Stat 153 Time Series Project
Spring2016

Philhoon Oh

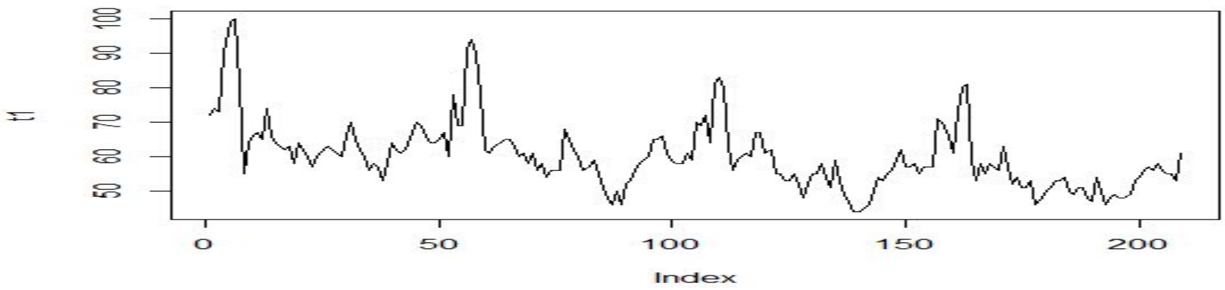
Abstract

The goal of this project is to predict the next 52 observations of the two given time series datasets. Each of these datasets is of length 209 and gives the google trends data (downloaded on 06 November, 2016) for a particular query from the week of November 06, 2011 to the week of November 11, 2015.

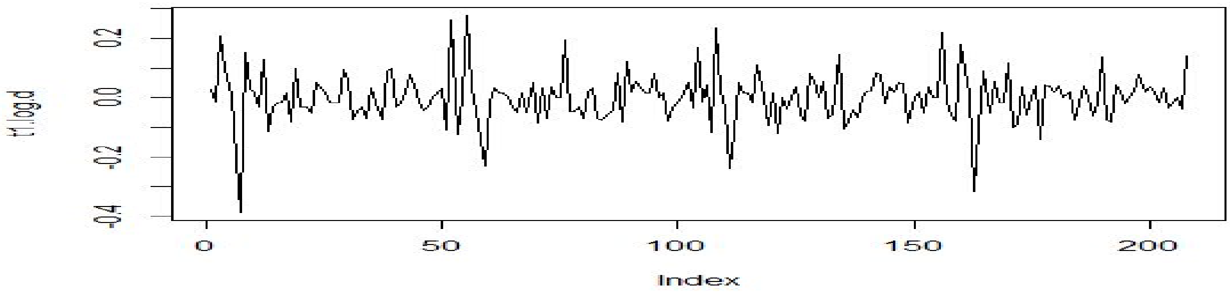
(data1: ballet, data2: DNA)

Exploratory Data Analysis(EDA)

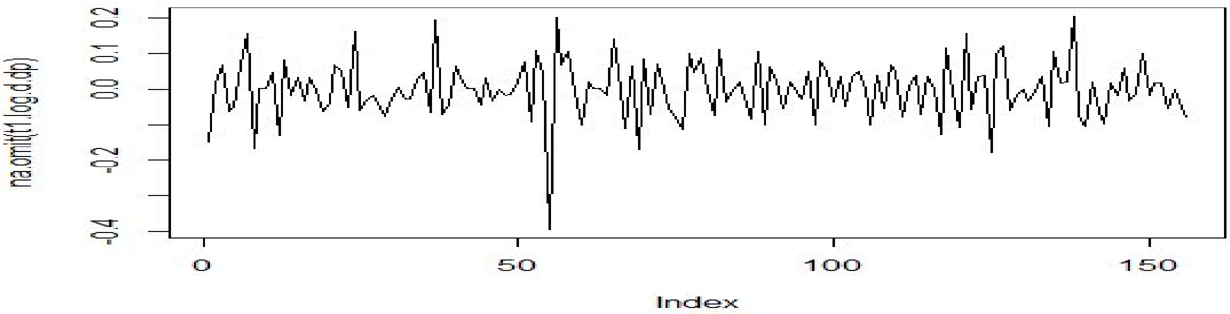
First, let's take a look at our given dataset: Q-1.csv



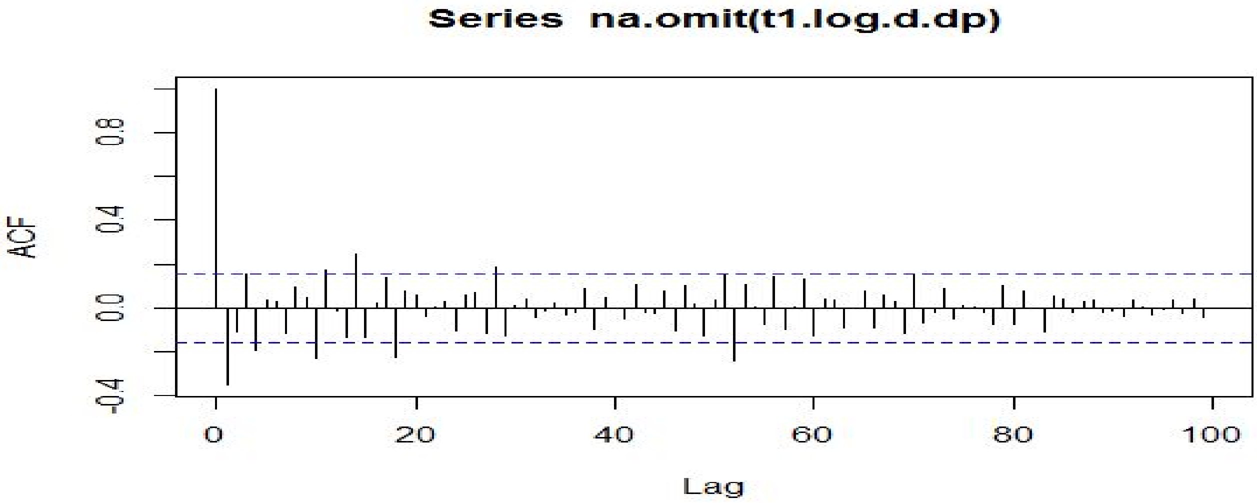
It seems that the resulting plot shows periodicity. Let's take a log to stabilize the variability and take a difference to remove the trend on the given dataset.

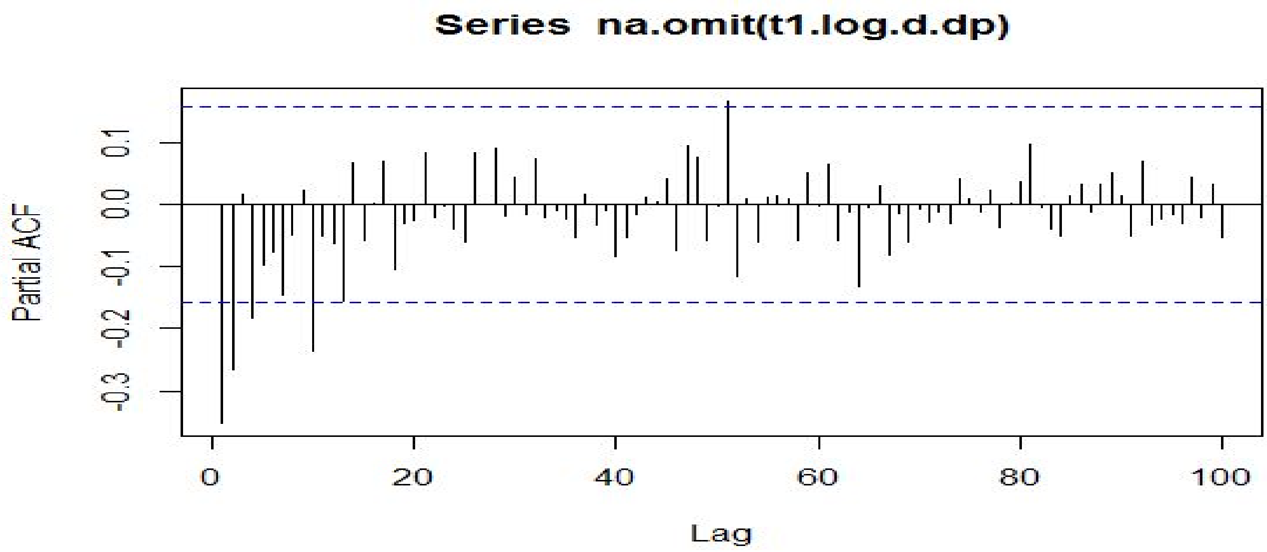


Resulting plot shows the seasonality of 52. We can remove the seasonality by differencing at lag = 52.



Now resulting plot seems to be white noise with one peculiar point around 52. That point might be an outlier; however, we are not going to remove that point since it's just one point, which is still probable, and our dataset is small. Let's take a look at ACF, PACF





ACF plot suggests that the data might follow ARMA(0,1)*(0,1)_52. PACF plot does not reveal much information but there might ARMA(1,0)*(0,1)_52 OR ARMA(2,0)*(0,1)_52 in our process.

Method

1. Differencing

According to EDA, PACF suggest that ARMA(1,0)*(0,1)_52 OR ARMA(2,0)*(0,1)_52 are suggested models. Let's check this out.

Model	AIC	BIC	CV score 1	CV score 2
a1.0.1	-369.8849	-360.7353	25.23343	26.74823
a1.1.1	-400.6762	-388.4768	17.13972	11.94141
a2.1.1	-398.9074	-383.6581	19.381	12.33244

- a1.0.1 = arima(t1.log, order=c(1,1,0), seasonal = list(order=c(0,1,1), period=52))
- a1.1.1 = arima(t1.log, order=c(1,1,1), seasonal = list(order=c(0,1,1),period=52))
- a2.1.1 = arima(t1.log, order=c(2,1,1), seasonal = list(order=c(0,1,1),period=52))
- CV score1 : mean of Cross-validation 1 - dividing the data set into 3 subgroups and do the cv.
- CV score2 : mean of Cross-validation 2 - starting from training set of 108, increasing by one, do the CV on the

resting of the dataset starting from 101, decreasing by one. EX) training set =108, testing set = 101. Then training set=109, testing set = 100.

It turns out among these three models, a1.1.1 is the best model.

According to EDA, ACF plot suggests that the data might follow ARMA(0,1)*(0,1)_52.

	m1.1	m1.2	m1.3	m2.1	m3.1	m4.1	m5.1	m6.1	m2.2	m2.3	m2.4
AIC	-383.5697	-382.3091	-380.6737	-397.1538	-396.7667	-400.4293	-398.8003	-396.9185	-397.6748	-395.6758	-394.5164
BIC	-374.4202	-370.1097	-365.4244	-384.9544	381.5174	-382.1302	-377.4513	-372.5197	-382.4255	-377.3767	-373.1674

- m1.1 = arima(t1.log, order = c(0, 1, 1), seasonal = list(order = c(0, 1, 1), period = 52))
- m1.2 = arima(t1.log, order = c(0, 1, 1), seasonal = list(order = c(0, 1, 2), period = 52))
- m1.3 = arima(t1.log, order = c(0, 1, 1), seasonal = list(order = c(0, 1, 3), period = 52))
- m2.1 = arima(t1.log, order = c(0, 1, 2), seasonal = list(order = c(0, 1, 1), period = 52))
- m3.1 = arima(t1.log, order = c(0, 1, 3), seasonal = list(order = c(0, 1, 1), period = 52))
- m4.1 = arima(t1.log, order = c(0, 1, 4), seasonal = list(order = c(0, 1, 1), period = 52))
- m5.1 = arima(t1.log, order = c(0, 1, 5), seasonal = list(order = c(0, 1, 1), period = 52))
- m6.1 = arima(t1.log, order = c(0, 1, 6), seasonal = list(order = c(0, 1, 1), period = 52))
- m2.2 = arima(t1.log, order = c(0, 1, 2), seasonal = list(order = c(0, 1, 2), period = 52))
- m2.3 = arima(t1.log, order = c(0, 1, 2), seasonal = list(order = c(0, 1, 3), period = 52))
- m2.4 = arima(t1.log, order = c(0, 1, 2), seasonal = list(order = c(0, 1, 4), period = 52))

Among these models, m1.1, m2.1, m4.1, m5.1, m6.1, m2.2 are selected as preferable models using AIC and BIC.

	m1.1	m2.1	m4.1	m5.1	m6.1	m2.2
CV score1	20.15904	19.11847	17.49653	17.32303	18.79556	27.51723
CV score2	18.37231	12.89052	12.12896	12.06189	12.14805	13.03457

Among these preferable models, the two best models are m4.1 and m5.1 using CV score1, and CV score2.

Finally, we come up with three candidates: a1.1.1, m4.1, m5.1. Let's see the summary statistics of each models.

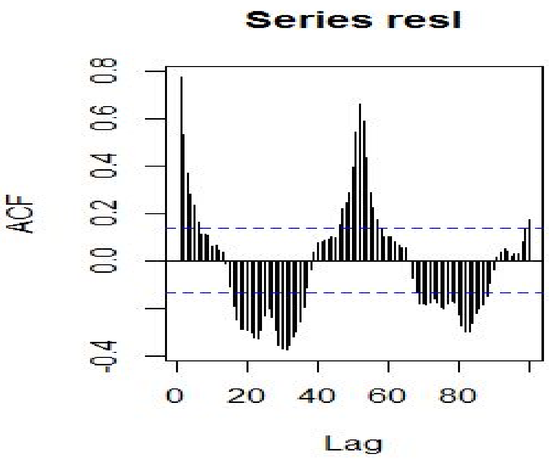
	AIC	BIC	CV score 1	CV score 2
m4.1	-400.4293	-382.1302	17.49653	12.12896
m5.1	-398.8003	-377.4513	17.32303	12.06189
a1.1.1	-400.6762	-388.4768	17.13972	11.94141

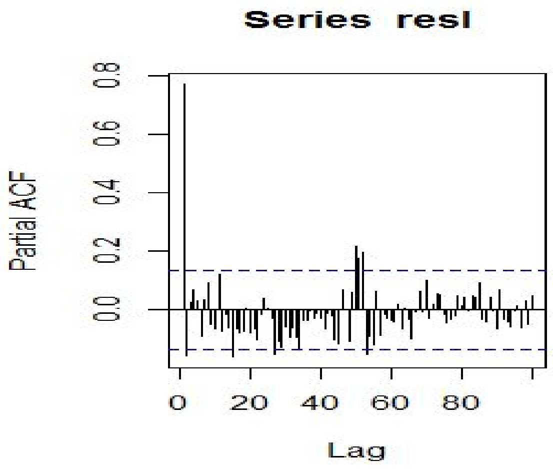
Every statistics of a1.1.1 prevails that of other models. Therefore, a1.1.1 would be the best model in differencing method

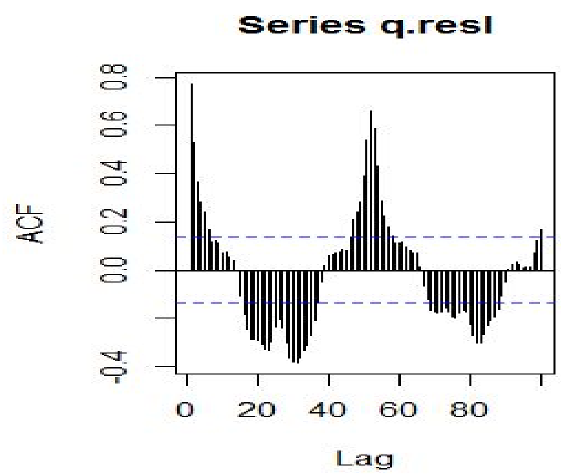
a1.1.1 = arima(t1.log, order=c(1,1,1), seasonal = list(order=c(0,1,1),period=52))

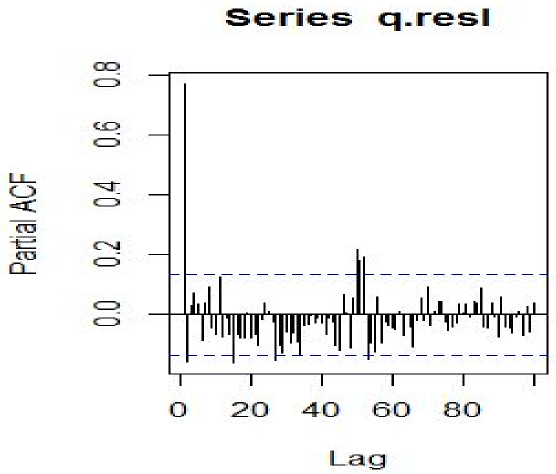
2. Parametric

To figure out the trend, we should know either quadratic trend or linear trend is appropriate for our given dataset. The first plot reveals that there is a seasonality after removing linear trend. The second plot also reveals that there is a seasonality after removing quadratic trend.





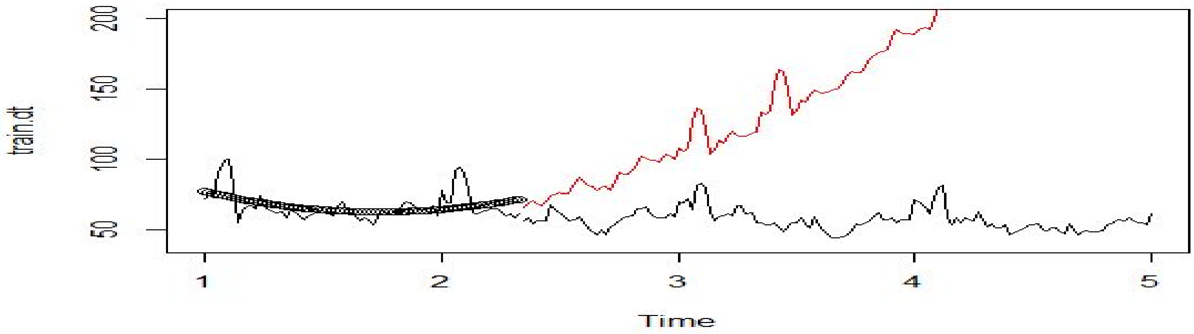




Finding appropriate trend model could be achieved by calculating the smallest MSE on the full dataset for each k=1,2,3,…,26. The smallest MSE occurred when k is equal to 26 for both methods. For the quadratic method, it turns out that the smallest MSE is 10.59176. For the linear method, it turns out that the smallest MSE is 10.50576. However, this result raises a question of overfitting.

Thus, using cross-validation, finding the appropriate k is essential. Using the CV score 1, subgrouping dataset into 3 groups, previously used in the difference method, the new k could be set.

For the quadratic method, depending on the curve of the training set, MSE varies a lot meaning if the training set is small, trend increases exponentially.



Therefore, we are looking the moderate k. Let's assumes that the training set is equal to 140 and test set is equal to 69 and applies to both methods compare the results. It turns out that when k=20, gives the minimum MSE of each method. The MSE of quadratic model is 26.19865 and MSE of linear model is 34.12356.

Let’s using Cross-validation 2: it starts at training set =108, and test set = 101, fitting the model and calculate the MSE. Then, increasing training set by one and decreasing test set by one and fitting the model calculate the MSE.

It calculating the 101 MSES on each k=1,2,3…26. Below tables show the frequency of minimum K when calculating 108 MSEs.

<Frequency table of best Ks in CV2 of linear model>

k	1	2	3	5	6	7	11	12	14	19	20	21	22	23	26
freq	21	4	12	6	2	3	1	22	4	2	19	2	1	1	1

<Frequency table of best Ks in CV2 of quadratic model>

k	1	2	3	4	5	6	7	12	14	18	20	22	23	26
freq	8	1	3	1	36	17	8	10	3	1	10	1	1	1

According to the frequency tables, best K for linear model might be 12 and best model for quadratic model might be 5. So let’s take at another look at the averaged MSE on CV2.

<CV score of CV2 on k=1,2,3…26 on linear model>

22.39999 23.67144 22.57821 24.38177 21.74659 22.39881 22.20601 22.89923 23.29585 23.46021 21.48131 20.80963 21.16337 20.63832 20.70451 20.77581 20.97849 20.86880 20.90090 20.63416 20.64150 20.78566 20.83956 20.97560 21.11327 21.08870

<CV score of CV2 on k=1,2,3…26 on quadratic model>

169.6583 167.4083 163.4297 160.0491 155.2374 155.6091 155.8905 158.1209 159.1425 159.9697 157.7506 156.9778 157.1033 156.3911 156.4466 156.5059 156.7894 156.6402 156.7059 156.3643 156.3870 156.5251 156.5772 156.7603 156.9411 156.9328

- The quadratic values are large compared to linear model because as mentioned previously, the MSE grows exponentially if the training dataset is small.

According to above figures, averaged MSE of linear model is minimum when k = 20 and frequencies of k=20 in above tables shows that k=20 could be a good choice. So I will use k=20 for linear model. For quadratic model, k=5 gives the lowest CV score and frequency table supports this idea. I will use k=5 for our quadratic model.

- If you think the frequency is more important, you could chose k=12 in the linear case, but I think CV value is more important. It's your judgement call.

For linear model with k=20, CV score is 20.63416. For quadratic model with k=5, CV score is 155.2374. Therefore, The best model of parametric method has the linear trend with k=20.

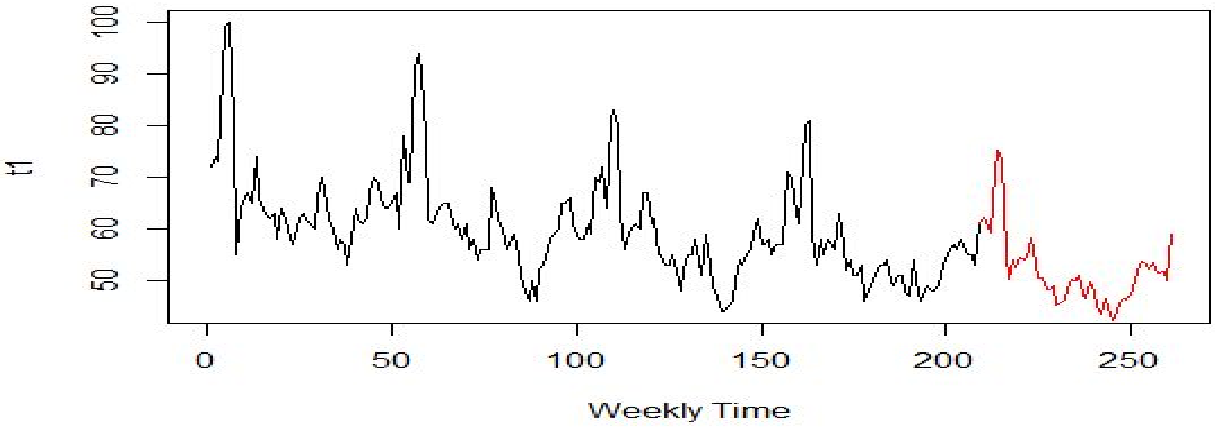
3. Comparing Differencing Method and Parametric Method using CV2

	CV score of CV1	CV score of CV2
Best differencing method	17.13972	11.94141
Best parametric method	87.83613	20.63416

The above table shows that differencing method performs better.

Prediction

Plots down here shows the best prediction using difference method and 52 predictions are listed below.



62.12105 61.43706 59.31496 70.85748 75.26499 73.51043 54.82079 50.13777 53.93798

52.50151 54.48660 54.00895 54.38317 58.18277 53.50330 50.44392 50.47574 48.13676
48.09002 48.85972 45.20529 45.70179 46.10955 48.51852 50.18688 49.92940 50.96000
47.41330 46.46947 49.60915 47.88644 44.84486 43.51229 46.38269 44.62489 42.22453
43.94140 45.66730 46.35802 46.23718 47.28851 49.98967 51.77283 53.69463 53.26153
52.18379 53.31671 51.42641 51.31554 51.61874 50.00245 59.05673