Data Intensive Computing Assignment 2 CSE 487/587

-Utkarsh Srivastava

Description: In this assignment, you are required to use R language/tools in CCR to do time-series forecast of stock price using the same data in hw#1. There are many approaches to forecasting. In this homework, you will compare three techniques, namely, Linear Regression Model, Holt-Winters Model, and ARIMA model.

Implementation:
Data of the stocks has been provided for 36 months .To compute the values and plot the graph, the data for each stock will be split into two parts: The first part with 744 trading days is used for training . The second part with 10 trading days is used for testing.

As specified in the problem statement , The MAE (Mean Absolute Error) is used to evaluate error in time series analysis for each stock.

$$MAE_i \text{ (each day)} = | \text{forecastData} - \text{testData} |$$
$$\text{sum of MAE} = \Sigma\, MAE_i$$

Based on this error, you are required to find stocks with best-forecasted performance, using the three techniques as follows:
-We first read the data from all files by using a file object
- if the length of the file is not 755, we skip it
- using **read.csv** function, the file is read and its timeseries Data is got using **ts** function, with the frequency set for 1 year or 365
- As discussed above, the data is then split into **trainData** and **testData** by setting the ending and starting values as 2014 and 2015

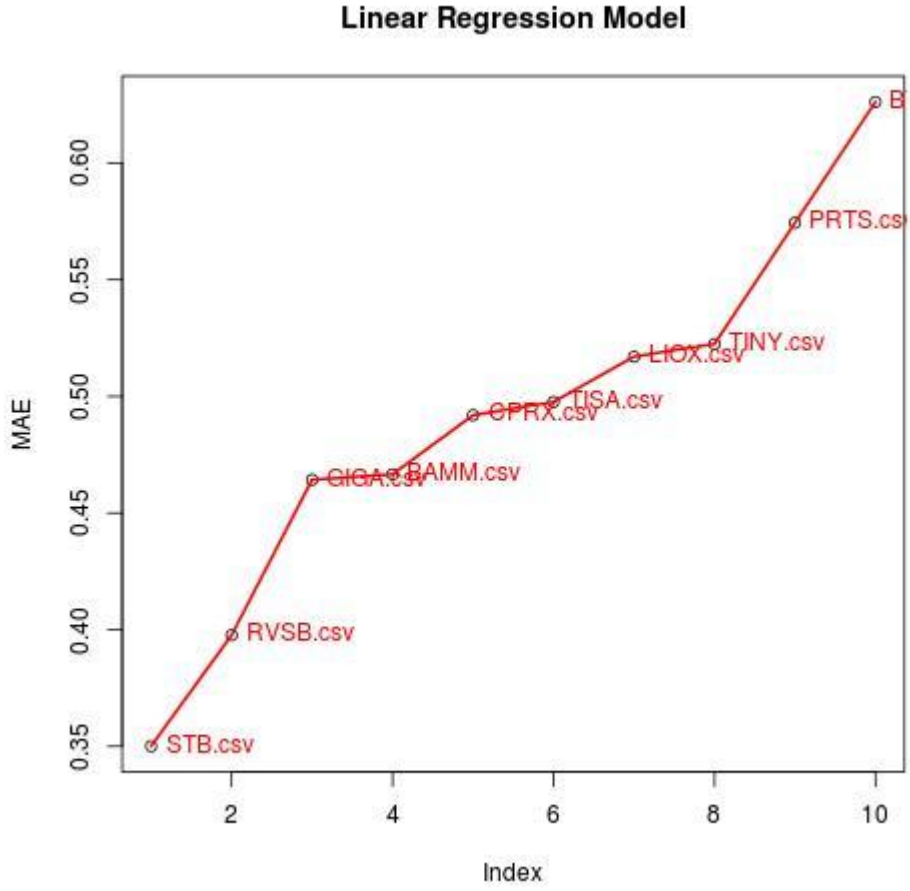
<u>The top 10 stocks with the minimum sum of MAE using Linear Regression Model</u>

-A linear regression model describes the relationship as a linear equation between 2 variables.
- The constant intercept of the equation is the error term
-We use the **tslm** function to fit the linear model. This function is a wrapper function of lm function

```
tslm(trainingData ~ trend + season)
```

- This data is then passed to the **forecast** method in tune and along with the length of testData
- MAE is then calculated and stored in an array
-The two arrays , each of stockname and the corresponding MAE are ordered using **order** method and the minimum 10 values are plotted on the graph

STB	0.3500859951
RVSB	0.3976781327
GIGA	0.4642997543
BAMM	0.4666093366
CPRX	0.4919287469
TISA	0.4976289926
LIOX	0.517002457
TINY	0.5224692875
PRTS	0.5744103194


Linear Regression Model
MAE
Index
0.35
0.40
0.45
0.50
0.55
0.60
2
4
6
8
10
STB.csv
RVSB.csv
GIGA.csv
BAMM.csv
CPRX.csv
TISA.csv
LIOX.csv
TINY.csv
PRTS.csv
B

The top 10 stocks with the minimum sum of MAE using Holt-Winters Model

-We use the **HoltWinters** function to fit the model and contains both trend and seasonal variations

HoltWinters(trainData, alpha=NULL, beta=NULL, gamma = FALSE)
seasonal = c("additive", "multiplicative"),
start.periods = 2, l.start = trainData, b.start = testData,
s.start = NULL,
optim.start = c(alpha = 0.3, beta = 0.1, gamma = 0.1),
optim.control = list())

- This data is then passed to the **forecast** method in tune and along with the length of testData
- MAE is then calculated and stored in an array
-There are two modes in this function

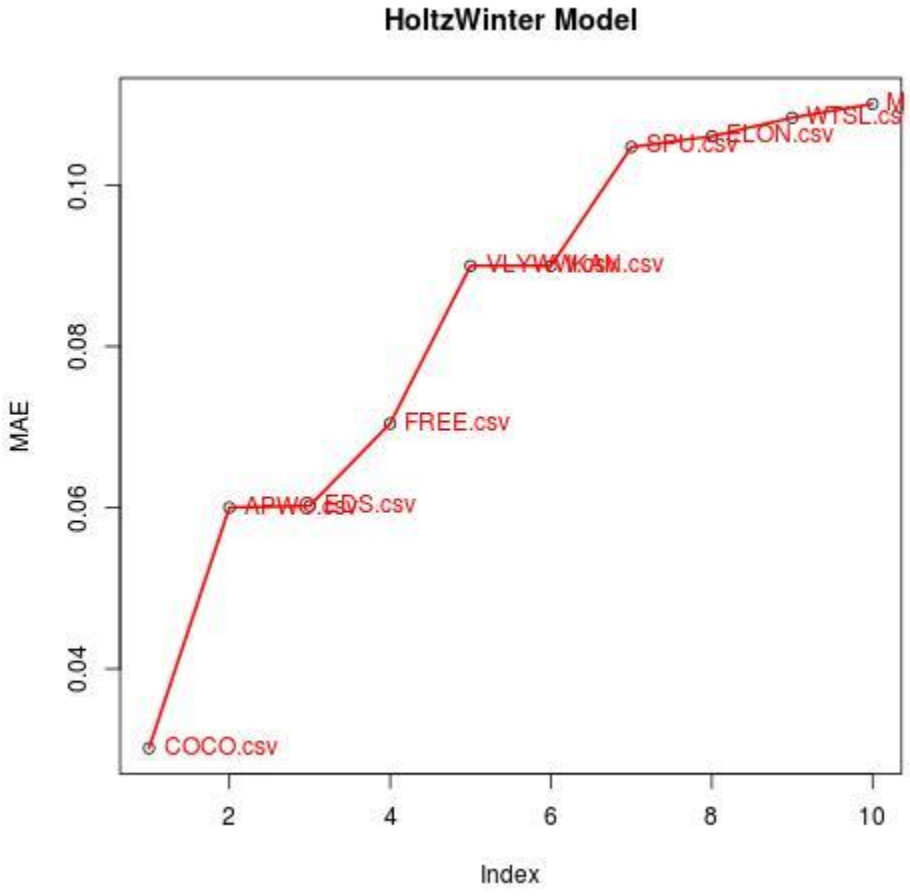
Beta : If set to 0, the function will do exponential smoothing.

Gamma : parameter used for the seasonal component. If set to 0, an non-seasonal model is fitted.

-The two arrays , each of stockname and the corresponding MAE are ordered using **order** method and the minimum 10 values are plotted on the graph

The values for beta=false and gamma=false

beta and gamma false->

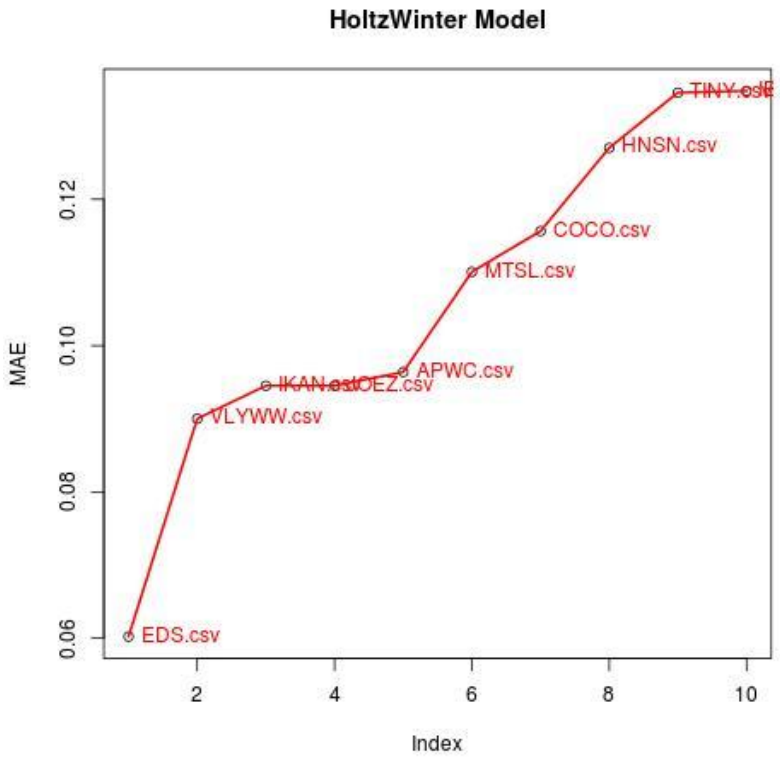


COCO	0.0301567581
APWC	0.0600065427
EDS	0.0602272459
FREE	0.0704327241
VLYWW	0.09
IKAN	0.09
SPU	0.1047463286
ELON	0.1060639647
WTSL	0.1083733868

The values for gamma=false

EDS.csv "EDS.csv" "0.0602270930678119"
VLYWW.csv "VLYWW.csv" "0.09"
IKAN.csv "IKAN.csv" "0.0945163102270198"
JOEZ.csv "JOEZ.csv" "0.0945248004932003"
APWC.csv "APWC.csv" "0.0963925582913947"
MTSL.csv "MTSL.csv" "0.110086724836609"
COCO.csv "COCO.csv" "0.115658980122067"
HNSN.csv "HNSN.csv" "0.127034131152494"
TINY.csv "TINY.csv" "0.134586325959772"
IBCA.csv "IBCA.csv" "0.134818345387194"

only gamma=False->



The top 10 stocks with the minimum sum of MAE using ARIMA Model

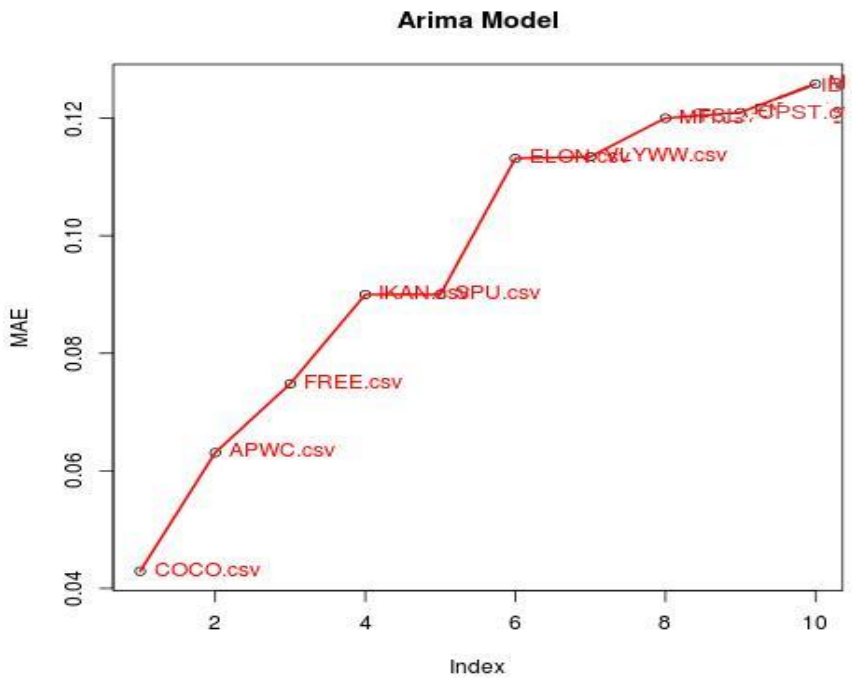
-The **auto.arima** function returns best ARIMA model according to either AIC, AICc or BIC value.

auto.arima(trainData)

- This data is then passed to the **forecast** method in tune and along with the length of testData
- MAE is then calculated and stored in an array
- The two arrays , each of stockname and the corresponding MAE are ordered using **order** method and the minimum 10 values are plotted on the graph

The values got are as follows :

0.04291029 COCO.csv
0.06308866 APWC.csv
0.07480337 FREE.csv
0.09000000 IKAN.csv
0.09000000 SPU.csv
0.11315610 ELON.csv
0.11343679 VLYWW.csv
0.12583623 MTSL.csv
0.13000000 CPST.csv
0.13363392 IBCA.csv



NOTE:
In order to make the arima model to work quicker , we can also add extra pair of parameters to it highighted below :

fitData = auto.arima(trainData,**seasonal=F, lambda=NULL, approximation=T**)

Although, this would change the last three values and the result obtained is

	stockname	arima_MAE
COCO.csv	"COCO.csv"	"0.0429102862857658"
APWC.csv	"APWC.csv"	"0.0630886648410769"
FREE.csv	"FREE.csv"	"0.0748033738137716"
IKAN.csv	"IKAN.csv"	"0.0900000000000001"
SPU.csv	"SPU.csv"	"0.0900000000000001"
ELON.csv	"ELON.csv"	"0.11315609717094"
VLYWW.csv	"VLYWW.csv"	"0.113436790470689"
MFI.csv	"MFI.csv"	"0.120000000000006"
ENZN.csv	"ENZN.csv"	"0.12096462378173"
MTSL.csv	"MTSL.csv"	"0.12583622799868"