Supervised Learning Assignment 1:
Five Model Evaluations on Two Datasets

Author Name
rjenez3@gatech.edu

***Abstract*—** This Assignment explores Supervised Learning techniques on five different models, and two datasets

1 INTRODUCTION

The goal of this Assignment is to take five different Supervised Learning models - Decision Trees, Boosted Decision Trees, Neural Networks, Support Vector Machines, and K-Nearest Neighbors, and apply them to two different data sets. Each of these models will be compared and contrasted against these two datasets.

The two datasets are an Airline Passenger Satisfaction survey and the CDC Behavioral Risk Factor Surveillance System dataset on Diabetes. In this analysis, I used all the tools available in the SciKit-Learn python ML learning module and the RandomizedSearchCV to optimize the models once I had found the key hyperparameters.

1.1 Datasets

The datasets I have selected are vital in determining the key differences between the models. I did an Exploratory Data Analysis (EDA) of each dataset below and arrived at a couple of posed problems that I hope these algorithms can answer.

1.2 Airline Dataset

The airline dataset was sourced from Kaggle (AirlinePassenger Satisfaction direct link in Jupyter notebook used to do this analysis Assignment1-DataAnalysis.ipynb). It has 103,904 samples in the training set and 25976 in the hold-out test set. The airline data seeks to find the relationship between customer satisfaction and some eighteen features related to the travel experience and the customers. The eighteen numerical features are a combination of factors that are attributable to the customer: Age; Aspects o the flight: Flight Distance, Departure Delay in Minutes, Arrival Delay in Minutes; And how the customer rated the service: Inflight wifi service, Departure/Arrival time

convenient, Ease of Online booking, Gate location, Food and drink, Online boarding, Seat comfort, Inflight entertainment, On-board service, Leg room service, Baggage handling, Checkin service, Inflight service, Cleanliness.

The non-numerical/categorical features transformed for the analysis were: Gender, Customer Type (disloyal/loyal), Type of Travel (personal/business), and Class of Service (Eco[Economy], EcoPlus[Economy Plus], and Business).

The categorical features were transformed to numerical ordinal and scaled to align with the other values since some models need values within the same scale to operate effectively.

There was no strong correlation between the features and satisfaction (in all cases < 0.3).

1.2.1 *Interest in the dataset*

This data is interesting because airlines are cost driven, and determining what key features drive satisfaction and loyalty for different classes of customers is a great operational example of using Machine Learning. Is there a simple way to improve the experience for some classes of customers?

After some detailed analysis, it was apparent that Business Passengers loyal to the airline had low satisfaction ratings (12.8 % in Economy Class satisfied and 20.4% in Economy Plus seating). What were the actions associated with the features that could remedy this situation?

1.2.2 *Hypothesis*

Due to the reasons above, I came up with the following Null Hypothesis:

No relationship exists between the features and the satisfaction of the Economy and Economy Plus, Business passengers.

We will score using an F1 metric, which balances Precision against Recall.

1.3 Diabetes Health Indicators Dataset

The Diabetes Dataset is taken from Kaggle (a link is in the Jupyter notebook used to do this analysis Assignment1-DataAnalysis.ipynb). This dataset is from The Behavioral Risk Factor Surveillance System (BRFSS), a health-related telephone survey collected annually by the CDC.

This dataset tries to link people with diabetes to several potential contributing health factors and socio-economic factors such as income. There are 253,680 samples, and I have split these into 80% training and 20% hold-out test set.

The features are: Diabetes_binary (do you have diabetes), HighBP (High Blood Pressure), HighChol (High Cholesterol), CholCheck (Cholesterol checked in the last five years), BMI (Body Mass Index), Smoker (have you smoked over 100 cigarettes in your lifetime), Stroke (ever told you had a stroke), HeartDiseaseorAttack(coronary heard disease or hear attack), PhysActivity (physical activity in the last 30 days), Fruits (1 or more fruits a day), Veggies (1 or more ad day), HvyAlcoholConsump (Heavy alcohol consumption - more than 14 drinks per week for men and 7 drinks per week women), AnyHealthcare (any healthcare coverage), NoDocbcCost (no doctor visits because of cost in the last 12 months), GenHlth (General health from 1 -excellent to 5 poor), MentHlth (mental health 1-30 scale, 1 excellent - to 30 poor), PhysHlth (physical health 1-30 scale - 1 excellent, 30 poor), DiffWalk (Difficulty walking or climbing stairs), Sex (female or mail), Age (13-level age category 1 = 18-24 9 = 60-64 13 = 80 or older), Education (Education level (EDUCA see codebook) scale 1-6 1 = Never attended school or only kindergarten 2 = Grades 1 through 8 …), Income (Income scale (INCOME2 see codebook) scale 1-8 1 = less than $10,000 5 = less than $35,000 8 = $75,000 or more)

I used a min-max scaler for most of the data to get everything in the same range of values.

In reviewing the correlation matrix, no features correlated higher than 0.3 with the outcome variable (Diabetes_binary).

1.3.1 *Interest in the dataset*

Diabetes is a common condition today, and screening is not often available to enough people to take care of this potentially deadly disease. Is there a simpler way to help screen vulnerable folks?

From an analysis standpoint, in most medical diagnoses, we want to be as accurate (accuracy) as possible so that when we say someone may wish to have further tests, we minimize the number of false positives (and false negatives).

1.3.2 *Hypothesis*

Null Hypothesis: No relationship exists between the feature (medical risk factors) and the ability to predict if someone has diabetes accurately.

The goal is to determine whether we can accurately recommend from the questions which people may have a higher risk of diabetes and encourage them to get further screening. We will use Accuracy to score the results.

2 ANALYSIS OF HYPERPARAMETERS FOR THE FIVE MODELS

In this section, we will discuss the analysis techniques that I used to determine how tune these models.

2.1 *Model Performance Based On Dataset*

For each Model, we will see how it performed on each dataset and on hyper-parameters that were most indicative of the performance and why.

2.1.1 *Decision Trees*

As we look at Decision Trees (Figures 1 and 2), let's look at the validation curves for our airline and diabetes data to orient ourselves concerning hyperparameters. Criterion (entropy, gini, log_loss) seem to impact the Airline data modeling but not so much on the Diabetes data. Pruning, when looking at the cc_alpha value (5th graph), it has a massive impact on the accuracy of the diabetes data, implying overfitting is possible, but less so with the airline data, hence a positive value for cc_alpha for the diabetes data.

A second pruning parameter, max_leaf_nodes, again shows that in the diabetes data constraining max_leaf_nodes more so than with the airline data prevents overfitting and improves results. The literature assumes that the dataset with fewer leaves is less complex. Still, I believe fewer good signals can be drawn from the dataset using Decision Trees.

3.2.2 Boosted *Decision Trees*

As we look at the Boosted Decision Trees (Figures 3 and 4), the validation curves point to hyperparameter values that are more pronounced. The learning rate (graph 2) seems to be smaller and less impactful in the Diabetes dataset vs. the Airline dataset.

This implies that we are less likely to overfit on the Airline dataset, whereas the Diabetes dataset may be more prone to overfitting. Picking the right value, in this case, seems easier since anything about 0.1 seems stable up to 0.2. I am still amazed by its dramatic effect on the Airline data's f1-score.

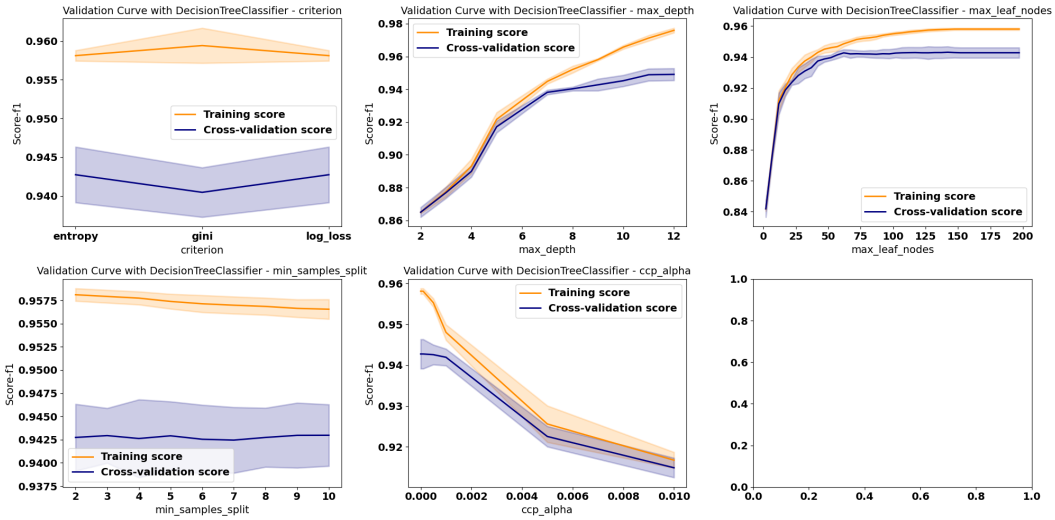

Figure 1— Decision Tree Airline Data Validation Curves.

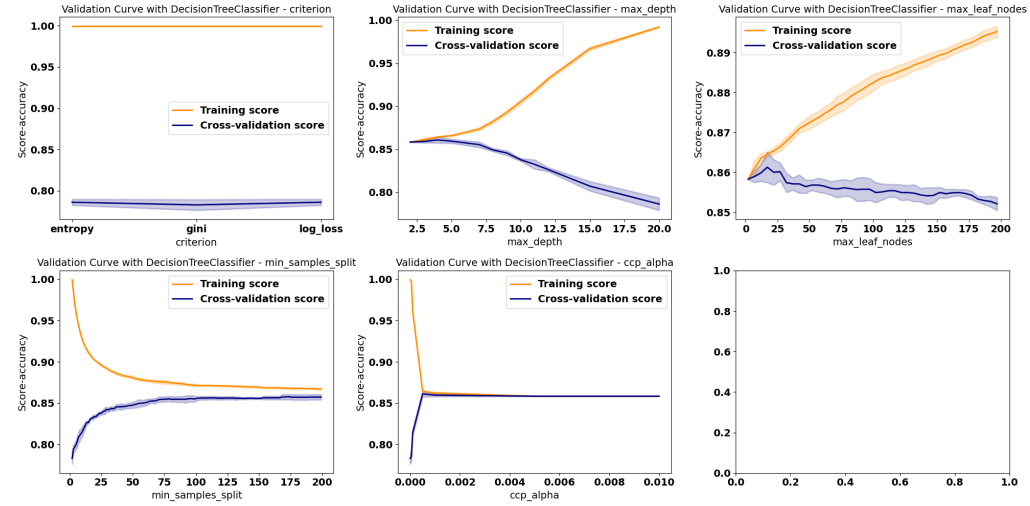

Figure 2—Decision Tree Diabetes Data Validation Curves.

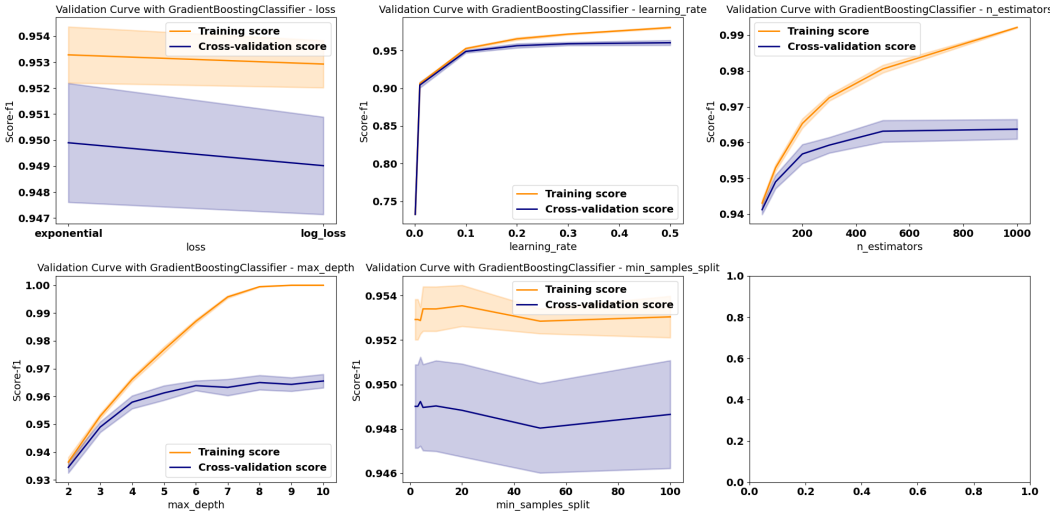

Figure 3— Boosted Decision Tree Airline Data Validation Curves.

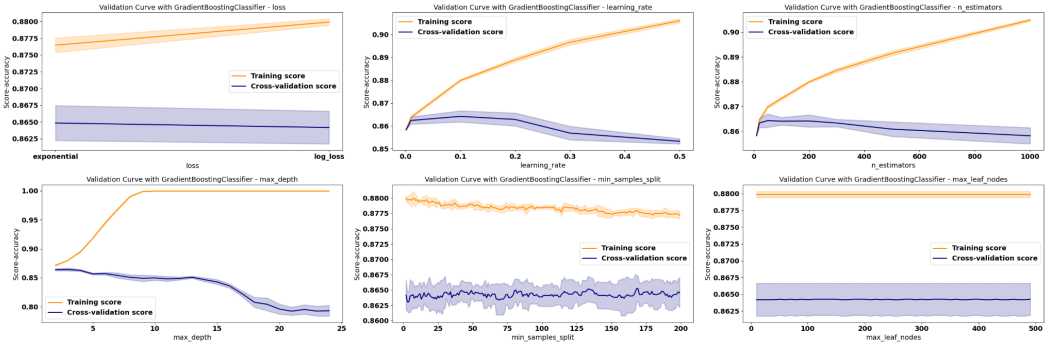

Figure 4—Boosted Decision Tree Diabetes Data Validation Curves.

3.2.2 K Nearest Neighbor

For K Nearest Neighbor (Figures 5 and 6), the number of neighbors, k, as we would assume, significantly impacts the model outcome. In the airline dataset, a relatively small value of k, between 3 and 7 (why do odd values seem to have higher results?), produce the best results, but in the Diabetes data, somewhere between 20 and 40 seems optimal. The high values of k for the Diabetes dataset could imply underfitting of the data. In contrast, a low value of k in the Airline data could suggest that we are more likely to overfit the data and be likely to adjust to succumb to noise.

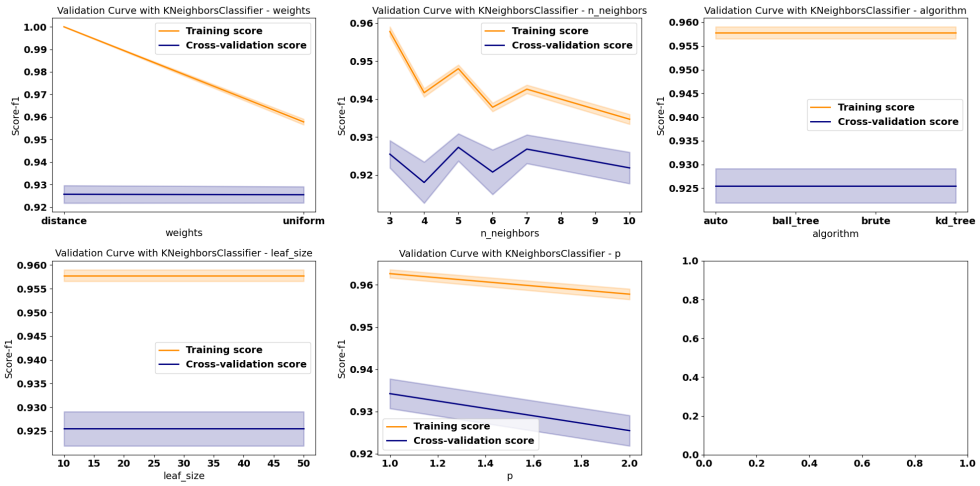

Figure 5 — K Nearest Neighbors Airline Data Validation Curves.

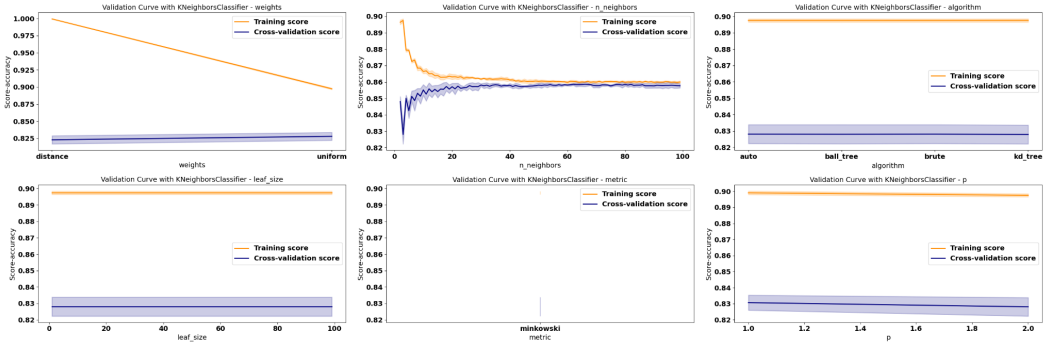

Figure 6 - K Nearest Neighbors Diabetes Data Validation Curves.

The weights (uniform/distance) hyperparameter, which determines whether we treat the points uniformly or weight them based on distance, seems to be equally set to the distance for both datasets, which means that not every point can be treated the same. It just means that some features are much more valuable to the model than others.

3.2.2 Support Vector Machines

For Support Vector Machines (Figures 7 and 8), the essential hyperparameter for these datasets is the value for C, the critical regularization parameter for the SKLEARN SVC model. A high C value means we are likely to overfit the data, and a low C value means we have higher bias and lower variance. With both datasets, we have values above 5, indicating that we are trending to overfitting (default values less than 1).

When we look at the kernels we use with the Airline dataset, Poly, and RBF seem to produce better results, while on the Diabetes dataset, we should consider Linear, Poly, and RBF. In both cases, sigmoid produces poor results. Certainly, it appears that the data in both cases (almost) is linearly separable, as there is not a significant difference between the Linear, Poly, and RBF. The Sigmoid function's poor performance certainly indicates higher dimensions are not strictly necessary in this model for these two datasets.

3.2.2 Neural Networks

As we look at Neural Networks (MLPClassifier - MultiLayer Perceptrons) (Figures 9 and 10), the validation curves for our Airline and Diabetes show the primary value I used first was the hidden layer sizes. In this case, I mainly focused on a single layer and scaled it up (I did try multiple layers but resisted going too deep for now- I just wanted to see how the models performed using too deep a number of layers).

For the Airline Data, I needed to use a higher number of perceptrons in each layer, up to 13, in a single layer, than for the Diabetes dataset. The expectation is that with more layers and Perceptrons, there is a chance of overfitting the data.

One other thing to consider was the activation function. There were four options (identity, logistic, tanh, relu), the relu function did best for the Airline Dataset, but the identity function seemed to do better for the Diabetes dataset. Again pointing to more of a linear relationship for accuracy with the Diabetes Dataset and a more complex relationship for the airline data.

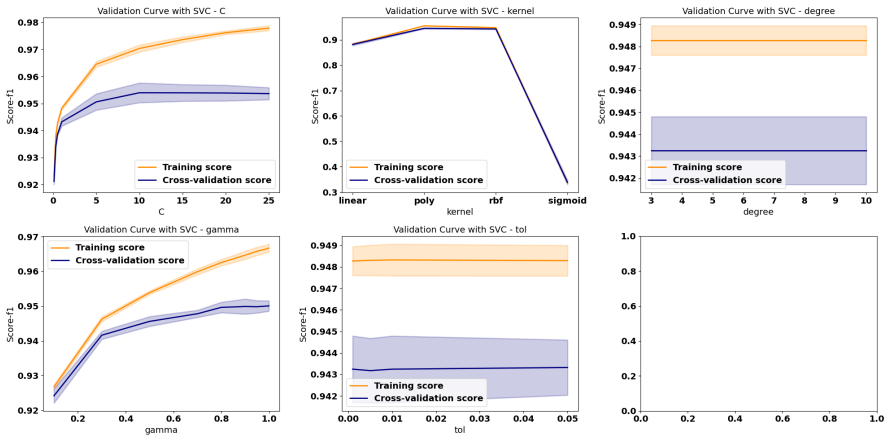

Figure 7— SVM Airline Data Validation Curves.

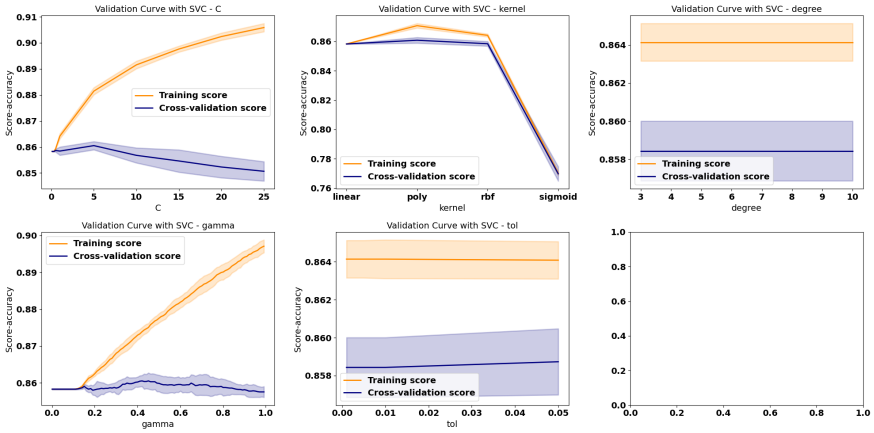

Figure 8—SVM Diabetes Data Validation Curves.

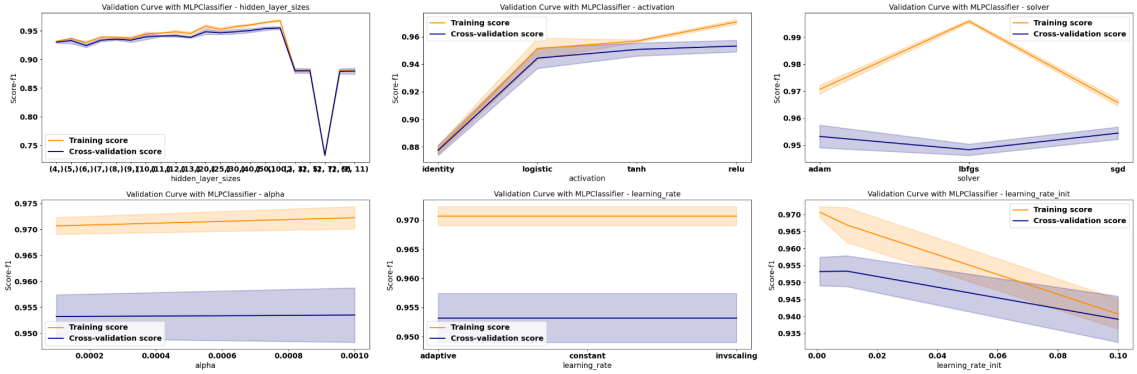

Figure 9— Neural Networks Airline Data Validation Curves.

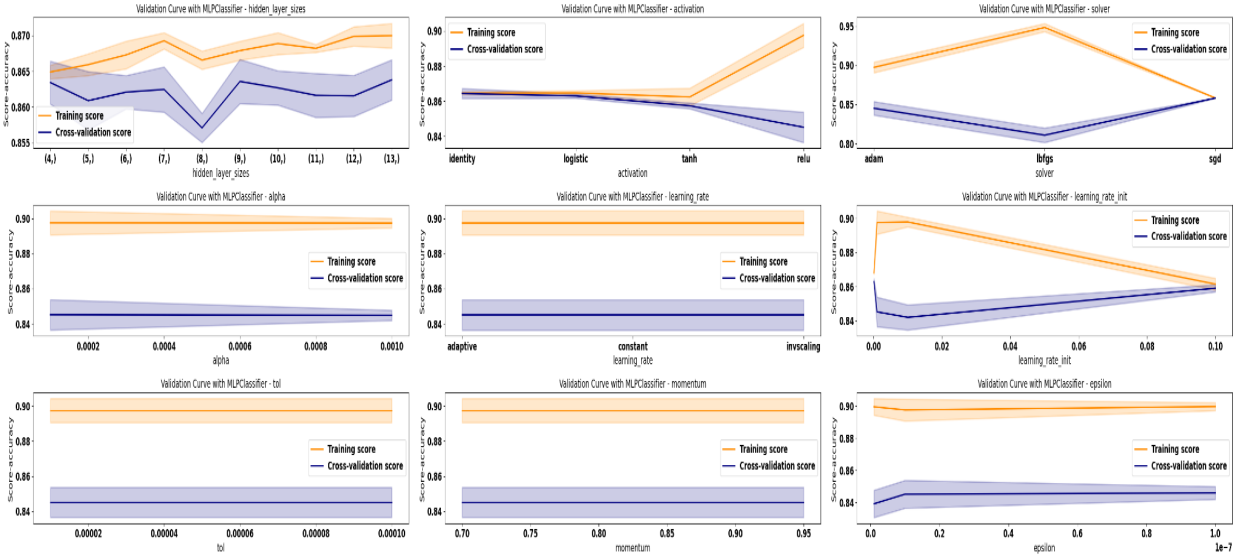

Figure 10—Neural Networks Diabetes Data Data Validation Curves.

4 OVERALL MODEL PERFORMANCE ON DATASETS

As we fine-tuned each model on the two datasets, it became apparent that these models each had somewhat similar performance. Still, in each case, one or two models seemed to do just a bit better on the test datasets.

I think I saw this kind of performance because these were well-curated datasets, and in each case, certain features predominated as reasons for the results that appeared.

For instance, in the Airline dataset, the main factor for business travelers achieving high satisfaction seems to be online boarding (the ability to check-in and complete boarding steps online, which improved the airport experience). In the case of the diabetes dataset, High Blood Pressure was a predominant factor that could indicate diabetes. I looked at the Receiver Operating Characteristic (ROC) and the Detection Error Tradeoffs (DET) curves for each model.

4.1 Airline Dataset ROC/DET Curves

We see that each model has a similar performance with AUCs above 0.98. Nonetheless, the Gradient Boosting Decision Tree had the highest AUC (0.99) and was better in determining the level of satisfaction, as shown in the DET curve.

This validates that Gradient Boosting does produce better results even when all models do well; it just works harder at finding the areas where the classification is weak and adjusting for this to produce a better overall model.

4.2 Diabetes Dataset ROC/DET Curves

As we look at the ROC/DET curves below, we can see that there is quite a difference in performance for each model when running against the Diabetes Dataset. Gradient Boosted Decision Trees and Neural Networks did much better (0.82 AUC) vs SVMs (worst - AUC of 0.70), Decision Trees (AUC 0.78), and K Nearest Neighbors (AUC 0.78).

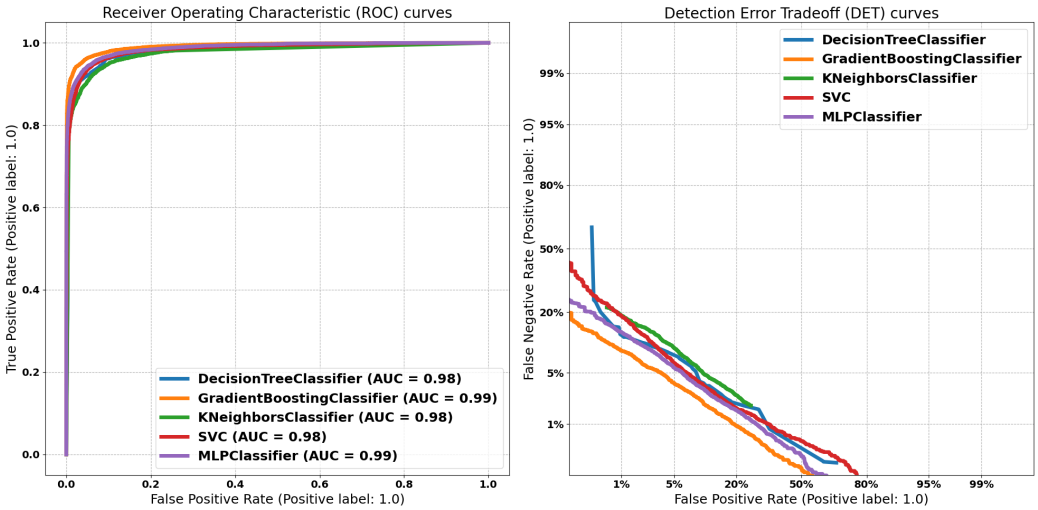

Figure 11—ROC/DET Curves For Airline Data

Here again, the Gradient Boosting Classifier outperforms the other models on this dataset because, again, this model digs in to create separation where the other models perform poorly. What is surprising is how badly SVC breaks down; more to consider her.

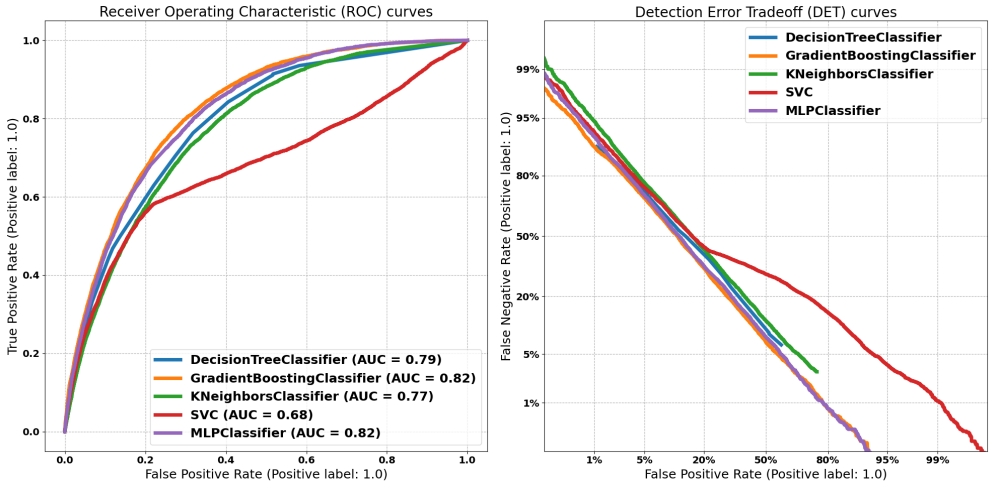

Figure 12—ROC/DET Curves For Diabetes Data

4.3 Model Time Performance

The last piece of the puzzle is to explore time complexity. From Table 1 we can see that GradientBoosting Decision Tree has equivalent training (fit) times to the SVM, but performs just as well NN/MLP in terms of prediction/Scoring time. In this case, maybe because of training time, we should have picked the MLP model?

Table 1—Model Fit (Training) and Score (Prediction) Times

Models	**Fit Times**	**Score Times**
Decision Trees	*10e-2 seconds*	*10e-3 second*
Gradient Boosted Decision Tree	*10e-0 second*	*10e-2 second*
K Nearest Neighbor	*10e-3 second*	*10e-1 second*
Support Vector Machine	*10e-0 second*+	*10e-1 second*
Neural Network/MLP	*10e-1 second*	*10e-2 second*

5 CONCLUSION

As a result of this process of comparing models, it cetainly has become apparent that most models will perform well on data if there are clear factors that are motivating outcomes - as in the case of the airline data. Where there are more complex datasets with multiple interrelated potential issues, as in the diabetes datasets, models that have tight feedback associated with their learning approach seem to do better as the Boosted Decision Trees, and the Neural network (MPLClassifier).

6 REFERENCES

1. SKlearn Documentation Manuals.
2. Mitchell, Tom: Machine Learning.

7 APPENDICES