ADA442 - Project Report

Classification, Comparison of Logistic Regression and Decision Tree with Carseats Dataset

Erdem Bakır, Gizem Güral

2022-06-06 22:14:07

Contents

1 Abstract **1**

2 Introduction **1**

3 Methodology **2**

4 Data Description **2**

5 Explaratory Data Aanalysis (EDA) and Pre-Processing **3**

6 Model Fit and Numerical Results **7**

6.1 Logistic Regression 7

6.2 Decision Tree Models 10

7 Conclusions **15**

8 References **15**

1 Abstract

In our classification problem, we compared results of logistic regression and decision tree models. While going on this process, we look at and describe the data to understand what we have. After that, we prepared the data in the Pre-processing part to make data ready to use. Also, the dependent variable is converted to categorical variable. We fit our logistic regression model with cross validation, and tree decision models. At the end, we compared the prediction accuracy of both models, and decide which one is better on prediction.

2 Introduction

Our data set containing sales of child car seats at 400 different stores. We choose this data set because it is focusing on analyzing determinants of sales on a special product. And we are dealing with a issue which is common and important for businesses. Classifying these sales and moving around these results can increase profits of businesses.

We have 400 stores as data with 11 variables. Some of them are categorical, the others are continuous. We focused on the classifying sales are good or bad in different stores.

3 Methodology

We use logistic regression model and tree decision models to predict and classify our dependent variable Sales. Logistic regression is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary). A Classification And Regression Tree (CART), is a predictive model, which explains how an outcome variable's values can be predicted based on other values. Firstly, after we look at data, we didn't do anything to deal with missing values, because there is no missing data. We visualized this with "visdat" library as well. In data set, Sales are representing with numbers in thousands, however, we need a categorical (factor) variable to use these models. That's why, after we determinant the mean of Sales as a point, and convert this variable to binary with Good which is higher than mean, and Bad which is lower than mean. Also we look at balance of our dependent variable. We used 4 different libraries in our project. ISLR2 is for our data set "Carseats". Caret is for cross-validation which is highly important to get appropriate results. Tree is for CART models, which is one of the main point of our project. Lastly, visdat has some visualizations at the data description part. Also we set seed for reproductubility.

After the steps, we fit our logistic regression and made predictions on cross-validated version of model. We used Leave One Out Cross Validation (LOOCV) method to apply to model. Also, after we get results, we created decision tree models, normal one and lured one. Also again, we used cross validation to prune our tree and decide the best number for tree.

4 Data Description

Describe the data set you used in the analysis.

```r
library(ISLR2)
library(caret)
```

```
## Zorunlu paket yükleniyor: ggplot2

## Zorunlu paket yükleniyor: lattice
```

```r
library(tree)
library(car)
```

```
## Zorunlu paket yükleniyor: carData
```

```r
library(visdat)
set.seed(4444)
data("Carseats")
```

We get our "Carseats" data from "ISLR2" package which is available in R. This data contains sales of child car seats at 400 different stores. There are 400 observations (stores) and 11 variables. Sales is unit sales in thousands at each location. It is an continuous variable in original data set. However, in our project, sales is converted to binary variable as 1 or 0. CompPrice is price charged by competitor at each location. Income is community income level in thousand of dollars. Advertising is local advertising budget for company at each location in thousands of dollar. Population represents size in region in thousands. Price is the price of company charges for car seats. ShelveLoc is about quality of shelving location. This is a 3 factor variable, Bad, Medium and Good. Age is average age of local population. Education is education level at each location. Urban is another categorical variable with 2 factors to indicate whether store is in urban or rural. (No = Rural, Yes = Urban) US is a factor with levels No and Yes to indicate whether the store is in the US or not. This is a simulated data.

5 Exploratory Data Aanalysis (EDA) and Pre-Processing

Simply mention the properties of data sets with descriptive statistics, additional figures with reference to what you will employ in the modeling part.

```
dim(Carseats)
```

```
## [1] 400  11
```

400 observations, 11 variables.

```
head(Carseats)
```

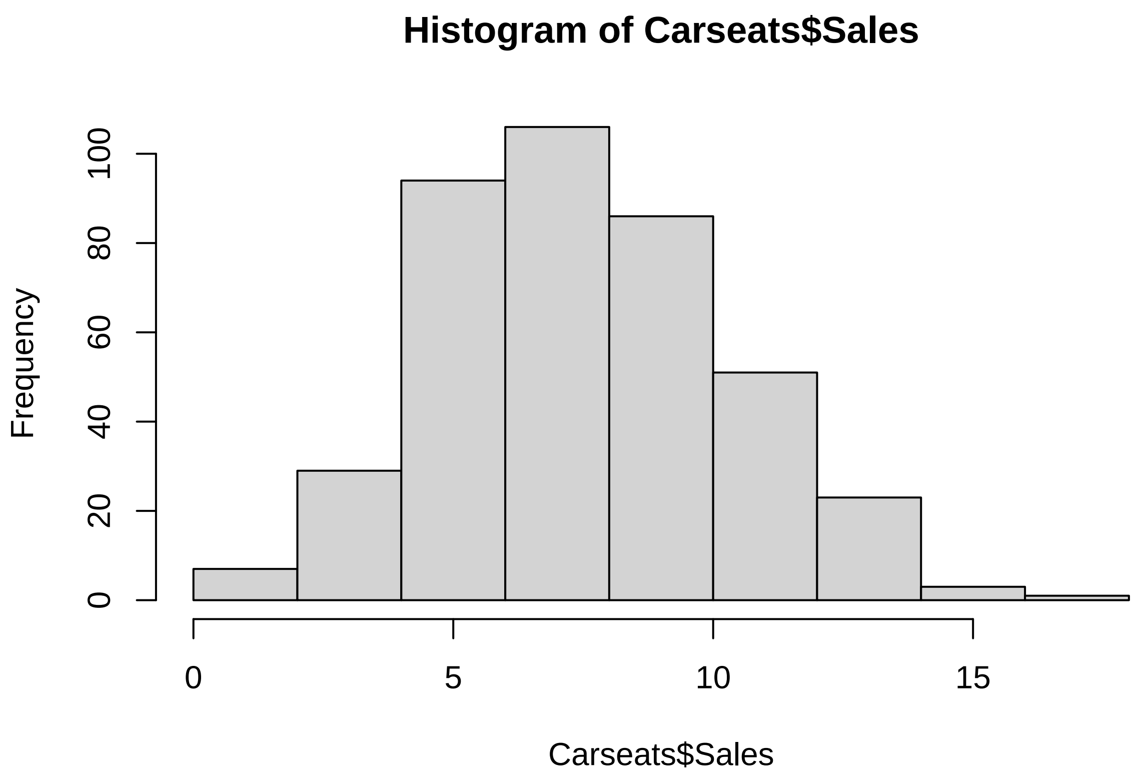
```
##   Sales CompPrice Income Advertising Population Price ShelveLoc Age Education
## 1  9.50       138     73          11        276   120       Bad  42        17
## 2 11.22       111     48          16        260    83      Good  65        10
## 3 10.06       113     35          10        269    80    Medium  59        12
## 4  7.40       117    100           4        466    97    Medium  55        14
## 5  4.15       141     64           3        340   128       Bad  38        13
## 6 10.81       124    113          13        501    72       Bad  78        16
##   Urban  US
## 1   Yes Yes
## 2   Yes Yes
## 3   Yes Yes
## 4   Yes Yes
## 5   Yes  No
## 6    No Yes
```

```
summary(Carseats)
```

```
##      Sales          CompPrice       Income        Advertising
##  Min.   : 0.000   Min.   : 77   Min.   : 21.00   Min.   : 0.000
##  1st Qu.: 5.390   1st Qu.:115   1st Qu.: 42.75   1st Qu.: 0.000
##  Median : 7.490   Median :125   Median : 69.00   Median : 5.000
##  Mean   : 7.496   Mean   :125   Mean   : 68.66   Mean   : 6.635
##  3rd Qu.: 9.320   3rd Qu.:135   3rd Qu.: 91.00   3rd Qu.:12.000
##  Max.   :16.270   Max.   :175   Max.   :120.00   Max.   :29.000
##    Population        Price        ShelveLoc        Age          Education
##  Min.   : 10.0   Min.   : 24.0   Bad   : 96   Min.   :25.00   Min.   :10.0
##  1st Qu.:139.0   1st Qu.:100.0   Good  : 85   1st Qu.:39.75   1st Qu.:12.0
##  Median :272.0   Median :117.0   Medium:219   Median :54.50   Median :14.0
##  Mean   :264.8   Mean   :115.8                Mean   :53.32   Mean   :13.9
##  3rd Qu.:398.5   3rd Qu.:131.0                3rd Qu.:66.00   3rd Qu.:16.0
##  Max.   :509.0   Max.   :191.0                Max.   :80.00   Max.   :18.0
##  Urban       US
##  No :118   No :142
##  Yes:282   Yes:258
##
##
##
##
```

Our dependent variable will be Sales. Minimum value is 0, which means there is no sales in at least one store. Maximum is 16.270. Average number of sales is 7.496. When we look at CompPrice and Price minimum values, our main product has lower minimum level and higher maximum value. It means that price of our main product is so elastic. %64.5 of stores are in United States. Also, %70.5 of all stores around the world is in urban areas. In some stores, our company doesn't allocate money for advertising and minimum value is 0.

```
hist(Carseats$Sales)
```



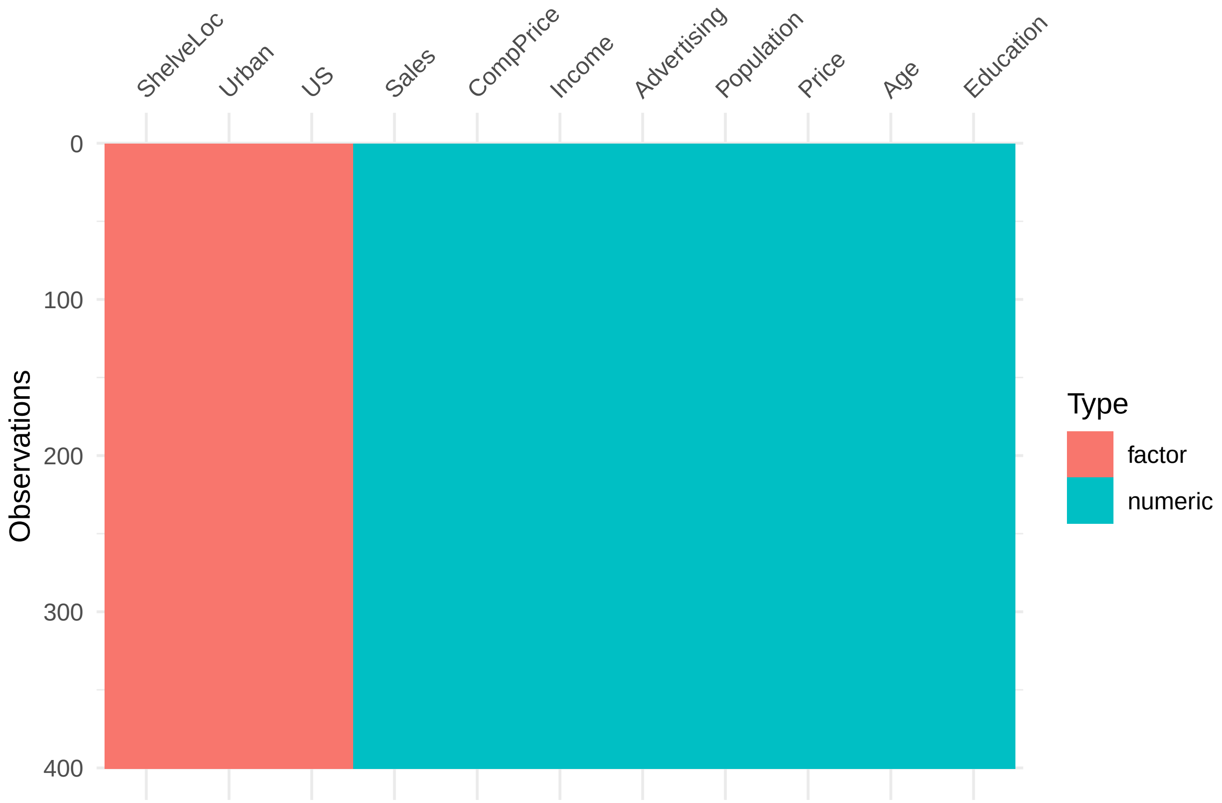
As we see, the distribution of our dependent variable is close to normal distribution.

```
sum(is.na(Carseats)) #No Null data
```

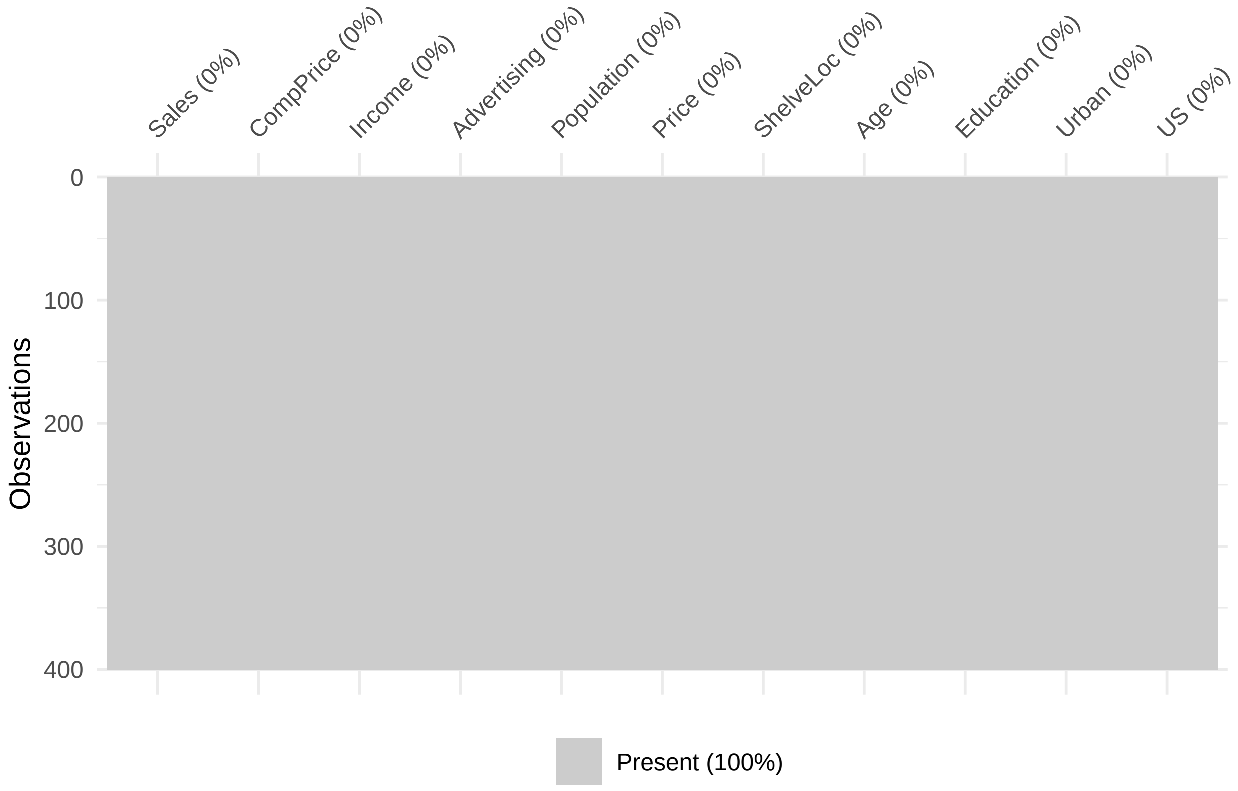
```
## [1] 0
```

```
vis_dat(Carseats)
```

```
## Warning: `gather_()` was deprecated in tidyr 1.2.0.
## Please use `gather()` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
```



```
vis_miss(Carseats)
```



```
df  = data.frame(Carseats)
```

Our data doesn't contain any null/missing data. We have 3 categorical variable.

```
summary(df)
```

```
##      Sales          CompPrice       Income        Advertising    
##  Min.   : 0.000   Min.   : 77   Min.   : 21.00   Min.   : 0.000  
##  1st Qu.: 5.390   1st Qu.:115   1st Qu.: 42.75   1st Qu.: 0.000  
##  Median : 7.490   Median :125   Median : 69.00   Median : 5.000  
##  Mean   : 7.496   Mean   :125   Mean   : 68.66   Mean   : 6.635  
##  3rd Qu.: 9.320   3rd Qu.:135   3rd Qu.: 91.00   3rd Qu.:12.000  
##  Max.   :16.270   Max.   :175   Max.   :120.00   Max.   :29.000  
##    Population        Price         ShelveLoc        Age          Education   
##  Min.   : 10.0   Min.   : 24.0   Bad   : 96   Min.   :25.00   Min.   :10.0  
##  1st Qu.:139.0   1st Qu.:100.0   Good  : 85   1st Qu.:39.75   1st Qu.:12.0  
##  Median :272.0   Median :117.0   Medium:219   Median :54.50   Median :14.0  
##  Mean   :264.8   Mean   :115.8                 Mean   :53.32   Mean   :13.9  
##  3rd Qu.:398.5   3rd Qu.:131.0                 3rd Qu.:66.00   3rd Qu.:16.0  
##  Max.   :509.0   Max.   :191.0                 Max.   :80.00   Max.   :18.0  
##  Urban       US     
##  No :118   No :142  
##  Yes:282   Yes:258  
##                     
## 
```

```
##
##
```

```
mean(df$Sales) #7.49 is mean of Sales.
```

```
## [1] 7.496325
```

```
df$Sales = as.factor(ifelse(df$Sales >= 7.5,"Good","Bad"))
```

We converted our dependent variable Sales to binary variable with two factor, Good and Bad. Also distribution is close to normal. While we are categorizing our dependent variable, we look at the mean of Sales which is 7.49, and called higher than mean as "Good", whether it is called "Bad".

```
ans_no <- length(df$Sales[df$Sales == "Bad"])
ans_yes <- length(df$Sales[df$Sales == "Good"])
ans_no / (ans_no + ans_yes)
```

```
## [1] 0.5025
```

As you see, our dependent variable is balanced. There is nothing we have to do.

```
smp_size <- floor(0.8 * nrow(df))
train_ind <- sample(seq_len(nrow(df)), size = smp_size)
train <- df[train_ind, ]
test <- df[-train_ind, ]
test_sales = df$Sales[-train_ind]
```

We are splitting our processed data to training and testing parts with 80-20 rule. We are going to use them in decision tree models.

6 Model Fit and Numerical Results

6.1 Logistic Regression

First Logistic Model:

```
gmodel.1 = glm(Sales ~ ., data = df, family = binomial(link  = "logit"))
summary(gmodel.1)
```

```
##
## Call:
## glm(formula = Sales ~ ., family = binomial(link = "logit"), data = df)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.47067  -0.30163  -0.00141   0.17169   2.37645
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -1.581172   2.138440  -0.739 0.459662
## CompPrice        0.193083   0.025843   7.471 7.94e-14 ***
## Income           0.025910   0.007225   3.586 0.000336 ***
## Advertising      0.265137   0.047703   5.558 2.73e-08 ***
## Population       0.001080   0.001354   0.797 0.425244
## Price           -0.205050   0.024333  -8.427  < 2e-16 ***
## ShelveLocGood    8.742602   1.111982   7.862 3.78e-15 ***
## ShelveLocMedium  3.074348   0.575835   5.339 9.35e-08 ***
## Age             -0.081594   0.014063  -5.802 6.55e-09 ***
## Education       -0.050655   0.075244  -0.673 0.500815
## UrbanYes        -0.277741   0.430084  -0.646 0.518420
## USYes           -0.702708   0.558521  -1.258 0.208334
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 554.51  on 399  degrees of freedom
## Residual deviance: 184.91  on 388  degrees of freedom
## AIC: 208.91
##
## Number of Fisher Scoring iterations: 7
```

```
vif(gmodel.1)
```

```
##                 GVIF Df GVIF^(1/(2*Df))
## CompPrice   4.466084  1        2.113311
## Income      1.161286  1        1.077630
## Advertising 2.949679  1        1.717463
## Population  1.167191  1        1.080366
## Price       6.751765  1        2.598416
## ShelveLoc   2.825474  2        1.296501
## Age         1.483359  1        1.217932
## Education   1.032533  1        1.016136
## Urban       1.068752  1        1.033804
## US          2.099009  1        1.448796
```

```
process = preProcess(as.data.frame(df),method=c("range"))
df_std = predict(process, as.data.frame(df))
gmodel.std = glm(Sales ~ ., data = df_std, family = binomial(link  = "logit"))
summary(gmodel.std)
```

```
##
## Call:
## glm(formula = Sales ~ ., family = binomial(link = "logit"), data = df_std)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.47067  -0.30163  -0.00141   0.17169   2.37645
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)         6.3735     1.3700   4.652 3.29e-06 ***
## CompPrice        18.9221     2.5326   7.471 7.94e-14 ***
## Income            2.5651     0.7153   3.586 0.000336 ***
## Advertising       7.6890     1.3834   5.558 2.73e-08 ***
## Population        0.5387     0.6757   0.797 0.425244    
## Price           -34.2434     4.0636  -8.427  < 2e-16 ***
## ShelveLocGood     8.7426     1.1120   7.862 3.78e-15 ***
## ShelveLocMedium   3.0743     0.5758   5.339 9.35e-08 ***
## Age              -4.4877     0.7735  -5.802 6.55e-09 ***
## Education        -0.4052     0.6020  -0.673 0.500815    
## UrbanYes         -0.2777     0.4301  -0.646 0.518420    
## USYes            -0.7027     0.5585  -1.258 0.208334    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 554.51  on 399  degrees of freedom
## Residual deviance: 184.91  on 388  degrees of freedom
## AIC: 208.91
##
## Number of Fisher Scoring iterations: 7
```

We created two version of GLM model, normal and standardized one. Because, we can see the importance level of variables from standardized one, however, the interpretation will be more complicated. That's why, the only reason we standardized our data between 0-1 is this. All the predictions and interpretations is going to normal data. After we are fitting our logistic regression model from data, we see the 7 variables are statistically significant with $p < 0$. These are CompPrice, Income, Advertising, Price, ShelveLocGood, ShelveLocMedium and Age. When we look at Price variable, 1 unit (dollars) increase in Price the log odds of admission decreases by %0.20. Second most important variable is CompPrice, when it increases 1 unit in dollars ,the log odds of being Sales High increases by 0.19. As we should expect, price and competitor's price are most important features in front of the consumers. If the Shelve Location is good, it has impact on sales of car seats changes the log odds of being Sales High by 8.74. Advertising is important also. While there are stores which doesn't allocate any money for advertising, company should consider on allocating money on there after this significant result.One unit increasing in advertising the log odds of admission increases by 0.26. Income is another variable that is significant and has impact but less than others. A one unit increase in income variable, increases the log odds by 0.02. Lastly, age has negative coefficient. It means that, when average age of population increases, probability of being Sales High decreases. Because generally young people have little kids who needs child car seats, not elders if they are not buying the product for their grandchild. Every change in age decreases in the log odds by 0.08. The other variables are not significant at any levels. We didn't create any logistic model which just contains significant variables, because the results will be so close. The main reason is coming from hypothesis of testing variables. The null hypothesis is Hn: ß = 0, Ha: ß != 0

Also, we tested multicollinearity in our model, and the mean of variable results are lower than 10. That's why, there is no multicollinearity problem in our model.

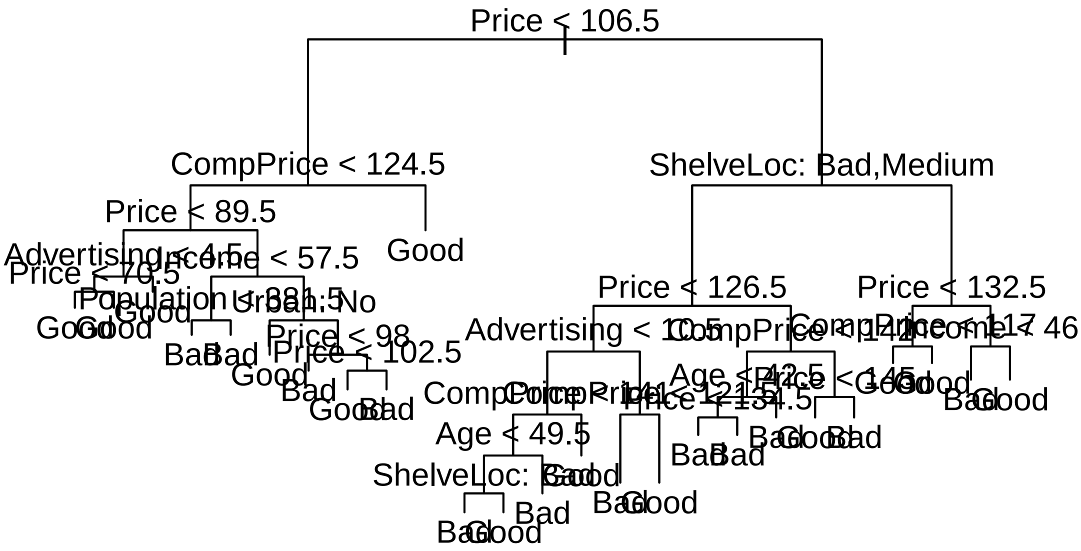
```
#5.1 Cross Validated Logistic Model with LOOCV method
data_ctrl <- trainControl(method = "LOOCV")
model_caret <- train(Sales ~ . ,
                     data = df,
                     trControl = data_ctrl,
                     method = "glm",
                     na.action = na.pass)
print(model_caret)
```

```
## Generalized Linear Model
##
## 400 samples
##  10 predictor
##   2 classes: 'Bad', 'Good'
##
## No pre-processing
## Resampling: Leave-One-Out Cross-Validation
## Summary of sample sizes: 399, 399, 399, 399, 399, 399, ...
## Resampling results:
##
##   Accuracy  Kappa
##   0.8975    0.7949795
```

Here is the cross-validated GLM model with LOOCV methodology. It is needed because of getting more accurate results and avoiding sampling problems. Basically, it splits data set to train model with 399 observations, and testing with 1 observation with different combinations of this sampling. As we see, our model works with %89.25 accuracy and prediction score. No need to compare models, because first model is created from whole data set and no test/trains. It was just to see which variables are important.

6.2 Decision Tree Models

```
tree_model_1 = tree(Sales ~ .,train)
plot(tree_model_1)
text(tree_model_1, pretty = 0)
```



Creating decision tree model with function tree is named same from library on train data set. We are using train-test splitting in this model which we did in pre-processing part. The tree is so confused because of including all variables.

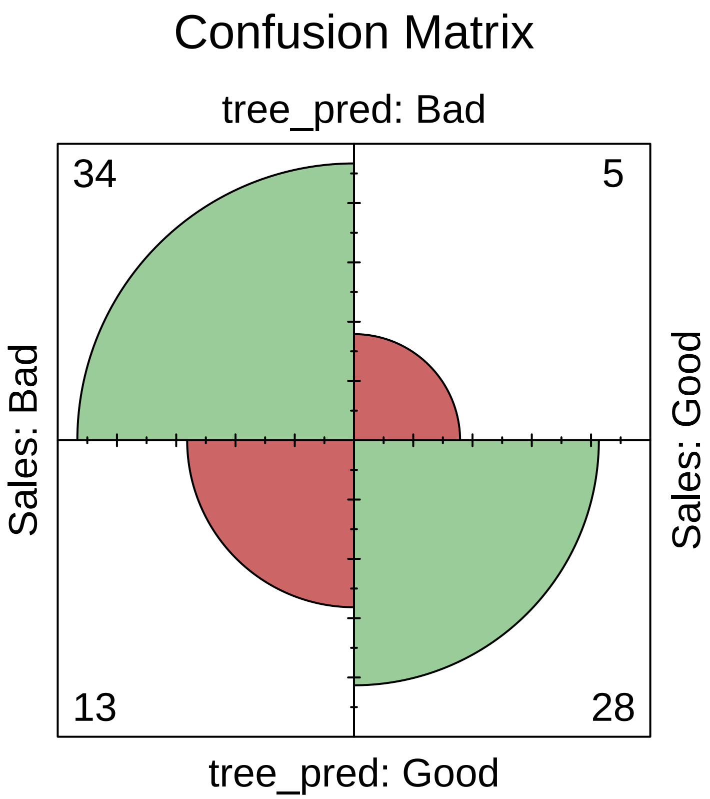
```
tree_pred = predict(tree_model_1, test, type = "class")
cfmatrix1=with(test,table(tree_pred,Sales));cfmatrix1
```

```
##           Sales
## tree_pred Bad Good
##      Bad   34    5
##      Good  13   28
```

```
mean(1-(tree_pred != test_sales))
```

```
## [1] 0.775
```

```
fourfoldplot(cfmatrix1, color = c("#CC6666", "#99CC99"),
             conf.level = 0, margin = 1, main = "Confusion Matrix")
```



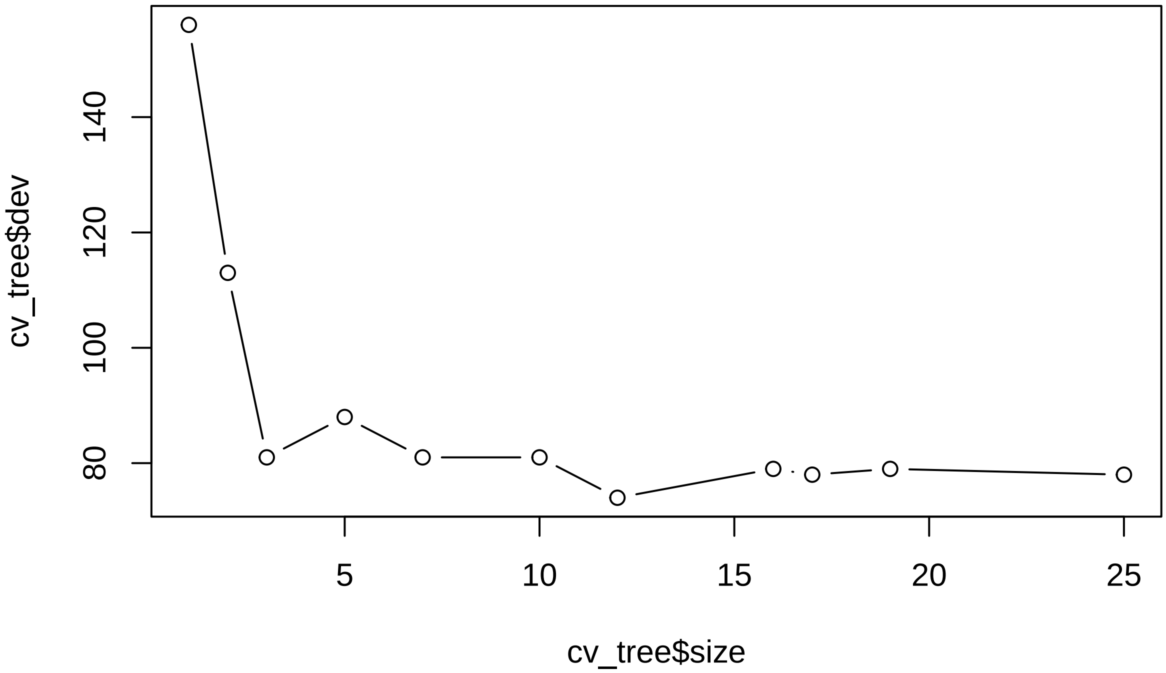
After creating model, we are testing and trying to make predictions on this model.The confusion matrix is created to see prediction results. Columns are real values, rows are predicted values. As you see, 34 value is labelled Bad in real data, and predicted Bad as well and predicted true. 5 of values is Good in our real classification, however, our model predicted it as Bad and it is an error. As same logic, 13 values are predicted wrong, and last 28 values which is Good in our data set and predicted Good as well are true. The ratio of true predictions in total is %77.5. Our model generally fails at prediction of “Good”

Lets prune the tree with cross-validation;

```
#Pruning the tree and cross validation for help
cv_tree = cv.tree(tree_model_1, FUN = prune.misclass)
names(cv_tree)
```

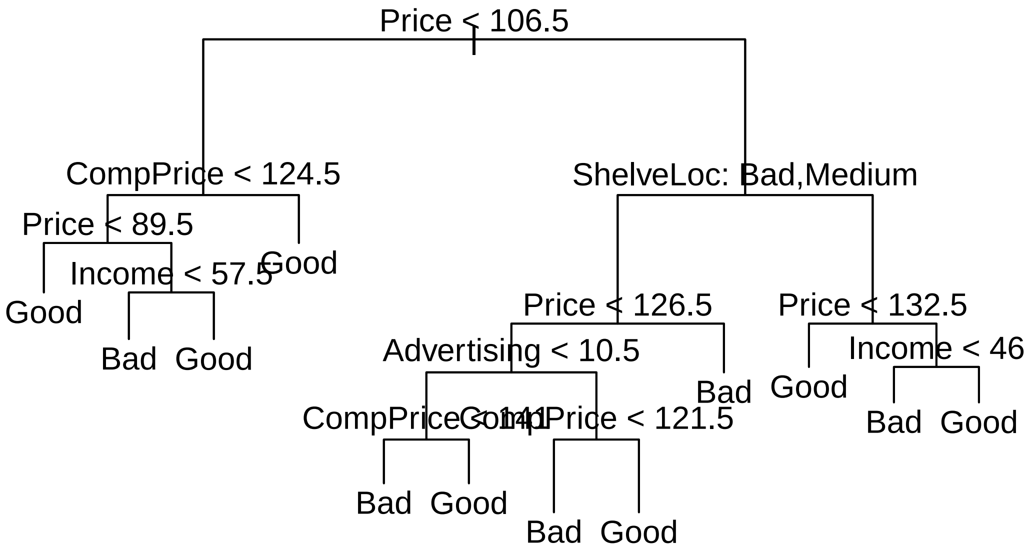
```
## [1] "size"   "dev"    "k"      "method"
```

```
plot(cv_tree$size,cv_tree$dev, type = "b")
```



With looking this plot, we are deciding optimal number of variables which should be used in prune decision tree. As we see in graphic, minimum point is on 12. Also, our next code will calculate the minimum to be sure.

```
#Pruned tree
bestprune <- cv_tree$size[which.min(cv_tree$dev)]
pruned_tree = prune.misclass(tree_model_1, best = bestprune)
plot(pruned_tree)
text(pruned_tree, pretty = 0)
```



Our new tree is more clear and pruned. There are many significant variables, and tree is not cleared so much.

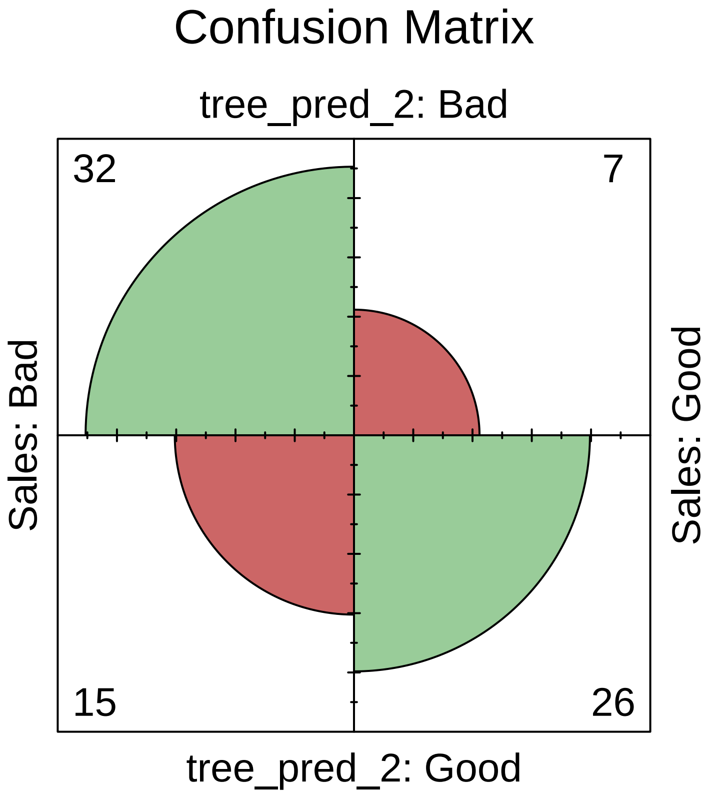
```
#Test pruned tree
tree_pred_2 = predict(pruned_tree, test,type = "class")
cfmatrix2= with(test,table(tree_pred_2,Sales));cfmatrix2
```

```
##            Sales
## tree_pred_2 Bad Good
##        Bad   32    7
##        Good  15   26
```

```
mean(1-(tree_pred_2 != test_sales))
```

```
## [1] 0.725
```

```
fourfoldplot(cfmatrix2, color = c("#CC6666", "#99CC99"),
             conf.level = 0, margin = 1, main = "Confusion Matrix")
```



We are testing pruned tree as well. While we are expecting higher accuracy predictions, our pruned tree has higher miss classification error. This model predicts with %72.5 correction, and we see the wrong predictions on confusion matrix again.

7 Conclusions

Our main goal was comparing logistic regression model and decision tree method on a classification problem. We classify Sales which are higher than mean as “Good”, otherwise “Bad”. In this problem, while we are making predictions, we used different methods. In our logistic regression model which conducted with all variables, we have %89.25 success ratio to predict original values from our cross-validated samples. After that, we created decision tree models, first one contains all variables, and second one is pruned Even our first decision tree model is better than pruned one, both of the models have lower prediction scores with %77.5 and %72.5. In our classification problem, logistic regression method is better to predict possible market car seat sales and the results on Good or Bad. One of the good side of data set is simulated, and give us expected results in regression. However, there could be different results in regression. Also, missing values can be a problem. Generally, decision tree models are giving lower accuracy from logistic regression, because they are not so good on predictions. Also, over-fitting is an another possible danger in these models, however, the prediction accuracy shows that we couldn’t have this kind a problem.

8 References

Give a list of the available works/papers that you used during finalizing your project.

-https://stats.oarc.ucla.edu/r/dae/logit-regression/ -https://www.r-bloggers.com/2018/11/interpreting-generalized-linear-models/ -https://www.statology.org/interpret-glm-output-in-r/ -https://www.guru99.com/r-decision-trees.html -https://rpubs.com/camguild/803096