COMPAS RECIDIVATE DATA

Multiple model comparison and prediction of recidivate offenders

Jinxi Luo

Table of Contents

Executive Summary

In this report we follow up from the last investigation on predicting general recidivism using the COMPAS data. SAS is the primary software used where five different classification models are constructed using the decision tree, random forest, SVM, neural network and model averaging ensemble framework. Parameter tuning showed that the best performing models did not incorporate categorical variables. With the decision tree plus neural network ensemble model resulting in the highest B.C.R of 69.57%, improving by 0.59% from the previous investigation's decision tree model.

Project Aim and Data Exploration

This project is a continuation from the last, with the same goal and data (Table 1) to predict all recidivation of criminals using the COMPAS decile score and offender covariates. The methodology in this project will differ with SAS being the primary software used to construct five separate predictive models using a decision tree, random forest, SVM, neural network and ensemble framework in the HPDM environment. Major parameter tuning for each model is done independently with a final comparison on the effectiveness of each model in predicting recidivation.

Table 1: Data dictionary for the current project.

Feature	Definition	Format
Age	Age of offender at recidivism charge date	Numerical: 18 to 96
Days before jail	Days between going in jail and offender arrest or offense date with the COMPAS screening	Numerical: -300 to 9,484
Days in jail	Days spent in jail for the crime with COMPAS screening	Numerical: 0 to 800
Decile score	The COMPAS evaluation score of offender recidivism risk	Numerical: 1 to 10
Juv counts	Sum of all juvenile offenses	Numerical: 0 to 21
Priors count	Count of all offender previous crimes (including juvenile counts) at COMPAS screening	Numerical: 0 to 38
Age category and crime charge degree *Age category and crime charge description* *Crime charge degree and race* *Crime charge description and race* *Crime charge description and sex*	Categorical variables that are the combinations of two of the original four categorical variables	Categorical: Var1level_Var2level

Previous exploration of the data is available in Appendix 1. In this report we will be more primarily interested in looking at the distribution of numerical variables plotted in two, three and higher dimensional spaces. As will be mentioned in the following section, categorical variables will be less significant to the analysis. We first begin looking at the distribution of values in the plot of decile score versus age (Figure 1). These two variables have been continuously found to have high feature importance in the previous decision tree models (Table 2).

Table 2: Variable importance of the T3 decision tree which had the best performance from the previous report.

Importance	*Variable*	*Importance Value*
1	age	1
2	decile_score	0.9538
3	priors_count	0.9368
5	juv_counts	0.8961
7	days_in_jail	0.8179

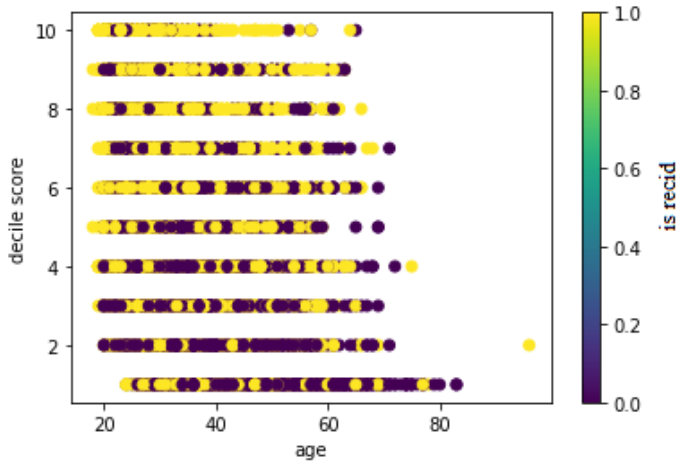

Figure 1: Scatter plot of decile score vs age. Yellow is positive recidivism with purple being negative.

So, it is hypothesized that these variables will be a good indicator of how the performance of other classification models may be affected. From Figure 1 we can see that there is a slight divide between recid and non recid criminals, where a higher density of yellow is above a decile score of 5 and higher density of purple below. There appears to be less correlation of recidivism with age, which contradicts what was seen in Figure 15 of Appendix 1 as there should be a higher density of recidivism in the younger age groups. Using a third variable to plot the data in 3D we can better view this characteristic.

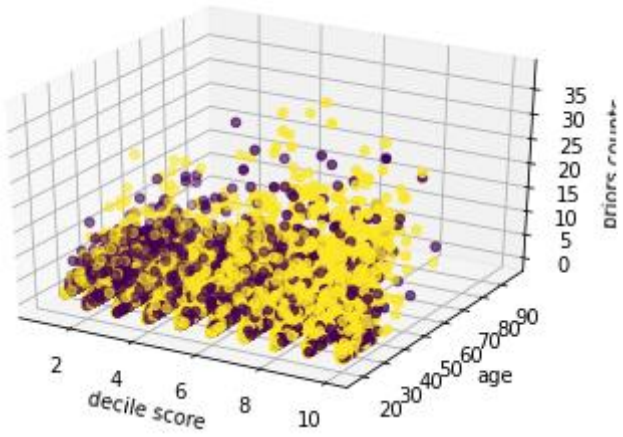

Figure 2: 3D scatter plot of the top 3 most important features.

Additionally, instead of a line, there now also appears to be a potential diagonal hyperplane that is able to roughly divide the yellow and purple circles. This is especially important when considering the SVM model where a nonlinear SVM may be needed to separate the data in a transformed scale. However, SVM work well with a high number of features and the data may be much more easily separated when considering the other three numerical variables in six dimensions. With neural networks the problem is similar in terms of separating data, where instead we may need to focus more on choosing additional hidden layers and a nonlinear activation function. To make predictions base on the division of data from multiple hyperplanes and potentially better model the data if there are nonlinear relationships. The latter however does not seem like it could be the case as a scatter plot matrix of the numerical variables is only represented by linear relationships (Figure 18 Appendix 1).

Lastly, as a random forest is generally just a lot of different decision trees. It should also be important to note what variables are used to train each tree, as decile score, priors count and age could potentially be features that are always used to train the trees due to their high importance resulting in a less robust forest.

Classification in SAS

Within the preliminary testing of each model using default settings. It was generally found that incorporating categorical variables significantly increased computation times, tended to cause overfitting of the training data with no added benefit in reducing validation classification errors. This was especially true for SVMs making it difficult to perform parameter tuning due to the high cardinality features. Thus, categorical variables will not be used in prediction. Continuing on from the last report for consistency, an 80:20 random split of training and validation data will also be used in each classification model. An example SAS node layout for this section can be seen in Figure 3 below.

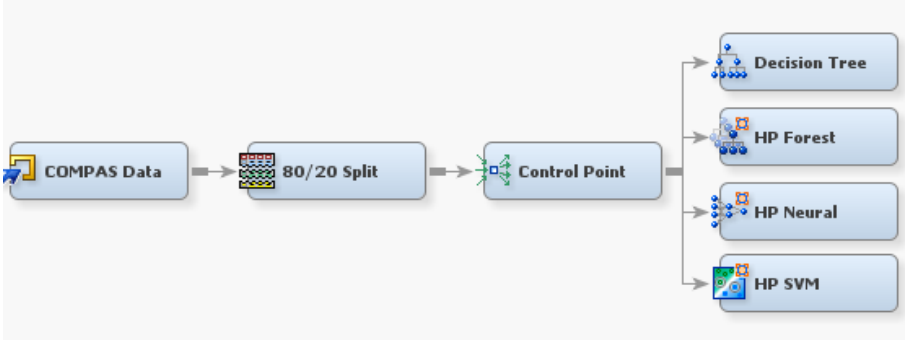

Figure 1: SAS node layout for the classification models.

It should be noted that while not visible in Figure 3 the results from the parameter tuning nodes has been found to differ from that of their respective final model for SVM and random forests. This issue is suspected to be caused by a difference in how data is partitioned in SAS code nodes but should not impact how the model is tuned.

Decision Tree

The tree model used in the classification is that of T3 from the previous report that had achieved the best results compared to similar trees through varying four major tuning parameters as shown in Table 3. Pruning categorical nodes and incorporating more high importance numerical variables in the decision tree by using ProbChisq, increasing surrogate rules while decreasing leaf size obtained higher B.C.R with the validation data.

Table 3: Major parameters for the decision tree.

Parameter	*Value*
Nominal Target Criterion	ProbChisq
Maximum depth	30
Leaf Size	10
Number of surrogate rules	20

Table 4: Performance measures for decision tree.

Measure	*Validation*	*Training*
Precision	0.689291	0.683232
Recall	0.655667	0.649243
F1-Score	0.672059	0.665804
B.C.R	0.689764	0.685316
Misclassification Error	0.309078	0.313291
Accuracy	0.690922	0.686709

Random Forest

Three major parameters (Table 5) are used for the random forest as suggested by the SAS guide (Wujek, 2015). These variables correspond to the number of trees in the ensemble which affects the prediction accuracy. Choosing how many randomly subset variables are used when splitting a node and the minimum leaf size of the nodes affect how generalized the individual tree models are; helping to prevent overall overfitting.

Table 5: Major parameters for the random forest.

Parameter	*Value*
Maximum Number of Trees	100
Number of Variables to Consider in Split Search	1
Smallest Percentage of Obs in Node	10

To tune the parameters, Figure 4 shows considering only one variable at each split provides the lowest error. While using 2 to 6 resulted in worse performance as the number of trees also increased. This could be explained by higher importance variables persistently being favoured when considering more than 2 variables, as similar results from the previous report had shown that five of the remaining six numerical variables had very high importance in the T3 decision tree. Then randomly choosing only one of the high importance features at random would be the best choice.

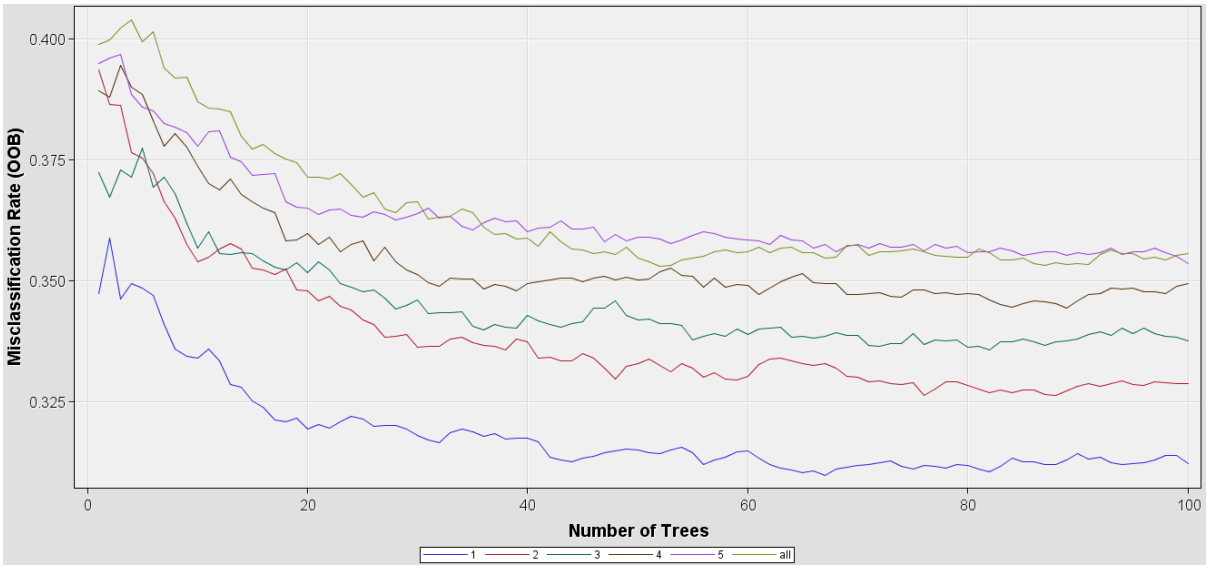

Figure 2: Out of bag misclassification error versus number of trees for each variable subset selection size.

As Figure 5 also shows, error rate starts to flat line near 100 trees suggesting that a choice of 100 for the maximum number of trees is an appropriate choice. Minimum node size of 10 also seem to work best as the number of trees increase.

Then model performance from those parameters can be seen in Table 6.

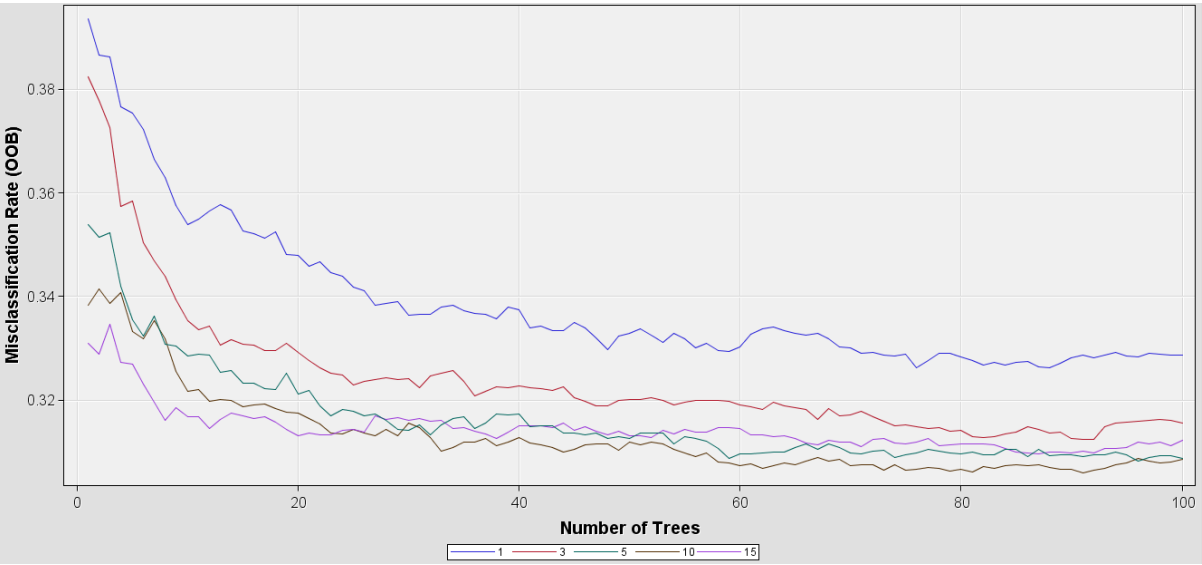

Figure 3: Out of bag misclassification verses number of trees for each minimum node size.

Table 6: Performance measures for random forest.

Measure	*Validation*	*Training*
Precision	0.719237	0.828856
Recall	0.595409	0.689618
F1-Score	0.651491	0.752853
B.C.R	0.689125	0.778910
Misclassification Error	0.307692	0.217640
Accuracy	0.692308	0.782360

Neural Network

Four major parameters (Table 7) are used for model tuning that are believed to have the most impact on prediction accuracy. One parameter is the network architecture, I will only focus on one and two hidden layer networks for simplicity, as it is hypothesised that a two-layer hidden network will perform better. Given that the data does not seem to be able to be separated by a single hyperplane, which in addition to changing the number of nodes in the hidden layers and the choice of activation function could improve predictions.

From experimentation it was confirmed that a two-layer architecture had achieved lower misclassification errors in both training and validation data. This was also seen when adjusting the number of hidden neurons by decreasing or increasing its value from a default of 3. Surprisingly when comparing the errors by changing activation functions there was minimal to no difference between runs. An explanation may lie from the linear functions computed by the neurons which have output values that are already easily divided before they are inputted into the activation function; with choices being identity, exponential, sine and tanh as provided in the HP Neural node. Final model performance measures are in Table 8.

Table 7: Major parameters for the neural network.

Parameter	*Value*
Architecture	Two Layer
Number of Hidden Neuron	3
Target Activation Function	Identity

Table 8: Performance measures for the neural network.

Measure	*Validation*	*Training*
Precision	0.704546	0.687870
Recall	0.622669	0.635540
F1-Score	0.661082	0.660670
B.C.R	0.689350	0.684310
Misclassification Error	0.308385	0.313810
Accuracy	0.691615	0.686190

SVM

The major parameters (Table 9) for the SVM will be the penalty, kernel, and kernel parameters, these were recommended by the SAS guide (Wujek, 2015) and accounts for the non-linearly separable data seen in exploration using the kernel trick.

Table 9: Major parameters for the SVM.

Parameter	*Value*
Penalty	5
Kernel	Polynomial (degree 2)

To tune parameters, a plot of penalty error for misclassified points in the margin versus each available kernel option is produced as seen in Figure 6. The polynomial kernel had the best results, and it can be seen in Figure 7, a penalty of 5 and 20 across different kernel parameter achieved the lowest validation errors. Thus, I will choose a penalty of 5 and keep the default kernel parameter at 2. The performance assessment measures of the final model are provided in Table 10.

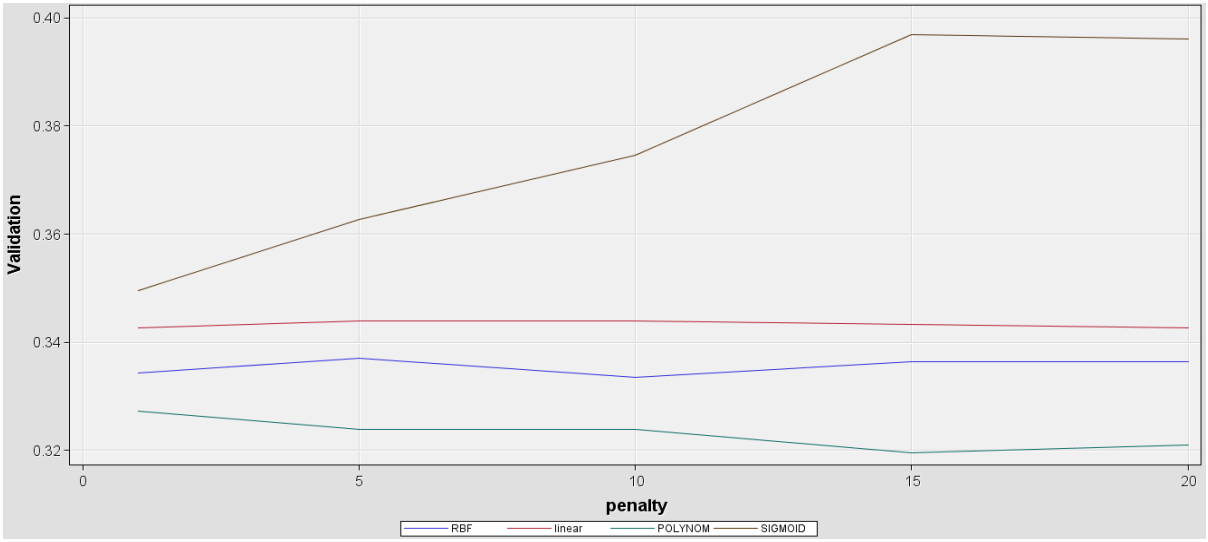

Figure 6: Validation error for each kernel.

Table 10: Performance measures for the SVM.

Measure	*Validation*	*Training*
Precision	0.715054	0.698488
Recall	0.572453	0.582913
F1-Score	0.635857	0.635488
B.C.R	0.679658	0.675007
Misclassification Error	0.316701	0.321435
Accuracy	0.683299	0.678565

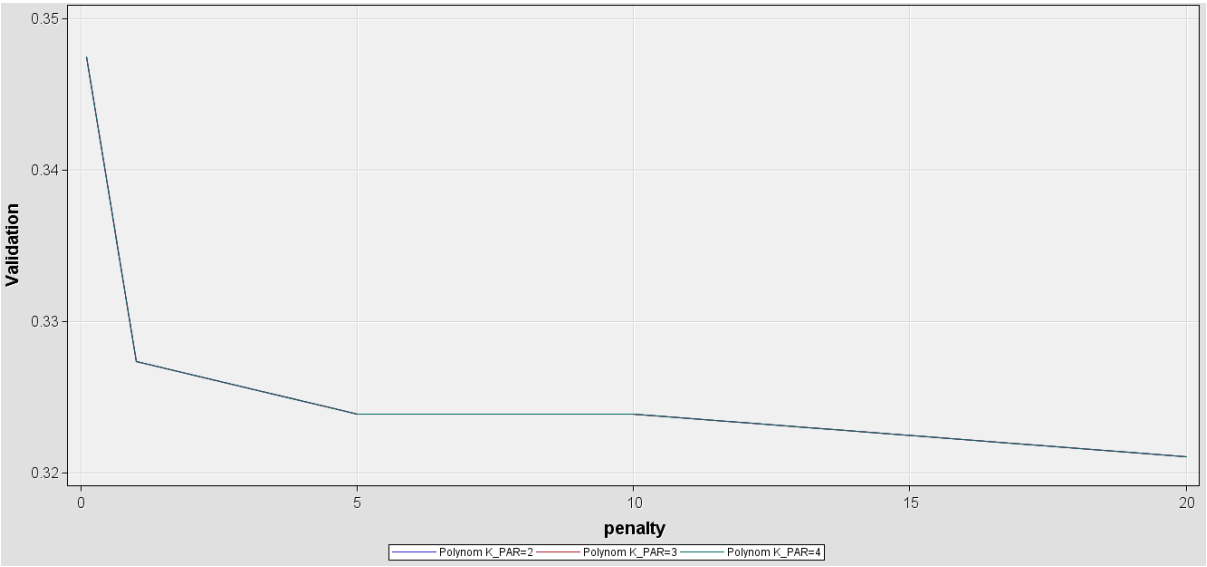

Figure 7: Validation error for each penalty and polynomial kernel parameter.

Creating an Ensemble Model

As we have already constructed four separate models to predict recidivism, for the ensemble, a model averaging approach with majority voting will be taken (model performance in Table 11). We will only be using the posterior probabilities from the decision tree and neural network models as the SAS ensemble node currently does not support output from the HP SVM and HP Forest nodes.

Figure 4: SAS node layout for the ensemble model.

Table 11: Performance measures for the ensemble.

Measure	*Validation*	*Training*
Precision	0.666667	0.657414
Recall	0.734577	0.725667
F1-Score	0.698976	0.689856
B.C.R	0.695707	0.687825
Misclassification Error	0.305613	0.313637
Accuracy	0.694387	0.686363

Model Comparison and Conclusion

Like the last report, the Balanced classification rate will be used as the main model performance evaluation metric for the validation data to account for the class imbalance (as seen in Figure 13 Appendix 1). A general view of the model performances in Table 12 shows that the Ensemble model performed the best with SVM doing the worse. The ensemble's performance is expected as it combines the posterior probabilities of the top two performing models that are independent of each other and have an accuracy greater than 50%. It would however be of interest to see if combining all four models together could benefit the bias variance trade off, when SAS adds data step code output support for the HP SVM and Forest nodes (Figure 19 Appendix 2).

Table 12: B.C.R and misclassification error comparison of validation and training data for each model.

Model	*B.C.R Validation*	*Misclass% Validation*	*Misclass% Training*
Ensemble (Tree/Neural)	0.695707	0.305613	0.313637
Decision Tree	0.689764	0.309078	0.313291
Neural Network	0.689350	0.308385	0.313810
Random Forest	0.689125	0.307692	0.217640
SVM	0.679658	0.316701	0.321435

Performance for the SVM and Random Forest is harder to explain as parameter tuning had found the optimal settings. Looking at the misclassification errors we can see that training errors in both validation and training were relatively high for the SVM model, this could indicate that there may be other parameters that should be tuned. Or that the data may not be the best for the SVM model, as it is known that one drawback for SVM is separating overlapping classes (with an example in Figure for 10) making it harder for a decision boundary to fully separate data points.

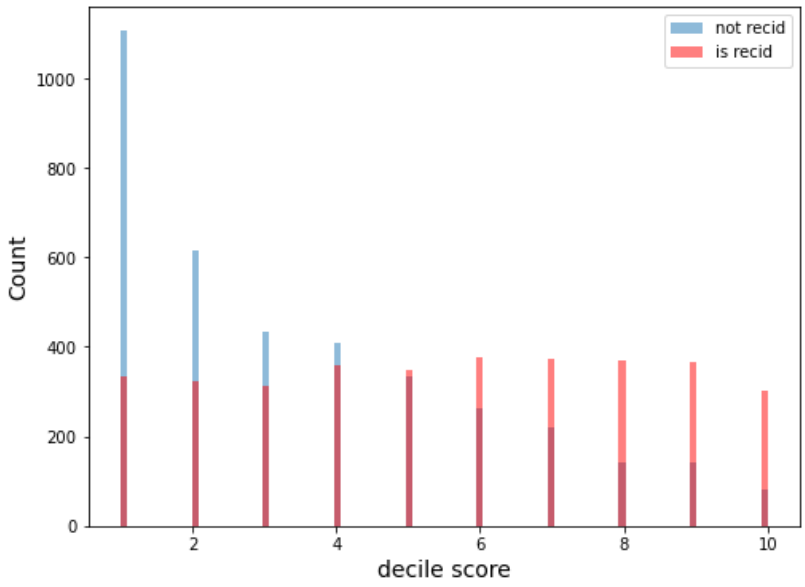

Figure 5: Example of target class overlapping with decile score.

The random forest model has the lowest misclassification rates in general but not a very good B.C.R. This would be due to how the B.C.R is calculated from a lower recall (and possibility specificity compared to the other models) which resulted from the model over fitting the training data. However, in general, the forest still performed better than most of the other models and so may still be regarded as valid.

We can then examine the validation ROC plots (Figure 11) for the data and it is seen that the ensemble model may perform worse than the other models when sensitivity and specificity is low or high, otherwise all the model's performance are hard to distinguish. The same performance characteristics are also seen in the lift charts through all proportions of the data as the plots for all the models tend to fluctuate (Figure 12).

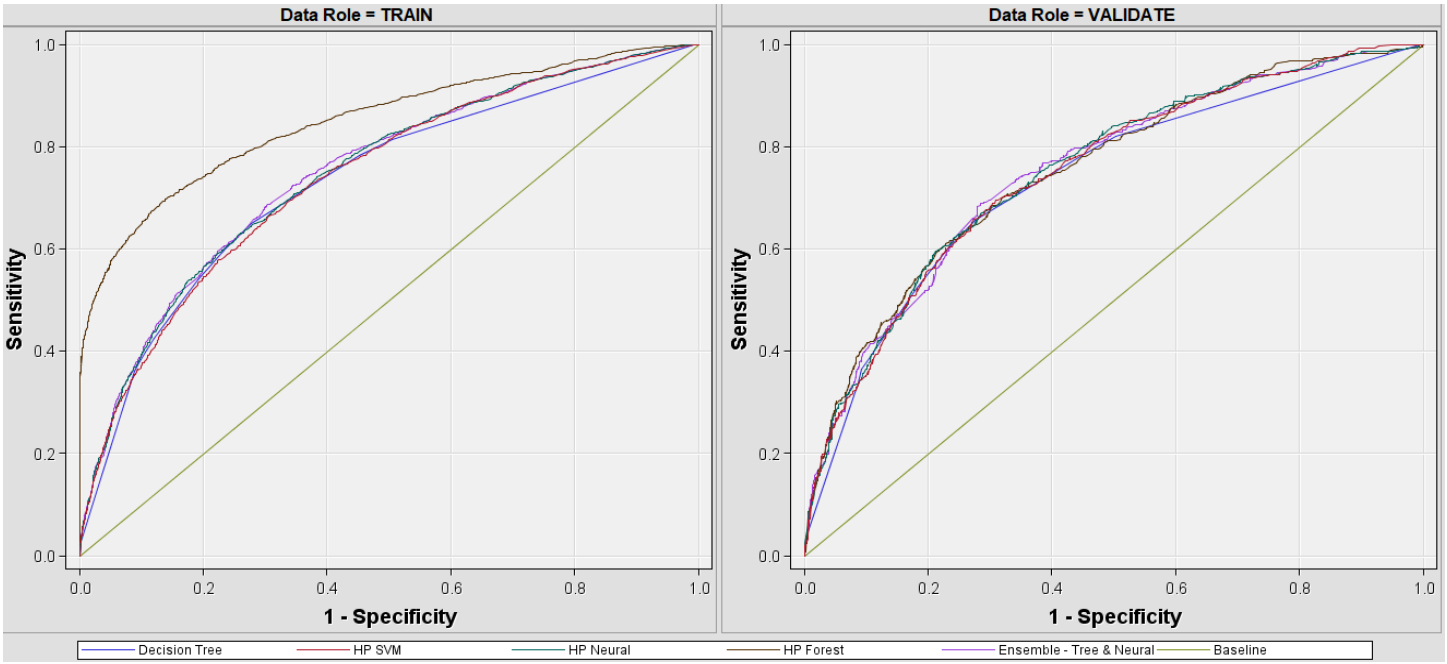

Figure 7: ROC plots of validation and training data for each model.

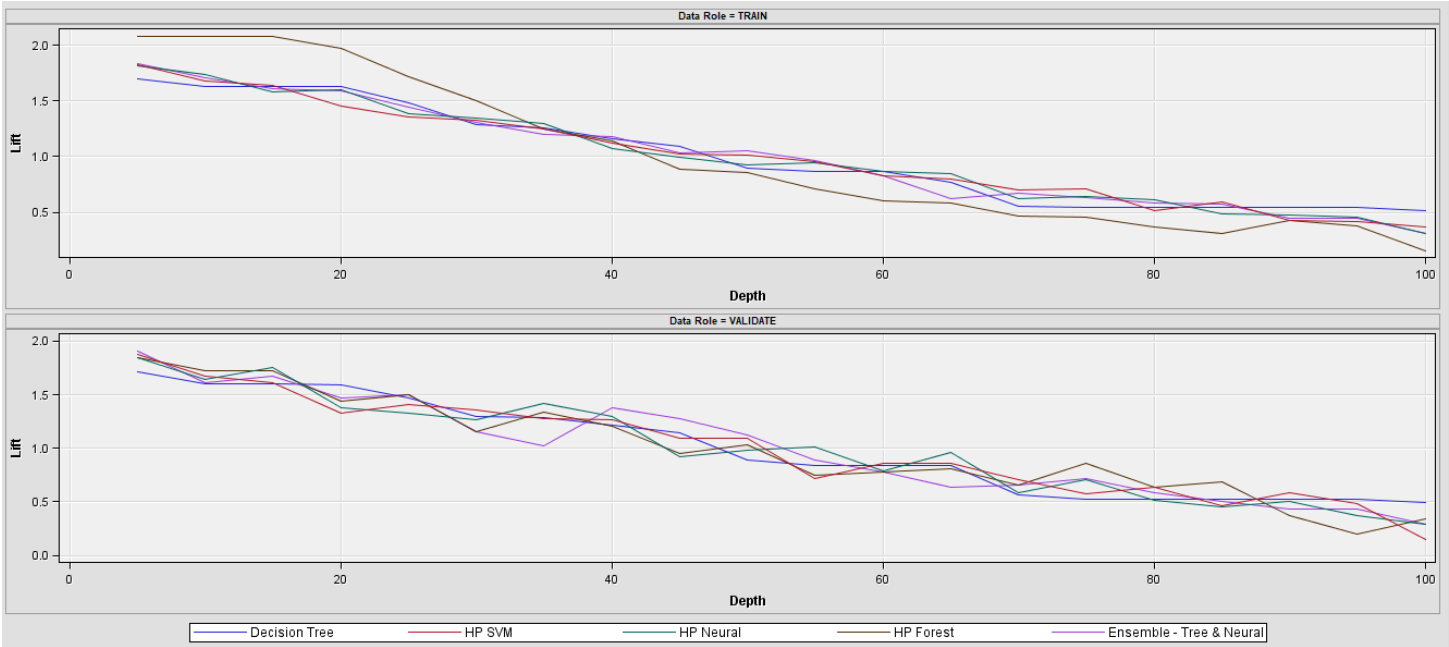

Figure 6: Lift plots of validation and training data for each model.

Appendix 1 – Previous Results

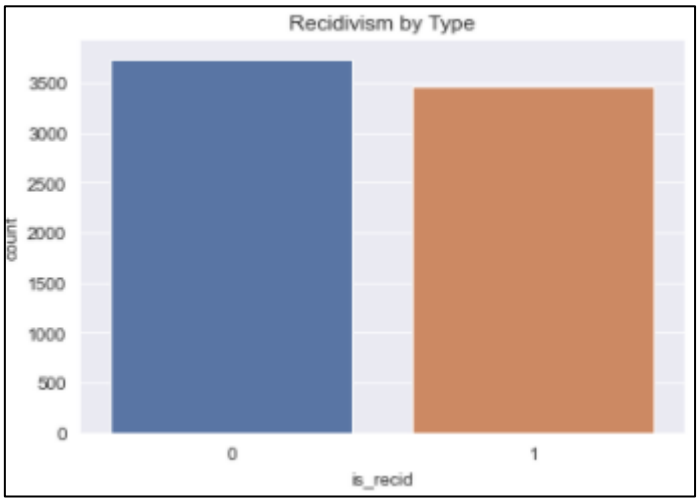

Figure 8: Bar plot of recidivism.

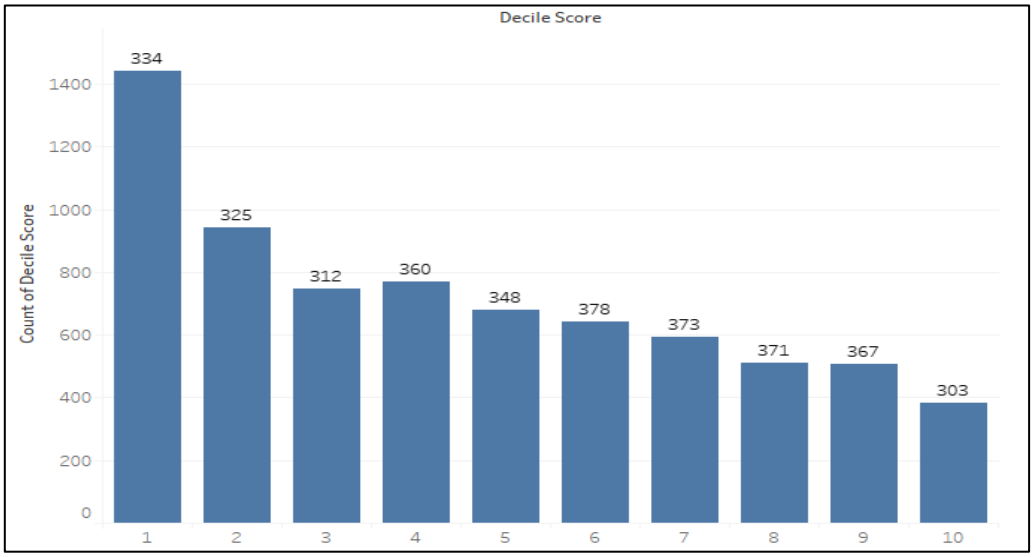

Figure 9: Bar plot of decile score with count of re-offense per bin.

Table 13: Summary statistics for the remaining numerical variables.

	Days Before Jail	Days in Jail	Juv Counts	Prior Counts
Mean	40.07	17.93	0.27	3.47
STD	291.85	50.12	0.95	4.88
Min	-300	0	0	0
25%	0	1	0	0
50%	0	1	0	2
75%	0	9	0	5
Max	9484	800	21	38

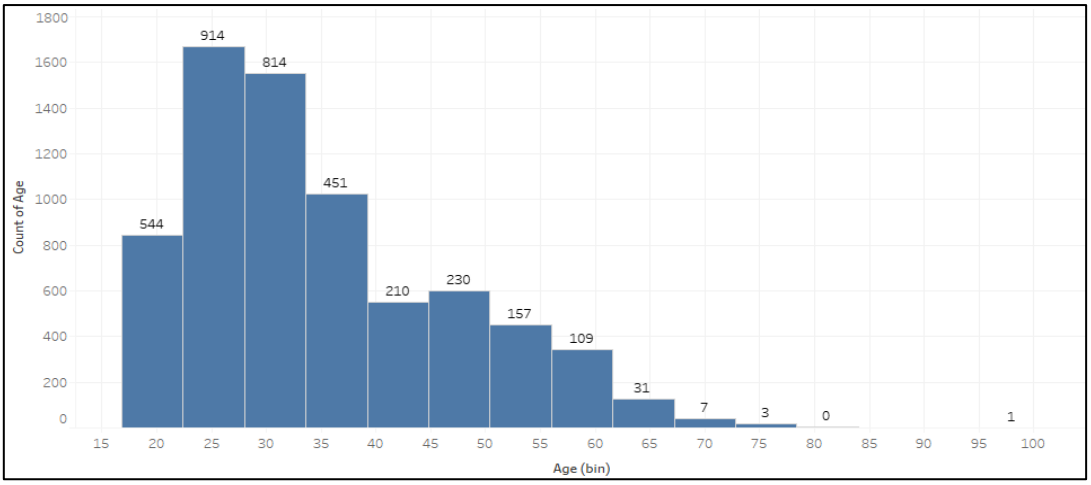

Figure 15: Histogram of age with numbering showing cases of recidivism.

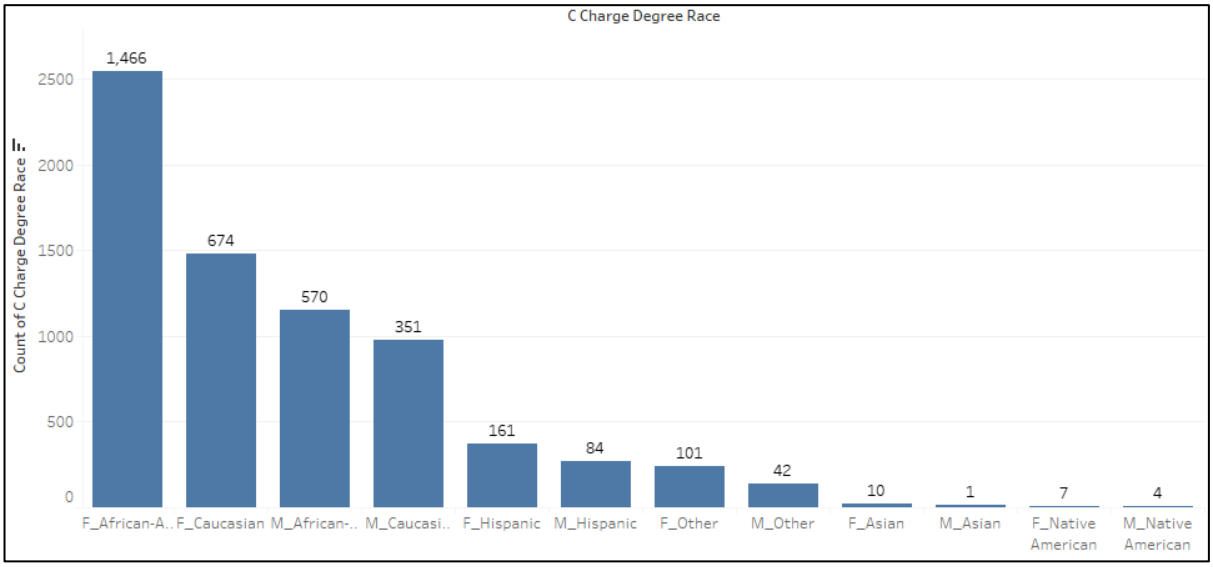

Figure 16: Histogram of crime charge degree and race.

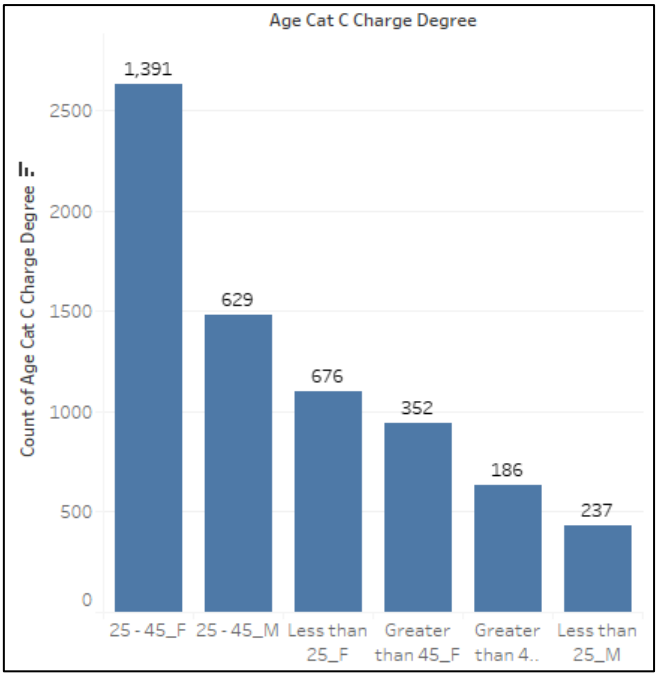

Figure 17: Histogram of Age category and crime charge degree.

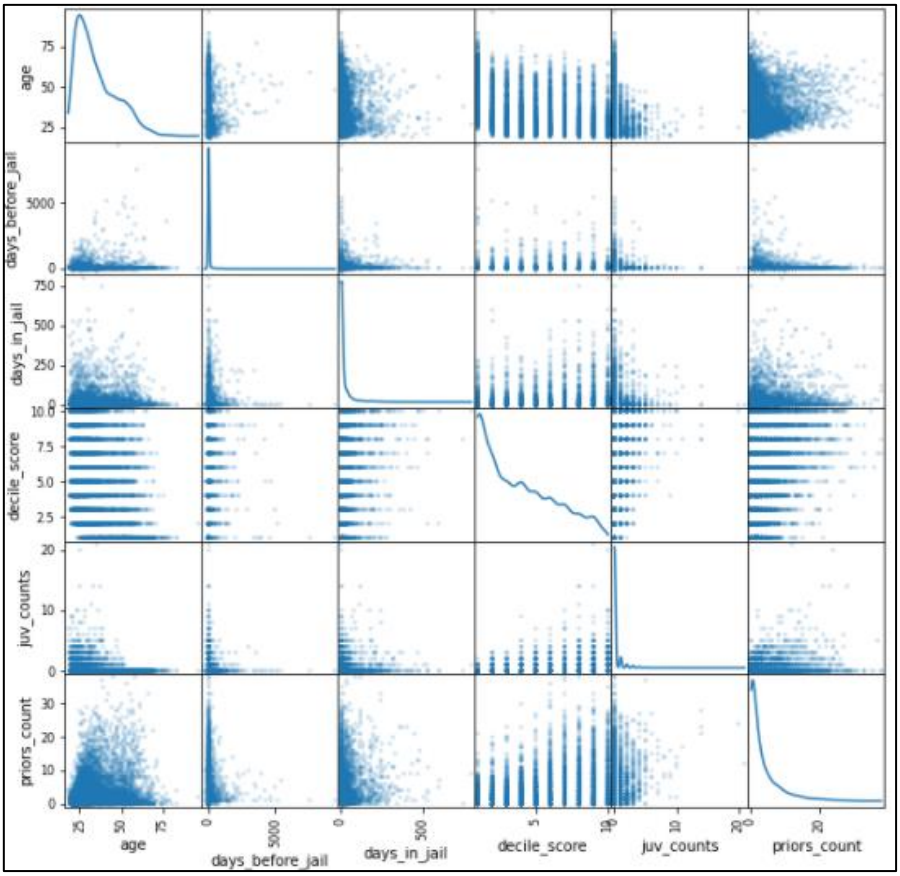

Figure 18: Scatter plot matrix for all numerical variables.

Table 14: Summary of the remaining categorical variables.

	C Charge Desc Race	C Charge Desc Sex	Age Cat C Charge Desc
Proportion Most Common %	8.9	13.0	9.5
Most Common	arrest case no change_African-American	arrest case no charge_Male	25-45_Battery
Cardinality	845	578	739

Appendix 2 – Present Results

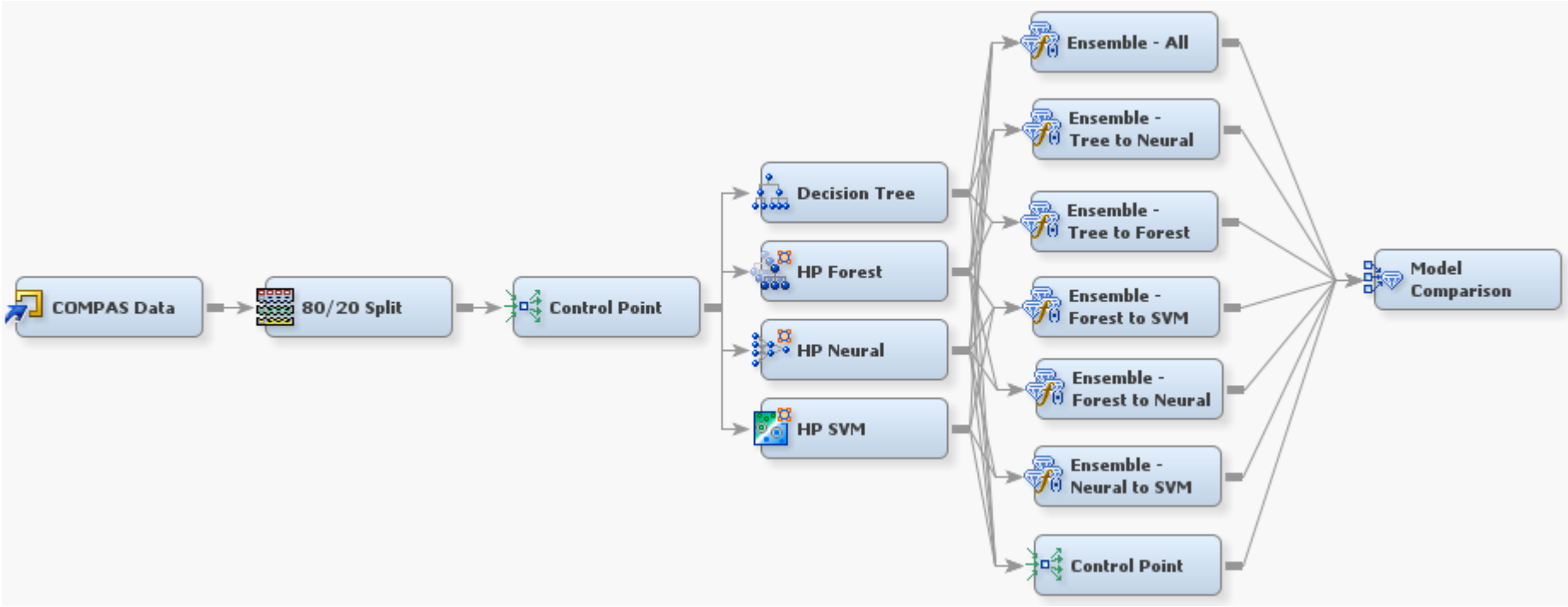

Figure 10: Example SAS node layout for ensembles using all the models.

Reference

Wujek, B., 2015. Tip: Getting the Most from your Random Forest. [Blog] *SAS Communities Library*, Available at: <https://communities.sas.com/t5/SAS-Communities-Library/Tip-Getting-the-Most-from-your-Random-Forest/ta-p/223949> [Accessed 7 November 2020].