James Taylor

Assignment 2

University of Maryland University College

DATA640

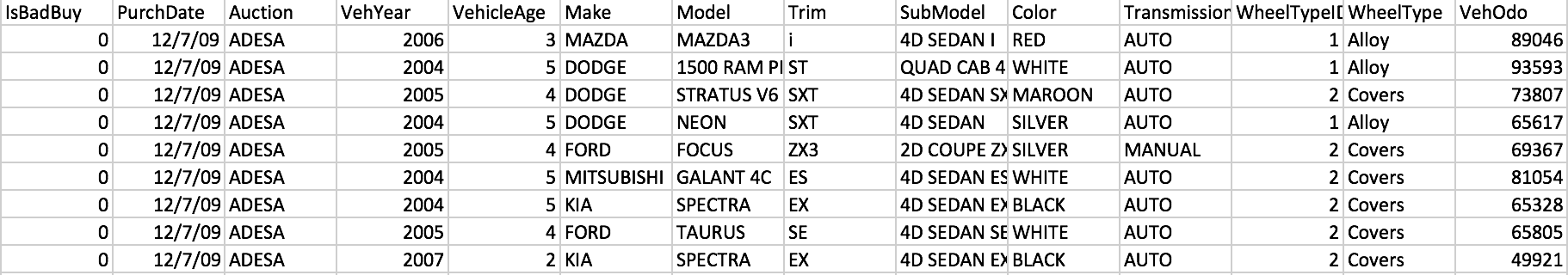
Dr. Knode

james.taylor.ps3@gmail.com

**Introduction**

The purpose of this paper is to use Support Vector Machines to make predictions on car lemon data. It is a supervised learning technique and its accuracy will be tested on unseen data. The model will attempt to make predictions not including the make and model of the vehicle, and focus more on the price paid vs the price it would normal cost for the same car. For example, if a car is sold at auction for 50% less than the average value of the same car with similar miles driven, is this a potential lemon? This will allow for a model to find the characteristics that indicate lemon vehicles around pricing, vehicles characteristics and sale.

The data is on car auctions sales. There are 72,983 observations on 34 variables. The variables are information on the vehicle sold, price information on that model of vehicle, details about the sale of the vehicle and if the vehicle turned out to be a lemon. In basic terms, a lemon vehicle is one that was sold with a significant fault that will cause its usable life to be shortened. The dataset had 12.3% of observations was a lemon vehicle, making the target variable significantly skewed. Below is a screenshot of some variables and values.



**Data Preparation**

The dataset had 34 variables included. There were two variables that were removed because they were unique identifiers that would bias the data. Specific details were removed because it was not the purpose of this model and would bias it. The date of purchase, the zip code of the auction and state were both removed for this reason. The reason behind excluding the date is that it doesn’t have a relationship to the likelihood a car is a lemon. Someone using this model in the future to predict if their car is a lemon will be using a model that was trained using dates from years ago. The bias may be introduced if sampling bias occurs, making the model treat certain months or days differently than it should. All the details around the make, model and trim of the vehicle sold were removed. The purpose of this model is to make predictions without these, which also will help with overfitting. Two variables about the wheels of the car were removed because they aren’t related to issues significant enough that would cause a car to be a lemon.

Observations with missing values in the remaining features were removed. There were 828 observations removed were MMRAcqusition variables had values of zero or one. There were 18 observations removed were MMRAcqusition variables had “NULL” values. There were 47 observations removed where MMRCurrency variables had zero or one. There were 293 observations removed where MMRCUrrency variables had “NULL” values. There were two types of observations removed, missing data and incorrect data. SVM’s cannot handle variables with missing values and incorrect data will bias our model. Because there is no way a car can average $0 or $1 auction price those observations with that were removed. That is a total of 1,186 observations removed for missing values.

Support vector machines need numerical values for its computation. This required the creation of dummy/binary variables in replacement of the string/categorical variables. This was done for the transmission variable. The remaining dataset had 16 variables with 71,797 observations. Support Vector Machines will treat features differently if their scales vary. This was handled by normalizing all the variables in a transformation node. The data was partitioned so that models were trained on 85% and validated on 15% of the dataset.

**Predictive Models Developed**

All models included all 15 input features. The logic was the higher the dimension, the better chance the SVM can find a hyperplane that separates the target variable.

There were three SVM models trained and validated on the large dataset with 16 variables with 71,797 observations where 12% of observations had a lemon car. The models consisted of a linear model with default *C* penalty parameter (1.0), a two-degree polynomial with default penalty and a linear model with penalty of 0.001. There were two models trained on validated on another dataset with 23,825 observations on the same 16 variables where the target variable was 37% a lemon. On that data, a linear SVM with 0.001 penalty parameter was trained as well as a logistic regression with rounding at .5 threshold.

Support Vector machines do not output a probability associated with each prediction, like a logistic regression would. SVM’s create a decision boundary that observations fall into, like binning, so someone cannot adjust where to round to get a model’s prediction. SVM’s do have a penalty parameter, which does something similar however. This lowers the priority of a hyperplane that makes the correct classifications, which makes the hyperplane’s margin more important. This would increase misclassifications. The first two models had issues accurately marking lemon vehicles, having nearly no specificity across both models on training and validation sets. Even when the penalty parameter was set at 0.001, a thousandth of the default value (1.0), a SVM was still not predicting any lemon cars. This third model has nearly the same predictions as the first two models, as seen in *Figure 1* and *Figure 2*.

In an attempt to get better SVM predictions there was another dataset created. This one was going to have a non-skewed target variable. There are 8,825 observations that have a lemon car. All of them were included in this new dataset along with 15,000 random observations with no lemon. A logistic regression was also included to compare the accuracy of the SVM.

**Results**

The models’ accuracy can be seen in *Figure 1* and *Figure 2*. The SVM’s had issues getting a hyperplane that separates the classification of lemons and non-lemon. There was only one SVM that made any lemon predictions, but even that model had a sensitivity below 0.001. Some of the latter models have very low penalty parameters in an attempt to get more lemon predictions, however it did not yield them. Setting the penalty parameter close to zero allows for more misclassifications. This would allow the SVM to draw a hyperplane that classifies more lemon vehicles but lowers the overall accuracy.

The imbalance in the target classification variable may have been playing a role. The other dataset was to ‘unskew’ the target variable, having 37% of observations be a lemon instead of the original 12%. A SVM will still attempt to separate the lemons from normal cars, but the skewness may be causing issues. This less-skewed data was still problematic for the SVM technique, making no lemon predictions again. Compare this to the logistic regression used on the same ‘less skewed’ data. It made 3,320 predictions an observation was a lemon on training data, getting a sensitivity of 25% and 24% on training and validation data respectively, however that is not a great model performance. There is a way to separate some of the lemons and non-lemons because a logit regression was able to when SVM was not. That being said, the logit still struggled with the overall accuracy, correctly predicting 66% of the time while the SVM (with no lemon cars predicting) had 63% on the less skewed data. That is hardly an overwhelming improvement.

**Conclusions and Takeaways**

The goal of this paper was to use prediction models to decide which vehicles will be lemons based of sale and value information on the vehicle. Support vector machines were the type to be used. These models on these variables failed to produce usable models for this purpose.

Although the accuracy was high for some models’ that is due to a skewed target variable. The models were able to achieve this accuracy without making any predictions a car is a lemon, and that was 12% of the first data set. The sensitivity, therefore, was zero, which means the model really isn’t providing any information to help predict a lemon. Essentially every prediction was non-lemon, which one would not need to build a model to do. Despite lowering the penalty parameters close to zero and using a less-skewed dataset, SVM technique could not draw a decision hyperplane that made lemon predications. A logistic regression was able to have low sensitivity but the overall accuracy was comparable to the SVM that made no lemon predications.

A theory is that SVM is not a good technique for lemon vehicles. There may be no good pattern in 16-dimensional space that can separate them. This may be evident by the incredibly low levels of penalty parameters without any sensitivity or lemon predictions. A logistic regression was able to make some correct lemon predictions.

More investigation should be done on which variables can deliver predictive validity of lemon vehicles. It appears sale and cost information is not enough.

Appendix A

**Bold figures means that model was trained and validated on the less-skewed dataset.**

Figure 1- Models

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Model number | Model kind (kernel) | Penalty  Parameter | T=training  V=validation | Accuracy | Error Rate | Sensitivity | Specificity |
| 1 | HP SVM (linear) | 1 | T | 0.8771 | 0.1229 | 0 | 1 |
| 1 | HP SVM (linear) | 1 | V | 0.8770 | 0.1230 | 0 | 1 |
| 2 | HP SVM (ploy^2) | 1 | T | 0.8772 | 0.1228 | 0.0005 | 1 |
| 2 | HP SVM (poly^2) | 1 | V | 0.8770 | 0.1230 | 0 | 1 |
| 3 | HP SVM (linear) | 0.001 | T | 0.8771 | 0.1229 | 0 | 1 |
| 3 | HP SVM (linear) | 0.001 | V | 0.8770 | 0.1230 | 0 | 1 |
| **4** | **HP SVM (linear** | **0.001** | **T** | **0.6296** | **0.3704** | **0** | **1** |
| **4** | **HP SVM (linear** | **0.001** | **V** | **0.6295** | **0.3705** | **0** | **1** |
| **5** | **Logit Reg.** | **N/A** | **T** | **0.6559** | **0.3440** | **0.2569** | **0.8907** |
| **5** | **Logit Reg.** | **N/A** | **V** | **0.6504** | **0.3495** | **0.2407** | **0.8916** |

Figure 2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Model Number | T=training  V=validation | False Negative | True Negative | False Positive | True Positive |
| 1 | T | 7,500 | 53,526 | 0 | 0 |
| 1 | V | 1,325 | 9,446 | 0 | 0 |
| 2 | T | 7,496 | 53,526 | 0 | 4 |
| 2 | V | 1,325 | 9,446 | 0 | 0 |
| 3 | T | 7,500 | 53,526 | 0 | 0 |
| 3 | V | 1,325 | 9,446 | 0 | 0 |
| **4** | **T** | **7,500** | **12,749** | **0** | **0** |
| **4** | **V** | **1,325** | **2,251** | **0** | **0** |
| **5** | **T** | **5,573** | **11,356** | **1,393** | **1,927** |
| **5** | **V** | **1,006** | **2,007** | **244** | **319** |