Jocelyne Walker

**Individual Assignment #4**

**The Lasso Laboratory**

Due: Monday Oct. 2 before 11:59pm

(40 pts. Total)

This homework assignment utilizes The Lasso to explore systematically the reduction/simplification of a more complete demand forecasting model of soft drinks. This is based on the dataset called “Soft Drink Sales.csv” which is an expanded version of the one we used in last week’s homework. As in last homework we assume we are the firm who owns “Product X” and “Product Y” competes for the same demand as our product, with pricing and promotion decisions for Product Y possibly affecting Product X.

The explicit form of the full model we are interested in exploring is:

* The variables “deal\_X” and “deal\_Y” correspond to the dummy variables and
* The variables “feat\_X” and “feat\_Y” are related to in-store feature advertising and correspond to the dummy variables and (taking a value of one if there is an in-store feature advertisement for the corresponding product on a given week)
* The “class” variable indicates if the soft drink is “reg” (regular) or “lit” (light).
* The “H\_Val” variable is a real estate index obtained from the values of the homes in the zone of the store, and it is related to the variable. The higher the index, the more expensive the real estate in the zone.
* There are seven stores in the data set, numbered from 1 through 7. We make this a categorical variable and create the dummy variables to control for store differences.
* “Sales\_oz\_X” are the sales of product X; this variable is in the model.
* The variables “pX” and “pY” correspond to and respectively.

To fit the parameters of this model we take the ln (log(…) in R) transformation on both sides of this equation to obtain:

It is not clear which variables have predictive value, and we want to use The Lasso to identify a subset of variables that can be used to forecast Product X demand. For this purpose we need to first load and pre-process the data as follows:

library(glmnet)

library(fpp)

library(dplyr) # Install and load dplyr package

SD <- read.csv("Soft Drink Sales.csv")[,-9] # Discard sales of Y

#

#Preprocess data

# and then eliminate unneeded columns

#

# First use the “dplyr” to create the additional variables needed

# For an overview of the “dplyr” package see

# <http://jules32.github.io/2016-07-12-Oxford/dplyr_tidyr/#2_dplyr_overview>

#

S <- mutate(SD, STORE=as.factor(STORE),

LpX = log(pX),

LpY = log(pY),

DLpX = deal\_X\*LpX,

DLpY = deal\_Y\*LpY,

LSales = log(Sales\_oz\_X))

S.T <- filter(S, WEEK <= 40) %>%

select(-pX, -pY, -WEEK, -Sales\_oz\_X)

#

# The variable S.T uses only the first 40 weeks of data to develop

# the model.

# Run the model.matrix to create all the dummy variables

# corresponding to the categorical variables.

#

x <- model.matrix(LSales ~ .,data=S.T)[,-1]

y <- S.T[,"LSales"]

#

# Set training and testing sets

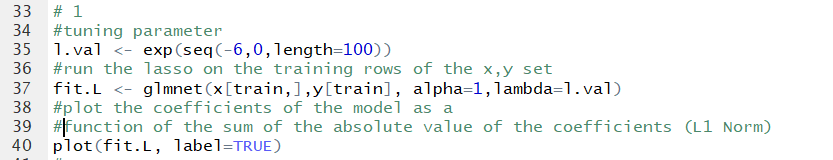
#

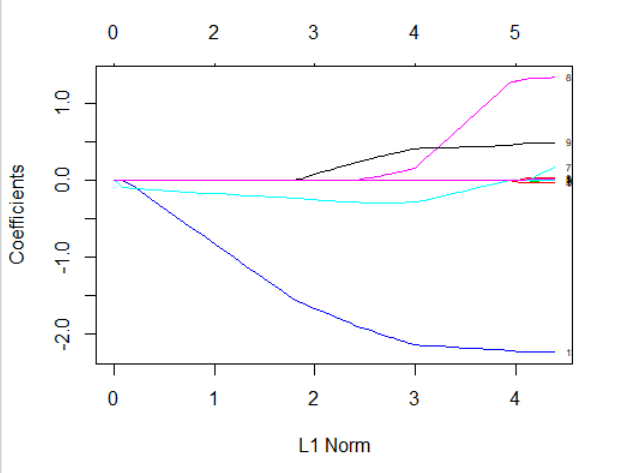
set.seed(1)

train <- sample(1:nrow(x),nrow(x)/2)

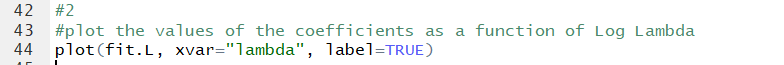
test <- -train

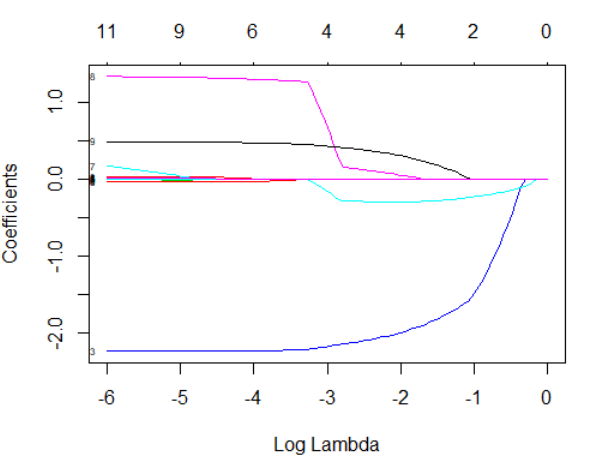
1. (5pts) Select the appropriate sequence of values of the tuning parameter , and run the Lasso on the training rows of the (x,y) set created above. Plot the coefficients of the model as a function of the sum of absolute values of the coefficients (L1 Norm).

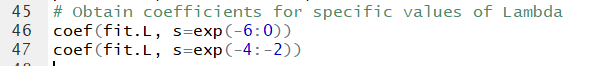


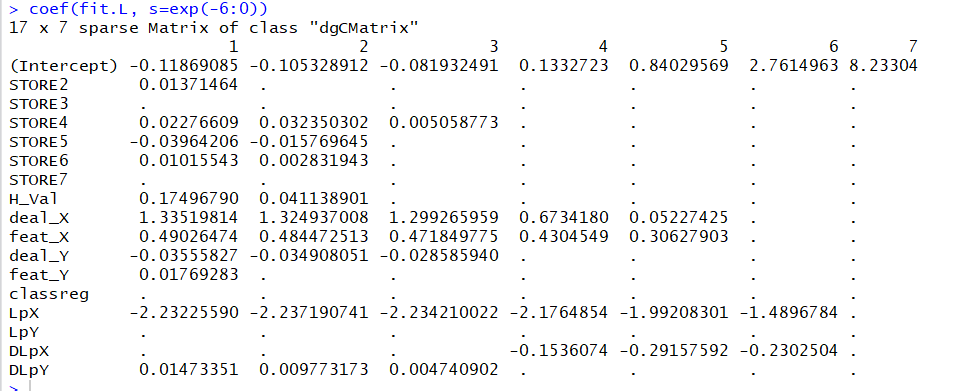


1. (5 pts) Plot the values of the coefficients as a function of the tuning parameter (Log Lambda). Print the values of the coefficients corresponding to the value of that yield 6 and 4 predictive variables in the model.

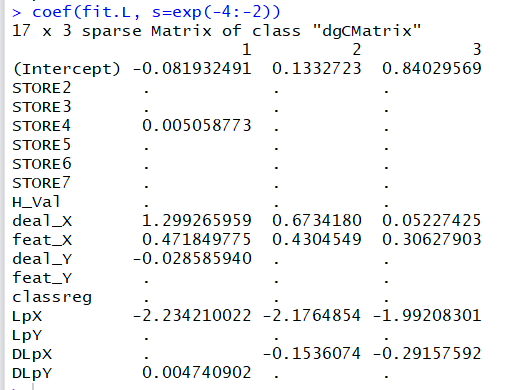






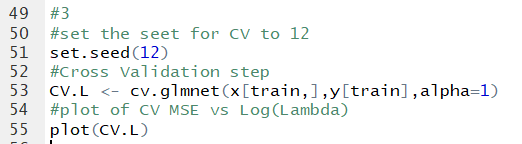


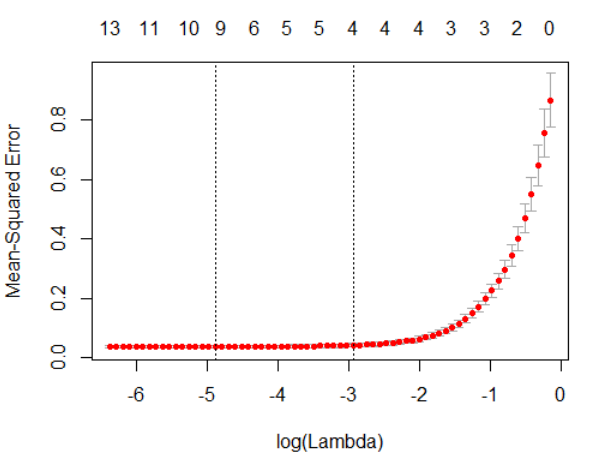
After seeing these coefficients, we see that when Lambda equals -4, we get 6 predictive variables. When Lambda equals -3 or -2, we get 4 predictive variables.



This shows just the 6 predictor variable and 4 predictor variable models.

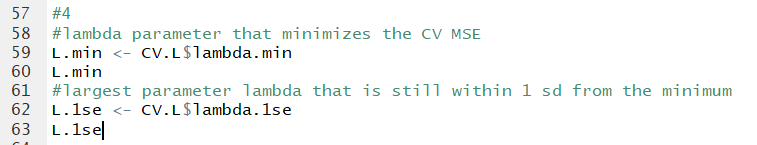
1. (5 pts) Fix the seed for the cross-validation step equal to 12 (run set.seed(12)) and then run the cross-validation step on the training set. Print the plot of cross-validation MSE vs log(Lambda).

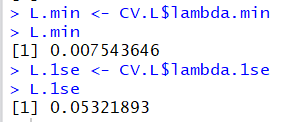




Plot of CV MSE versus Log(Lambda) is above.

1. (5 pts) Identify and extract the value of the parameter that minimizes the cross validation MSE, and also identify and extract the value of the largest parameter that is still within one standard deviation from the minimum. What are the predictive variables identified by these two values of ?

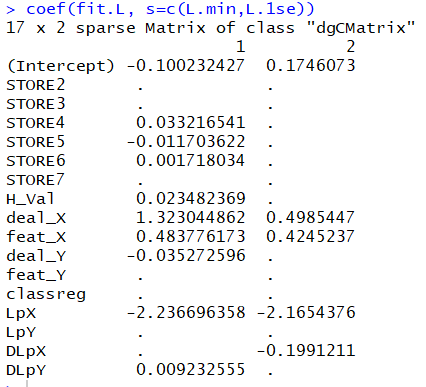




The value of lambda that minimizes CV MSE is 0.00754. The value of lambda that is largest but still within 1 standard deviation of the minimum is 0.0532.

The predictive variables identified by these two values of Lambda:

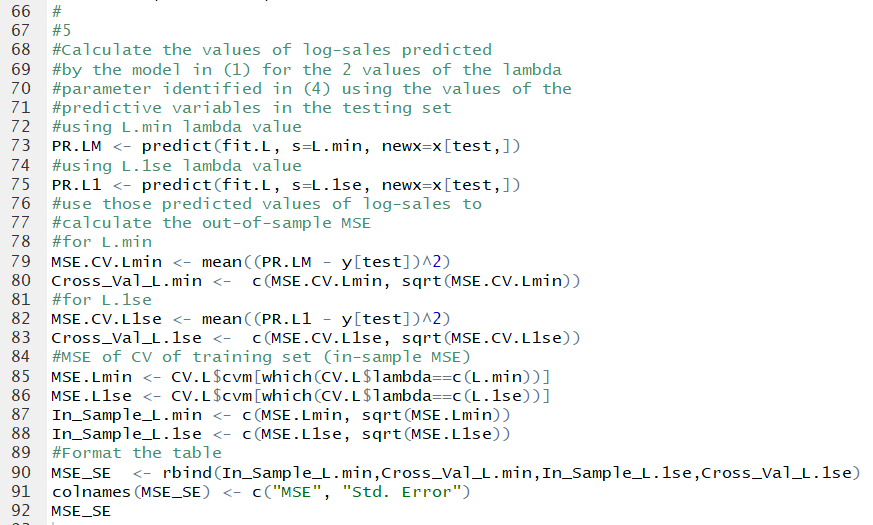


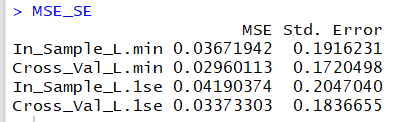


The predictor variables identified by L.min are Store 4, Store 5, Store 6, H\_Val (Home values in the area), deal\_x, feat\_x (in-store feature advertising for X), deal\_Y, LpX (log of price of x), and DLpY (deals of Y times log of price of Y).

The predictor variables identified by L.1se are deal\_X, feat\_X, LpX, and DLpX.

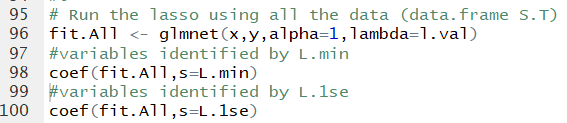
1. (5 pts) Calculate the values of log-sales predicted by the model in (1) for the two values of the parameter identified in (4) using the values of the predictive variables in the **testing** set.; then use these predicted values of log-sales to calculate the out-of-sample MSE, and compare it with the MSE obtained in cross-validation of the **training** set.



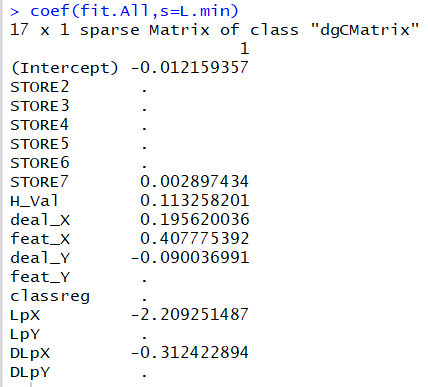


The MSE is lowest for the cross-validation of L.min data. The highest MSE is for the in-sample for the training set for L.1se.

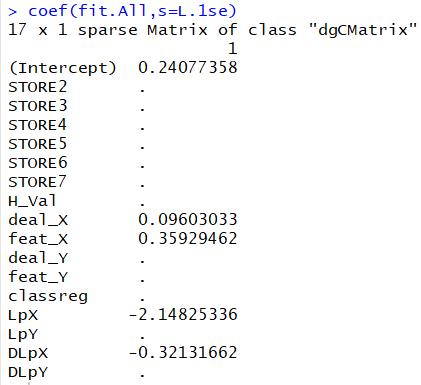
1. (5 pts) Next run The Lasso (as you did in (1)), but now use the entire set (the data.frame **S.T**). What are the variables identified by the two values of the parameter selected in (4)?



Variables Identified by L.min Lambda parameter: Store7, H\_Val (Home Value), deal\_x, feat\_x, deal\_Y, LpX (log-price X), and DLpX (deal X \* log(PriceX)



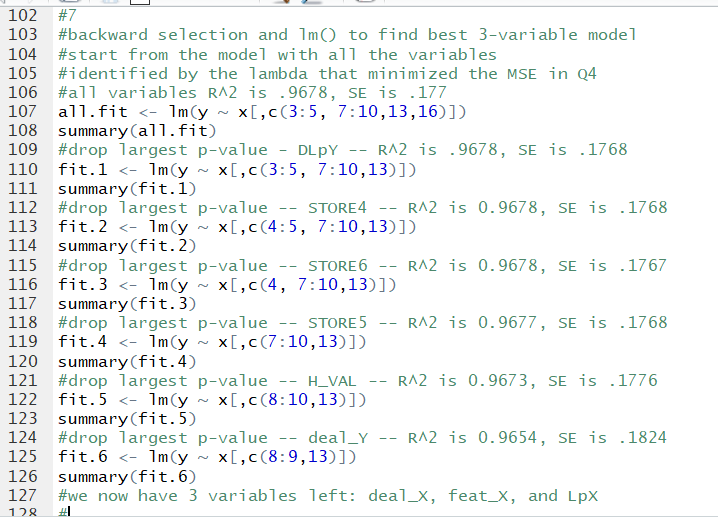
Variables identified by L.1se lambda parameter: dealX, deatX, Log-price X, and DLpX (deal X \* log(Price X).

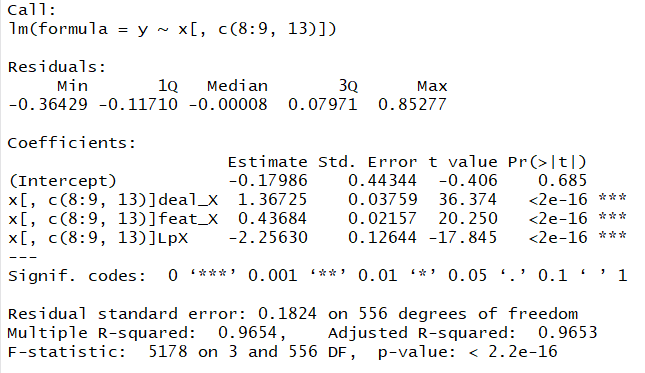


1. (5 pts) Now use Backward Selection and the lm(…) function to find the best three-variable model starting from the model with all the variables identified by the value of the parameter that minimized the MSE in question (4). What is the of this model?

The three variables left in the model are deal\_X, feat\_X, and LpX (log of price of X).

The R^2 of this model is 0.9654.





1. (5 pts) Report in explicit form (with two decimals) the model you obtained in (7). What is the in-sample standard error of the residuals? What is the standard error of the residuals obtained under cross-validation (by the CV(…) function)?

Log(Sales of X) = -0.18 + 1.37(deal\_X) + 0.44(feat\_X) – 2.26(Log(Price\_X))

In-sample standard error of the residuals is 0.1824.

CV(fit.4) is 3.37e-02.