (each fold) and the model is trained on the rest of the data. The score metric is calculated on both training set and validation set. The final score is the average of the metric on each fold. The metric for wine dataset is accuracy while the metric for diabetes is f1 due to imbalance. This process was used to find the optimal hyperparameters for the model.

Final model is train using all the available train data or any oversampling data and test it with our reserved testing dataset. We each model and dataset, we come up with a hypothesis and attempt to support it.

III. KNN

A. Diabetes dataset

Hypothesis: Using oversampling technique like SMOTE is an effective way to improve performance of KNN model in dealing with imbalance dataset.

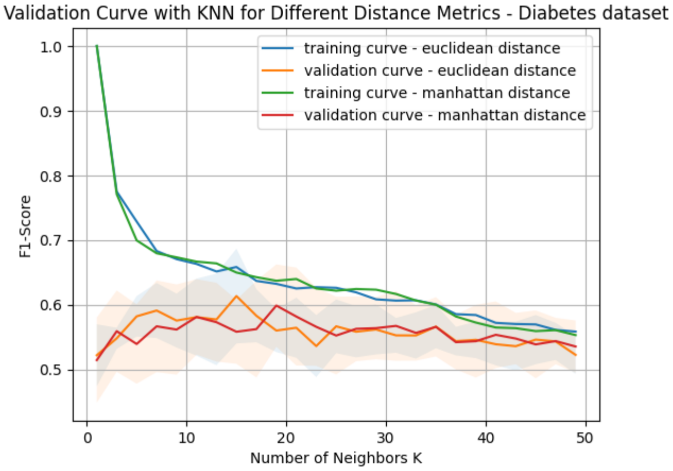

Fig. 1: Training and validation curve of KNN using euclidean and manhattan distance on Diabetes dataset

We applied various number of neighbor k to our KNN model. We also use 2 different distance function of the KNN when calculating distance of the new point to the nearest neighbor. They are euclidean and manhattan distance. Our result showed that there is not much different between using euclidean and manhattan distance function.

With k value less than 5, the F1-score of the training curve is high but the validation curve is low, this indicate overfitting (fig.1). Euclidean distance variance (blue highlighted) is lower than the manhattan variance (orange highlighted have wider range). Our optimal is around k = 15 although there is not much different in performance compared with other k value close to 15.

When k less than 15, we can see the variance (highlighted area) is high compared to when k is over 40. The prediction score f1 is lower as k increase but variance is lower is a sign of bias-variance trade-off. Given our prediction score is low, we opt to choose k value with high f1 score but lowest variance possible. That is k equal 15, using Euclidean distance.

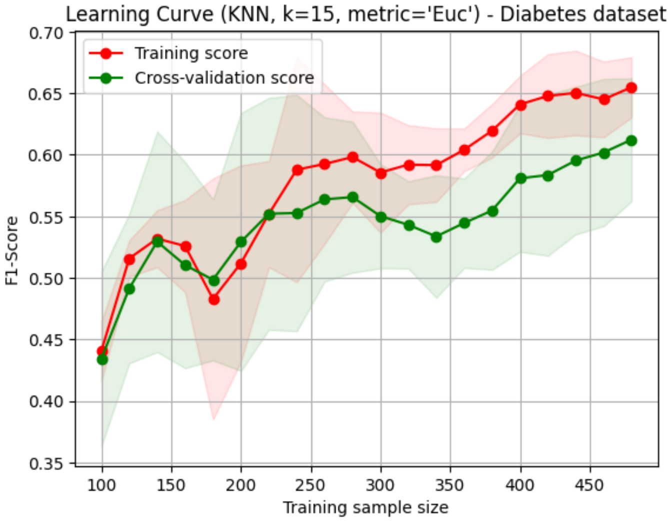

Fig. 2: Learning curve of KNN (k=15, Euclidean distance) on Diabetes dataset

The cross-validation or learning curve pointing up indicating that the model can benefit from more data (fig.2). In this case, oversampling like SMOTE will improve model performance.

Original data	Recall	F1-Score
No Diabetes	0.84	0.79
Diabetes	0.45	0.52
With SMOTE	Recall	F1-Score
No Diabetes	0.74	0.82
Diabetes	0.87	0.74

TABLE I: Classification report on Diabetes dataset with and without oversampling SMOTE technique

After applying SMOTE to increase the training dataset to 1,000 records. The prediction result on the testing dataset is shown on table 1. The Recall and F1 score of predicting diabetes in case of original training dataset is quite low only 0.45 and 0.52 respectively. With oversampling technique like SMOTE, the performance increase significantly to 0.87 and 0.74. This prove that SMOTE improve the KNN model for Diabetes dataset.

B. White wine quality dataset

Hypothesis: KNN model using euclidean distance function have similar accuracy with KNN model using manhattan distance function in wine dataset.

Figure 3: When k value is low, accuracy score on training set is high while score on valuation set is low because of overfitting. The gap between training and validation curve is closer when k increases indicating that variance is decreasing. As both line converge and not showing much improvement after k is higher than 30, we can say that the model has low variance and bias. The model generalize well with any k above 30.

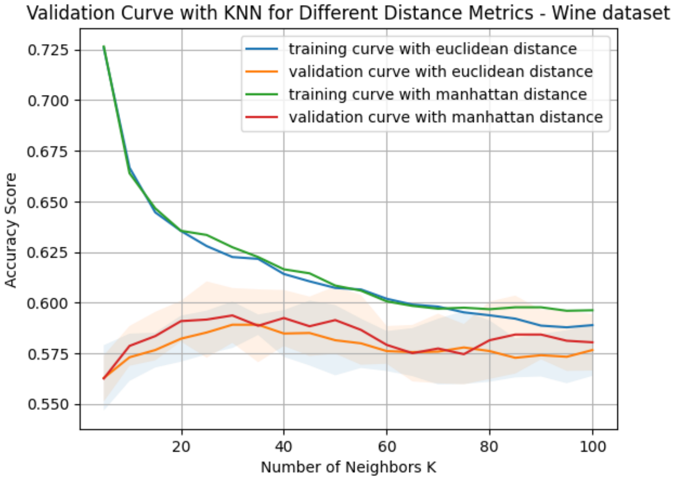

Fig. 3: Validation curve of KNN using euclidean and manhattan distance on Wine dataset

In the same graph, line red and orange go almost together same as line blue and green. This indicates that there is not much different between Euclidean and Manhattan distance in term of model performance. Both Euclidean and Manhattan curve show low bias and low variance after k equal 30. It is safe to say that choosing either Euclidean or Manhattan will produce equally good result as long as k is high enough. For this dataset, we chose k equal 30 and using Manhattan distance function.

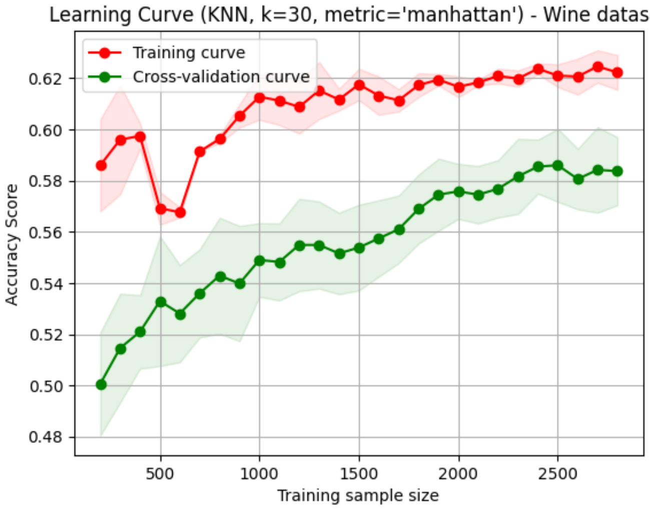

Fig. 4: Learning curve of KNN using euclidean and manhattan distance on Wine dataset

The cross validation curve start slowing down as training sample size increase, indicating that the model benefits less from more data inputs and a convergence is happening. Given the size of the data, we think current data is sufficient and oversampling is not necessary. When we look at the prediction result on the test dataset. There is not much difference between prediction in all three classes. The imbalance data does not affect the prediction of minority class, this also indicate of sufficient data.

IV. Neural Networks

A. Diabetes Dataset

Hypothesis: Neural network learner with ReLU activation function is a better model than neural network with Sigmoid activation function in predicting diabetes.

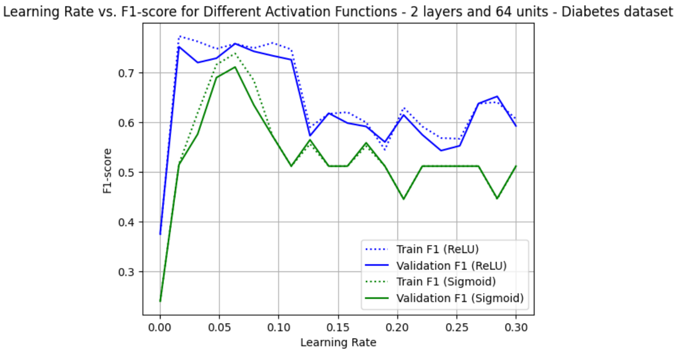

Fig. 5: Validation curve of NN under different learning rate on Diabetes dataset

In figure 5, the model with ReLU (blue) is better than the model with Sigmoid regardless of value of the learning rate. This prove that NN with ReLU is better in Diabetes dataset. Also, the train and validation curve travel very close to each other indicating that Neural network achieve low variance. Learning rate is best between 0.02 and 0.10.

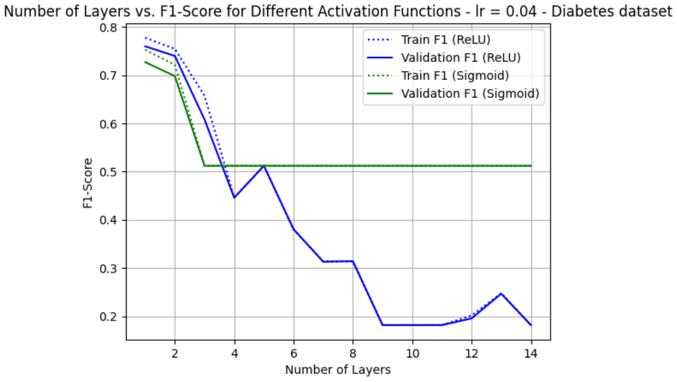

Fig. 6: Validation curve of NN under different layer on Diabetes dataset

In figure 6, once again, we saw that ReLU model perform better than Sigmoid model when number of network layers is below 4. NN with Sigmoid function perform no better when the number of layer above 3, and therefore the number of layer above 3 is hardly a choice. Under 4 layers, the ReLU model always performed better. Under Occam's Razor theorem, we should choose the simplest model that produce similar best result. Therefore, we will chose 2 layers and 64 hidden units, activation function is ReLU and learning rate is 0.04.

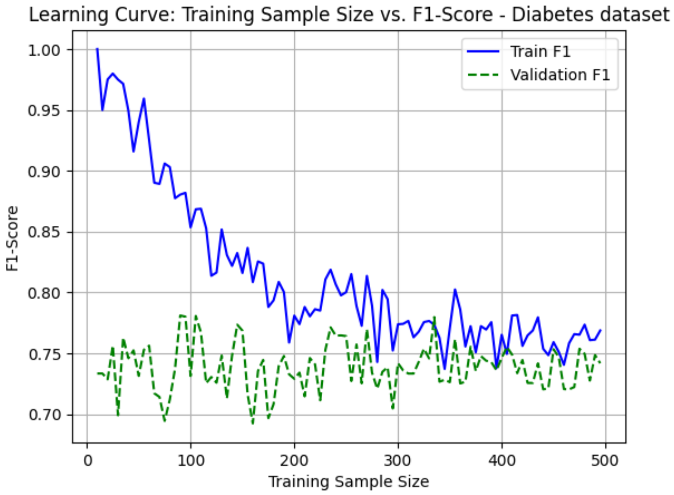

Fig. 7: Learning curve of NN under different sample size on Diabetes dataset

In figure 7, we can see that 200 samples are sufficient for the NN model to train. The blue and green line are close after 200 indicating low variance. There is no significant improvement after sample size equal 200. Therefore, more data will not help the model.

Original data	Recall	F1-Score
No Diabetes	0.67	0.76
Diabetes	0.83	0.68
With SMOTE	Recall	F1-Score
No Diabetes	0.76	0.83
Diabetes	0.85	0.74

TABLE II: Classification report on Diabetes dataset with and without oversampling SMOTE technique

In figure 8, the green and blue line diverge when epoch is 20 or higher. This indicate that more iterative attempt to update the weights of corresponding neural network nodes is counter-productive. It provides no prediction improvement and only increase variance. For this reason, our final model keep epoch equal 20.

Again, the loss curve showed no improvement after epoch is higher than 10. Therefore, it it not necessary to have higher epoch value. Our final model is a neural netowrk with 2 layers and 64 nodes at each layer, learning rate is 0.04 and epoch is 20. Our prediction score on testing dataset before oversampling is shown in table 2. With SMOTE, we can see the improvement in prediction of the minority class.

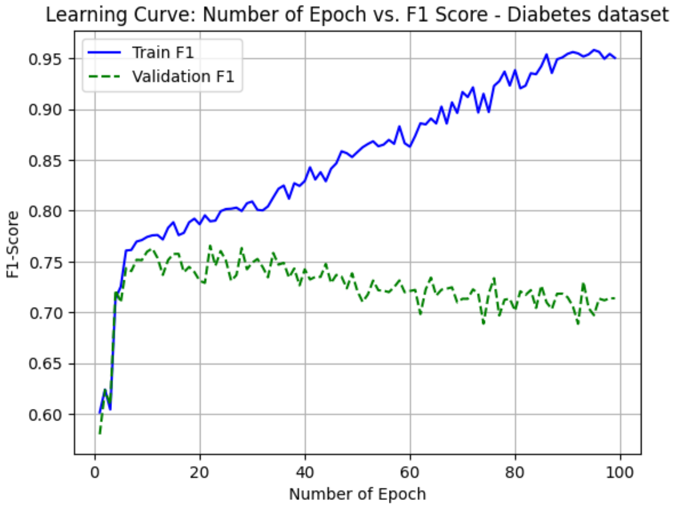

Fig. 8: Validation curve of NN under different epoch on Diabetes dataset

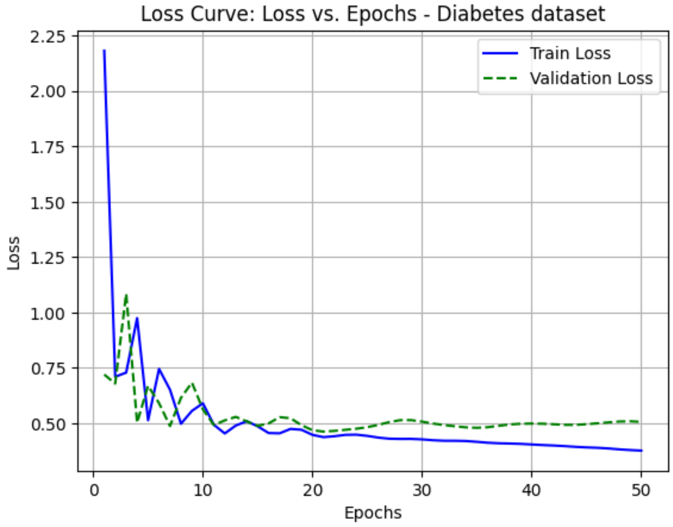

Fig. 9: Loss curve of NN under different epoch on Diabetes dataset

B. *White wine quality dataset*

Hypothesis: Epoch higher than 25 times do not improve the Neural network learning machine on wine dataset. Similar to Diabetes dataset, we analysed 2 different activation functions ReLU and Sigmoid. On figure 10, validation curve conclude that higher learning rate does not result in better model. The optimal value for learning rate between 0.01-0.03. The variance is low.

In figure 11, add more layer, making the model more complex do not always result in higher outcome. Model performance using ReLU significantly drop after the number of layers reach 2.5. To pick the most optimal network structure, We go with layer of 2 and 200 nodes.

In figure 12, the training and learning curve came together after training sample size reaches 1,000. This indicates that over 1,000 sample is sufficient for the model to train. The line going side-way indicating that more data

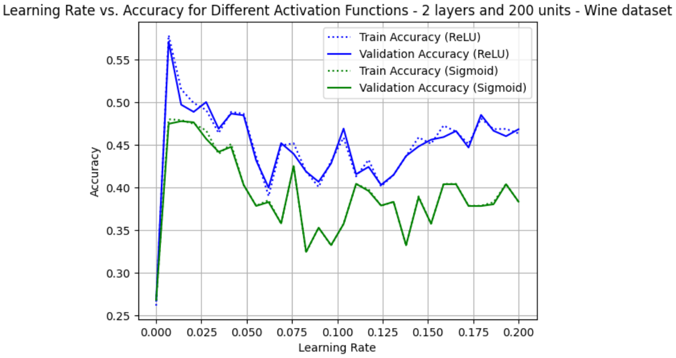

Fig. 10: Validation curve of NN under different learning rate on wine dataset

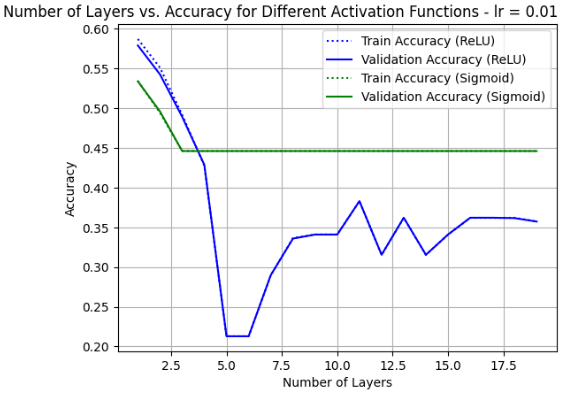

Fig. 11: Validation curve of NN under different layer on wine dataset

is not needed. Due to small and simple 2 layers structure, the data required to train the model is small. Due to the curse of dimensionality, as number of dimension increases (more layers), the required volume of data increases exponentially. Due to the size of the dataset, choosing simple 2 layers structure is appropriate.

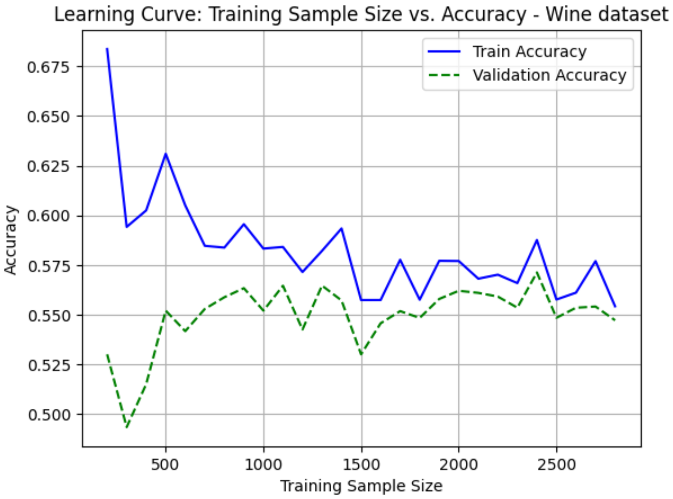

Fig. 12: learning curve of NN under training sample size on wine dataset

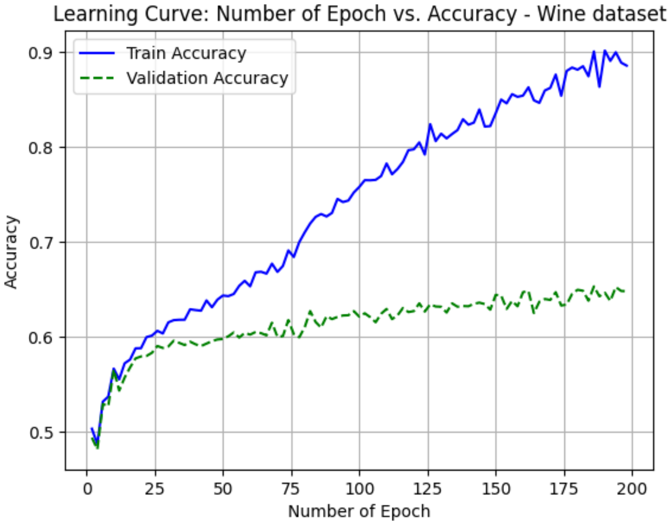

Fig. 13: learning curve of NN under different epochs on wine dataset

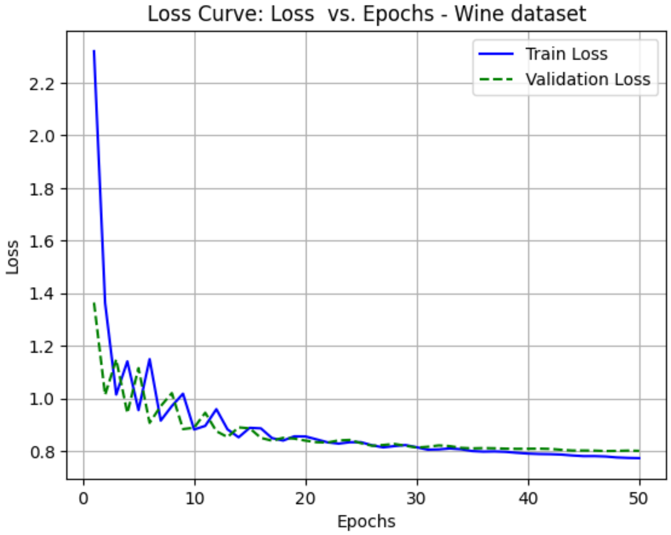

Fig. 14: Loss curve of NN under different epochs on wine dataset

In figure 13, increase number of epoch does not guarantee better performance. As the number of epoch increases, the distance between training and learning curve increases indicating higher variance. The learning curve (green dash line)increases as number of epoch increases, indicating a lower bias, but higher variance. This is a bias-variance trade off. After epoch is greater than 25, the speed that bias decreases is slowing down, while variance increases rapidly. The trade off of increasing variance does not accompany by similar decrease in bias. It is once again confirmed in the loss curve in figure 14 (green line). Loss does not improve much after epoch is higher than 20. Therefore, it is concluded that the performance of learning model does not improve with epoch higher than 25.

V. SVM

Support Vector Machine (SVM) works by finding the optimal hyperplane that seperates different classes in the feature space (multi-dimension). To handle the non-linear data, SVM employ kernel functions to calculate distance instead of actually transforming all the data. The Radial Basis Function (RBF) kernel maps data to an infinite-dimensional space to capture complex relationships, while polynomial kennel maps data to polynomial feature space. They both handle the non-linear data effectively but RBF usually require more computational power due to infinite-dimensional space calculation. For SVM with polynomial kernel, we explored different value of C, and polynomial degree, we did fine-tune a coefficient - coef0 but will not discuss in this report. For SVM with RBF kernel, we explored different values of C, and gamma. A small C creates wider margins but allow some misclassification (high bias, low variance), while a large C result in low bias, high variance. We decide to analyze C equal either 1 or 10, two common value for this parameter.

A. Diabetes Dataset

Hypothesis: Among the three learning model being exploring, SVM is the best solution to predict Diabetes case.

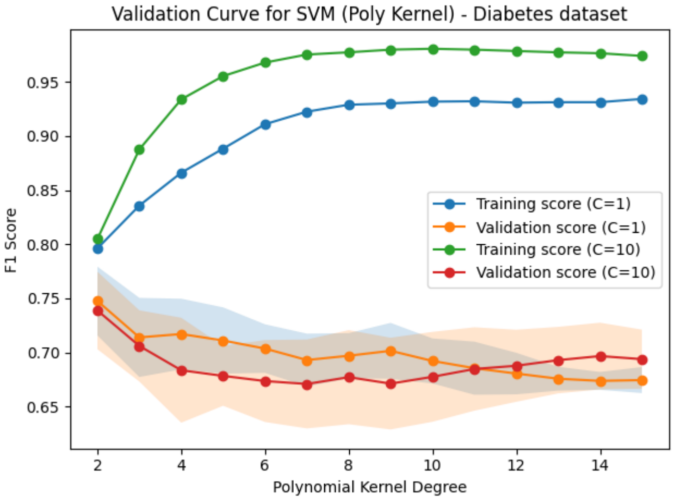

Fig. 15: Validation curve of SVM for Polynomial Kernel on Diabetes dataset

In figure 15, result of (C=10) model produced high score in training curve and the gap between training and validation curve is high, indicating very overfitted. Validation curve (C=10) only produce better score than (C=1) at high polynomial degree but model with (C=1) results in lower variance. At low polynomial degree, (C=1) also produce better f1 score (low bias). For this reason, we opted to choose C equal 1.

Following Occam's Razor theorem, we also preferred simpler model with lower polynomial degree so we decided to explore polynomial degree of 2 and coef0 equals 2. The result of this model is shown in figure 16.

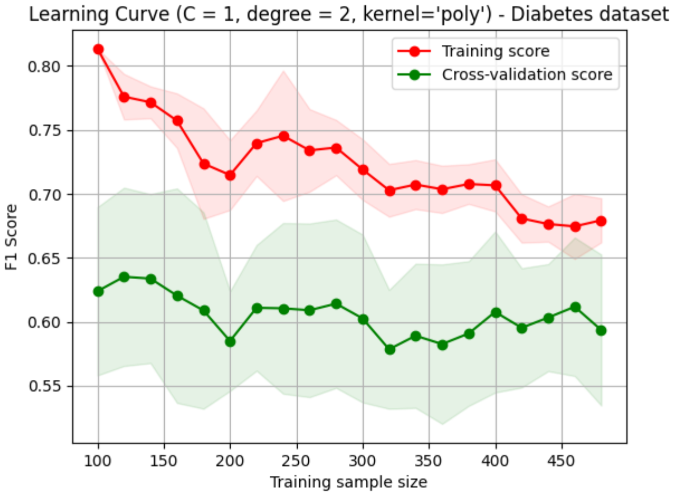

Fig. 16: Learning curve of SVM for Polynomial Kernel on Diabetes dataset

In figure 16, the gap between training and learning curve is big. This is indication of overfitting. Learning curve does not show significant increase as sample size increase, this means the model might not benefit from getting access to more train data. The learning curve has been stable very early. This is because SVM can achieve good performance with relatively less data than others model due to their use of support vectors, which represent critical elements of the data. Although we did see that training curve is approaching learning curve, indicating lower variance so more data can help with lower variance but the difference between two curve are less than 10 percent, so more data might not be necessary.

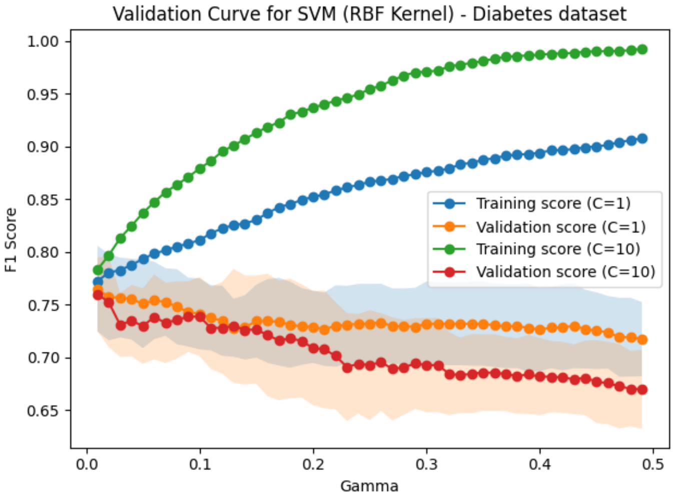

Fig. 17: Validation curve of SVM for RBF Kernel on Diabetes dataset

To continue, we explore the SVM model with RBF kernel. In figure 17, it is clearly that (C=1) produced much better results than (C=10), the validation curve of (C=1) is higher than (C=10) in any value of gamma. In additional, distance between training and validation curve of (C=1) is also smaller indicates lower variance. We chose the value of C to be 1 with low value of gamma.

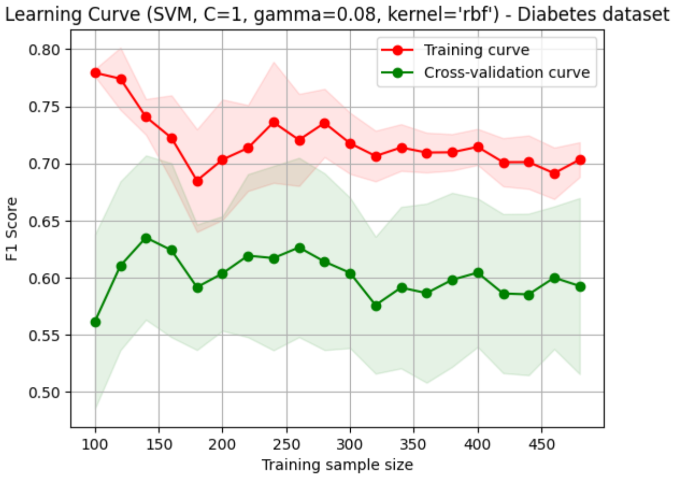

Fig. 18: Learning curve of SVM for RBF Kernel on Diabetes dataset

After fine-tuning, we decided to analyze with gamma equal 0.08. Similar to figure 16, the SVM model produce a promising result (figure 18). More sampling data might not improve the model's performance. Initially, training and learning curve are far apart, showing high variance. As they approach and go side-way, indicating lower variance and stable bias. Our final model is SVM with RBF kernel (C = 1, gamma =0.08). The model is trained and evaluated on our testing dataset. The result is shown in table III.

Original data	Recall	F1-Score
No Diabetes	0.89	0.83
Diabetes	0.53	0.61
With SMOTE	Recall	F1-Score
No Diabetes	0.81	0.87
Diabetes	0.89	0.79

TABLE III: Classification report on Diabetes dataset with and without oversampling SMOTE technique

The SVM with RBF kernel is proven to be best model for the diabetes dataset. Initially, the recall and F1 scores for predicting minority class were low as expected. However, after applying SMOTE technique to balance the dataset, the prediction results improved significantly. In fact, SVM achieved highest score among the three model. In addition, SVM is efficient and require less data, which is suitable for the Diabetes dataset given its small available data. This combination of high performance and efficiency makes SVM with RBF kernel the optimal choice.

B. *White wine quality dataset*

Hypothesis: Given three models (KNN, Neural Networks and SVM), SVM is the optimal choice to predict white wine quality.

For wine dataset, the SVM with poly kernel (C=10) produced slightly better result than model with (C=1), but has worse variance. At low poly kernel degree, where variance is low, model with (C=10) could be meaningful. However, similar to Diabetes dataset, we would lean toward simpler model to pick C equal 1 unless higher C produce significant improvement. (Figure 19)

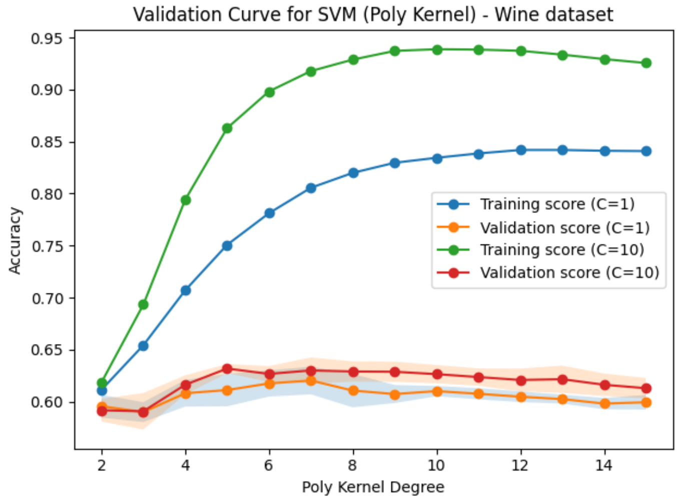

Fig. 19: Validation curve of SVM with poly kernel on wine dataset

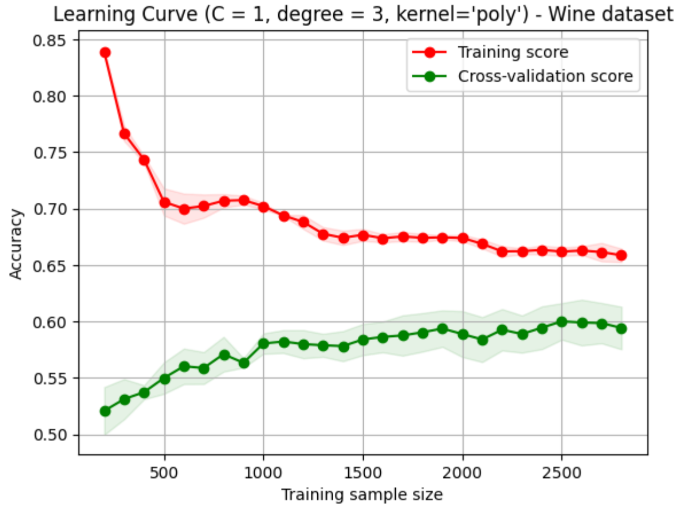

Fig. 20: Learning curve of SVM with poly kernel on wine dataset

Choosing C = 1, we went with poly kernel degree of 3 to add a bit of complex to the model. The learning curve is shown in figure 20. The learning curve converged. Both training and cross-validation curve approached each other. Adding more data to the training process do not improve the performance significantly. We are satisfied with the current result of low bias and low variance. SVM model is an excellent choice for wine quality prediction.

In addition, we explored the SVM model with RBF kernel on wine dataset. Compared to poly kernel, the RBF kernel model produced sightly better result. THere is no noticeable difference between C = 1 or C = 10. For this reason, we leaned toward simple model (C=1). Model (C=10) also resulted in high variance which is not desired.

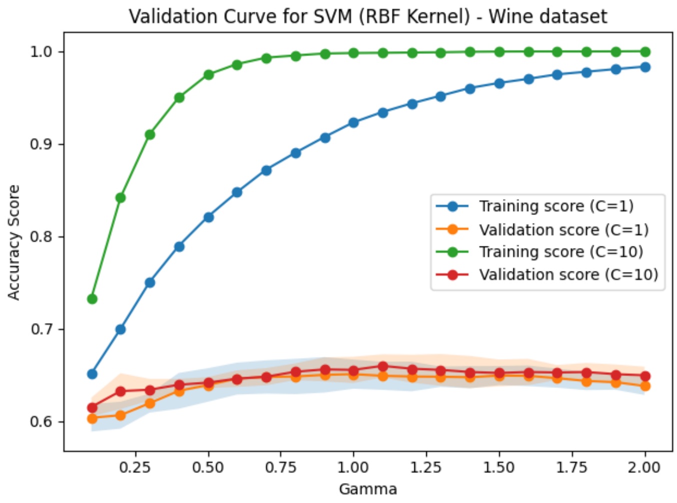

Fig. 21: Validation curve of SVM with RBF kernel on wine dataset

Based on figure 21, a lower gamma value is better choice.

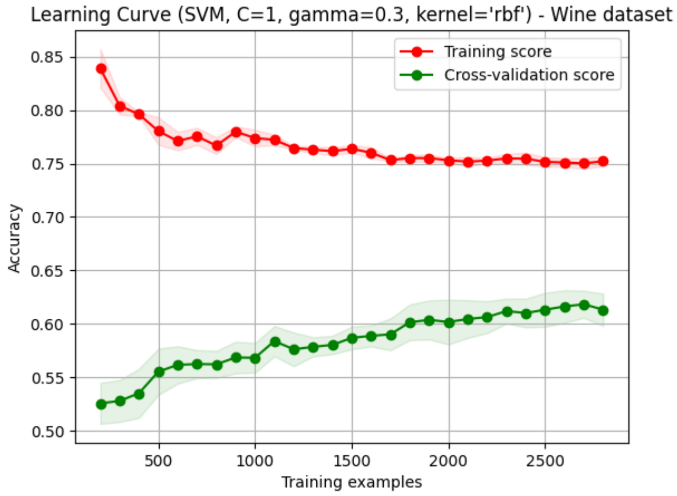

Fig. 22: Learning curve of SVM with RBF kernel on wine dataset

Our final model is SVM model with RBF kernel and gamma equal 0.3. Figure 22 showed that more training data does not improve performance further. This is as expected because SVM model is efficient with available data. Current model achieve low bias with variance is in acceptable range.

Table IV showed the prediction result of all three models. The accuracy score of KNN, NN and SVM are 0.60, 0.62 and 0.65 respectively. SVM outperformed the other model in overall prediction and accuracy. However, it produced worse prediction for 'low' quality class (low recall score). This is comprehensible because wine dataset is imbalanced. But it is unexpected because SVM performed very good on Diabetes dataset which is also imbalanced. Given this result, we should have applied SMOTE oversampling technique on wine quality dataset.Overall, SVM remains optimal choice due to its best prediction and high score in almost all categories.

KNN Model	Recall	F1-Score
low	0.51	0.55
medium	0.62	0.64
high	0.63	0.60
NN Model	Recall	F1-Score
low	0.63	0.66
medium	0.69	0.64
high	0.49	0.55
SVM Model	Recall	F1-Score
low	0.47	0.56
medium	0.66	0.69
high	0.73	0.66

TABLE IV: Classification report on Wine dataset without SMOTE

VI. Limitations

Despite of the SVM with RBF kernel superior overall performance, several limitations were identified. First, SVM struggled with the 'low' quality class in imbalanced white wine quality dataset, underscoring the need for oversampling or SMOTE. Second, RBF kernel is computationally demanding and require significant processing power and memory. Training and tuning SVM model in wine quality dataset took significantly longer time than for Diabetes dataset due to much bigger sample size. Lastly, while tuning hyperparameter, we found out that SVM model's performance is highly sensitive to the choice of hyperparameters like C or gamma. Therefore, tuning SVM model is crucial for optimal performance.

VII. Conclusion

In this study, we explored hyperparameter tuning and evaluated the performance of three supervised learning model (K-Nearest Neighbors - KNN, Neural Network - NN, and Support Vector Machine - SVM) on two imbalanced dataset: Diabetes and white wine quality. SVM with the RBF kernel emerged as the best model, demonstrating superior prediction accuracy and generalization ability. Despite its computational cost and sensitivity to hyperparameters tuning, SVM outperformed and can be better if combined with SMOTE to address class imbalance in wine dataset. These findings highlighted the importance of model selection and hyperparamater tuning in achieving optimal result for classification problem.

VIII. Resources

[1] *API Reference.* Scikit-learn. https://scikit-learn.org.
[2] UCI. Wine quality dataset https://archive.ics.uci.edu/dataset/186/wine+quality
[3] Kaggle. Diabetes dataset https://www.kaggle.com/datasets/mathchi/diabetes-data-set/data