Business Analytics - Assignment 2 **Nicolas Tachet - nt2479**
Prof. Jacob Leshno
Due Date: 09/22/17

As suggested on the assignment, I submitted two files (my R code and this pdf with my answers). Don't hesitate to refer to my R code for more clarity.

1) Open the file in R and use the command summary to print a summary of the data.
Let's print a summary of our dataset. This dataset provides us the evolution of IBM stock during a year.

```
> summary(ibmData)
      Date                Return              X1D          
 Min.   :2012-07-02   Min.   :-8.280000   Min.   :-8.28000  
 1st Qu.:2012-09-28   1st Qu.:-0.620000   1st Qu.:-0.60000  
 Median :2012-12-31   Median :-0.060000   Median :-0.06000  
 Mean   :2012-12-30   Mean   : 0.004618   Mean   : 0.02273  
 3rd Qu.:2013-04-02   3rd Qu.: 0.660000   3rd Qu.: 0.67000  
 Max.   :2013-06-28   Max.   : 4.400000   Max.   : 4.40000  
      X3D                 X1W                 X2W          
 Min.   :-3.54000   Min.   :-2.08000   Min.   :-1.04000  
 1st Qu.:-0.31000   1st Qu.:-0.24000   1st Qu.:-0.15000  
 Median : 0.02000   Median : 0.04000   Median : 0.02000  
 Mean   : 0.02032   Mean   : 0.01839   Mean   : 0.02048  
 3rd Qu.: 0.47000   3rd Qu.: 0.34000   3rd Qu.: 0.22000  
 Max.   : 1.89000   Max.   : 1.32000   Max.   : 0.78000  
      X3W                 X1M                 X6W          
 Min.   :-0.79000   Min.   :-0.60000   Min.   :-0.40000  
 1st Qu.:-0.12000   1st Qu.:-0.10000   1st Qu.:-0.07000  
 Median : 0.04000   Median : 0.03000   Median : 0.07000  
 Mean   : 0.02478   Mean   : 0.02936   Mean   : 0.03193  
 3rd Qu.: 0.22000   3rd Qu.: 0.21000   3rd Qu.: 0.16000  
 Max.   : 0.58000   Max.   : 0.53000   Max.   : 0.34000  
      X2M                 X3M                 X4M          
 Min.   :-0.26000   Min.   :-0.20000   Min.   :-0.14000  
 1st Qu.:-0.06000   1st Qu.:-0.07000   1st Qu.:-0.03000  
 Median : 0.04000   Median : 0.02000   Median : 0.01000  
 Mean   : 0.02863   Mean   : 0.02261   Mean   : 0.02289  
 3rd Qu.: 0.12000   3rd Qu.: 0.11000   3rd Qu.: 0.07000  
 Max.   : 0.30000   Max.   : 0.24000   Max.   : 0.20000  
      X5M                 X6M                 X9M          
 Min.   :-0.06000   Min.   :-0.07000   Min.   :-0.04000  
 1st Qu.:-0.02000   1st Qu.: 0.01000   1st Qu.: 0.00000  
 Median : 0.01000   Median : 0.02000   Median : 0.03000  
 Mean   : 0.02357   Mean   : 0.02546   Mean   : 0.02908  
```

```
 3rd Qu.: 0.06000   3rd Qu.: 0.04000   3rd Qu.: 0.05000
 Max.   : 0.15000   Max.   : 0.11000   Max.   : 0.09000
      X1Y
 Min.   :-0.01000
 1st Qu.: 0.02000
 Median : 0.03000
 Mean   : 0.03847
 3rd Qu.: 0.05000
 Max.   : 0.11000
```

2) Divide your data into two parts: a training set (75%) and a test set (25%). Note that we cannot split the data randomly, why?

We divide our set into two parts as suggested in the assignment. We notice that we can not pick the data randomly as they are time series. We thus have to pick them in a chronological order. Indeed, one data depends on what happened in the past. They would be meaningless if picked randomly.

3) Create 4 validation tests where you use 4 months of data to fit the model and then measure the performance on the following month. For each, use best subset selection to find the best model. Consider subsets of sizes from 1 to 8. Which subset size is best? What is your final model?

To split our training set into 4 sets (each one containing a training set and a validation one), we are going to use masks that will allow us to create subsets starting and ending at a precise date. We are going to use the validation set approach: we divide the training data into four sets of five months. In each of this set we have four months for the training and one for the validation. In the training set it goes from month July 2012 to March 2013 (9 months).

Doing this, we see all the training data set are of size 81 (approx.) and all the validation data set are of size 20 (approx.). Thus, making this chronological split makes us choose a 80% - 20% ratio between training and validation.

Now, for each of these four sets, we are looking for the best adjusted R-squarred using various predictors. For each set, we obtain the appropriate number of predictors (minimizing the Adjusted R2).

```
bestadjR2[[i]]= which.max(regfit.summary$adjr2)

[1] 5 6 6 8
```

We then estimate the MSE for each validation set and different numbers of predictos (from 1 to 8).

```
> val_mse
      [,1]   [,2]   [,3]   [,4]  [,5]   [,6]  [,7]   [,8]
[1,] 1.329 1.4963 1.5773 1.6897 2.076 2.1874 2.237 1.8913
[2,] 0.427 0.4272 0.4126 0.4299 0.427 0.4957 0.487 0.5141
```

```
[3,] 1.874 2.7342 2.0532 1.9760 1.710 1.7218 1.801 1.7209
[4,] 1.384 1.2882 1.6310 1.6985 3.376 4.8249 7.892 6.9025
```

We are going to keep the second model that uses 3 predictors and has the smallest MSE (MSE=0.427). This is our t^*.
After this, we train our model on the (training + validation) set. Our new MSE is 0.978 (with the validation test).
Eventually **on the test set we find a MSE of 2.0744.**

4) On the same 4 subsets, we use lasso regression to find the best model. Consider the values 0, .001, .01, .1, 1, 10, 100, 1000 for λ. Which choice of λ is the best? What is your final model?

```
lambda <- c(0, .001, .01, .1, 1, 10, 100, 1000)
lambda_star =list()
mse_lasso_best = rep(0,4)

for (i in 1:4){
  mse_lasso = rep(0,8)
  for (j in 1:8){

    ##Solve lasso for each value of ? on training data (50 %)

    x_train <- model.matrix(Return~., training[[i]])[,-1]
    lasso.mod <- glmnet(x_train, training[[i]]$Return, alpha = 0, lambda[j]

    ##Compute MSE of each model on validation data (25% of data)

    x_valid <- model.matrix(Return~., validation[[i]])[,-1]
    y_valid <- validation[[i]]$Return
    lasso.pred <- predict(lasso.mod, newx = x_valid)
    mse_lasso[j] = mean((lasso.pred[,1]-y_valid)^2)
  }
  lambda_star[i]=which.min(mse_lasso)
  mse_lasso_best[i]=min(mse_lasso)
}
```

These are the MSE for each set (found on validation test, using the training set to train our model for each subset and each value of lambda):
1) 0.911 2) 0.441 3) 1.496 4) 0.628

On the second set, we have the minimum MSE for Lambda star equal to 0.1. Thus $\lambda^* = 0.1$ is the choice of λ corresponding to the smallest MSE on the validation data. On the test set, we find a MSE equal to 2.219 which is better than with the best subset selection method

5) Pick one of the two models final models from the previous two questions. What is the MSE of your model on the test data? How does that compare to the MSE on the validation tests?

We pick the first model (*Best Subset Selection*), the **MSE was equal to 2.074 for the test set.**

6) Create a trading strategy from the model you picked. Start with $1of investment and every day select to go either long or short according to the prediction of the model. What is the return of your trading strategy on the test data? Based on the results, should you invest using this strategy?

We are going to pick our best subset selection model which had a lower MSE than model using best subset selection. Then, we are going to predict every return for all the data we did not use to build our model (the 25% test data). Indeed, if we were using this model to predict the all data set it would be meaningless... This would mean that we are trying to forecast the data we used to build our model.

Thus, we are going to consider that we are a trader who can start investing on the 2013-04-02 (meaning at the beginning of the test set).

> Nico_fund

Date	Predicted_return	Nature_Trade	Fund_return	Fund_value
2013−04−02	−0.3817743265	Short	−0.93	0.9907000
2013−04−03	−0.1780716934	Short	−0.79	0.9828735
2013−04−04	−0.3866695590	Short	−0.64	0.9765831
2013−04−05	−0.4132284096	Short	−0.90	0.9677938
2013−04−08	−0.4031455073	Short	−0.04	0.9674067
2013−04−09	−0.3233596008	Short	−0.05	0.9669230
2013−04−10	−0.3248574143	Short	−1.33	0.9540629
2013−04−11	−0.2700332411	Short	−0.44	0.9498651
2013−04−12	−0.5013907462	Short	−0.73	0.9429310
2013−04−15	−0.5737868891	Short	−1.00	0.9335017
2013−04−16	−0.5161757475	Short	−1.31	0.9212729
2013−04−17	−0.2240028154	Short	−1.10	0.9111389
2013−04−18	−0.4821278296	Short	−1.20	0.9002052
2013−04−19	−0.4528758404	Short	−8.28	0.8256682
2013−04−22	−0.8077793303	Short	−1.14	0.8162556
2013−04−23	0.2616594937	Long	2.02	0.8327439
2013−04−24	0.5830944661	Long	0.05	0.8331603
2013−04−25	0.2880252108	Long	1.17	0.8429083
2013−04−26	0.4605762492	Long	0.18	0.8444255
2013−04−29	0.2632666620	Long	2.49	0.8654517
2013−04−30	0.4083614360	Long	1.71	0.8802510
2013−05−01	0.1934798792	Long	1.44	0.8929266
2013−05−02	−0.2293053140	Short	−1.38	0.8806042

2013−05−03	0.1440520344	Long	1.05	0.8898505
2013−05−06	0.0408022081	Long	0.85	0.8974143
2013−05−07	−0.1457302475	Short	−0.42	0.8936451
2013−05−08	0.0444920644	Long	1.06	0.9031177
2013−05−09	−0.0115219599	Short	−0.77	0.8961637
2013−05−10	−0.2365957739	Short	−0.61	0.8906971
2013−05−13	−0.0789235665	Short	−0.98	0.8819683
2013−05−14	−0.2190238042	Short	−0.37	0.8787050
2013−05−15	−0.0168189845	Short	−0.05	0.8782657
2013−05−16	−0.0647490158	Short	−0.67	0.8723813
2013−05−17	−0.0209106329	Short	−1.83	0.8564167
2013−05−20	0.0009616438	Long	0.40	0.8598424
2013−05−21	−0.3868727435	Short	−0.51	0.8554572
2013−05−22	−0.2995977697	Short	−0.80	0.8486135
2013−05−23	−0.3929633012	Short	−0.40	0.8452191
2013−05−24	−0.3820768148	Short	−0.21	0.8434441
2013−05−28	−0.3045923060	Short	−1.00	0.8350097
2013−05−29	−0.0791267510	Short	−0.07	0.8344252
2013−05−30	−0.2674494573	Short	−0.69	0.8286676
2013−05−31	−0.2236110743	Short	−0.64	0.8233642
2013−06−03	−0.4228202670	Short	−0.45	0.8196590
2013−06−04	−0.3624066320	Short	−1.32	0.8088395
2013−06−05	−0.5784935651	Short	−1.67	0.7953319
2013−06−06	−0.4790429499	Short	−0.52	0.7911962
2013−06−07	−0.0481737638	Short	−1.25	0.7813062
2013−06−10	−0.1397335195	Short	−0.64	0.7763059
2013−06−11	−0.3199722328	Short	−0.51	0.7723467
2013−06−12	−0.3052965864	Short	−1.36	0.7618428
2013−06−13	−0.3297626979	Short	−1.28	0.7520912
2013−06−14	0.0118380791	Long	0.77	0.7578823
2013−06−17	−0.1923656499	Short	−0.42	0.7546992
2013−06−18	0.2358003465	Long	0.90	0.7614915
2013−06−19	0.2048473593	Long	1.43	0.7723808
2013−06−20	0.0077309057	Long	2.27	0.7899139
2013−06−21	0.0337886604	Long	0.96	0.7974970
2013−06−24	0.2252062974	Long	0.98	0.8053125
2013−06−25	0.2760326519	Long	0.74	0.8112718
2013−06−26	0.4308085361	Long	0.06	0.8117586
2013−06−27	0.3600095105	Long	0.41	0.8150868
2013−06−28	0.3813806913	Long	2.32	0.8339968

Our $1 investment is now only 83 cents! We lose 17% of our inital investment in 3 months. This strategy is definitely not a good one. This might be explained by the fact that past results are not the only explanation to the evolution of a stock. If we wanted to beat the market, we should try to find strategies/factors that are not used by other traders in the market (trading edges).
Indeed, if we look at Morgan Stanley's BEST model (Lecture 5), we see that past results are just one factor among fifty other relevant ones.