**TEAM AGREEMENT TEMPLATE & GUIDELINES**

**For**

**<Data Scientist>**

**Prepared by:**

***Team Member:***

***Minwoo kang n9913351***

***Mary Rose n10086820***

***LunLu n0000000***

***Project Due Date***

***<07/04/2019>***

# Sign-off and Approvals

|  |  |  |
| --- | --- | --- |
| **Team Agreement Sign-Off:** | | |
| The undersigned members of this team agree to abide by this team agreement to ensure the successful completion of the ***<insert project name>*** project to meet the client’s requirements and timeframes. | | |
| Student number & name | Signature | Date |
| 1. ***Minwoo kang / n9913351*** | *Minwoo kang* | *8/03/2019* |
| 1. ***Mary Rose / n10086820*** |  |  |
| 1. ***LunLu / n0000000*** |  |  |

# Document Change History

| Revision Number | Date of Issue | Author(s) | Brief Description of Change |
| --- | --- | --- | --- |
| 1 | 8-Apr-2019 | SK | Initial working draft, based upon semester 2 template |
| 2 | 8-Apr-2019 | SK | Incorp. RN feedback – Communications and Code of Conduct moved to sep doc. |
| 3 | 8-Apr-2019 | RN | Align project details with project overview and background briefing notes. |
| 4 | 8-Apr-2019 | RN | Updated to incorporate revised project requirements |
| 5 | 8-Apr-2019 | RN | Separate Team Agreement developed. |
| 0.6 | 8-Apr-2019 | SK | Team Agreement updated. |
| 0.7 | 8-Apr-2019 | RN | Agreement & Guidelines Merged |
|  |  |  | ***Add intermediate versions as required*** |
| *1.0* | ***dd-mm-yyyy*** | ***Initials*** | ***Version to be submitted*** |
|  |  |  | ***Add subsequent versions if required*** |

***Instructions: You should use this template to plan and discuss your team agreement by substituting and adding your own ideas and text wherever there are italics throughout the document.***

Table of Contents

[Sign-off and Approvals 2](#_Toc4752973)

[Document Change History 2](#_Toc4752974)

[1 Task 1. Data Selection and Distribution. 4](#_Toc4752975)

[2 Task 2. Predictive Modelling Using Decision Trees 4](#_Toc4752976)

[3 Task 3. Predictive Modelling Using Regression 5](#_Toc4752977)

[4 Task 4. Predictive Modelling Using Neural Networks 5](#_Toc4752978)

[5 Task 5. Generating an Ensemble Model and Comparing Models 6](#_Toc4752979)

[6 Task 6. Final Remarks: Decision Making 7](#_Toc4752980)

# Task 1. Data Selection and Distribution.

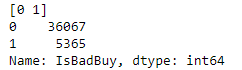
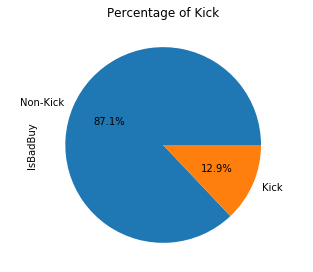
**1. What is the proportion of cars who can be classified as a “kick”?**

Figure 1

12.9% (5365) is categorised into a “kick” while 87.1% (36067) is categorised into a “non-kick”

**2. Did you have to fix any data quality problems? Detail them.**

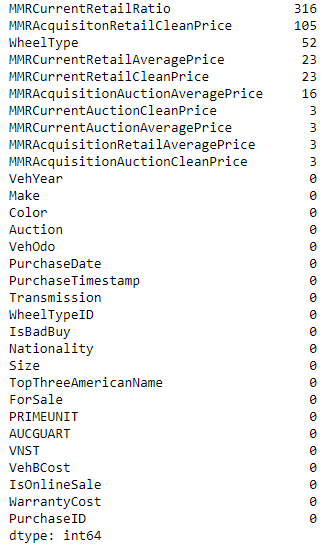
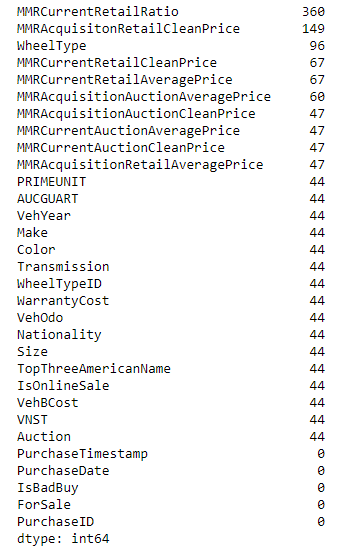


Figure 2 Figure 3

1. By using“MissingData = df1.isnull().sum()” codes, we identified there are some missing values in various columns depicted in figure 2. At first, we explored dataset where 44 null values appeared and decided to remove entire rows depicted in figure 3.
2. In Colour column, “?” and “NOT AVAIL” are replaced to OTHER
3. In Transmission column, “?” is replaced into “AUTO” as “AUTO” is representing majority in a Transmission column. “Manual” is replaced into “MANUAL” for consistency
4. “?” is replaced into 1 as “1” is representing majority in a “WheelTypeID” column. “WheelTypeID” column is transformed into a categorical type.
5. “nan value” and “?” are replaced into “Alloy” as it is representing majority in “WheelType” column.
6. “?”, “USA”, and “OTHER” are replaced into AMERICA as it is representing majority in a NATIONALITY column.
7. “?” is replaced into “MEDIUM” as it is representing majority in a SIZE column.
8. “?” is replaced into “GM” as it is representing majority in a TOPAMERICANNAME column
9. MMRAcquisitionAuctionAveragePrice: “?” is replaced placed and then it is filled with mean values calculated in the column.
10. MMRAcquisitionAuctionCleanPrice: “?” is replaced placed and then it is filled with mean values calculated in the column.
11. MMRAcquisitionRetailAveragePrice: “?” is replaced placed and then it is filled with mean values calculated in the column.
12. MMRAcquisitonRetailCleanPrice: “?” is replaced placed and then it is filled with mean values calculated in the column.
13. MMRCurrentAuctionAveragePrice: “?” is replaced placed and then it is filled with mean values calculated in the column.
14. MMRCurrentAuctionCleanPrice: “?” is replaced placed and then it is filled with mean values calculated in the column.
15. MMRCurrentRetailAveragePrice: “?” is replaced placed and then it is filled with mean values calculated in the column.
16. MMRCurrentRetailCleanPrice: “?” is replaced placed and then it is filled with mean values calculated in the column.
17. MMRCurrentRetailRatio: “?” is replaced placed and then it is filled with mean values calculated in the column.
18. “?” is replaced into 0 in VehBCost column
19. IsOnlineSale: ‘?’,’2’,’4,’ are replaced into ‘0’ as it is representing majority and ‘-1.0’,’0.0’ are replaced into ‘1’ and ‘0’ for consistency. Afterwards, the type of the column is changed into categorical type
20. ForSale: ‘yes’,’YES’ are replaced into ‘Yes’ for consistency. ‘?’ is replaced into “Yes” representing majority values. ‘0’ is replaced into ‘No” as it can be understood as no in binary.

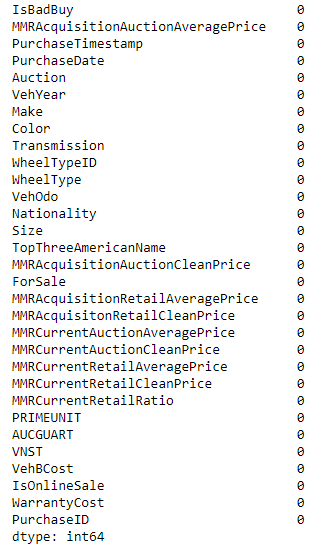
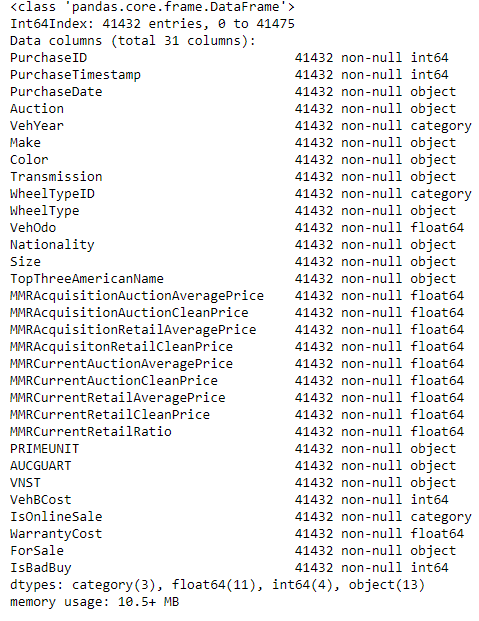


Figure 4 Figure 5

**3. Can you identify any clear patterns by initial exploration of the data using histogram or box plot?**

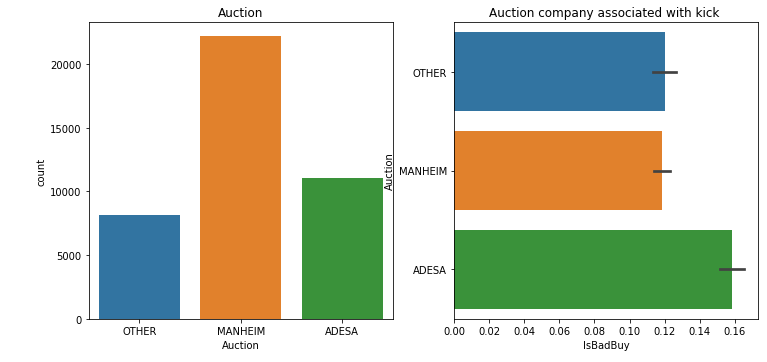


Figure 6

In figure 6, ADESA shows higher probability to be categorised as a kick in consideration to number of cars.

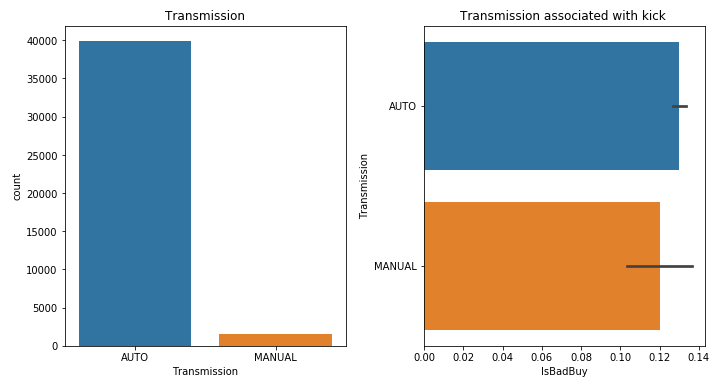


Figure 7

In figure 7, Auto has higher probability to be categorised as a kick in consideration to number of cars.

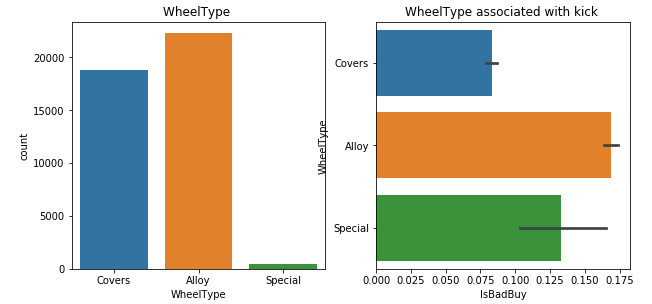


Figure 8

In figure 8, Alloy wheel has higher probability to be categorised as a kick in consideration to number of cars.

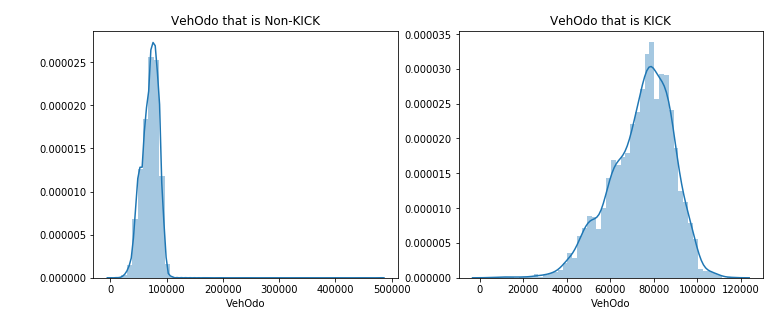


Figure 9

In figure 9, cars drive between 70000 and 90000 has higher probability to be categorised as a kick.

**4. What variables did you include in the analysis and what were their roles and measurement level**

**set? Justify your choice.**

We include all except for ’PurchaseID', 'PurchaseTimestamp', 'PurchaseDate' , 'WheelTypeID', 'PRIMEUNIT', 'AUCGUART', 'IsOnlineSale', 'ForSale'

* ’PurchaseID', 'PurchaseTimestamp', 'PurchaseDate' are excluded as it is not necessary information for training model
* 'WheelTypeID' is excluded as ‘WheelType’ exist
* 'PRIMEUNIT', 'AUCGUART' are excluded as majority data are missing
* 'IsOnlineSale', 'ForSale' are excluded as one values in two are extremely surpassing.

**5. What distribution scheme did you use? What data partitioning allocation did you set? Explain your selection.**

# Task 2. Predictive Modelling Using Decision Trees

**1. Python: Build a decision tree using the default setting.**

a. What is the classification accuracy on training and test datasets?

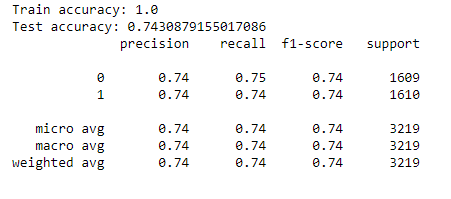


Figure 10

b. What is the size of tree (i.e. number of nodes)?

The depth of a decision tree is the length of the longest path from a root to a leaf.

The size of a decision tree is the number of nodes in the tree.

c. How many leaves are in the tree that is selected based on the validation data set?

d. Which variable is used for the first split? What are the competing splits for this first split?

e. What are the 5 important variables in building the tree?

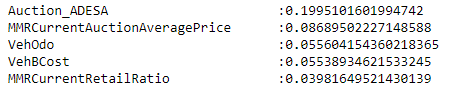


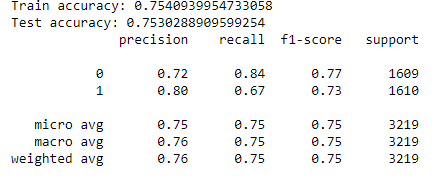
Figure 11

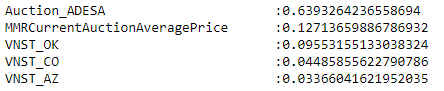
f. Report if you see any evidence of model overfitting.



Figure 12

g. Did change the default setting (i.e., only focus on changing the setting of the number of splits to create a node) help improving the model? Answer the above questions on the best performing tree.





2. Python: Build another decision tree tuned with GridSearchCV

a. What is the classification accuracy on training and test datasets?

b. What is the size of tree (i.e. number of nodes)? Is the size different from the maximal tree or the tree in the previous step? Why?

c. How many leaves are in the tree that is selected based on the validation data set?

d. Which variable is used for the first split? What are the competing splits for this first split?

e. What are the 5 important variables in building the tree?

f. Report if you see any evidence of model overfitting.

g. What are the parameters used? Explain your choices.

3. What is the significant difference do you see between these two decision tree models (steps 2.1 & 2.2)? How do they compare performance-wise? Explain why those changes may have happened.

4. From the better model, can you identify which cars could potential be “kicks”? Can you provide some descriptive summary of those cars?

# Task 3. Predictive Modelling Using Regression

1. In preparation for regression, is any imputation of missing values needed for this data set? List the variables that needed this.

2. Apply transformation method(s) to the variable(s) that need it. List the variables that needed it

3. Build a regression model using the default regression method with all inputs. Once you done it, build another one and tune it using GridSearchCV. Answer the followings:

h. Name the regression function used.

i. How much was the difference in performance of two models build, default and optimal?

j. Show the set parameters for the best model. What are the parameters used? Explain your decision. What are the optimal parameters?

k. Report which variables are included in the regression model.

l. Report the top-5 important variables (in the order) in the model.

m. What is classification accuracy on training and test datasets?

n. Report any sign of overfitting.

3. See whether you can further improve the performance by applying transformation to regularize input distributions. Report the variables that required transformation. What transformation function did you use and why?

4. Choose the best model in previous step (step 3.2: SAS) to apply transformation of variables. Does it improve the performance?

5. Using the best regression model, which cars could potential be “kicks”? Can you provide some descriptive summary of those cars?

# Task 4. Predictive Modelling Using Neural Networks

1. Build a Neural Network model using the default setting. Answer the following:

a. What is the network architecture?

*The default network architecture for the neural network consist of the following:*

*Input : 126*

*Hidden layers: 100*

*Output : 2*

b. How many iterations are needed to train this network?

*A total of 115 iterations are needed to train the default neural network.*

*­*

c. Do you see any sign of over-fitting?

*A sign of overfitting can be seen in the accuracies of the model. The train accuracy results in 96% whilst the test accuracy is only 79%. A clear sign that the model performs well on the training set but diminishes on the test set.*

d. Did the training process converge and resulted in the best model?

*The training process converged as there was no warning shown during training. At this phase, it is early to assume that the default setting of the neural network is the best model. Moreover, as there is a huge difference between the training and test accuracy, this model is unlikely the best model.*

e. What is classification accuracy on training and test datasets?

*Figure xxx shows the accuracy of training and test datasets (0.966 and 0.799 respectively).*

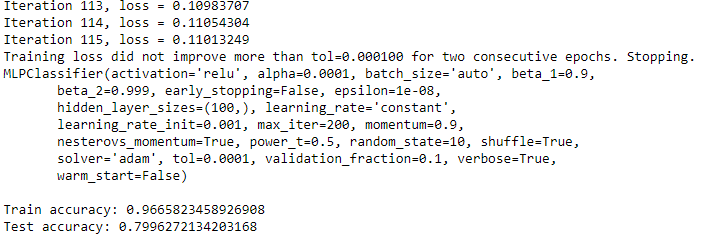


Figure 1XXX

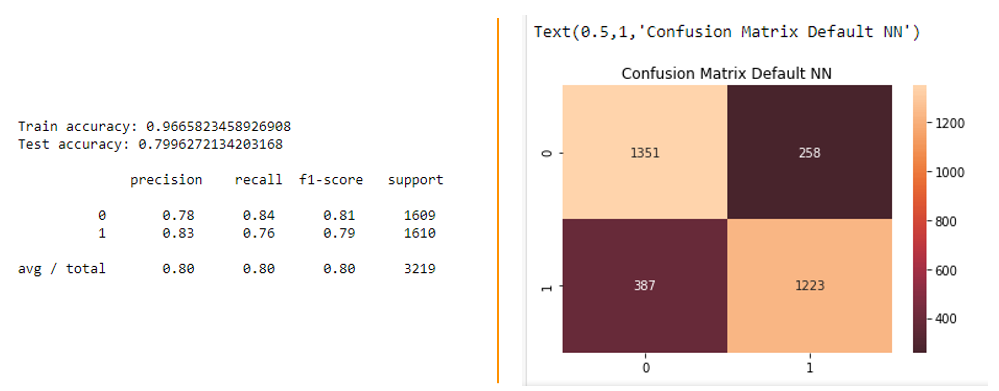


Figure 2 Neural network with default setting

2. Python: Refine this network by tuning it with GridSearchCV.

*A total of three GridSearchCV models were created in this task. Each model has different parameter numbers. The first model has only one parameter—the hidden\_layer\_sizes (figure xxx). The second model tunes the hidden\_layer\_sizes and added alpha as another parameter (figure xxx). Finally, the last GridSearchCV model for this neural network has six parameters as illustrated in figure xxx.*

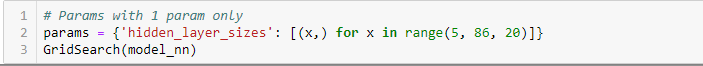


Figure 3

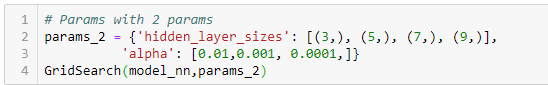


Figure 4



Figure 5

a. What is the network architecture?

*Using the best parameters produced by the GridSearchCV, the hidden layer sizes, for each of the three model (GridSearchCv with one parameters, GridSearchCV with two parameters, and GridSearchCv with six parameters) training consist of 5, 3, and 4 respectively. All input and output are the same with that of the default neural network (126 and 2).*

b. How many iterations are needed to train this network?

*Training the models take 119, 143 and 34 iterations.*

c. Sign of overfitting?

*Signs of overfitting are not seen on the three GridSearch models. The train and test accuracy are relatively close to its other with the GridSearchCV bearing two paramaters shows the closest accuracy.*

d. Did the training process converge and resulted in the best model?

*Yes, the training process for the three GridSearchCV model converged and resulted in best model so far compared to the default setting of the neural network.*

e. What is classification accuracy on training and test datasets? Is there any improvement in the outcome?

*The classification accuracy of these models is shown in figures xx, xx, and xx. With train and test accuracy of (0.852,0.836) for one parameter, (0.852,0.836) for two parameters and (0.849,0.828) for GridSearch with six parameters. Although the measurement of accuracy are merely similar across the three GridSearchCV models, the last model with six parameters can be assumed to have the greatest improvement as shown in the confusion matrix. It detects the highest value for non-kick (true negative) and kick (true positive), 1426 and 1242 respectively, with regards to other neural network models including the default setting.*

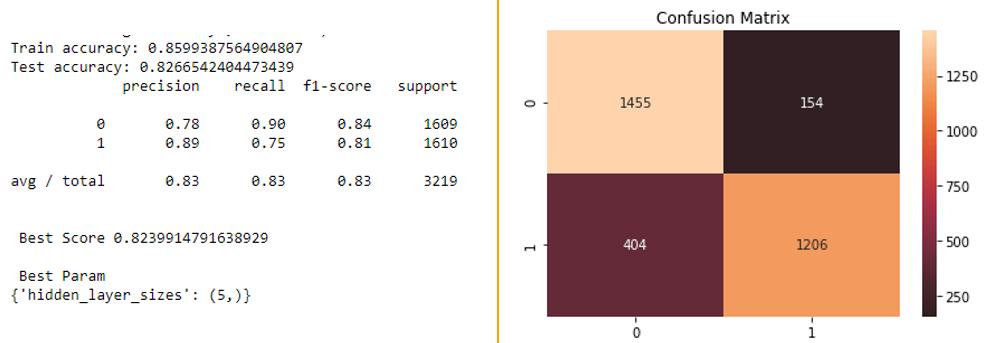


Figure 6 GridSearchCV result with 1 parameter

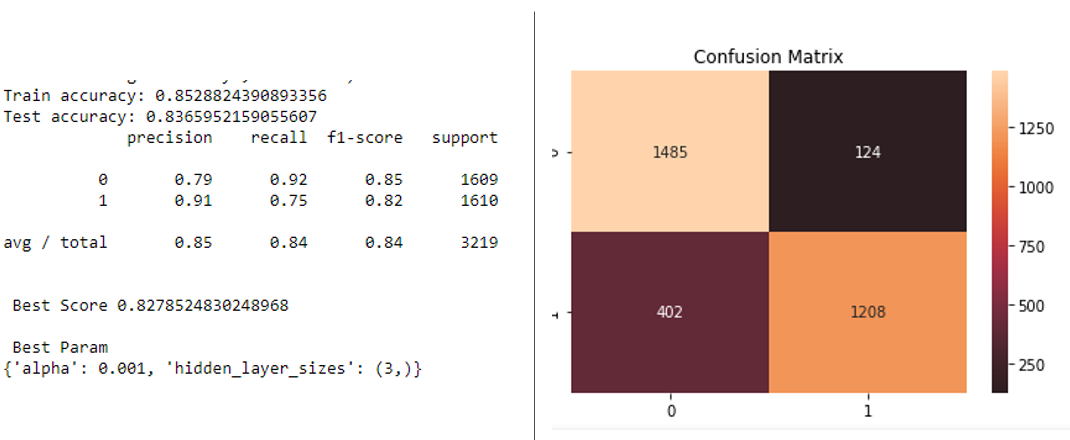


Figure 7 GridSearchCV result with 2 parameters

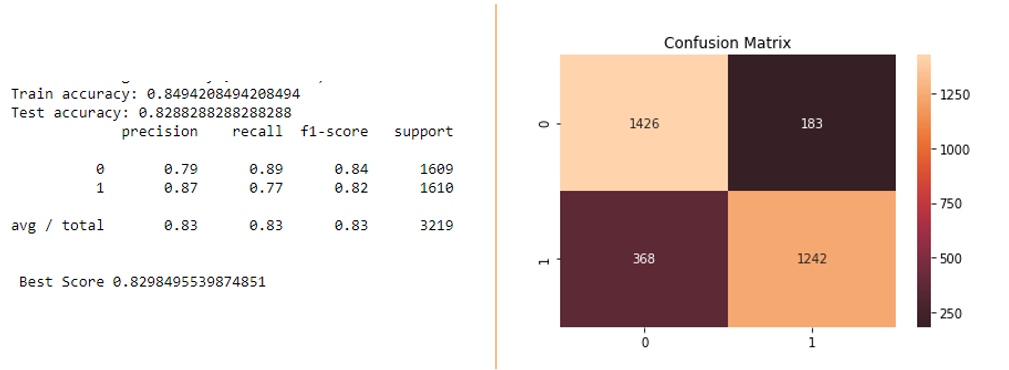
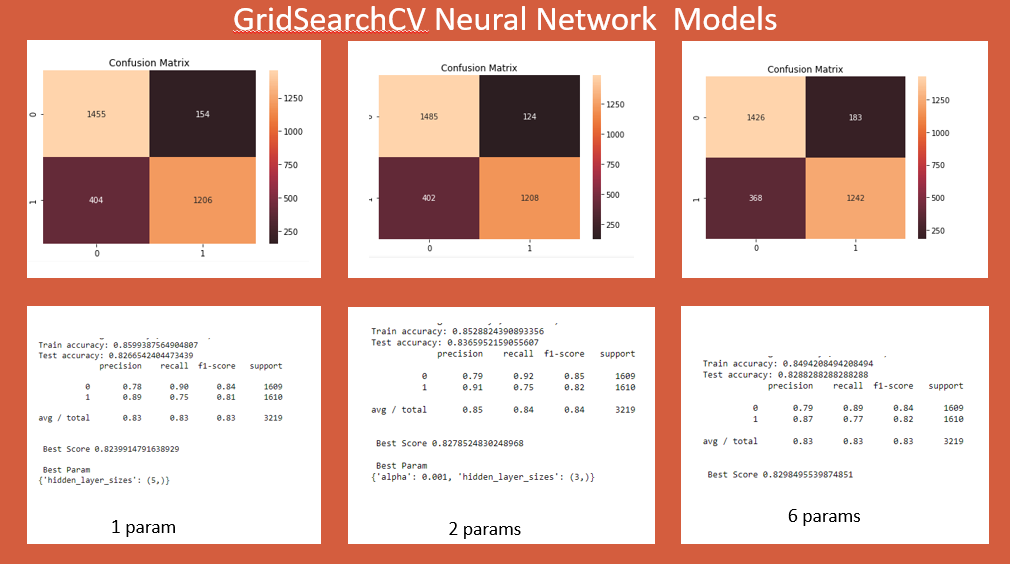


Figure 8 GridSearchCV result with 6 parameters



3. Would feature selection help here?

Build another Neural Network model with inputs selected from RFE with regression (use the best model generated in Task 3) and selection with decision tree (use the best model from Task 2).

*Figure 9 shows the result of neural network with feature selection using the best model in regression and decision tree. In the Neural network GridsearchCV, the model with two parameters (hidden layer sizes and alpha value), the implementation of feature selection is based on this.*

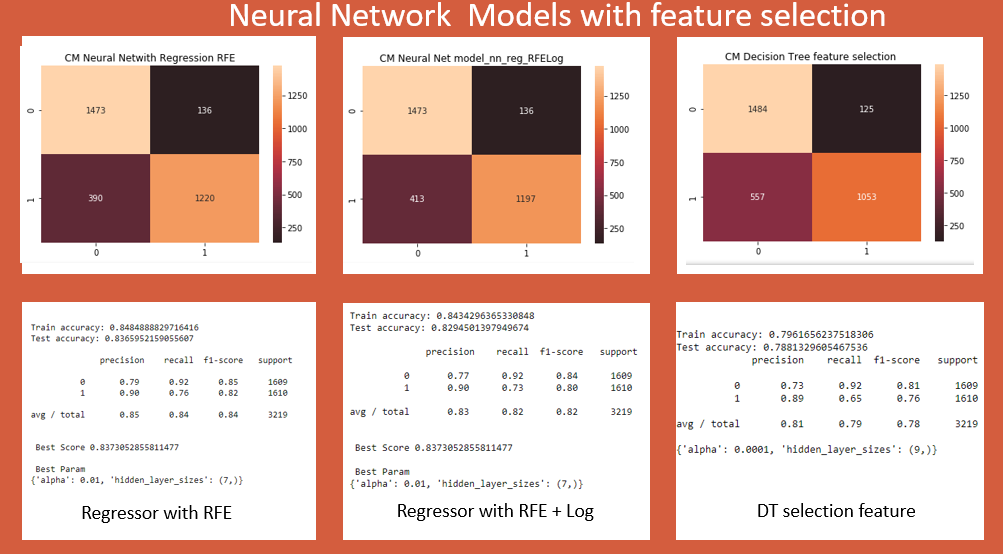


Figure 9 Neural network with feature selection

a. Did feature selection help here? Any change in the network architecture? What inputs are being used as the network input?

*No, feature selection in neural network appears to have little to none effects on the improvement of the model with regards to the GridSearchCV neural network models. Neural networks using the regressor recursive feature elimination (RFE) has fifty inputs, 7 hidden layer size and 2 outputs. The neural network using both of the regressor’s RFE and log transformation has 47 inputs, 7 hidden layer sizes and 2 outputs. The neural network using the decision tree feature selection has 12 feature inputs, 9 hidden layers and 2 outputs.*

b. What is classification accuracy on training and test datasets? Is there any improvement in the outcome?

*The classification accuracy of the models are summarised in table 1. The result did not improve the outcome of training with regards to the neural network’s GridSearchCV models.*

|  |  |  |  |
| --- | --- | --- | --- |
| Model | Regressor with RFE | Regressor with RFE and Log | Decision tree feature selection |
| Training accuracy | 0.848 | 0.843 | 0.796 |
| Testing accurary | 0.836 | 0.829 | 0.788 |

c. How many iterations are now needed to train this network?

*Table 2 shows the number of iterations each model takes to train.*

|  |  |  |  |
| --- | --- | --- | --- |
| Model | Regressor with RFE | Regressor with RFE and Log | Decision tree feature selection |
| Number of iterations | 123 | 117 | 80 |

d. Do you see any sign of over-fitting? e. Did the training process converge and resulted in the best model?

*There were no signs of overfitting in the recursive feature elimination (RFE) or the decision tree feature selection as the accuracy for training and testing are quite close to each other. Yes, the process did converge in these models. However, these cannot be the best models as it fails to improve the performance for classifying kick and non-kick cars in comparison with neural networks using all the features assign in the training set.*

4. Using the comparison methods, which of the models (i.e one with selected variables and another with all variables) appears to be better?

*Using all the variables in neural network appears to be the better model.*

From the better model, can you identify cars those could potential be “kicks”?

Can you provide some descriptive summary of those cars? Is it easy to comprehend the performance of the best neural network model for decision making?

*In neural network, comprehending the reasons why a certain car is kick is difficult. We cannot determine how the input or nodes were selected for learning such as assigning weights and backpropagating to learn better feature. The outcome is certain by the reasons are hidden*

# Task 5. Generating an Ensemble Model and Comparing Models

1. Generate an ensemble model to include the best regression model, best decision tree model, and best neural network model. .

a. Does the Ensemble model outperform the underlying models? Resonate your answer.

2. Use the comparison methods (or the comparison node) to compare the best decision tree model, the best regression model, the best neural network model and the ensemble model.

a. Discuss the findings led by (a) ROC Chart (and Index); (b) Score Ranking (or Accuracy Score); (c) Fit Statistics; (or Classification report) and (4) Output.

b. Do all the models agree on the cars characteristics? How do they vary?

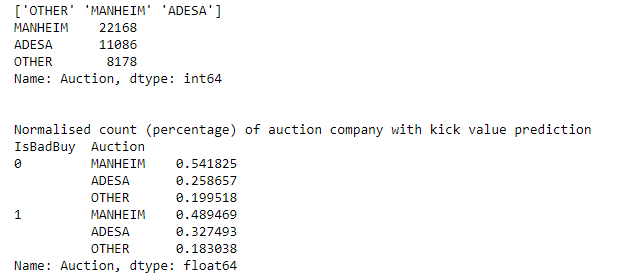
# Task 6. Final Remarks: Decision Making

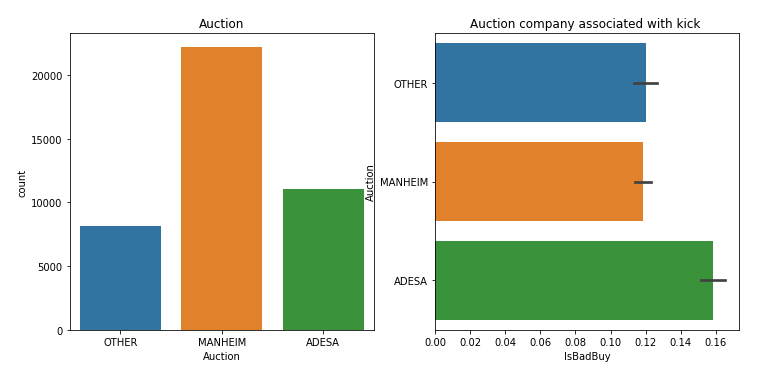
1. Finally, based on all models and analysis, is there a particular model you will use in decision making? Justify your choice.

2. Can you summarise positives and negatives of each predictive modelling method based on this analysis?

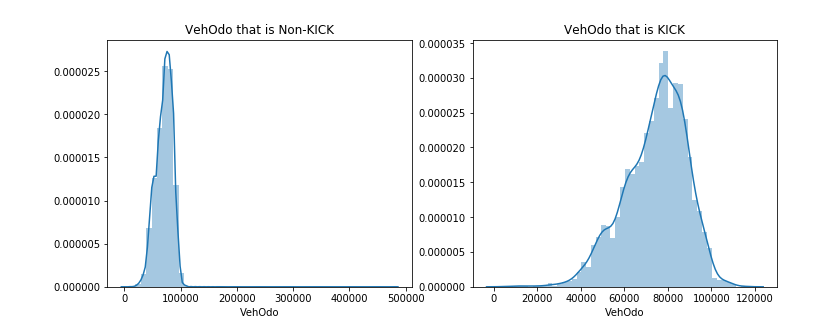
3. How the outcome of this study can be used by decision makers?

Can you identify any clear patterns by initial exploration of the data using histogram or box plot?





ADESA and Other shows higher probability of a kick while it occupied less volume of cars



More drive, more kick