Car Evaluation: How's you car doing?

Manil Lakabi

Introduction

- **Problem:**
  - How can I determine the condition of a vehicle based on its common features?

- **Goal:**
  - To analyze and explore data on used cars in the US over the last few years.
  - To create a Machine Learning model that can predict the condition of a vehicle based on features about its main characteristics

Clients: Who Cares?

- Used car dealerships:
- Potential new car buyers:
- You:
  - Buying a car long distance
  - People buying cars in an auction

Dataset

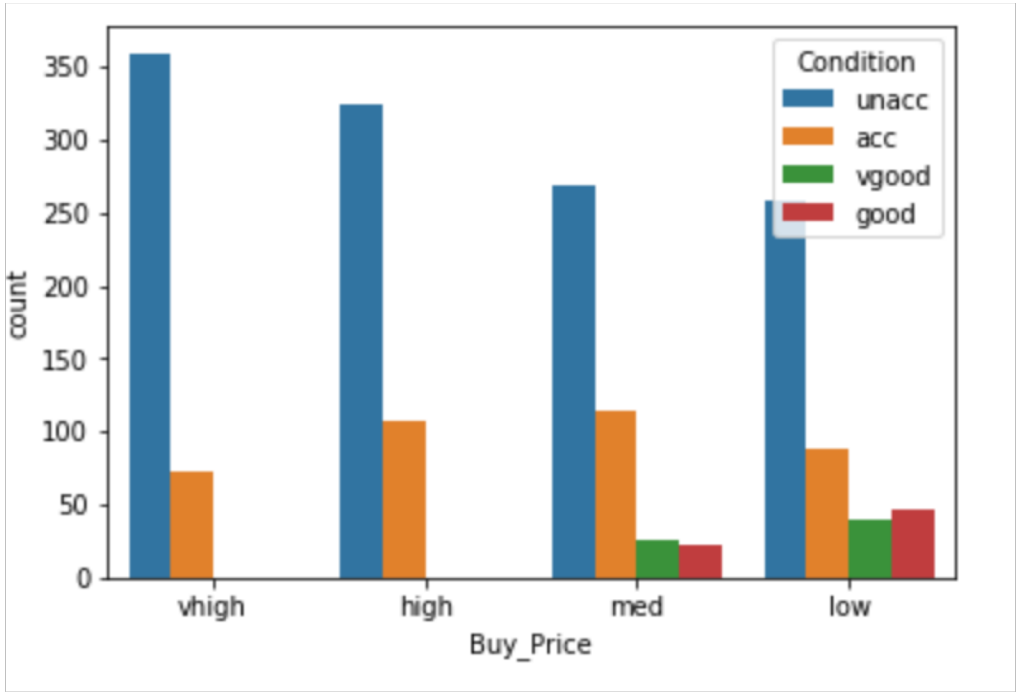
Car Evaluation Data Set (UCI Machine Learning Repository)

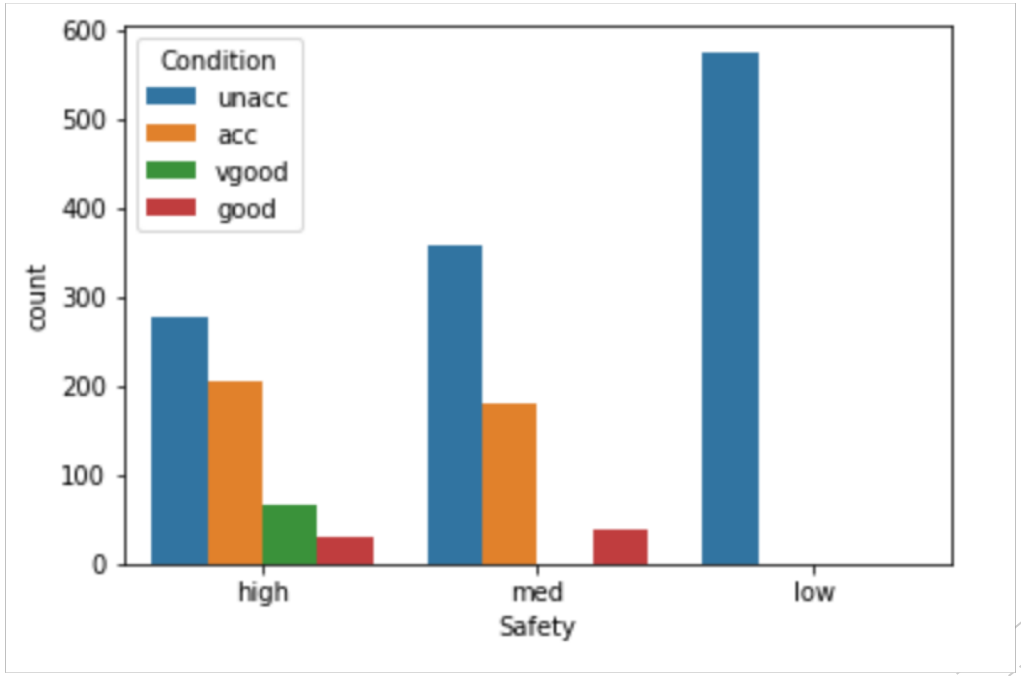
	Buy_Price	Maint_Price	Doors	Passengers	Trunk	Safety	Condition
0	vhigh	vhigh	2	2	small	med	unacc
1	vhigh	vhigh	2	2	small	high	unacc
2	vhigh	vhigh	2	2	med	low	unacc
3	vhigh	vhigh	2	2	med	med	unacc
4	vhigh	vhigh	2	2	med	high	unacc
5	vhigh	vhigh	2	2	big	low	unacc
6	vhigh	vhigh	2	2	big	med	unacc
7	vhigh	vhigh	2	2	big	high	unacc
8	vhigh	vhigh	2	4	small	low	unacc
9	vhigh	vhigh	2	4	small	med	unacc
10	vhigh	vhigh	2	4	small	high	unacc
11	vhigh	vhigh	2	4	med	low	unacc
12	vhigh	vhigh	2	4	med	med	unacc
13	vhigh	vhigh	2	4	med	high	unacc
14	vhigh	vhigh	2	4	big	low	unacc
15	vhigh	vhigh	2	4	big	med	unacc
16	vhigh	vhigh	2	4	big	high	unacc
17	vhigh	vhigh	2	more	small	low	unacc
18	vhigh	vhigh	2	more	small	med	unacc
19	vhigh	vhigh	2	more	small	high	unacc
20	vhigh	vhigh	2	more	med	low	unacc
21	vhigh	vhigh	2	more	med	med	unacc
22	vhigh	vhigh	2	more	med	high	unacc
23	vhigh	vhigh	2	more	big	low	unacc
24	vhigh	vhigh	2	more	big	med	unacc
25	vhigh	vhigh	2	more	big	high	unacc
26	vhigh	vhigh	3	2	small	low	unacc
27	vhigh	vhigh	3	2	small	med	unacc
28	vhigh	vhigh	3	2	small	high	unacc
29	vhigh	vhigh	3	2	med	low	unacc

* unacc: Unacceptable Condition
* acc: Acceptable Condition
* good: Good Condition
* vgood: Very Good Condition

Initial Insights

- The number of passengers a vehicle can hold, as well as safety seem to be strong predictors in the overall condition of a vehicle
- All features seem to have an effect on the predictor variable ('Condition')
- Buying price and maintenance price seem to affect condition very similarly





Statistical Testing: Challenges

Because the data was categorical in nature there were limited options in creating statistical tests. To combat this, the problem was tackled in two different manners

1) Chi-Squared test on the independent variables
2) Vectorization by converting the categorical inputs into a numerical format

Correlation between buying price and condition

```
Buy_con = pd.crosstab(df['Buy_Price'], df['Condition'])
```

```
chi2_contingency(Buy_con)
```

```
(188.8892399516746,
 7.029416116370655e-36,
 9,
 array([[ 96.05558772,  17.25998842, 302.42501448,  16.25940938],
        [ 96.05558772,  17.25998842, 302.42501448,  16.25940938],
        [ 96.05558772,  17.25998842, 302.42501448,  16.25940938],
        [ 95.83323683,  17.22003474, 301.72495657,  16.22177186]]))
```

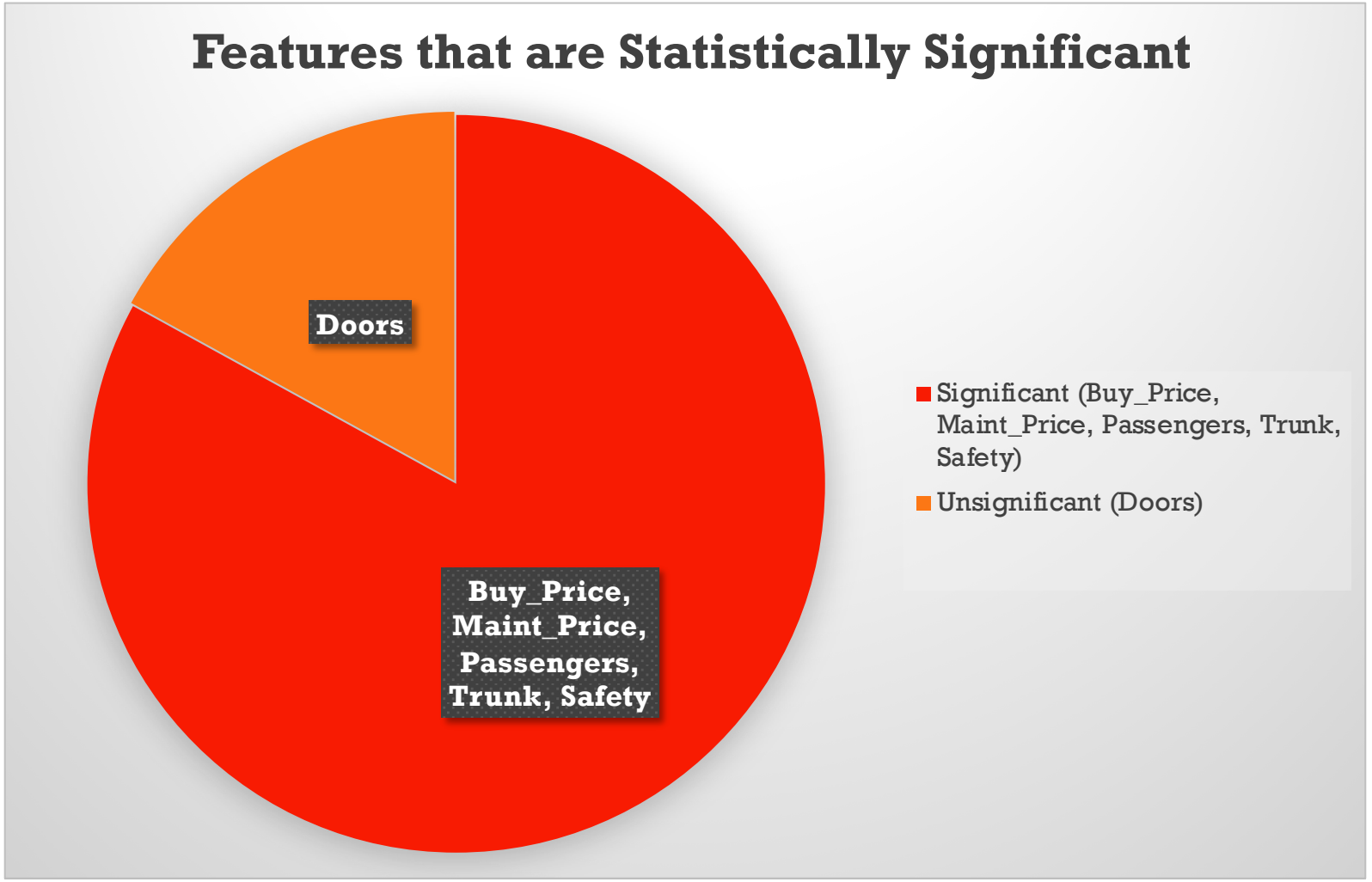
```
# Because the p value is small (less then 0.05) we can say that the two variables have significan
t relationships between them.
```

```
df['Buy_Price'] = df['Buy_Price'].map({'low': 1, 'med': 2, 'high' : 3, 'vhigh': 4})
df['Maint_Price'] = df['Maint_Price'].map({'low': 1, 'med': 2, 'high' : 3, 'vhigh': 4})
df['Safety'] = df['Safety'].map({'low': 1, 'med': 2, 'high' : 3})
df['Trunk'] = df['Trunk'].map({'small': 1, 'med': 2, 'big' : 3})
df['Passengers'] = df['Passengers'].map({'2': 2, '4' : 4, 'more' : 5})
df['Doors'] = df['Doors'].map({'2': 2, '3':3, '4':4, '5more' : 5})
df['Condition'] = df['Condition'].map({'unacc': 1, 'acc': 2, 'good' : 3, 'vgood': 4})
```

```
print (df)
```

	Buy_Price	Maint_Price	Doors	Passengers	Trunk	Safety	Condition
0	4	4	2	2	1	2	1
1	4	4	2	2	1	3	1
2	4	4	2	2	2	1	1
3	4	4	2	2	2	2	1
4	4	4	2	2	2	3	1
5	4	4	2	2	3	1	1
6	4	4	2	2	3	2	1
7	4	4	2	2	3	3	1
8	4	4	2	4	1	1	1

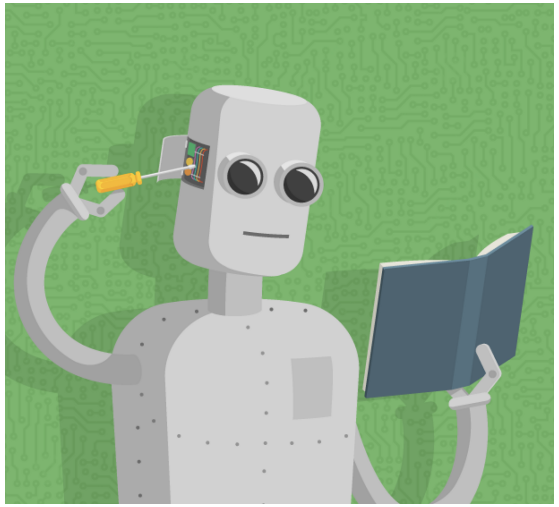
Statistical Testing: Continued



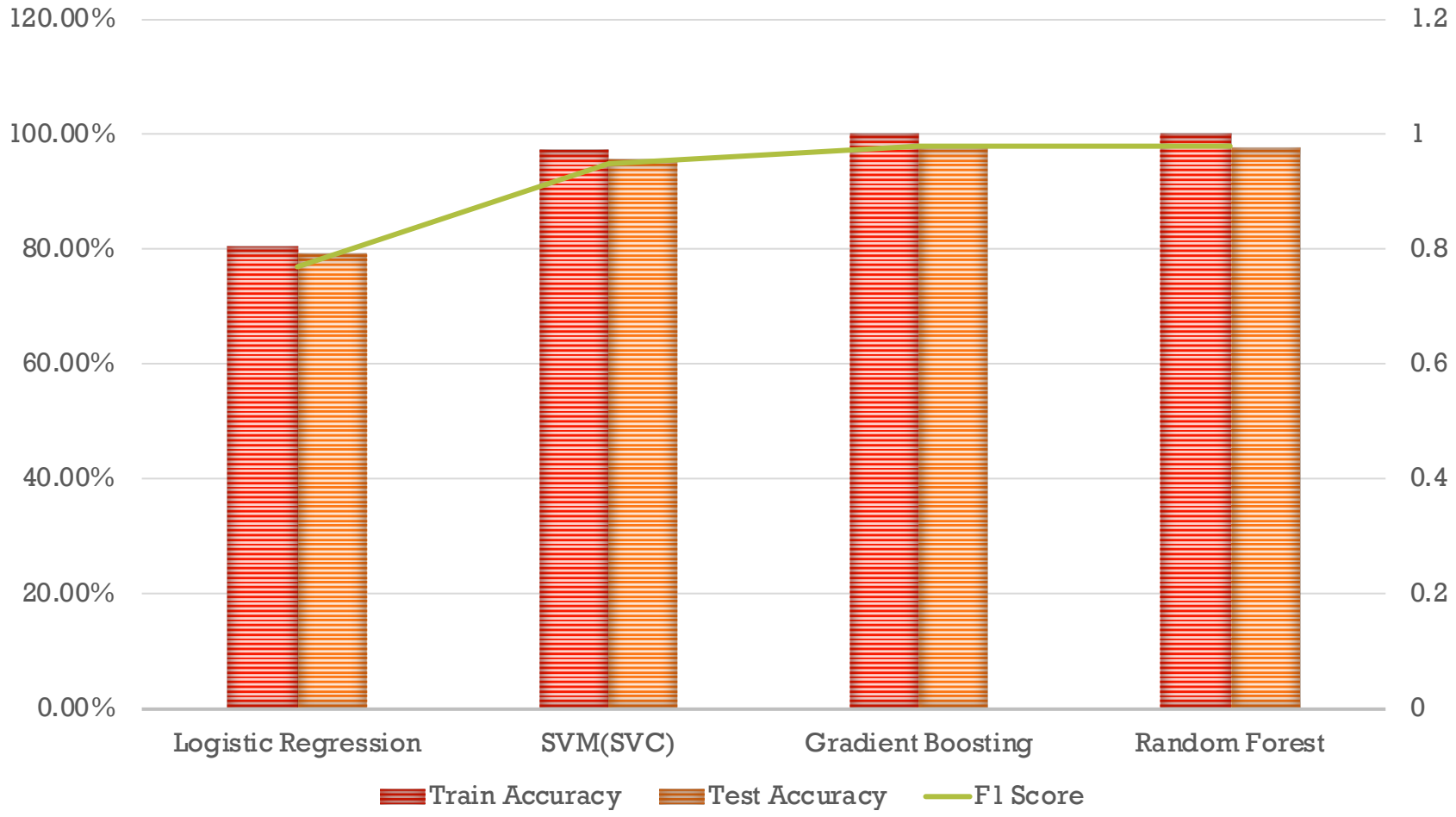
Machine Learning: Creating a Predictive Model

- Decision tree based models such as Gradient Boosting and Random Forest seemed to have yielded the best results. This is because rather then single (or base) decision tree modes, which have an inherent problem of overfitting where the model might perform very well on the training data, but don't generalize well enough to perform well on the test data.

- Ensemble models like gradient boosting or random forest, use several decision trees to learn from the training data. And in this case created far more robust models.

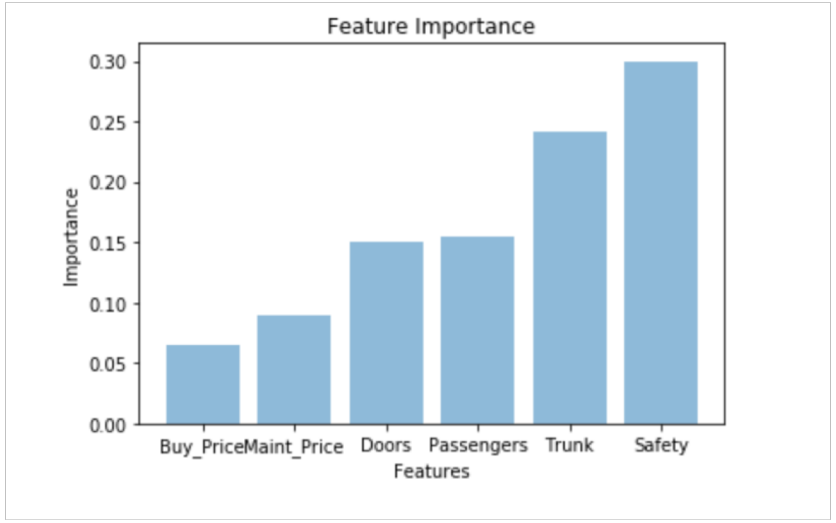


MACHINE LEARNING: MODEL PERFORMANCE

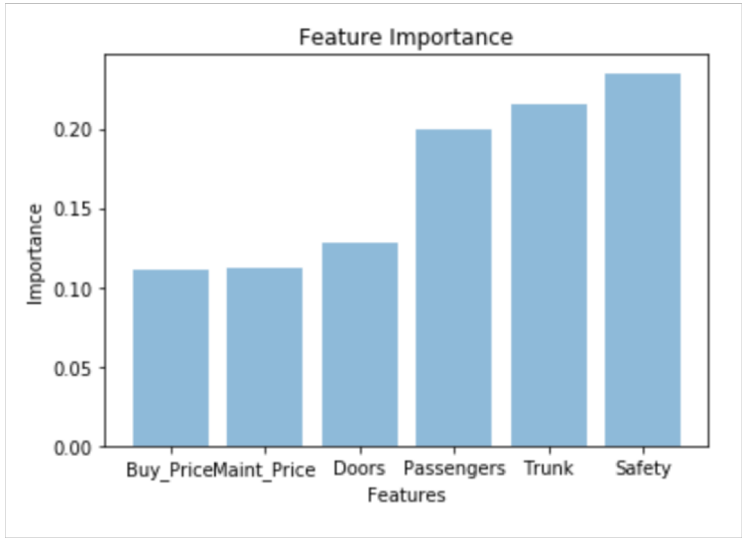


Feature Importance

RANDOM FORREST CLASSIFIER



GRADIENT BOOSTING



Important Feature: Safety

Least Important: Buy_Price

Conclusion

- Safety of a vehicle proved to be the best predictor variable for evaluating the condition of a car. This could be attributed to the fact that safer cars would most likely have a solid build quality and stronger body structure than vehicles that were deemed unsafe.

- It is also important to note that buying price and maintenance seemed to have little importance in determining the condition of a vehicle. This could be because the notion of money in this context is relatively arbitrary in the sense that without accounting for make, year, and mileage of a vehicle we are not really comparing cars at face value. "Buying price" is also vague, as it could imply current buying price of a vehicle or the original buying price of a vehicle.

- Some actionable insights to further improve the model would be to add more features such as make, year, type of vehicle, number of owners, etc… Also converting half of the existing features to numeric values would provide more real world practicality to the model.