**Decision based on older data**

Based on the data I created from the midterm, I cannot accurately recommend that the car in question is a good fit. The data that I gather was data about the most abundant results, which were:

* Unacc rating having the highest count of cars
* Low priced vehicles have the highest count of cars
* The maintenance costs very high and high come from low priced cars with acc rating

With, based on the information I did have regarding the data around what the automobile on offer has, I would recommend this car to the buyer on an assumption only.

**Final Data**

**Exploring Results**

The accuracy of the models on average was high at 95.8%. The most accurate model was the gradient boosted tree at 99.4% accuracy, and it took 57 seconds to complete. The fasted total time was a decision tree was 1 seconds but had an accuracy of 95.1%.

My impression of the ROC Comparison graph is that in most instances, a positive instance is passed resulting in and upward movement in the ROC graph. Once the models reached about 0.9, that is when they started to pass negative instances resulting in a step to the right. There was a model that was perfect on the ROC graph, however, the chosen colors make it hard to decipher which model it actually was (maybe a gradient boosted tree).

**Model Exploration – all models performed very well on the ROC comparison graph being very close to 1 with the lowest being Naïve Bayes at .98 and highest was Gradient Boosted Trees at 1**

Naïve Bayes

* Was the fastest model when focusing on rating “unacc” and rating “acc”
* Precision was 93.8 +/- 2.8 for “unacc” and 92.5 +/- 2.8 for “acc”
* Recall was 96.7 +/- 1.7 for “unacc” and 85.8 +/- 6.2 for “acc”
* Safety was the highest weighted attribute at 0.295 followed closely by persons at 0.292
* The model seems to have calculated that “acc” occurs more often for the middle two attributes in each category and “unacc” on the outer two attributes for each category
* The simulator for Naïve Bayes model is 60% for “unacc” so the car is not recommended to the customer
* The supporting factor is maintenance costs for “unacc” and the contradicting factors are persons, safety, purchase price, luggage boot, and doors

Generalized Linear Model

* Precision was 97.1 +/- 1.8 for “unacc” and 91.0 +/- 4.5 for “acc”
* Recall was 95.9 +/- 2.5 for “unacc” and 93.3 +/- 3.7 for “acc”
* Safety was weighted the highest at 0.253 and person was close with 0.252
* The model doesn’t really tell anything just looking at it
* The simulator for GLM says that “acc” is most likely 55% of the time. The car would not be recommended due to not reaching the threshold
* The supporting factors for “acc” are persons, safety, purchase price, luggage boot, and doors. The factor contradicting it is maintenance cost

Logistic Regression

* Precision was 96.0 +/- 2.5 for “unacc” and 93.0 +/- 2.5 for “acc”
* Recall was 97.1 +/- 1.0 for “unacc” and 90.8 +/- 4.7 for “acc”
* Safety was weighted the highest with 0.248 and persons was seconds with 0.245
* The model tells us the std. error, z-score, p-value which could be graphed, and statistical calculation could be utilized to obtain further information that is not noted. Other than that, it does not tell us much else besides numbers
* The simulator predicts “acc” at 58% which is not high enough to recommend the car to the customer
* The factor supporting “acc” are persons, safety, purchase price, luggage boot, and doors. The factor contradicting it in maintenance cost

Fast Large Margin

* Precision was 96.0 +/- 2.4 for “unacc” and 91.7 +/- 2.9 for “acc”
* Recall was 96.5 +/- 1.3 for “unacc” and 90.8 +/- 4.7 for “acc”
* The highest weighted attribute was “safety=low” at 0.341 follow by “persons=4” and “persons=more” both with weights of 0.140
* The model does not really tell me anything that is understandable. I can assume that the lower the number the less weighted that attribute is and the higher the number the attribute has is more weighted. I can infer this because “safety=low” has a number of +3.630 and correlates to the highest weighted attribute
* The simulator has determined that the most likely rating of “acc” was at 51% which is not enough to recommend this car to this customer
* The factors supporting “acc” are persons = 4, persons = more, maintenance = low, maintenance = med, and purchase price = low. The factors contradicting it are safety = low, and luggage boot = low

Deep Learning

* Precision was 97.9 +/- 1.7 for “unacc” and 93.9 +/- 2.8 for “acc”
* Recall was 97.1 +/-1 for “unacc” and 95.4 +/- 3.5 for “acc”
* The highest weighted attribute was safety at 0.243 followed by persons at 0.242
* The model tells me that the AUC is 0.9981 which is very good. Most of the graph is under the curve. It gives a confusion matrix of the data with gains, lifts, and errors. It tells the status of neuron layers and gives the scoring history.
* The simulator gives most likely to “acc” with a 52% accuracy. It is not enough to confidently recommend the car to the customer.
* The factors support “acc” are persons, safety, purchase price, luggage boot, and doors, the factor contradicting it is maintenance cost

Decision Tree

* Precision was 96.2 +/- 1.2 for “unacc” and 90.7 +/- 4.5 for “acc”
* Recall was 96.8 +/- 1.8 for “unacc” and 87.7 +/- 1.5 for “acc”
* The highest weighted attribute was safety at 0.235 followed by persons at 0.230
* The model tells me that everything stems from the safety attribute and get pruned from there. The strongest part of the tree would be the persons part of the tree because it has the most weight.
* The simulator is 100% in favor of “acc” and can accurately recommend this car to the customer based on the characteristics.
* The factors supporting “acc” are persons, safety, purchase price, and luggage boot. The factors contradicting it are doors and maintenance cost

Random Forest

* Precision was 95.8 +/- 2.4 for “unacc” and 93.4 +/- 3.2 for “acc”
* Recall was 97.5 +/- 1.1 for “unacc” and 89.9 +/- 6.0 for “acc”
* The highest weighted attribute was safety at 0.298 followed by persons weighted at 0.266
* The model I very much like the decision tree model. There is a difference in the random forest model though. Sometimes the model on has one branch and the other times it has multiple branches, but always stems from safety since it is the highest weighted attribute
* The simulator has determined that the most likely rating is “unacc” with a rating of 58%. The car cannot be recommended to the customer
* The factors supporting “unacc” are maintenance cost and doors. The factors contradicting it are persons, safety, luggage boot, and purchase price

Gradient Boosted Tree

* This was the most accurate model for precision and recall for rating “unacc” and fasted for precision for rating “acc”
* Precision was 100 +/- 0.0 for “unacc” and 97.5 +/- 2.5 for “acc”
* Recall was 99.1 +/- 1.4 for “unacc” and 98.8 +/- 1.7 for “acc”
* The highest weight attribute was safety with a weight of 0.241 followed by persons with a weight of 0.238
* The model is like that of the previous 2 models, decision tree and random forest. All trees stem from safety and a pruned accordingly
* The simulator determined that the car could be recommended to the customer base on the attributes with the most likely rating of “acc” with 82%
* The factors supporting “acc” are persons, safety, purchase price, luggage boot, and doors. The factor contradicting it is maintenance cost

Support Vector Machine

* This was the most accurate model for recall when focusing on rating “acc”
* Precision was 99.7 +/- 0.7 for “unacc” and 93.5 +/- 6.3 for “acc”
* Recall was 97.2 +/- 2.5 for “unacc” and 99.4 +/- 1.4 for “acc”
* The highest weighted attribute was “safety = low’ with a weight of 0.311 followed by “persons = more with a weight of 0.145
* The model does not tell me much. I could guess that the model lists each instance with a certain support vector and then score it accordingly for each. The model used 99 support vectors for “unacc” and 87 support vectors for “acc”
* The simulator determined “acc” with a 64% rating that the most likely would be acceptable. Not rated high enough to recommend the car to the customer
* The factors supporting “acc” are person = more, person = 4, maintenance = low, maintenance = med, purchase price = low, and purchase price = med. The factor contradicting it is safety = low

**Finding Report**

As stated at the beginning of the paper and based on the data I created from the midterm, I cannot accurately recommend that the car in question is a good fit. The data that I gather was data about the most abundant results. Rating “unacc has the highest number of cars at 63% of the cars in the dataset, “acc” has 23% of the cars. Low- and medium-priced cars has ~75% of the data points. The lower the price of the car with an “acc” rating the more it makes up for in maintenance cost.

Where the association rules are concerned, there is no evidence to support the car that the customer wants to purchase. All the association rules were with rating “unacc” and seemingly the opposite of what the customer is looking for (2 door, small luggage, 2 persons, etc). The following are the association rules from the midterm(k-means clusters look very similar):

1. Unacc and lowsafety has a support of 0.333 and confidence of 0.476
2. Unacc and 2person has a support of 0.333 and confidence of 0.476
3. Unacc and smallluggage has a support of 0.260 and confidence of 0.372
4. Unacc, lowsafety, and smallluggage has a support of 0.111 and confidence of 0.427
5. Unacc, 2person, and smallluggage has a support of 0.111 and confidence of 0.427
6. Unacc, lowsafety, and moreperson has a support of 0.111 and confidence of 0.596

It is safe to say that the car can not be recommended to the customer.

From the data on the final I can more strongly recommend this car with these attributes, but I can not 100% recommend this car to the customer. The models were very accurate in their predictions when focusing on the “unacc” with a tight +/- range. When focusing on “acc” they were not as accurate and had a greater +/- range to them. I would recommend changing the safety factor because it was the highest weighted attribute in most of the models. Safety = low has the highest correlation overall with a weight of 0.463. the next closest weight was persons = 4 at 0.245.

So changing safety to low, keeping persons at 4, maintenance costs at vhigh, changing purchase price to either low or vhigh, changing luggage boot to small, and keeping safety = med would yield a car that I could strongly recommend to the customer.