UNIVERSITY OF MARYLAND
ROBERT H. SMITH SCHOOL OF BUSINESS

ASSIGNMENT 3

Your objective is to develop models to predict the outcome variable "BadBuy", which labels whether a car purchased at an auction was a "bad buy" (lemon). Your task is to build a model to guide auto dealerships in their decisions on whether to bid for and purchase a vehicle. You can also apply your learning from this analysis to make more data-informed car-buying decisions!

You will use **carvana.csv** which contains data from 10,062 car auctions as provided by Carvana. Auto dealers purchase used cars at auctions with a plan to sell them to consumers, but sometimes these auctioned vehicles can have severe issues that prevent them from being resold at a profit (hence, lemons). The data contains information about each auctioned vehicle.

Data Dictionary

Variable	Definition
Auction	Auction provider where vehicle was purchased
Age	The years elapsed since the manufacturer's year (how old is the vehicle)
Make	Vehicle manufacturer
Color	Vehicle color
WheelType	Vehicle wheel type description (Alloy, Covers)
Odo	Vehicle odometer reading
Size	Size category of the vehicle (Compact, SUV, etc.)
MMRAauction	Auction price for this vehicle (in average condition) at the time of purchase
MMRAretail	Retail price for this vehicle (in average condition) at the time of purchase
BadBuy	Whether the vehicle is a bad purchase / lemon ("YES") or a good investment ("NO")

Before you start:

- Load the following libraries in the given order: *tidyverse, tidymodels, plotly, skimr, caret*
- Load the Carvana data and call it *dfc*
- Explore the dataset using `skim()` etc.

Assignment Instructions

There are two main objectives. The first is to predict the variable BadBuy as a function of the other variables. The second is to build alternative models, measure, and improve performance.

1) (~5 points) Data preparation

a) Load the dataset into R and call it *dfc*. Inspect and describe the data.
There are many missing or irregular data values in the dataset for example in "WheelType" which has "NULL" as a value. There are a total of six factor variables in the dataset namely "Auction", "Make", "Color", "WheelType", "Size" and "BadBuy". Some of these, like "Make" have a lot of different types.

b) Set the seed to **52156**. Randomly split the dataset into a training dataset and a test dataset. Use **65%** of the data for training and hold out the remaining **35%** for testing.

2) (~10 points) Exploratory analysis of the *training* data set

a) Construct and report boxplots of the (1) auction prices for the cars, (2) ages of the cars, and (3) odometer of the cars broken out by whether cars are lemons or not. Does it appear that there is a relationship between either of these numerical variables and being a lemon? Describe your observations from the box plots. Please also pay attention to the outliers detected by the box plots and make sense of them.
There seems to be a clear relationship between age of the car and the car being a lemon. Older the car, more is the chance of it being a lemon. The difference in auction price and odometer reading between lemons and not lemons is very little so it is not a clearly defined relationship. The outliers tell us that the auction prices for a lot of the lemons are extremely high indicating a lot of variance in the auction prices. Additionally, there are a lot of odometer readings which are unexpectedly very low for lemons which again shows variance in the readings.

b) Construct and report a table for the count of good cars and lemons broken up by Size (i.e., How many vehicles of each size are lemons?).
Hint: Remember `tally()`? That's one way to do it. You may want to think more systematically and use a combination of summarize(), length(), mutate(), arrange()

i) Which size of vehicle contributes the most to the number of lemons? (That is, which vehicle size has the highest *percentage* of the total lemons?)
Compact size contributes the most to the number of vehicles.

ii) Because the vehicles of the size you identified in (i) contribute so much to the number of lemons, would you suggest the auto dealership stop purchasing vehicles of that size? Why or why not?
No, I wouldn't suggest stopping the purchase of vehicles of compact size as there is also a large number of the same size vehicles which were good.

Taking into account the other variables one can make an informed decision to choose a good vehicle.

3) (~20 points) Run a linear probability model to predict a lemon using all other variables.

a) Compute and report the RMSE using your model for both the training and the test data sets. Use the predicted values from the regression equation. **Do not** do any classifications yet.
Training data RMSE – 0.4479165
Testing data RMSE - 0.4528846

b) For which dataset is the error smaller? Does this surprise you? Why or why not?
Training dataset has a smaller error. This is not surprising as the initial model is built using the values of this dataset and decision variables.

c) Use a cutoff of 0.5 and do the classification in the test data. Compute and report the confusion matrix (recall to convert BadBuy into a factor for the confusion matrix).
   i) Which type of errors (false positives and false negatives) occur more here?
   False negatives occur more.
   ii) For this problem, do you think a false positive or a false negative is a more serious error? Based on your answer, which metric makes a better objective?
   False negative is a more serious error as this would suggest that the car is good while in fact the car is a lemon. Precision makes a better objective because it will tell us of all the cars what fraction of them are actually good.

d) What is the testing accuracy of your model? Based on accuracy, does the model perform better than using a random classifier (i.e., the baseline accuracy)?
Hint 1: Calculate manually if you like, or use the `confusionMatrix()` function.
Hint 2: The baseline accuracy is the accuracy you would achieve if you classified every single class as a member of the most frequent class in the actual test dataset.
The testing accuracy is 0.6731. In case of random classifier, we take every single class to be 0. From this we get an accuracy of 0.5061. We can conclude that the original model does performs better than the random classifier.

e) Compute and report the predicted "probability" that the following car is a lemon:
Auction="ADESA" Age=1 Make="HONDA" Color="SILVER"
WheelType="Covers" Odo=10000 Size="LARGE"
MMRAauction=8000 MMRAretail=10000
Does the probability your model calculates make sense? Why or why not?
The probability is -0.141.
This probability does not make sense as it is negative while the chance of this combination of variables is perfectly viable and should occur between 0 and 1. This can be either because the model is not entirely accurate or the data point for this

particular scenario would be an outlier in the dataset.

4) **(~25 points) Run a logistic regression model to predict a lemon using all other variables.**
Hint 1: Don't forget to convert your dependent variable BadBuy to a factor in both datasets.
Hint 2: If you haven't yet, switch to using *caret* at this point.

a) Did you receive a rank-deficient fit error? Why do you think so? Figure out the variables causing the problem by running tally() for all your factor variables, and recode them in a way to prevent the error.
Hints: You will need to recode two factor variables:
1. *Color* has two redundant levels that need to be combined.
2. Create a new category for *Make*, call it OTHER, and recode any of the makes with less than 10 observations as OTHER.

Make sure to make the changes in the full dataset, convert BadBuy to a factor, repeat the process of setting the seed to 52156 and splitting the data.
Run your logistic regression again to confirm the rank-deficient fit error is gone.
Color and Make are causing the problem. The error can be because of a very small number of observations for some levels in a factor variable while some of the levels in it contain a very large number of observations.

b) What is the coefficient for Age? Provide an exact numerical interpretation of this coefficient.
0.2785. On an average, with every change in one year of age of a car, its probability of being a lemon increases by 0.28, keeping all the remaining variables constant.

c) What is the coefficient for SizeVAN? Provide an exact numerical interpretation of this coefficient.
-0.5982. On an average, a car of size "Van" has a probability of 0.6 less than size "Compact" of being a lemon, keeping all other variables constant.

d) Use a cutoff of 0.5 and do the classification in the test data. Compute and report the confusion matrix for your test data predictions.

BadBuy		
predictedClass	0	1
0	1341	721
1	441	1018

e) Compute and report the predicted probability using your logistic model for the same car from 3(e). What does the resulting value tell you about this particular car now? Does the result make more sense than the result in Question 3(e)? Why or why not?

Pro tip: Pipe a confusion matrix (from any model) into tidy() and see what happens!

The predicted probability is 0. This makes more sense as instead of a negative probability, this particular set of variables falls under one class i.e. 0. Hence, it will be safe to predict that this car will not be bad buy.

(5) (~40 points) Explore alternative classification methods to improve your predictions.

- In the models below, use a 10-fold cross validation to make the results consistent across.
- Use the same training and test data you created and used after recoding the data in Q4.
- Make all comparisons to the logistic model you have run in Q4 after recoding the data.

a) Set the seed to **123** and run a linear discriminant analysis (LDA) using all variables.

i) Compute the confusion matrix and performance measures for the test data, and compare them **with the logistic regression results**. Discuss your findings.
While the accuracy and sensitivity of this model is low, specificity is high indicating that this model is more efficient at predicting the good cars then the logistic model. On the other hand, the logistic model is better at predicting the overall lemons.

b) Set the seed to **123** and run a kNN model using all variables.

i) Create a plot of the k vs. cross-validation accuracy.

ii) What is the optimal k? What else do you infer from the plot?
The optimal value of k is 19 as it is the value from where the cross validation accuracy increases. It is the highest for k = 41. But here there will be a lot of variance.

Hint: To inspect the details of any model, you will need to train the model and store it before piping it into predict(). See the GitHub repository for guidance.

iii) Compute the confusion matrix and performance measures for the test data, and compare them **with the logistic regression and LDA model** results. Discuss your findings.
The accuracy, specificity and sensitivity for this model is lower than either of the above two models, indicating that this is not a good model at predicting either good cars or lemons.

c) Set the seed to **123** and build a lasso model using all variables.

i) Set the seed to **123** and run a Lasso model using all variables. Report the table of variable importance in a tibble format and share your observations.
Hint: See the Github repo for help. Use a 100-point grid between 10^{-5} and 10^{2}

ii) Report the plot of variable importance for the 25 most important variables.

iii) What is the optimum lambda selected by the model? What does it mean that the algorithm chooses this particular lambda value?
0.0005857021 is the optimum lambda. This means that at this value of lambda the model will have the least variance or error rate.

iv) Compute the confusion matrix and performance measures for the test data, and compare them **with the logistic regression, LDA, and kNN model** results. Discuss your findings.

The accuracy and sensitivity of this model are higher than LDA and KNN and lower than logistic regression. The specificity of this model is next to that of LDA. Overall this model does a better job at predicting the lemons and good cars than all of the models.

d) Set the seed to **123** and build a (I) ridge and (II) elastic net[1] model using all variables.

i) Compute the confusion matrix and performance measures for the test data, and compare them **only with the lasso model** results. Discuss your findings.

Accuracy is most for lasso whereas the sensitivity and specificity for elastic net and lasso are almost equal. This shows that elastic net is as good a model at correctly predicting lemons and good cars out of the actual dataset. The optimum level for elastic net is a little lower than lasso meaning it is more lenient at penalizing the variables.

Hint: Use the same grid for lambda. Notice the different optimum value!

e) Set the seed to **123** and run a quadratic discriminant analysis (QDA) with all variables

i) Have you received an error? What do you think the error you received means? Do some research and explain what you think it is about.

The error is present because there are two or more variables which are highly correlated in the model. This can be removed by keeping just one of these variables in the model.

ii) Why is the rank deficiency a problem for QDA, but not for LDA?

In case of LDA, covariance matrices across classes are identical which implies that the correlations are taken to be equal for them. This will take care of the correlation problem in case of QDA.

iii) Compute the confusion matrix and performance measures for the test data, and compare them **only with the LDA model** results. Discuss your findings.

While the accuracy and sensitivity of this model is lower than that of LDA, the specificity is higher which means this model is the best at ruling out the true negatives i.e. it can correctly predict the not lemons or good cars.

f) Among all the models you have studied, which model do you think is better for the given business case/problem? Discuss why you think it is better than the others. Also report the ROC curves for the models you have developed on the same chart.

Among all the models, I think the logistic regression model is the best for the given business case as apart from the high accuracy this model also efficiently predicts the

[1] Naive elastic net. Feel free to run a grid search but be careful not to hit the limits of your computational power!

true positives which is required to obtain the false negatives in the model. Since, the false negatives is a bigger concern of error for this scenario, sensitivity will be very useful.

Bonus question: You may have noticed that lasso drops certain levels of Make and Color such as “Brown”, keeping the other levels of the same variable (“Blue” etc.). This may not be helpful, so you may want to use a grouped lasso. Set the seed to 123 and try grouped lasso with the lambda values 50 and 100. Do the results make more sense now? Why or why not?

Hint: Run a plain lasso again with a lambda value of 0.01 and print the coefficients this time. Compare them with the coefficients from group lasso.