**Using the GASTURBINE data given on the Beachboard, find the following:**

1. Use stepwise regression (with stepwise selection) to find the “best” predictors of heat rate.

lm(formula = GASTURBINE\_2$HEATRATE ~ RPM + `INLET-TEMP` + `EXH-TEMP` +

AIRFLOW + ENGINE\_Advanced, data = GASTURBINE\_2)

Residuals:

Min 1Q Median 3Q Max

-985.54 -265.80 -52.38 260.74 1409.22

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 1.500e+04 1.113e+03 13.482 < 2e-16 \*\*\*

RPM 8.930e-02 1.321e-02 6.759 6.01e-09 \*\*\*

`INLET-TEMP` -9.965e+00 8.733e-01 -11.410 < 2e-16 \*\*\*

`EXH-TEMP` 1.326e+01 2.305e+00 5.752 3.03e-07 \*\*\*

AIRFLOW -8.597e-01 4.295e-01 -2.002 0.0498 \*

ENGINE\_Advanced 3.840e+02 2.145e+02 1.790 0.0784 .

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 447.2 on 61 degrees of freedom

Multiple R-squared: 0.9273, Adjusted R-squared: 0.9214

F-statistic: 155.7 on 5 and 61 DF, p-value: < 2.2e-16

According to the above R output the best coefficients for heat rate using stepwise selection are RPM, INLET-TEP, EXH-TEMP, and AIRFLOW. I would not include ENGINE\_Advanced since it is not a significant coefficient.

1. Use stepwise regression (with backward elimination) to find the “best” predictions of heat rate.

lm(formula = GASTURBINE\_2$HEATRATE ~ RPM + `INLET-TEMP` + `EXH-TEMP` +

AIRFLOW + ENGINE\_Advanced, data = GASTURBINE\_2)

Residuals:

Min 1Q Median 3Q Max

-985.54 -265.80 -52.38 260.74 1409.22

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 1.500e+04 1.113e+03 13.482 < 2e-16 \*\*\*

RPM 8.930e-02 1.321e-02 6.759 6.01e-09 \*\*\*

`INLET-TEMP` -9.965e+00 8.733e-01 -11.410 < 2e-16 \*\*\*

`EXH-TEMP` 1.326e+01 2.305e+00 5.752 3.03e-07 \*\*\*

AIRFLOW -8.597e-01 4.295e-01 -2.002 0.0498 \*

ENGINE\_Advanced 3.840e+02 2.145e+02 1.790 0.0784 .

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 447.2 on 61 degrees of freedom

Multiple R-squared: 0.9273, Adjusted R-squared: 0.9214

F-statistic: 155.7 on 5 and 61 DF, p-value: < 2.2e-16

Using the above R output, we can see that backward elimination using AIC gives us the same predictors as the stepwise selection. The best predictors for HEATRATE using backward AIC selection are RPM, INLET-TEP, EXH-TEMP, and AIRFLOW. I would not include ENGINE\_Advanced since it is not a significant predictor.

lm(formula = GASTURBINE\_2$HEATRATE ~ RPM + `INLET-TEMP` + `EXH-TEMP`,

data = GASTURBINE\_2)

Residuals:

Min 1Q Median 3Q Max

-1025.8 -297.9 -115.3 225.8 1425.1

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 1.436e+04 7.333e+02 19.582 < 2e-16 \*\*\*

RPM 1.051e-01 1.071e-02 9.818 2.55e-14 \*\*\*

`INLET-TEMP` -9.223e+00 7.869e-01 -11.721 < 2e-16 \*\*\*

`EXH-TEMP` 1.243e+01 2.071e+00 6.000 1.06e-07 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 465 on 63 degrees of freedom

Multiple R-squared: 0.9189, Adjusted R-squared: 0.915

F-statistic: 237.9 on 3 and 63 DF, p-value: < 2.2e-16

Using the above R output, we can see that backward elimination using BIC gives us different predictors than the AIC model. The best predictors for HEATRATE using backward BIC selection are RPM, INLET-TEP, and EXH-TEMP.

1. Use all-possible-regression-selection to find the “best” predictors of heat rate.

best\_cp best\_r2 best\_adjr2

SHAFTS FALSE TRUE FALSE

RPM TRUE TRUE TRUE

CPRATIO FALSE TRUE FALSE

INLET-TEMP TRUE TRUE TRUE

EXH-TEMP TRUE TRUE TRUE

AIRFLOW TRUE TRUE TRUE

POWER FALSE TRUE FALSE

ENGINE\_Traditional FALSE TRUE FALSE

ENGINE\_Advanced TRUE TRUE TRUE

Using all-possible-regressions we see:

The lowest CP has the coefficients: RPM, INLET-TEMP, EXH-TEMP, AIRFLOW, ENGINE\_Advanced

The highest R2 has all the coefficients in the model

The Highest Adj R2 value has coefficients RPM, INLET-TEMP, EXH-TEMP, AIRFLOW, ENGINE\_Advanced

1. Compare the results, parts a-c. Which independent variables consistently are selected as the “best” predictors?

The variables consistently selected as the “best” coefficients are RPM, INLET-TEP, and EXH-TEMP. The coefficients AIRFLOW and ENGINE\_Advanced are not selected in the Backward BIC model.

# Import Packages

library("MASS")

library("leaps")

library("dplyr")

library("fastDummies")

library("ggplot2")

library("GGally")

# Check integrity of the data and its structure

head(GASTURBINE)

summary(GASTURBINE)

sum(is.na(GASTURBINE))

str(GASTURBINE)

# Change Engine to Factor

GASTURBINE$ENGINE = as.factor(GASTURBINE$ENGINE)

str(GASTURBINE)

# Create Dummy variables

GASTURBINE\_2 = dummy\_cols(GASTURBINE, select\_columns = c("ENGINE"))

str(GASTURBINE\_2)

View(GASTURBINE\_2)

# Clean Data. Remove one of the dummy variables and the ENGINE variable

GASTURBINE\_2 = GASTURBINE\_2 %>% select(c(HEATRATE, everything()))

GASTURBINE\_2 = GASTURBINE\_2 %>% select(-c(ENGINE, ENGINE\_Aeroderiv))

str(GASTURBINE\_2)

# Setup full model

full\_model = lm(GASTURBINE\_2$HEATRATE ~ ., data = GASTURBINE\_2)

summary(full\_model)

n = nrow(GASTURBINE\_2)

# Stepwise Selection

Stepwise\_model = stepAIC(full\_model, direction = "both", trace = TRUE)

summary(Stepwise\_model)

# Backward Selction

Backward\_Model\_AIC = stepAIC(full\_model, direction = "backward", trace = FALSE)

summary(Backward\_Model\_AIC)

Backward\_Model\_BIC = stepAIC(full\_model, direction = "backward", trace = FALSE, k = log(n))

summary(Backward\_Model\_BIC)

# All Possible Subsets

mycp = leaps(x = GASTURBINE\_2[,2:10], y = GASTURBINE\_2$HEATRATE, names = names(GASTURBINE\_2)[2:10], method = "Cp")

myr2 = leaps(x = GASTURBINE\_2[,2:10], y = GASTURBINE\_2$HEATRATE, names = names(GASTURBINE\_2)[2:10], method = "r2")

myadjr2 = leaps(x = GASTURBINE\_2[,2:10], y = GASTURBINE\_2$HEATRATE, names = names(GASTURBINE\_2)[2:10], method = "adjr2")

# Select best CP, R^2 and AdjR^2

best\_cp = mycp$which[which((mycp$Cp == min(mycp$Cp))),]

best\_adjr2 = myadjr2$which[which((myadjr2$adjr2 == max(myadjr2$adjr2))),]

best\_r2 = myr2$which[which((myr2$r2 == max(myr2$r2))),]

# Create dataframe of model

options(warn = -1)

model\_1 = data.frame(best\_cp, best\_r2, best\_adjr2)

# View Model

model\_1

**Using the Liver Cancer data given on the Beachboard, fit a logistic regression model for the probability of severity based on various predictors (BMI, Age, Time, Markers, Hepatitis, Jaundice). Describe your findings.**

From the R output below we can see that the only significant coefficients of the GLM model is Age.

Call:

glm(formula = Severity ~ ., family = binomial(), data = Liver\_Cancer\_2)

Deviance Residuals:

Min 1Q Median 3Q Max

-1.4653 -0.9856 -0.7049 1.1943 2.1731

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) -3.250362 1.243213 -2.614 0.00894 \*\*

BMI 0.035372 0.035388 1.000 0.31754

Age 0.034537 0.014183 2.435 0.01488 \*

Time -0.003347 0.011746 -0.285 0.77569

Markers1 0.305331 0.411172 0.743 0.45773

Hepatitis1 -0.272271 0.456639 -0.596 0.55101

Jaundice1 -0.331563 0.383023 -0.866 0.38668

Validation2 0.055988 0.405972 0.138 0.89031

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 180.94 on 135 degrees of freedom

Residual deviance: 168.83 on 128 degrees of freedom

AIC: 184.83

Number of Fisher Scoring iterations: 4

This is unusual, so I checked the VIF of our model to check for multicollinearity. If the variables are highly correlated our model will not be accurate.

BMI Age Time Markers Hepatitis Jaundice Validation

1.047107 1.022236 1.015306 1.039504 1.043218 1.068888 1.04581

All the VIF values are less than 10 and since all other coefficients have a P-Value larger then .05 we can say they are not significantly different than zero. We will refit the model with only the coefficient Age.

Call:

glm(formula = Severity ~ Age, family = binomial(), data = Liver\_Cancer\_2)

Deviance Residuals:

Min 1Q Median 3Q Max

-1.310 -1.021 -0.761 1.210 2.118

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) -2.59395 0.82401 -3.148 0.00164 \*\*

Age 0.03672 0.01365 2.689 0.00717 \*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

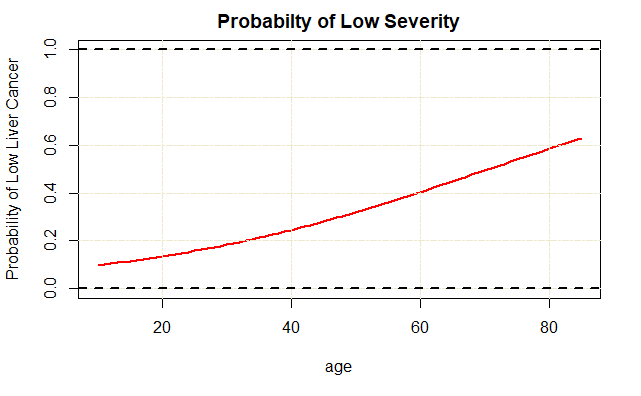
(Dispersion parameter for binomial family taken to be 1)

Null deviance: 180.94 on 135 degrees of freedom

Residual deviance: 172.63 on 134 degrees of freedom

AIC: 176.63

Number of Fisher Scoring iterations: 4



The interpretation of our model is interesting. The model tells us that as your age increase the odds of being diagnosed with high severity liver cancer decreases. Specifically, it says that for every one unit increase in age (one year) the odds of someone being diagnosed with low severity liver cancer increases by a factor of

I would recommend that more information be provided on where the dataset was obtained from, how the data was collected, and how the data was interpreted by the collector. I do not believe the data was properly labeled since it is counterintuitive to suggest that a person’s risk of severe cancer decreases as they age.

--

title: 'STAT 595 Midterm 1 Question 2'

output: word\_document

---

```{r setup, include=FALSE}

knitr::opts\_chunk$set(echo = TRUE)

```

Using the Liver Cancer data given on the Beachboard, fit a logistic regression model for the probability of severity based on various predictors (BMI, Age, Time, Markers, Hepatitis, Jaundice). Describe your findings.

```{r}

# Load needed Libraries

library("readr")

library("tidyverse")

library("car")

```

```{r}

# Read in dataset

Liver\_Cancer = read.csv("Liver\_Cancer.csv")

```

Check data structure and integrity

```{r}

head(Liver\_Cancer)

summary(Liver\_Cancer)

str(Liver\_Cancer)

```

We will start our logistic regression by re-coding the severity variable.High = 0, Low = 1.

```{r}

Liver\_Cancer\_2 = Liver\_Cancer %>% mutate(Severity = ifelse(Severity == "High", 0, 1))

Liver\_Cancer\_2$Markers = as.factor(Liver\_Cancer\_2$Markers)

Liver\_Cancer\_2$Hepatitis = as.factor(Liver\_Cancer\_2$Hepatitis)

Liver\_Cancer\_2$Jaundice = as.factor(Liver\_Cancer\_2$Jaundice)

Liver\_Cancer\_2$Validation = as.factor(Liver\_Cancer\_2$Validation)

view(Liver\_Cancer\_2)

str(Liver\_Cancer\_2)

```

We will now fit the logisitc regresion model. Note that R uses logistic regression to predict the P(1) or in our model the P(Low)

```{r}

logit\_model = glm(Severity ~ ., data = Liver\_Cancer\_2, family = binomial())

summary(logit\_model)

```

From the above results we can see that the only significant coeffecient is Age. All other coeffecients are not significantly different from zero.

We want to check for multicoliniarity

```{r}

vif(logit\_model)

```

We will refit the model with just age

```{r}

logit\_model\_2 = glm(Severity ~ Age, data = Liver\_Cancer\_2, family = binomial())

summary(logit\_model\_2)

```

We want to plot the model to validate our choice of logistic regression

```{r}

newx = seq(10, 85, by = 1)

fm = predict(logit\_model\_2, newdata = data.frame(Age = newx), type = "response")

plot(newx, fm, type = "l", xlab = "age", ylab = "Probability of Low Liver Cancer", main= "Probabilty of Low Severity", ylim = c(0,1), lwd = 2, col = "red")

grid(NULL, NULL, lty = 6, col = "cornsilk2")

abline(h = 1, lty = 2, lwd = 2)

abline(h = 0, lty = 2, lwd = 2)

```

**In your text (An Introduction to Statistical Learning with Applications in R), do exercise #10 a, b, c, d, and i on page 171**

This question should be answered using the Weekly data set, which is part of the ISLR package. This data is similar in nature to the Smarket data from this chapter’s lab, except that it contains 1*,* 089 weekly returns for 21 years, from the beginning of 1990 to the end of 2010.

1. Produce some numerical and graphical summaries of the Weekly data. Do there appear to be any patterns?

Year Lag1 Lag2 Lag3

Min. :1990 Min. :-18.1950 Min. :-18.1950 Min. :-18.1950

1st Qu.:1995 1st Qu.: -1.1540 1st Qu.: -1.1540 1st Qu.: -1.1580

Median :2000 Median : 0.2410 Median : 0.2410 Median : 0.2410

Mean :2000 Mean : 0.1506 Mean : 0.1511 Mean : 0.1472

3rd Qu.:2005 3rd Qu.: 1.4050 3rd Qu.: 1.4090 3rd Qu.: 1.4090

Max. :2010 Max. : 12.0260 Max. : 12.0260 Max. : 12.0260

Lag4 Lag5 Volume Today

Min. :-18.1950 Min. :-18.1950 Min. :0.08747 Min. :-18.1950

1st Qu.: -1.1580 1st Qu.: -1.1660 1st Qu.:0.33202 1st Qu.: -1.1540

Median : 0.2380 Median : 0.2340 Median :1.00268 Median : 0.2410

Mean : 0.1458 Mean : 0.1399 Mean :1.57462 Mean : 0.1499

3rd Qu.: 1.4090 3rd Qu.: 1.4050 3rd Qu.:2.05373 3rd Qu.: 1.4050

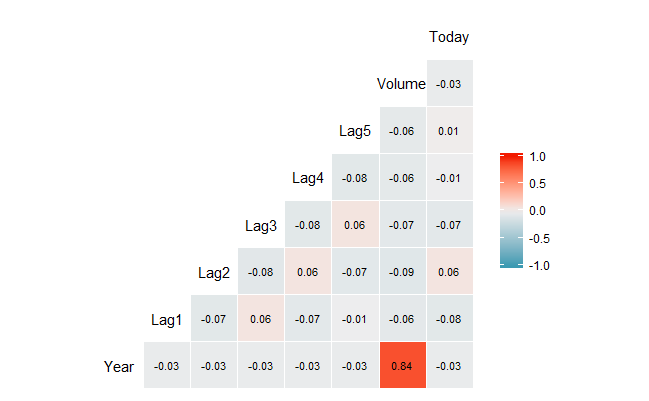
Max. : 12.0260 Max. : 12.0260 Max. :9.32821 Max. : 12.0260

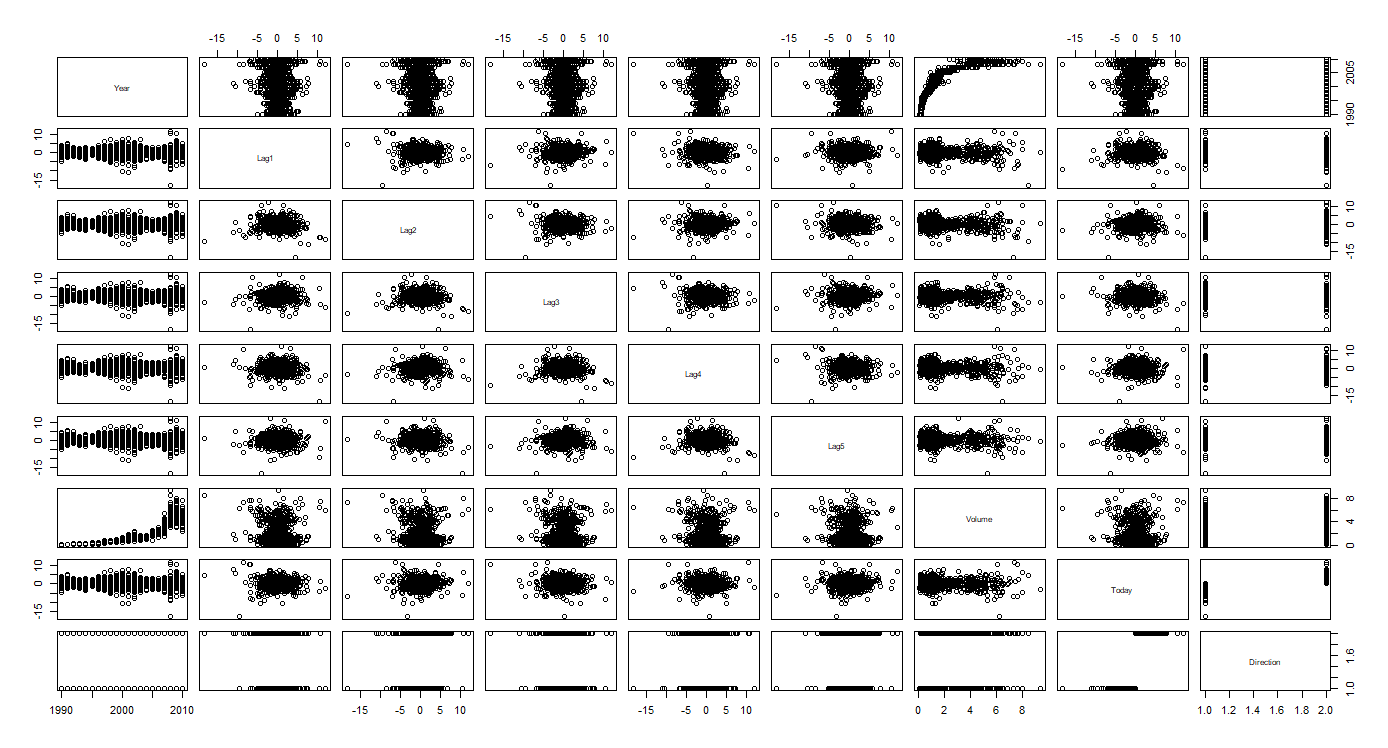
Direction

Down:484

Up :605

Correlation Between Variables and Plots





From the above plots we can see there is a strong positive correlation (0.84) between Year and Volume. Volume is described as the volume of shares traded (average number of daily shares traded in billions) and year is the year the observation was recorded from 1990 – 2010. We can also see that direction is binary (up/down) and can be used in a logistic model.

1. Use the full data set to perform a logistic regression with Direction as the response and the five lag variables plus Volume as predictors. Use the summary function to print the results. Do any of the predictors appear to be statistically significant? If so, which ones?

glm(formula = Direction ~ . - Year - Today, family = binomial(),

data = weekly\_2)

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) 0.26686 0.08593 3.106 0.0019 \*\*

Lag1 -0.04127 0.02641 -1.563 0.1181

Lag2 0.05844 0.02686 2.175 0.0296 \*

Lag3 -0.01606 0.02666 -0.602 0.5469

Lag4 -0.02779 0.02646 -1.050 0.2937

Lag5 -0.01447 0.02638 -0.549 0.5833

Volume -0.02274 0.03690 -0.616 0.5377

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

From the above output we can see that the Intercept is statistically significant with a P-Value of .0019 and Lag 2 is statistically significant with a P-Value of 0.0296. All other coefficients are not significant.

1. Compute the confusion matrix and overall fraction of correct predictions. Explain what the confusion matrix is telling you about the types of mistakes made by logistic regression.

actual

predicted Down Up

Down 54 48

Up 430 557

From the above confusion matrix, we can create a new table that gives us the percentages of True Negative, True Positive, False Positive, and False Negative

actual

predicted Down Up

Down 4.9% 4.4%

Up 39.49% 51.15%

From the above calculations we can see that the logistic model correctly predicted the results 56% of the time and was incorrect the other 44%. The model has difficulty correctly classifying a decrease in the market and incorrectly classified a decrease as an increase 39.49% of the time.

The model tends to classify the majority of the observations as Up which increases the sensitivity of our model but decreases the specificity. This resulted in 430 false positives. Since this is financial data a false positive is potentially worse than a false negative.

A false positive will indicate to a seller that the market value has risen when it fell. This might cause the buyer to sell his or her stock at a loss. A false negative will indicate to a seller the market will lose value so they should not sell.

1. Now fit the logistic regression model using a training data period from 1990 to 2008, with Lag2 as the only predictor. Compute the confusion matrix and the overall fraction of correct predictions for the held out data (that is, the data from 2009 and 2010).

Training Data

actual\_train

predicted Down Up

Down 23 20

Up 418 524

Testing Data

actual\_test

predicted Down Up

Down 9 5

Up 34 56

After we use the training data to help refine our model we can see that the accuracy of the model has increased from 56% to 86% and the misclassification rate has reduced from 44% to 14%. From this we can conclude that by using the training data we have improved the logistic model.

1. Experiment with different combinations of predictors, including possible transformations and interactions, for each of the methods. Report the variables, method, and associated confusion matrix that appears to provide the best results on the held out data. Note that you should also experiment with values for *K* in the KNN classifier.

Logistic Regression with Lag 1 as the only predictor

|  |  |
| --- | --- |
| actual\_train  predicted Down Up  Down 24 23  Up 417 521 |  |
| actual\_test  predicted Down Up  Down 4 6  Up 39 55 |  |

The training model also over predicts the amount of “Up” results. This indicates that the model is too sensitive and not specific enough. I would recommend that we find a new model that balances sensitivity and specificity.

Logistic Regression with interaction between Lag 2 and Lag 1

|  |  |
| --- | --- |
| actual\_train  predicted Down Up  Down 4 3  Up 437 541 |  |
| actual\_test  predicted Down Up  Down 1 1  Up 42 60 |  |

We can see that the interaction model will over predict the “Up” result. This indicates that the model is too sensitive and not specific enough. I would recommend that we find a new model that balances sensitivity and specificity.

Using the full model, we get the below confusion table

|  |  |
| --- | --- |
| actual\_train  predicted Down Up  Down 80 70  Up 361 474 |  |
| actual\_test  predicted Down Up  Down 31 44  Up 12 17 |  |

We can see from the above confusion table that the full model does not do a good job predicting our data. The full model tends to predict more “Down” results. This results in a higher specificity percentage but low accuracy and sensitivity.

# Training with Lag 1

#split the datset

weekly\_trn = filter(weekly, Year <=2008)

weekly\_test = filter(weekly, Year > 2008)

# Fit model and confusion matrix

Lag1\_logit\_model = glm(Direction ~ Lag1, data = weekly\_trn, family = "binomial")

# Convert to classifications

trn\_pred = ifelse(predict(Lag1\_logit\_model, type = "response")>0.5, "Up", "Down")

# Make Predictions on Training Data

trn\_tab = table(predicted = trn\_pred, actual\_train = weekly\_trn$Direction)

#Make Prediction on Test Data

test\_pred = ifelse(predict(Lag1\_logit\_model, newdata = weekly\_test, type = "response")>0.5, "Up", "Down")

test\_tab = table(predicted = test\_pred, actual\_test = weekly\_test$Direction)

#View Tables

trn\_tab

test\_tab

# Interaction Between Lag1 and Lag2

#split the datset

weekly\_trn = filter(weekly, Year <=2008)

weekly\_test = filter(weekly, Year > 2008)

# Fit model and confusion matrix

Lag\_logit\_model = glm(Direction ~ Lag2:Lag1, data = weekly\_trn, family = "binomial")

# Convert to classifications

trn\_pred = ifelse(predict(Lag\_logit\_model, type = "response")>0.5, "Up", "Down")

# Make Predictions on Training Data

trn\_tab = table(predicted = trn\_pred, actual\_train = weekly\_trn$Direction)

#Make Prediction on Test Data

test\_pred = ifelse(predict(Lag\_logit\_model, newdata = weekly\_test, type = "response")>0.5, "Up", "Down")

test\_tab = table(predicted = test\_pred, actual\_test = weekly\_test$Direction)

#View Tables

trn\_tab

test\_tab

# Full Model

Direction ~ . -Year - Today

#split the datset

weekly\_trn = filter(weekly, Year <=2008)

weekly\_test = filter(weekly, Year > 2008)

# Fit model and confusion matrix

Lag\_logit\_model = glm(Direction ~ . -Year - Today, data = weekly\_trn, family = "binomial")

# Convert to classifications

trn\_pred = ifelse(predict(Lag\_logit\_model, type = "response")>0.5, "Up", "Down")

# Make Predictions on Training Data

trn\_tab = table(predicted = trn\_pred, actual\_train = weekly\_trn$Direction)

#Make Prediction on Test Data

test\_pred = ifelse(predict(Lag\_logit\_model, newdata = weekly\_test, type = "response")>0.5, "Up", "Down")

test\_tab = table(predicted = test\_pred, actual\_test = weekly\_test$Direction)

#View Tables

trn\_tab

test\_tab