Inspire…Educate…Transform.

Statistics and Probability in Decision Modeling

Time Series Forecasting

Dr. Sridhar Pappu
Executive VP – Academics, INSOFE

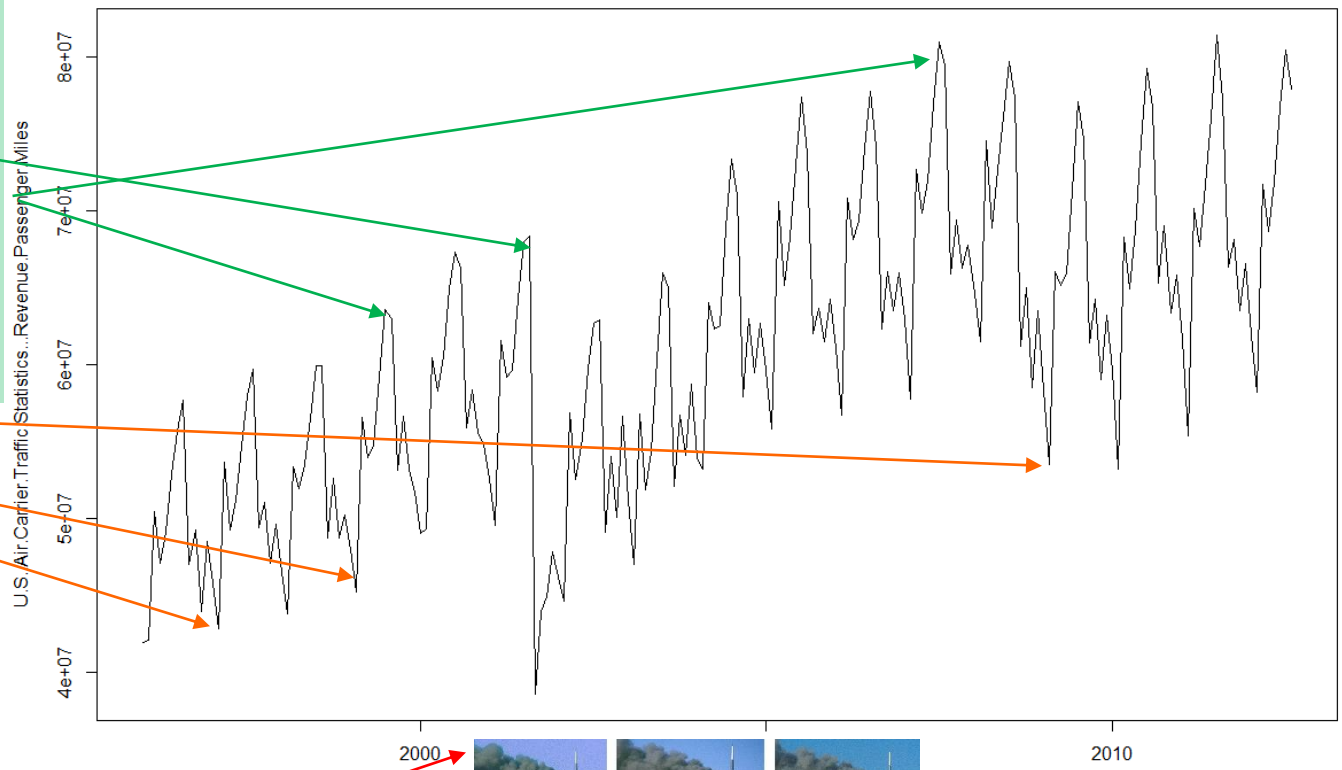
January 21, 2018

US Air Carrier Traffic – Revenue Passenger Miles ('000)

R code: milestimeseries <- ts(miles, frequency = 12, start = c(1996,1))

RPM

```
> milestimeseries <- ts(miles, frequency = 12, start = c(1996,1))
> milestimeseries
          Jan      Feb      Mar      Apr      May      Jun      Jul
1996 41972194 42054796 50443045 47112397 49118248 52880510 55664750
1997 45850623 42838949 53620994 49282817 51191842 54707221 57995025
1998 46514139 43769273 53361926 51968480 53515798 56460422 59939170
1999 47988560 45241211 56555731 53920855 54674958 59213000 63572248
2000 49045412 49306303 60443541 58286680 60533783 64903295 67346377
2001 52634354 49532578 61575055 59151645 59662416 64353323 67965298
2002 46224031 44615129 56897729 52542164 55116060 59745343 62664511
2003 51197175 47040806 56766580 51857453 54335598 60272900 65962215
2004 53979786 53179693 64035864 62340117 62530704 68866398 73335888
2005 59629608 55795165 70595861 65145552 68268899 72952959 77432998
2006 61035027 56729212 70799794 68120559 69352606 74099239 77798621
2007 63016013 57793832 72700241 69836156 71933109 76926452 80988340
2008 64667106 61504426 74575531 68906882 72725750 76162105 79707545
2009 58373786 53506580 66027341 65166300 65868254 71350227 77136799
2010 59651061 53240066 68307090 64953250 68850904 74474550 79304441
2011 61630362 55391206 70158268 67683558 71711448 76057910 81423215
2012 61940180 58243763 71696039 68669228 71887523 76760759 80499353
          Aug      Sep      Oct      Nov      Dec
1996 57723208 47035464 49263120 43937074 48539606
1997 59715433 49418190 51058879 47056048 49654209
```



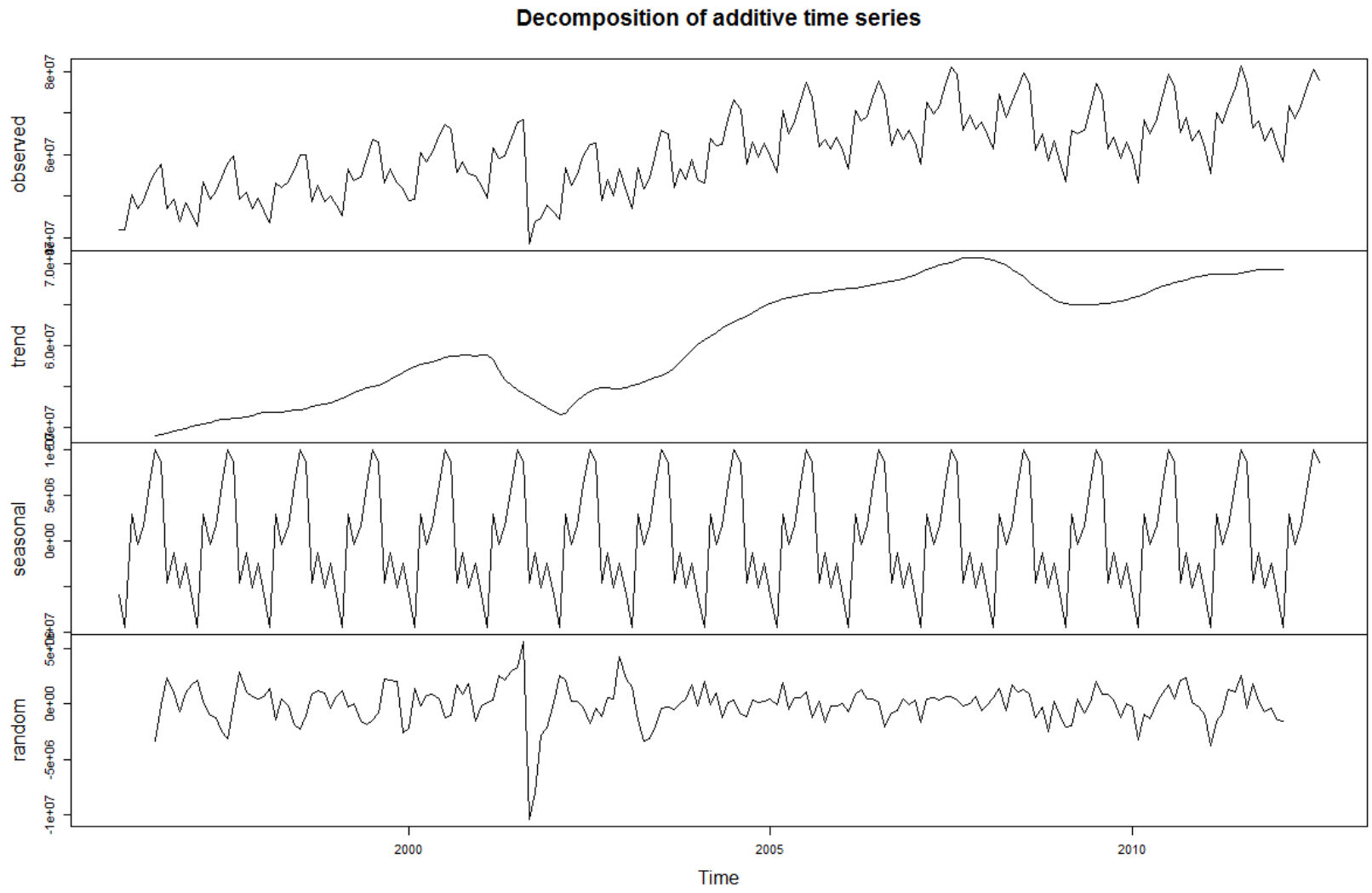
Data sources:
http://www.bts.gov/xml/air_traffic/src/index.xml
and https://datamarket.com/data/set/281x/us-air-carrier-traffic-statistics-revenue-passenger-miles
Last accessed: 31-Mar-2016

	Aug	Sep	Oct	Nov	Dec
1996	57723208	47035464	49263120	43937074	48539606
1997	59715433	49418190	51058879	47056048	49654209
1998	59927214	48751280	52578217	48734375	50208641
1999	63003663	53131972	56653901	53215500	51746821
2000	66256804	55900504	58373996	55590325	54822970
2001	68377080	38601868	43964788	44915764	47836501
2002	62944816	49096035	54019748	50106814	56656594
2003	64989766	52121480	56724551	54128776	58739845
2004	70961522	57881042	63021142	59453943	62680310

E 73026

Decomposing Time Series into the 3 Components – Revenue Passenger Miles (RPM)

R code: decompose(milestimeseries, type = "additive")



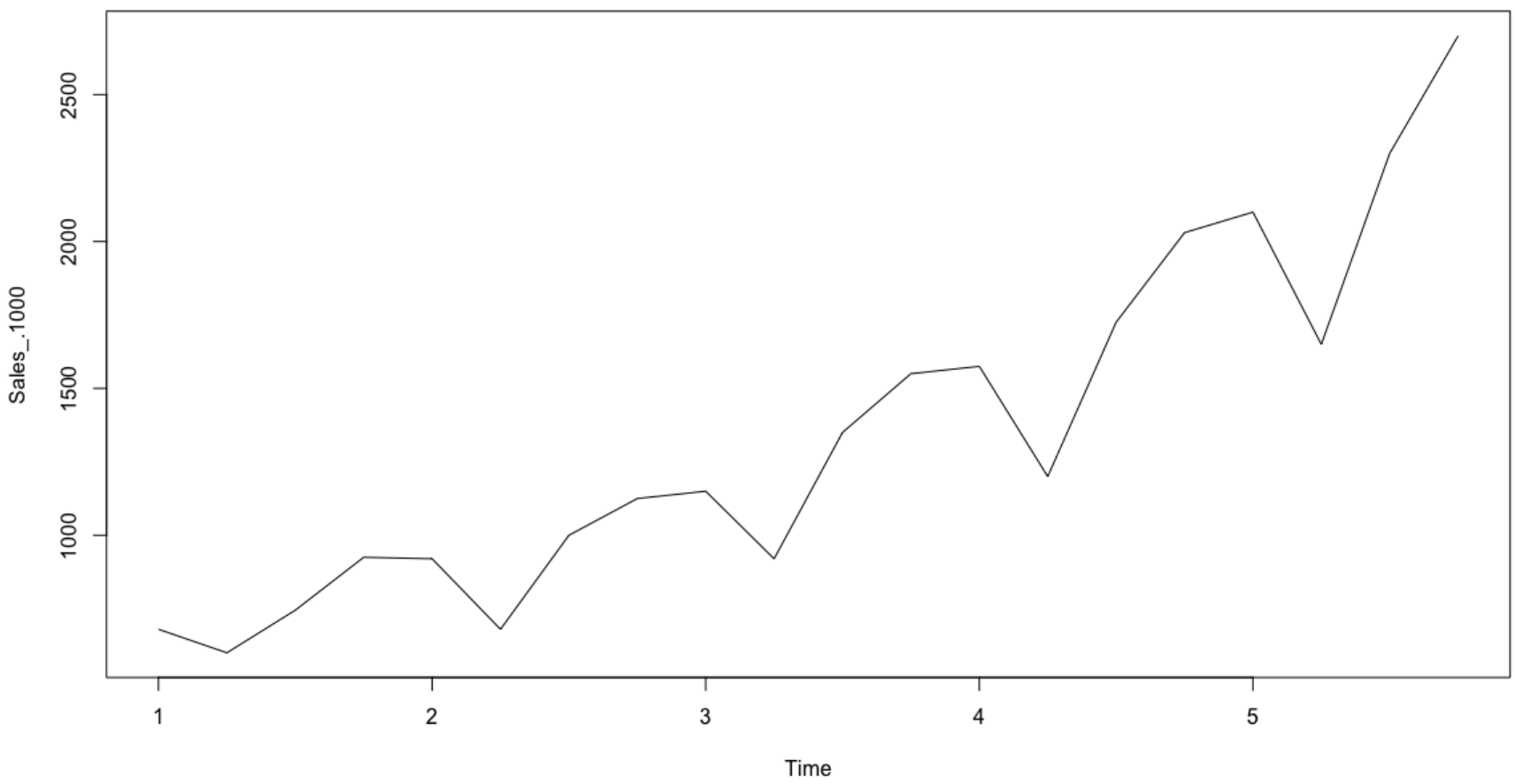
Time Series

CURVE FITTING / REGRESSION ON TIME METHODS

Regression on Time

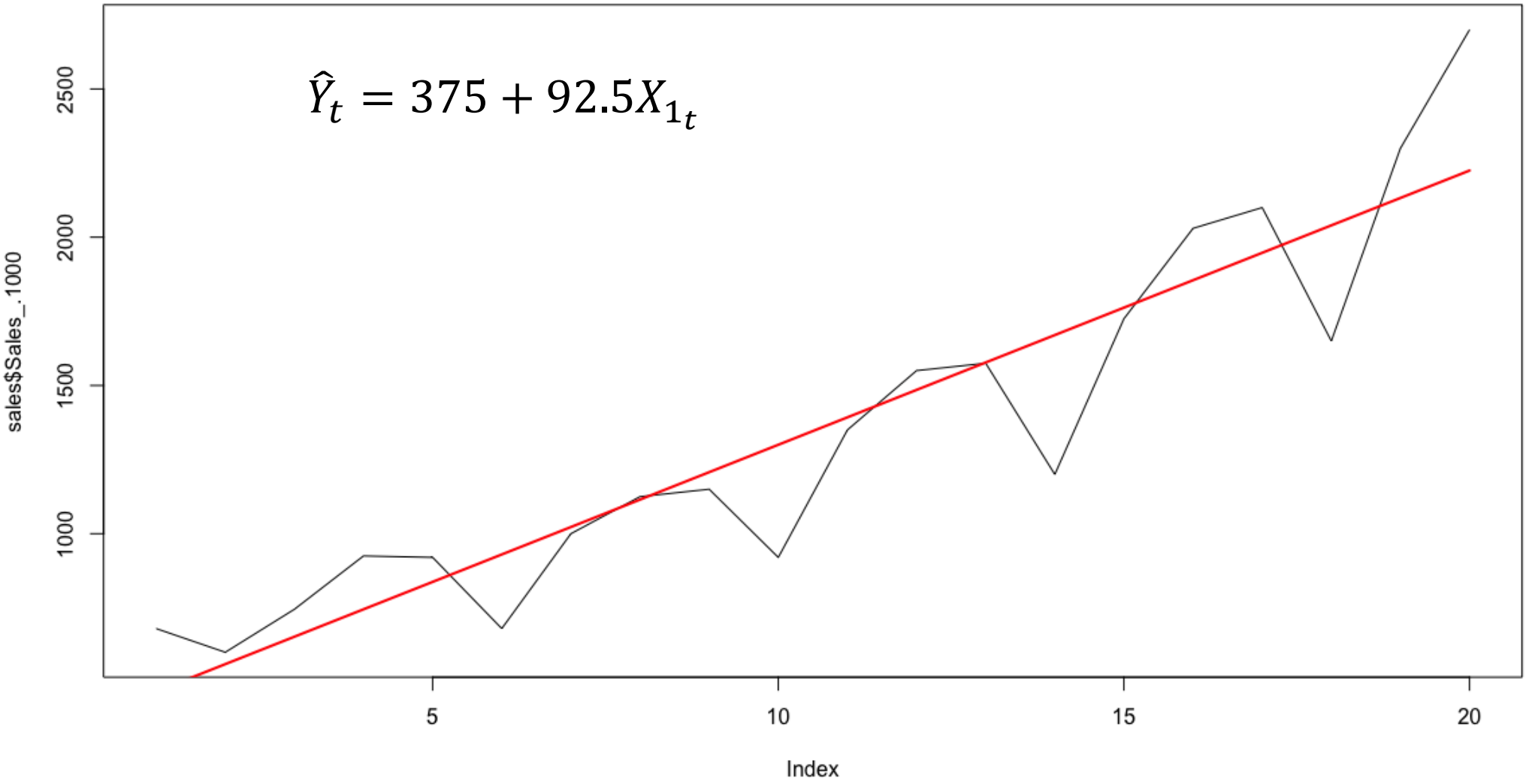
Use when trend is the most pronounced

Quarter	Sales_$1000
1	680
2	600
3	745
4	925
5	920
6	680
7	1000
8	1125
9	1150
10	920
11	1350
12	1550
13	1575
14	1200
15	1725
16	2030
17	2100
18	1650
19	2300
20	2700

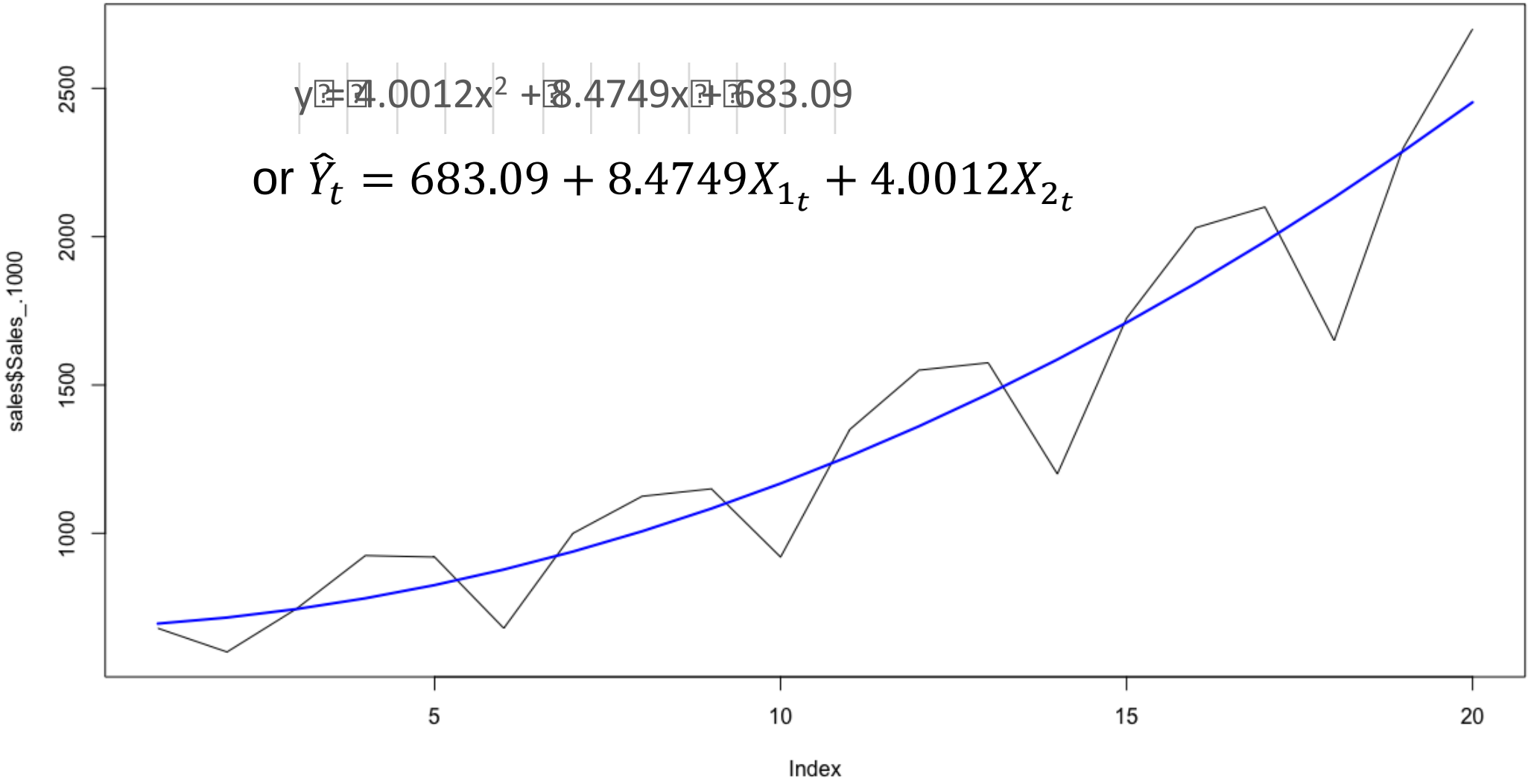


Regression Analysis

$$\hat{Y}_t = 375 + 92.5X_{1_t}$$



Quadratic Trend

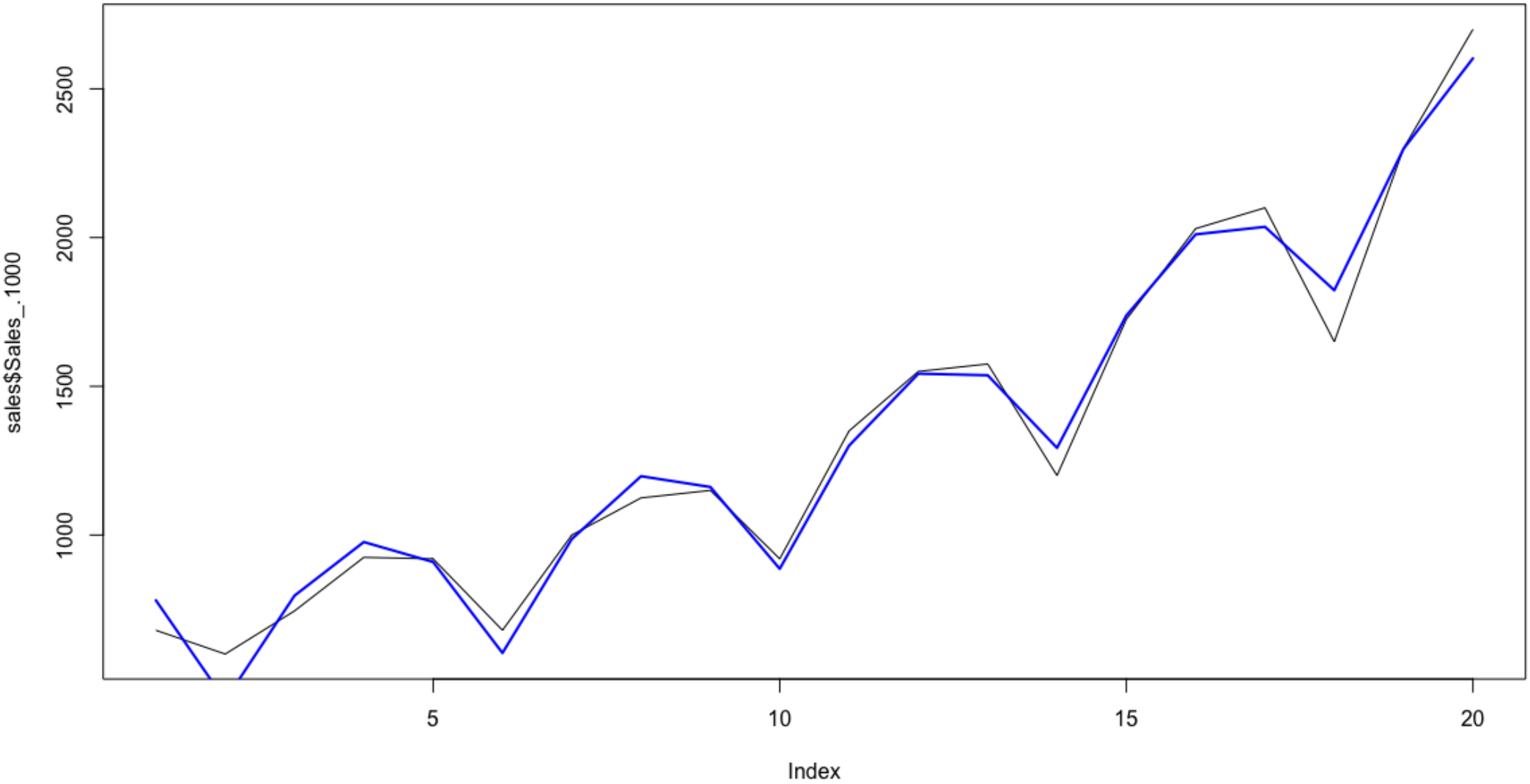


Incorporating Seasonality – Dummy Variable Approach

	Value of		
Quarter	X_{3_t}	X_{4_t}	X_{5_t}
1	1	0	0
2	0	1	0
3	0	0	1
4	0	0	0

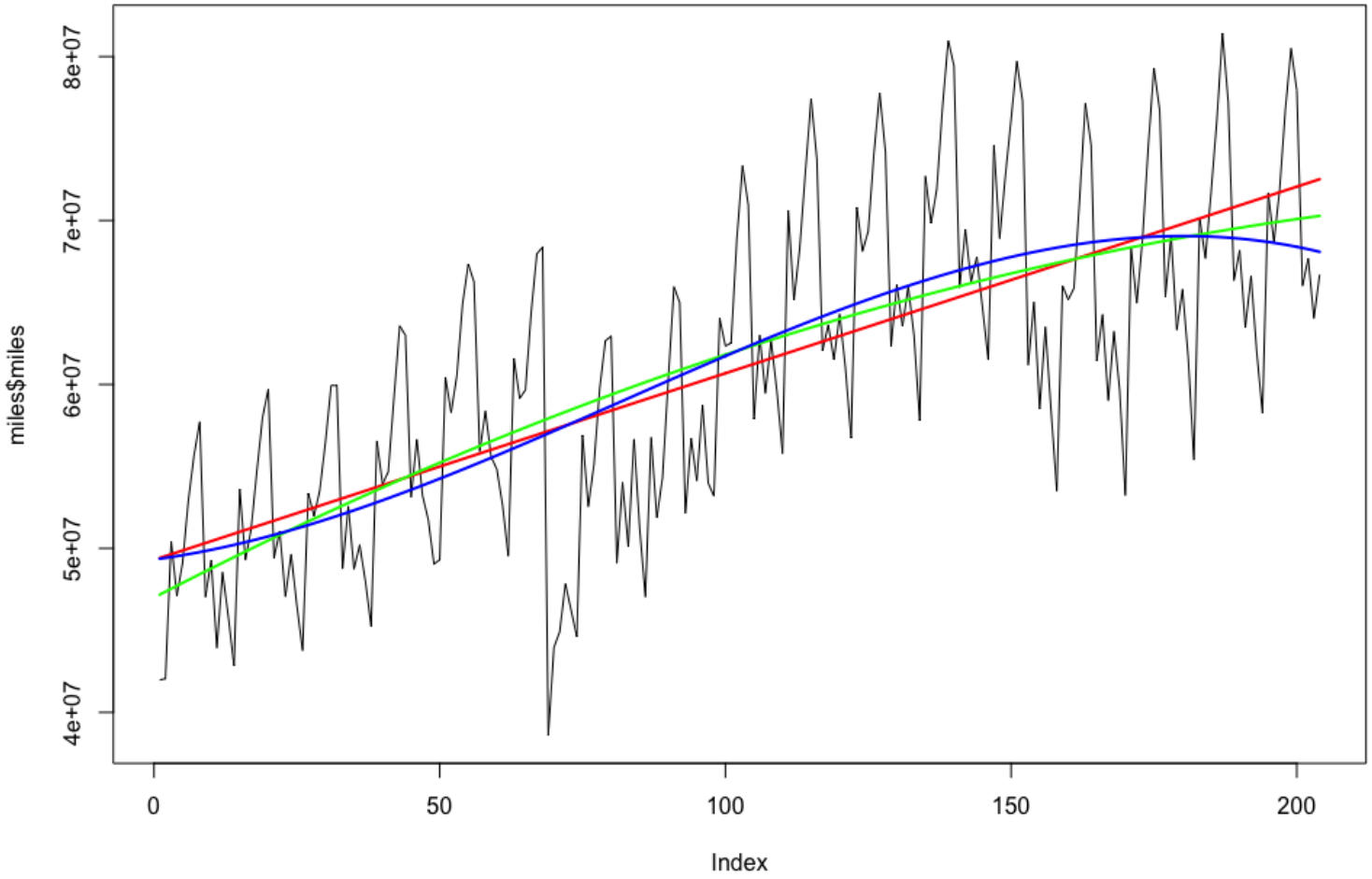
$$\hat{Y}_t = \beta_0 + \beta_1 X_{1_t} + \beta_2 X_{2_t} + \beta_3 X_{3_t} + \beta_4 X_{4_t} + \beta_5 X_{5_t} + \varepsilon_t$$

Seasonal Regression Models



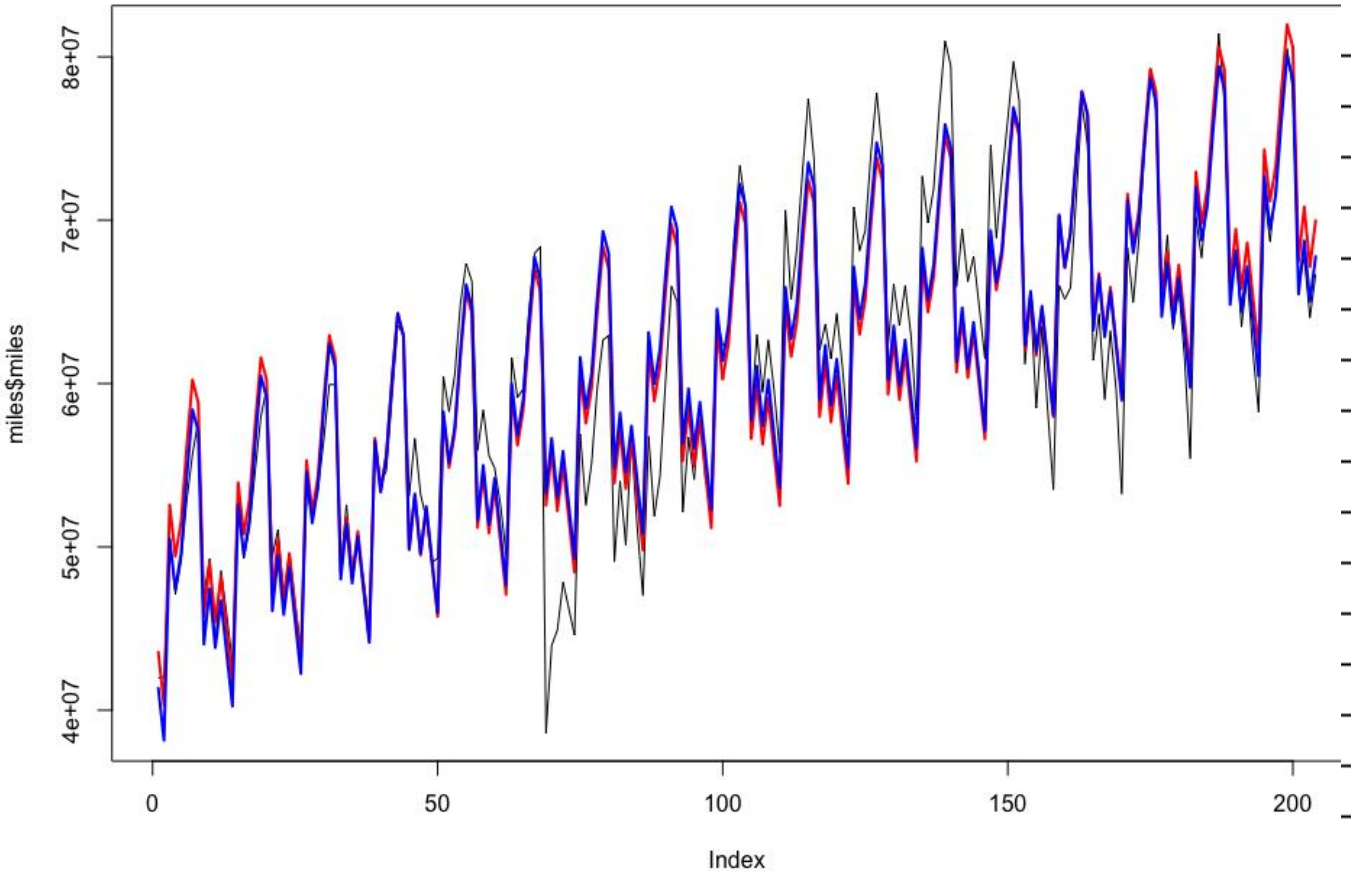
Seasonal Regression Models

Seasonal Regression Models - RPM



	miles	time	var3
1	41972194	1	
2	42054796	2	
3	50443045	3	
4	47112397	4	
5	49118248	5	
6	52880510	6	
7	55664750	7	
8	57723208	8	
9	47035464	9	
10	49263120	10	
11	43937074	11	
12	48539606	12	
13	45850623	13	
14	42838949	14	
15	53620994	15	

Seasonal Regression Models - RPM



	miles	time	seasonal
1	41972194	1	1
2	42054796	2	2
3	50443045	3	3
4	47112397	4	4
5	49118248	5	5
6	52880510	6	6
7	55664750	7	7
8	57723208	8	8
9	47035464	9	9
10	49263120	10	10
11	43937074	11	11
12	48539606	12	12
13	45850623	13	1
14	42838949	14	2
15	53620994	15	3

Case

Year	Quarter	Time variable (this is created)	Revenues (in $M)
2008	I	1	10.2
	II	2	12.4
	III	3	14.8
	IV	4	15
2009	I	5	11.2
	II	6	14.3
	III	7	18.4
	IV	8	18

```
Call:
lm(formula = y ~ x)

Residuals:
    Min      1Q  Median      3Q     Max
-3.5595 -0.9384  0.4405  1.3265  1.9286

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  10.0393     1.5531   6.464  0.00065 ***
x             0.9440     0.3076   3.069  0.02196 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.993 on 6 degrees of freedom
Multiple R-squared: 0.6109,     Adjusted R-squared: 0.5461
F-statistic: 9.422 on 1 and 6 DF,  p-value: 0.02196
```

What is the Regression equation?

$$y = 10.0393 + 0.9440x$$

Incorporating Seasonality – Another Approach

- Take the trend prediction and actual value.
- Depending on additive or multiplicative model compute the deviation and map it as seasonality effect for each prediction.
- Take averages of the seasonality value. Use this to make future predictions.

Seasonality: Multiplicative

Time	Observed values TSI* (assuming no impact of cyclicality)	Predicted values (per the regression) T*	SI* = TSI/T
1	10.2	10.983	0.929
2	12.4	11.927	1.040
3	14.8	12.871	1.150
4	15.0	13.815	1.086
5	11.2	14.759	0.759
6	14.3	15.703	0.911
7	18.4	16.647	1.105
8	18.0	17.591	1.023

* T: Trend; S: Seasonal; I: Irregular

Quarterly Seasonality

Time	Average seasonality factor
Q1	$0.844 \left(= \frac{0.929+0.759}{2}\right)$
Q2	0.975
Q3	1.127
Q4	1.054

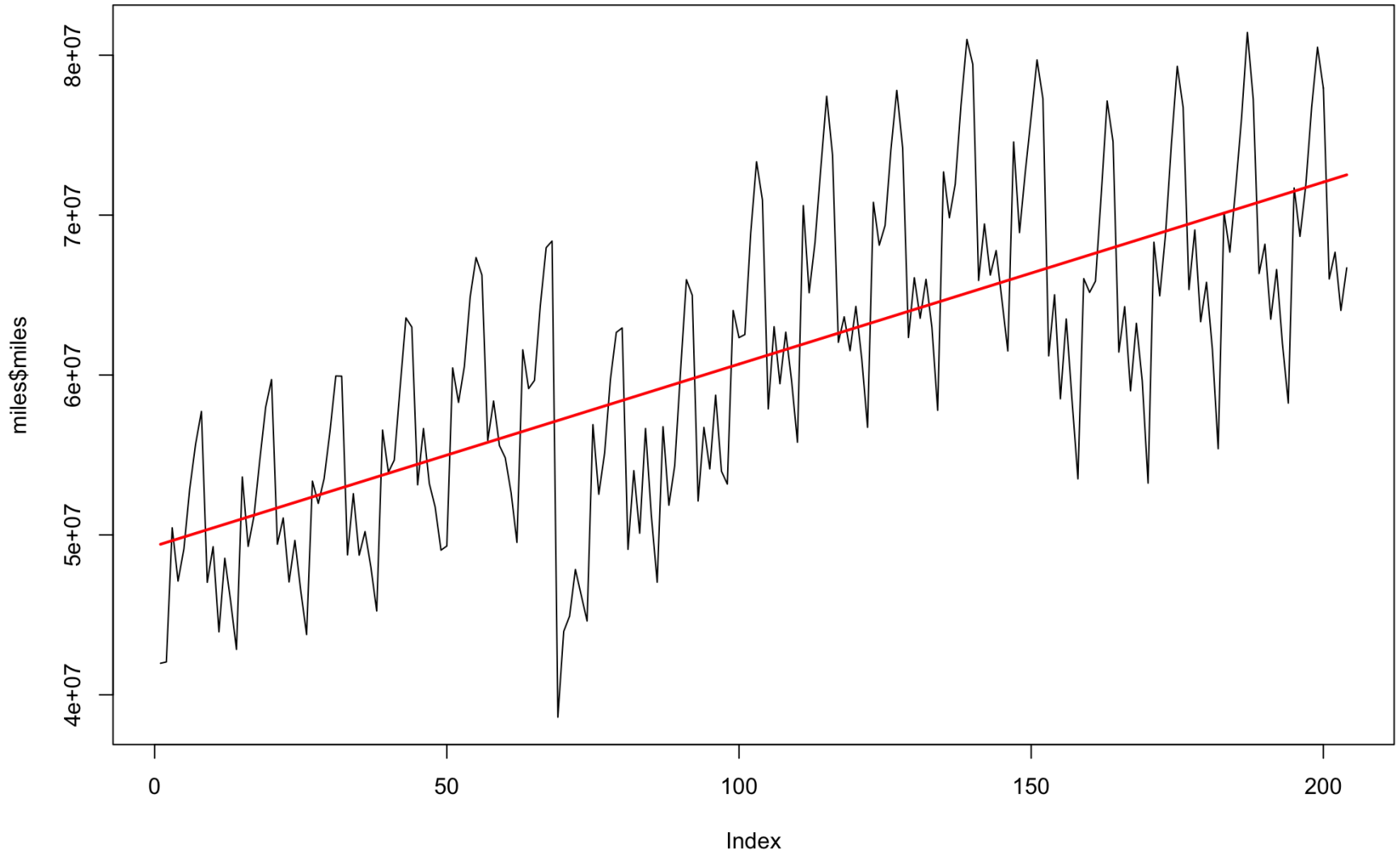
Time	Observed values TSI* (assuming no impact of cyclicality)	Predicted values (per the regression) T*	SI* = TSI/T
1	10.2	10.983	0.929
2	12.4	11.927	1.040
3	14.8	12.871	1.150
4	15.0	13.815	1.086
5	11.2	14.759	0.759
6	14.3	15.703	0.911
7	18.4	16.647	1.105
8	18.0	17.591	1.023

Computations

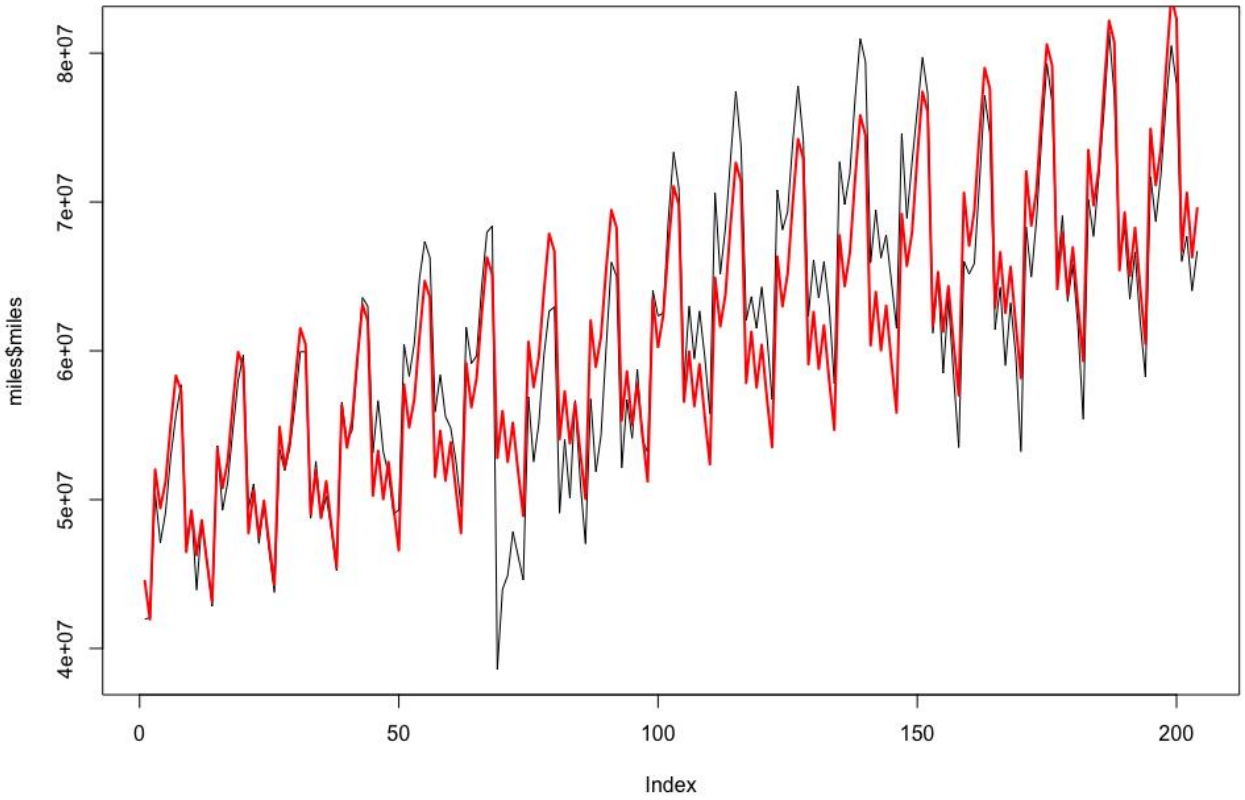
- Trend Y_9 =10.039 + 0.944(9) = 18.535

- Corrected for seasonality and randomness: 18.535 * 0.844 = 15.643

Regression on Time - RPM

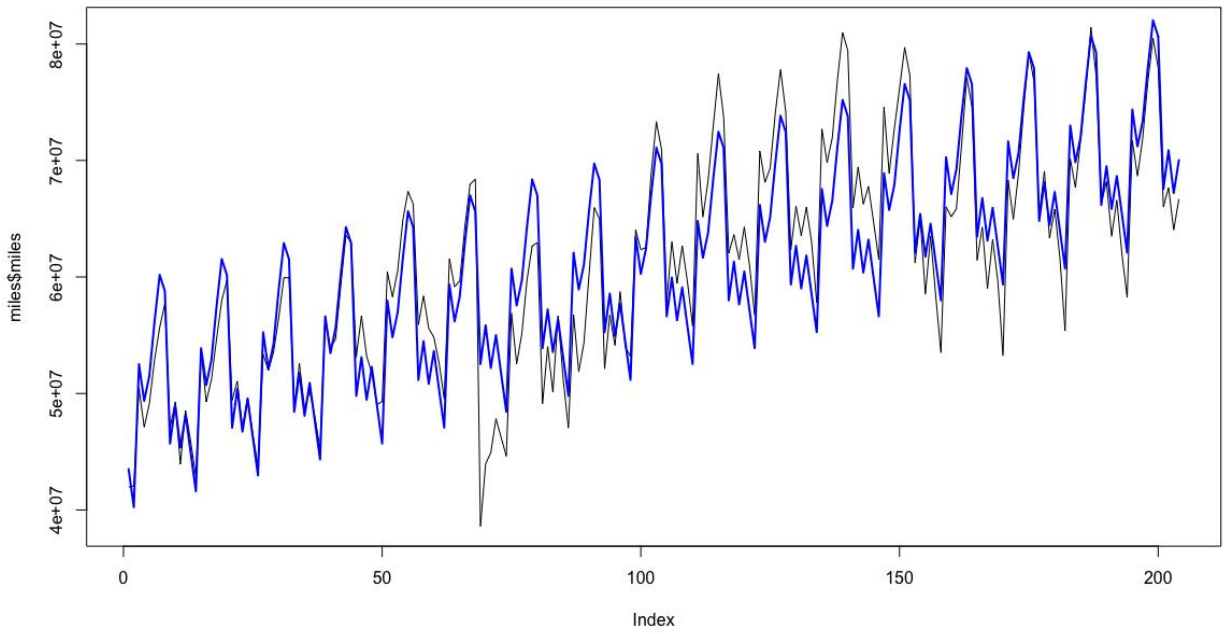


Seasonality: Multiplicative



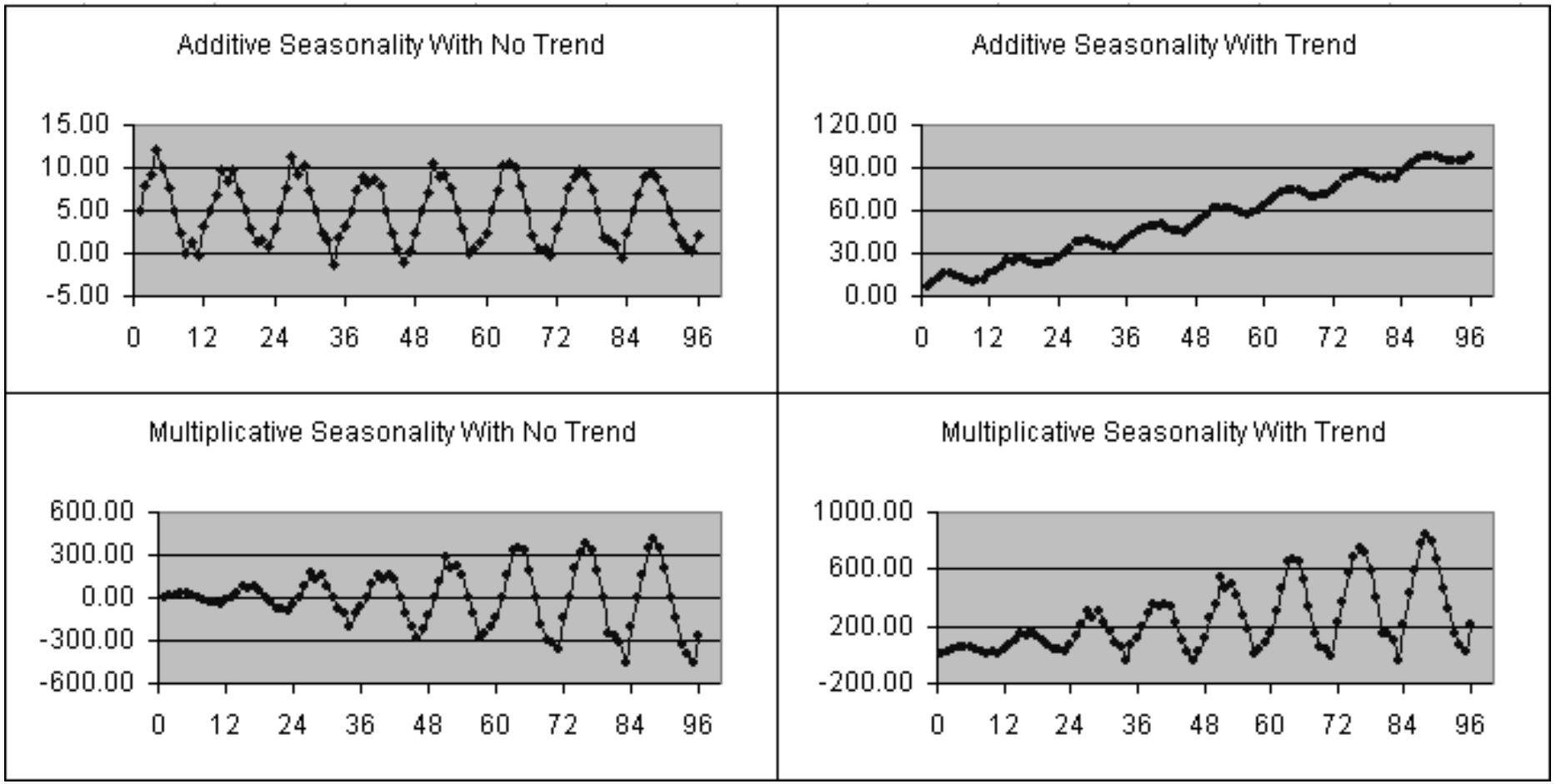
	miles	time	seasonal	mae
1	41972194	1	1	0.849386
2	42054796	2	2	0.8491019
3	50443045	3	3	1.016129
4	47112397	4	4	0.946865
5	49118248	5	5	0.9849257
6	52880510	6	6	1.057953
7	55664750	7	7	1.111125
8	57723208	8	8	1.149602
9	47035464	9	9	0.9346292
10	49263120	10	10	0.9766855
11	43937074	11	11	0.8691307
12	48539606	12	12	0.9580177
13	45850623	13	1	0.9029174
14	42838949	14	2	0.8417232
15	53620994	15	3	1.051224

Seasonality: Additive



	miles	time	seasonal	mae
1	41972194	1	1	-7442550
2	42054796	2	2	-7473763
3	50443045	3	3	800670.5
4	47112397	4	4	-2643793
5	49118248	5	5	-751757.1
6	52880510	6	6	2896690
7	55664750	7	7	5567114
8	57723208	8	8	7511757
9	47035464	9	9	-3289802
10	49263120	10	10	-1175962
11	43937074	11	11	-6615823
12	48539606	12	12	-2127106
13	45850623	13	1	-4929905
14	42838949	14	2	-8055394
15	53620994	15	3	2612836

Additive or Multiplicative



Issues with Regressing on Time

If there is no trend or if seasonality and fluctuations are more important than trend, then the coefficients behave weirdly

Advanced Time Series Methods

HOLT-WINTERS

Holt-Winters – Important Concepts

SMOOTHING

Simple Moving Averages

	Number of products sold	SMA (K=2)	SMA (K=3)
2	15		
3	20	17.5	
4	16	18	17
5	13	14.5	16.333333
6	18	15.5	15.666667
7	14	16	15
8	15	14.5	15.666667
9	17	16	15.333333
10	20	18.5	17.333333
11	20	20	19

Weighted Moving Averages

	Number of products sold	WMA (K=2)	WMA (K=3)
1			
2	15		
3	20	=(A3*2+A2*1)/3	
4	16	=(A4*2+A3*1)/3	=(A4*3+A3*2+A2*1)/6
5	13	=(A5*2+A4*1)/3	=(A5*3+A4*2+A3*1)/6
6	18	=(A6*2+A5*1)/3	=(A6*3+A5*2+A4*1)/6
7	14	=(A7*2+A6*1)/3	=(A7*3+A6*2+A5*1)/6
8	15	=(A8*2+A7*1)/3	=(A8*3+A7*2+A6*1)/6
9	17	=(A9*2+A8*1)/3	=(A9*3+A8*2+A7*1)/6
10	20	=(A10*2+A9*1)/3	=(A10*3+A9*2+A8*1)/6
11	20	=(A11*2+A10*1)/3	=(A11*3+A10*2+A9*1)/6
12	18	=(A12*2+A11*1)/3	=(A12*3+A11*2+A10*1)/6
13	20	=(A13*2+A12*1)/3	=(A13*3+A12*2+A11*1)/6

1	Number of products sold	WMA (K=2)	WMA (K=3)
2	15		
3	20	18.3333333	
4	16	17.3333333	17.1666667
5	13	14	15.1666667
6	18	16.3333333	16
7	14	15.3333333	15.1666667
8	15	14.6666667	15.1666667
9	17	16.3333333	15.8333333
10	20	19	18.1666667
11	20	20	19.5
12	18	18.6666667	19
13	20	19.3333333	19.3333333

Exponential ~~Weighted Moving Averages~~ or Exponential Smoothing

Averaging over long periods dampens fluctuations, removing not only the noise but also trend and seasonality.

Moving averages over short recent periods maintain trend and seasonality but determining an optimum number for periods is tricky, even when using metrics like MAE. If averaged over too few periods, irregularities continue to remain and if averaged over long periods, dampening again becomes a problem.

Exponential smoothing **retains all older periods** while giving a greater weight to more recent periods (hence not a MOVING average).

Caution: It doesn't make any one method superior for all situations.

Stationary Model: Case 3 – Exponential ~~Weighted Moving Averages~~ or Exponential Smoothing

$$\hat{Y}_{t+1} = \hat{Y}_t + \alpha\left(Y_t - \hat{Y}_t\right)$$

Above equation indicates that the predicted value for time period t+1 $(\hat{Y}_{t+1})$ is equal to the predicted value for the previous period $(\hat{Y}_t)$ plus an adjustment for the error made in predicting the previous period's value $(\alpha\left(Y_t - \hat{Y}_t\right))$.

The parameter α can assume any value between 0 and 1 ($0 \leq \alpha \leq 1$).

Exponential Smoothing in Other Ways

$\hat{Y}_{t+1} = \hat{Y}_t + \alpha(Y_t - \hat{Y}_t)$ can be rewritten variously as

$$= \alpha Y_t + (1-\alpha)\hat{Y}_t$$

$$= Y_t - (1-\alpha)\left(Y_t - \hat{Y}_t\right)$$

$$= \alpha Y_t + \alpha(1-\alpha)Y_{t-1} + \alpha(1-\alpha)^2 Y_{t-2} + \cdots + \alpha(1-\alpha)^n Y_{t-n} + \cdots$$

Various Ways of Understanding Exponential Smoothing

- Forecast
  - Interpolation between previous *forecast* and previous *observation*
  - Previous *forecast* plus fraction of previous error
  - Previous *observation* minus fraction 1- of previous error
  - *Exponentially weighted (i.e., discounted) moving average*

$$= \alpha Y_t + (1-\alpha)\hat{Y}_t$$

$$= \hat{Y}_t + \alpha(Y_t - \hat{Y}_t)$$

$$= Y_t - (1-\alpha)(Y_t - \hat{Y}_t)$$

$$= \alpha Y_t + \alpha(1-\alpha)Y_{t-1} + \alpha(1-\alpha)^2 Y_{t-2} + \cdots + \alpha(1-\alpha)^n Y_{t-n} + \cdots$$

Holt-Winters

MODEL BUILDING

Holt-Winters Method

- 3 components – Trend, Seasonality and Level.

Additive Seasonality

$$\hat{y}_t = l_{t-1} + b_{t-1} + s_{t-m}$$

$$\hat{y}_{t+h|t} = l_t + hb_t + s_{t+h-m}$$

Multiplicative Seasonality

$$\hat{y}_t = (l_{t-1} + b_{t-1})s_{t-m}$$

$$\hat{y}_{t+h|t} = (l_t + hb_t)s_{t+h-m}$$

Holt-Winters Method

- 3 weights – smoothing parameters – are used to update components at each period.

The 3 smoothing equations are:

Additive

$$l_t = \alpha(y_t - s_{t-m}) + (1-\alpha)(l_{t-1} + b_{t-1})$$

$$b_t = \beta(l_t - l_{t-1}) + (1-\beta)b_{t-1}$$

$$s_t = \gamma(y_t - (l_{t-1} + b_{t-1})) + (1-\gamma)s_{t-m}$$

Multiplicative

$$l_t = \alpha\frac{y_t}{s_{t-m}} + (1-\alpha)(l_{t-1} + b_{t-1})$$

$$b_t = \beta(l_t - l_{t-1}) + (1-\beta)b_{t-1}$$

$$s_t = \gamma\frac{y_t}{l_{t-1} + b_{t-1}} + (1-\gamma)s_{t-m}$$

Recall Exponential Smoothing form: $\hat{Y}_t = \alpha Y_{t-1} + (1-\alpha)\hat{Y}_{t-1}$

Initial values for level, trend and seasonal components are obtained using the Decomposition method *(implemented in R's HoltWinters() function and described next)*.

Holt-Winters Implementation in R's HoltWinters() function

STEP 1: Fit a *2 X m* moving average smoother to the first 2 years (periods) of data. Moving averages remove some randomness and leave a smooth trend component at that point.

Note: For shorter time series, a simple linear model with time trend is used in place of the MA smoothing. First order Fourier approximation is used for the seasonal component. The rest of the steps remain same as follows.

U.S. Air Carrier Traffic Statistics	12-MA	2X12-MA Trend
41972194		
42054796		
50443045		
47112397		
49118248		
52880510	48812034	48973636
55664750	49135237	49167910
57723208	49200583	49332997
47035464	49465412	49555846
49263120	49646280	49732680
43937074	49819080	49895193
48539606	49971306	50068400
45850623	50165495	50248505
42838949	50331514	50430794
53620994	50530075	50604898
49282817	50679721	

Holt-Winters Implementation in R's HoltWinters() function

STEP 2: Get de-trended data by subtracting (for additive Holt-Winters) or dividing (for multiplicative Holt-Winters) the smooth trend from the original data. The **initial seasonal values** are then obtained by averaging the de-trended values from the same season.

U.S. Air Carrier Traffic Statistics	12-MA	2X12-MA Trend	Detrended Series	Seasonal Component
41972194				
42054796				
50443045				
47112397				
49118248				
52880510	48812034			
55664750	49135237	48973636	6691114	10086225
57723208	49200583	49167910	8555298	8613219
47035464	49465412	49332997	-2297533	-4652131
49263120	49646280	49555846	-292726	-1338310
43937074	49819080	49732680	-5795606	-5113908
48539606	49971306	49895193	-1355587	-2406656
45850623	50165495	50068400	-4217777	-5951588
42838949	50331514	50248505	-7409556	-9546987
53620994	50530075	50430794	3190200	2875281
49282817	50679721	50604898	-1322081	-364949

Average of all July de-trended values

Holt-Winters Implementation in R's HoltWinters() function

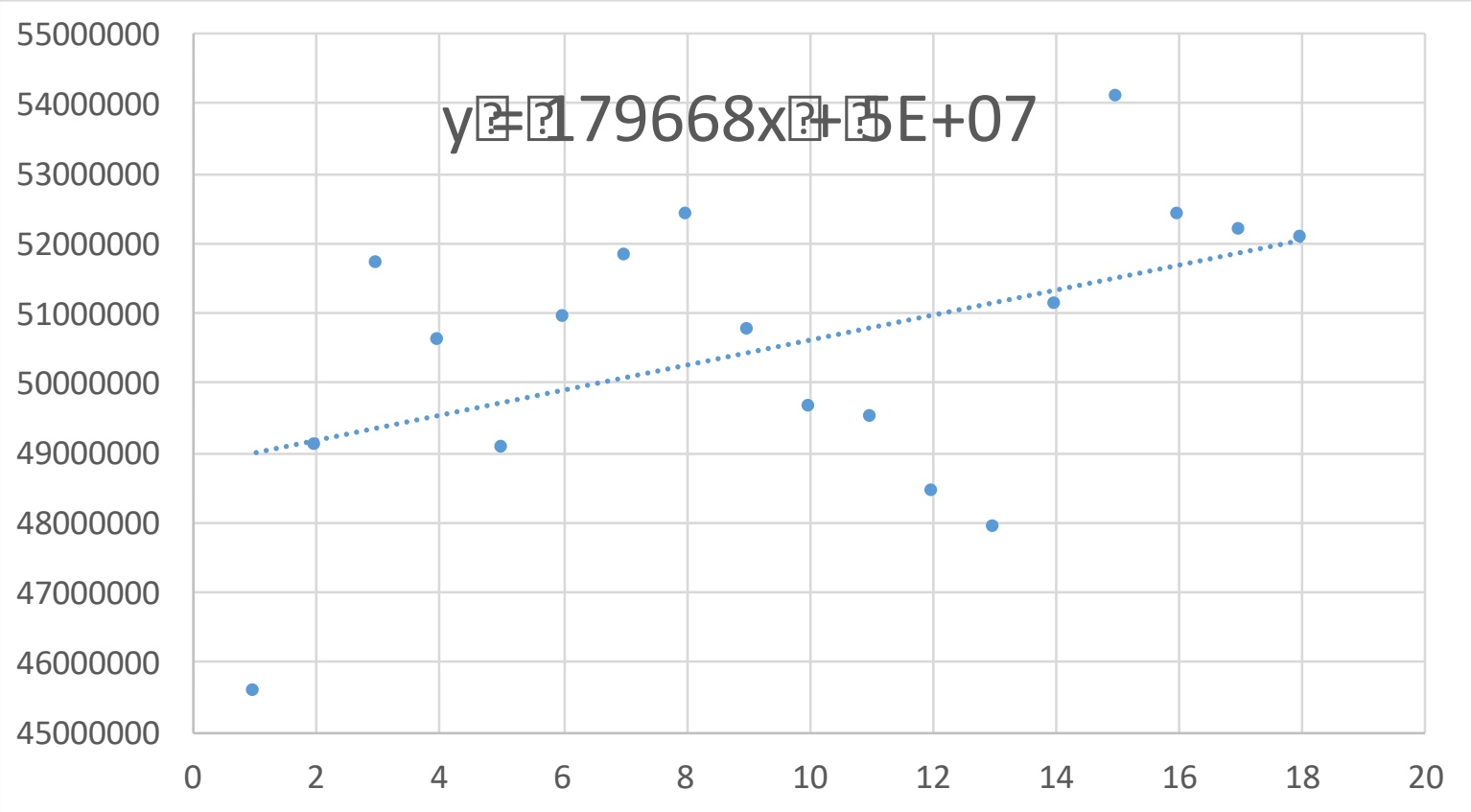
STEP 3: Get seasonally adjusted data by subtracting (for additive Holt-Winters) or dividing (for multiplicative Holt-Winters) the seasonal component from the original data.

U.S. Air Carrier Traffic Statistics	12-MA	2X12-MA Trend	Detrended Series	Seasonal Component	Seasonally Adjusted Data
41972194					
42054796					
50443045					
47112397					
49118248					
52880510	48812034				
55664750	49135237	48973636	6691114	10086225	45578525
57723208	49200583	49167910	8555298	8613219	49109989
47035464	49465412	49332997	-2297533	-4652131	51687595
49263120	49646280	49555846	-292726	-1338310	50601430
43937074	49819080	49732680	-5795606	-5113908	49050982
48539606	49971306	49895193	-1355587	-2406656	50946262
45850623	50165495	50068400	-4217777	-5951588	51802211
42838949	50331514	50248505	-7409556	-9546987	52385936
53620994	50530075	50430794	3190200	2875281	50745713
49282817	50679721	50604898	-1322081	-364949	49647766

Holt-Winters Implementation in R's HoltWinters() function

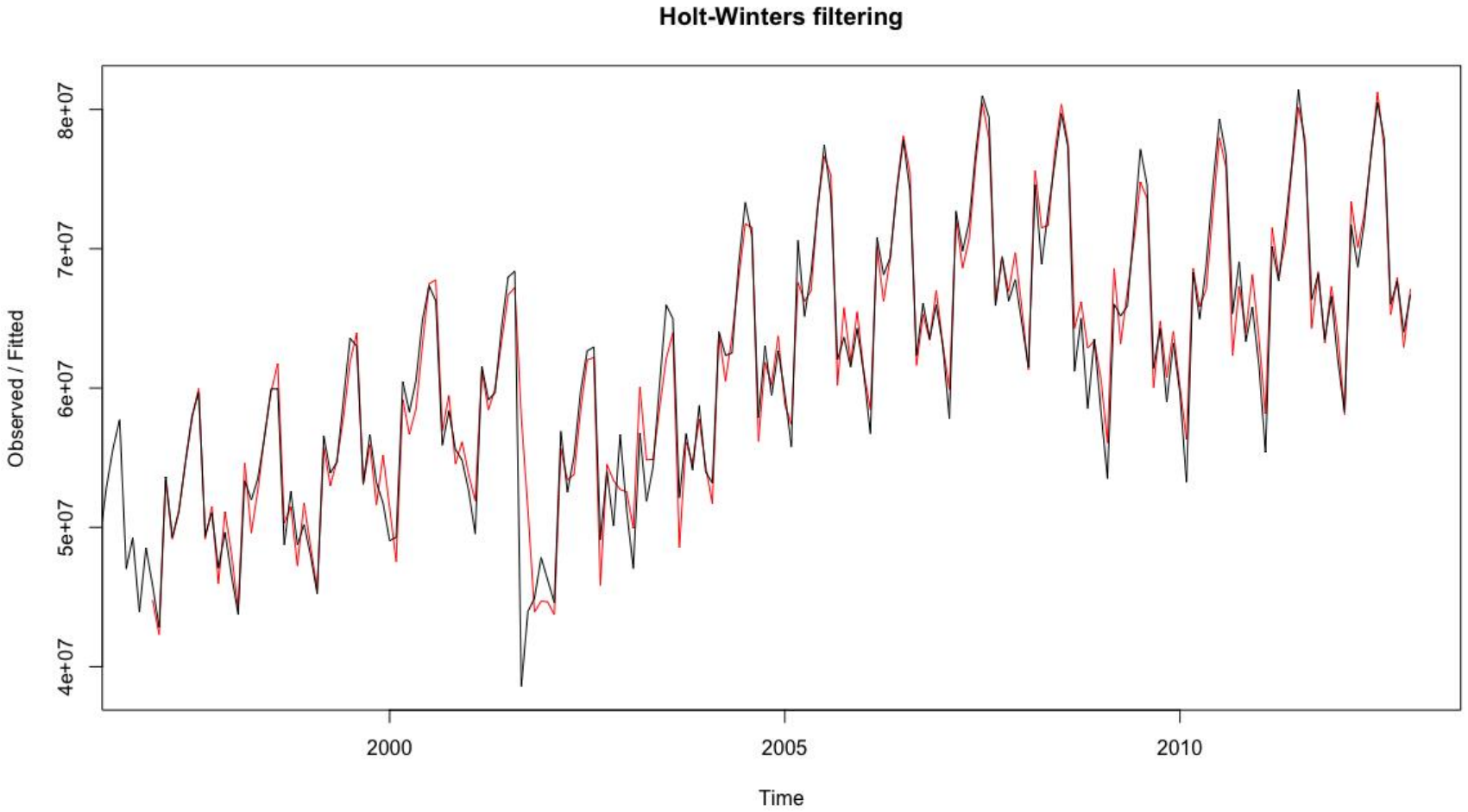
STEP 4: Fit a linear trend to the seasonally adjusted data to get **initial level** l_0 (the intercept) and the **initial slope** b_0 values.

Seasonally adjusted data
45578525
49109989
51687595
50601430
49050982
50946262
51802211
52385936
50745713
49647766



Holt-Winters Method: *RPM*

R code: HoltWinters(milestimeseries)



milesforecast$SSE

[1] 8.5203e+14

Holt-Winters Method: *RPM*

```
> milesforecast
Holt-Winters exponential smoothing with trend and additive seasonal component.

Call:
HoltWinters(x = milestimeseries)

Smoothing parameters:
 alpha: 0.5024519
 beta : 0
 gamma: 0.6698853

Coefficients:
           [,1]
a   71153143.7
b     181508.7
s1   -8052713.5
s2  -12314754.8
s3    2106026.2
s4    -447843.5
s5    2758176.7
s6    7232862.9
s7   11309888.8
s8    8022549.6
s9   -4490660.8
s10  -2715650.2
s11  -7315682.9
s12  -4384446.7
> milesforecast$fitted
              xhat    level    trend       season
Jan 1997 44780388 48804078 181508.7  -4205198.60
Feb 1997 42307860 49523328 181508.7  -7396976.76
```

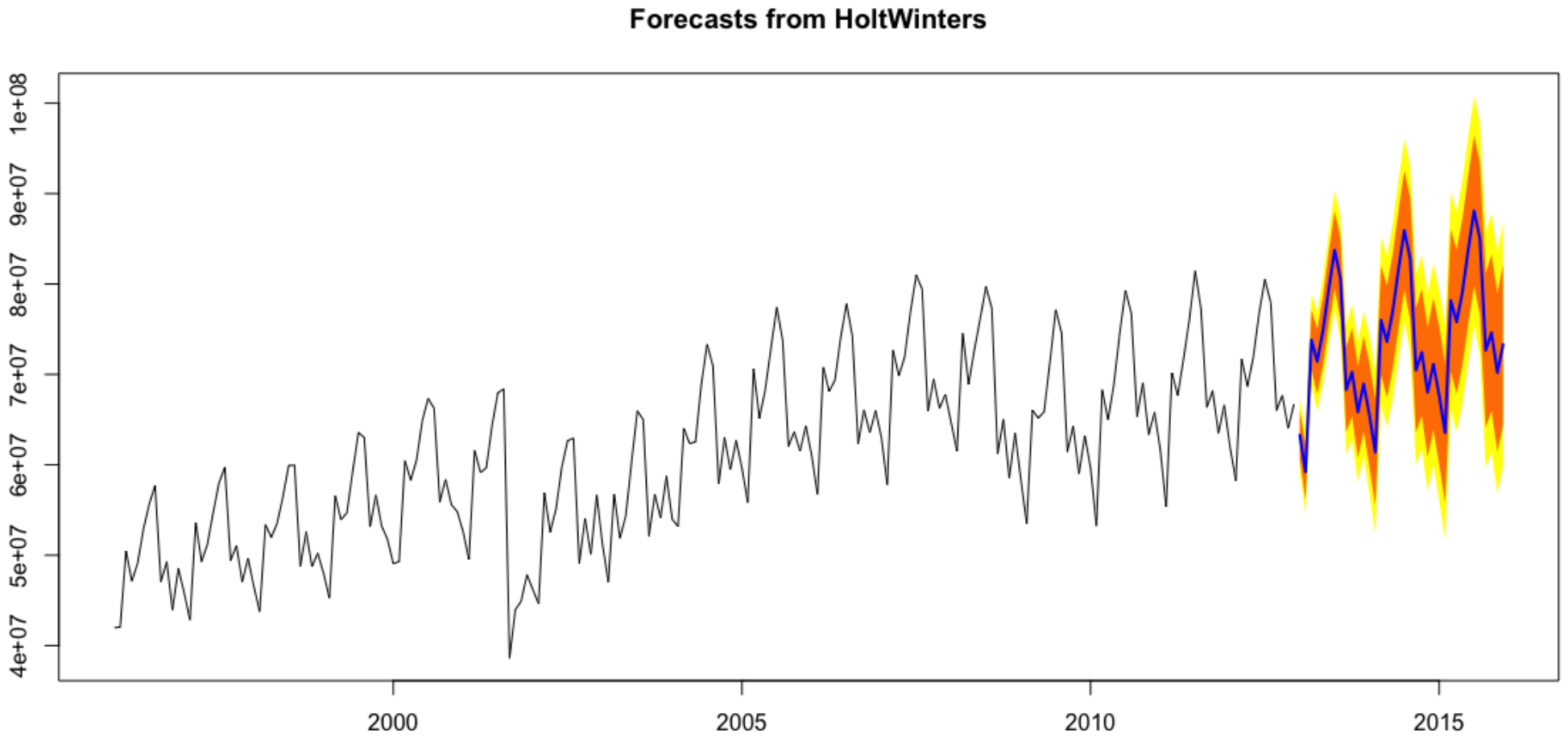
$$l_t = \alpha(y_t - s_{t-m}) + (1-\alpha)(l_{t-1} + b_{t-1})$$
$$b_t = \beta(l_t - l_{t-1}) + (1-\beta)b_{t-1}$$
$$s_t = \gamma(y_t - (l_{t-1} + b_{t-1})) + (1-\gamma)s_{t-m}$$

$$\hat{y}_{t+h|t} = l_t + hb_t + s_{t+h-m}$$

R refers to *l* as *a*

Holt-Winters Method: Forecasting

R code: forecast.HoltWinters(milesforecast, h=36)



Advanced Time Series Methods

ARIMA

ARIMA – Important Concepts

ACF-PACF AND STATIONARITY

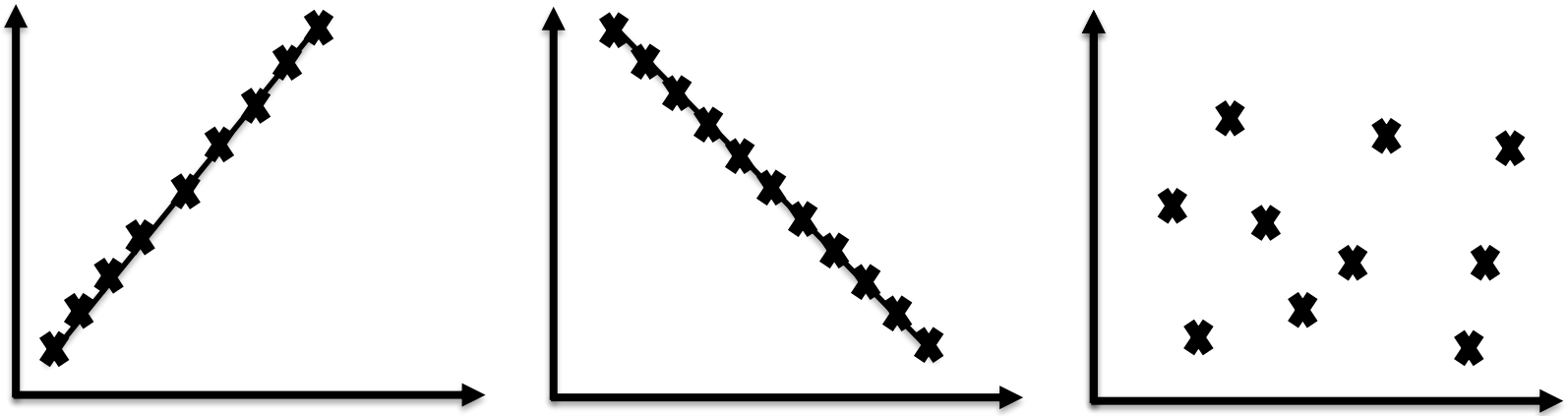
Time Series Descriptive Statistics

- In descriptive statistics covered earlier (central tendencies, measures of variability, distributions, correlations, etc.), the order of observations in the data was of no consequence.

- In time series descriptive statistics, order of observations is of primary importance and so **autocorrelations** play a vital role in identifying the models and their characteristics.

Correlation Coefficient

Correlation coefficient, *r*, is a number between -1 and 1 and tells us how well a regression line fits the data.



r = 1 r = -1 r =0

It gives the strength and direction of the relationship between two variables.

Autocorrelation (ACF) and Partial ACF (PACF)

- ACF: n^{th} lag of ACF is the correlation between a day and n days before that.

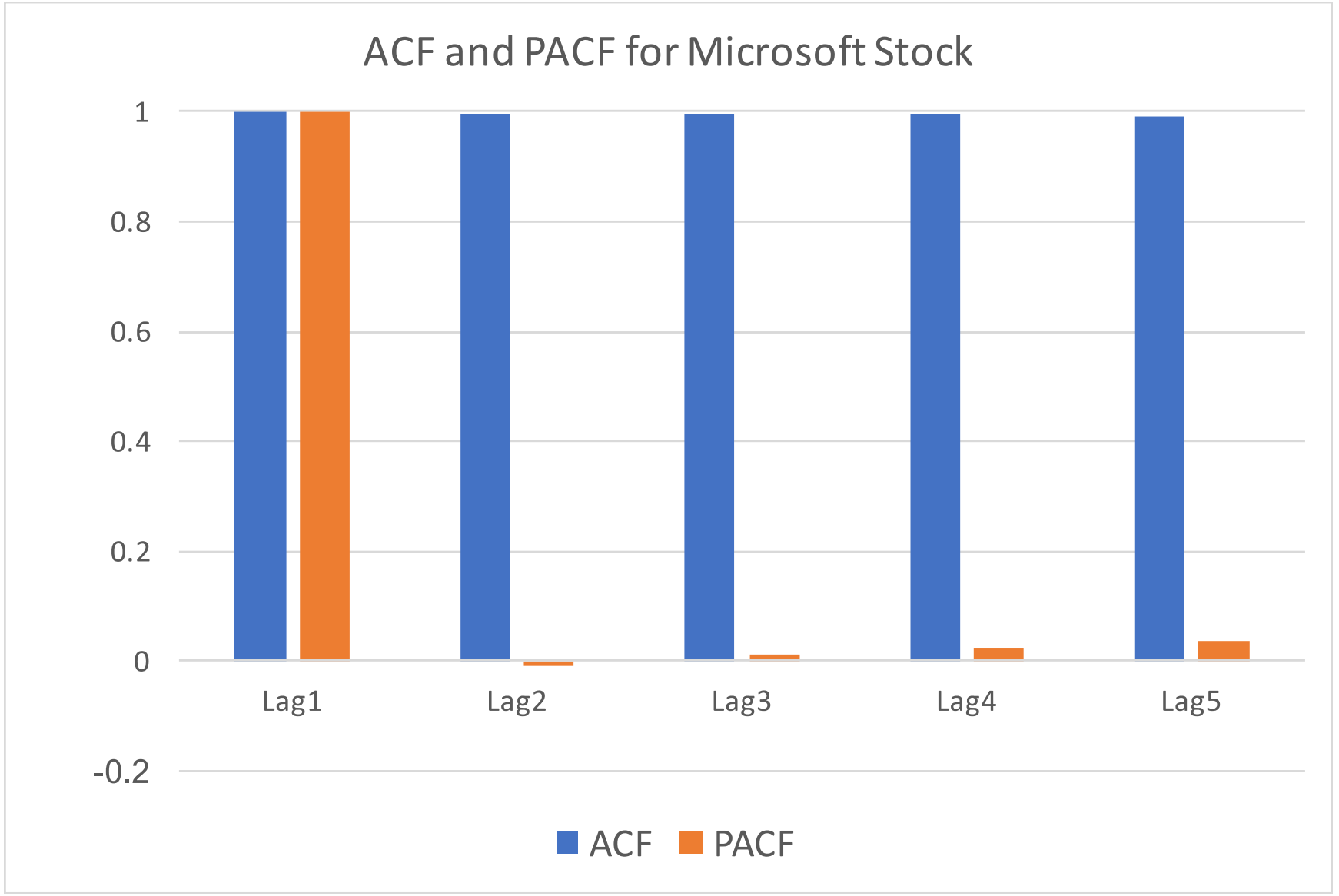
- PACF: The same as ACF with all intermediate correlations removed. It is the k_{th} coefficient of the ordinary least squares regression.

$$[y_t] = \beta_0 + \sum_{i=1}^{k} \beta_i [y_{t-i}] \text{ where}$$

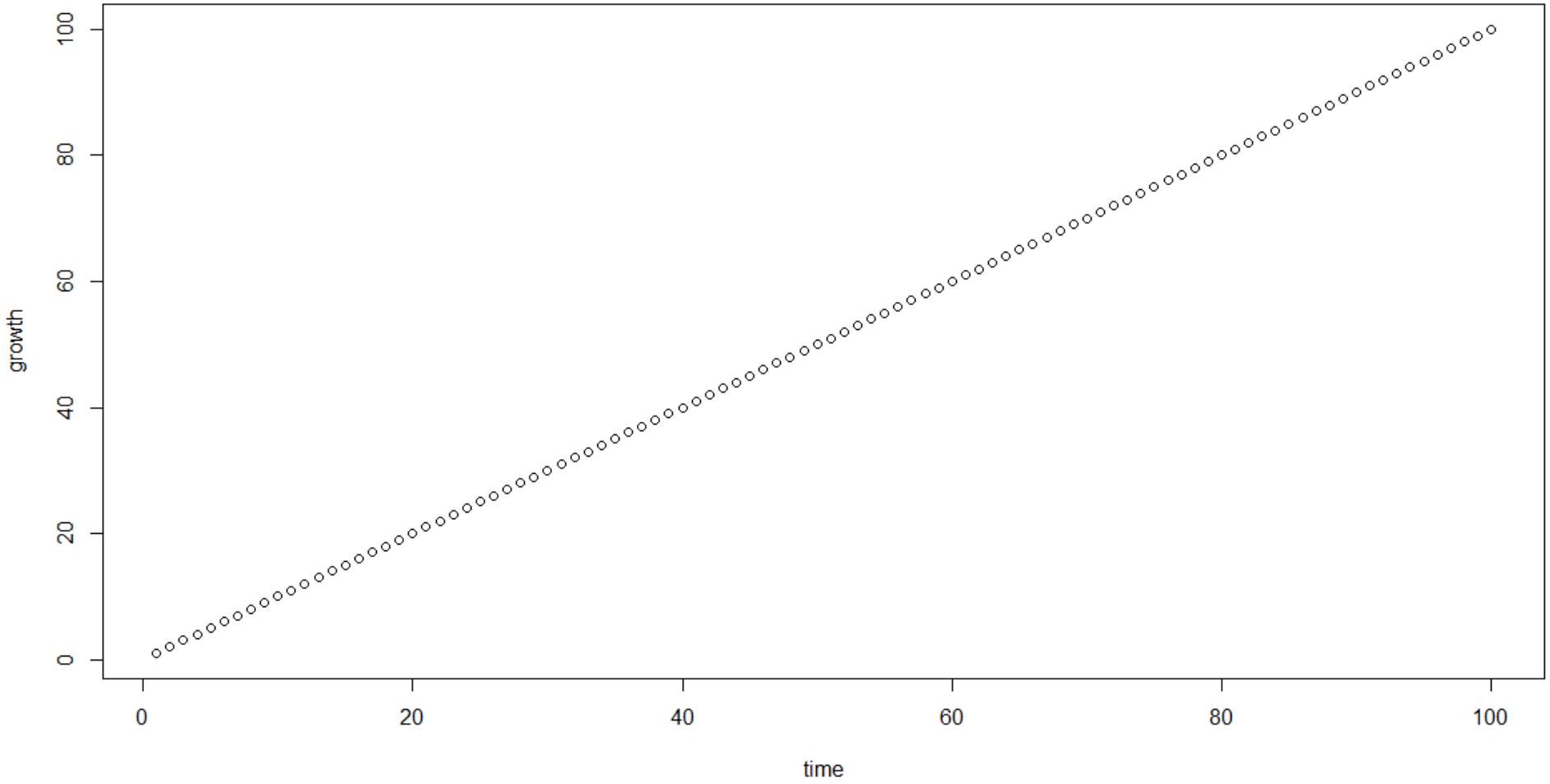
$[y_t]$ is the input time series, k is the lag order and β_i is the i_{th} coefficient of the linear multiple regression.

EXCEL ACTIVITY

Autocorrelation (ACF) and Partial ACF (PACF)

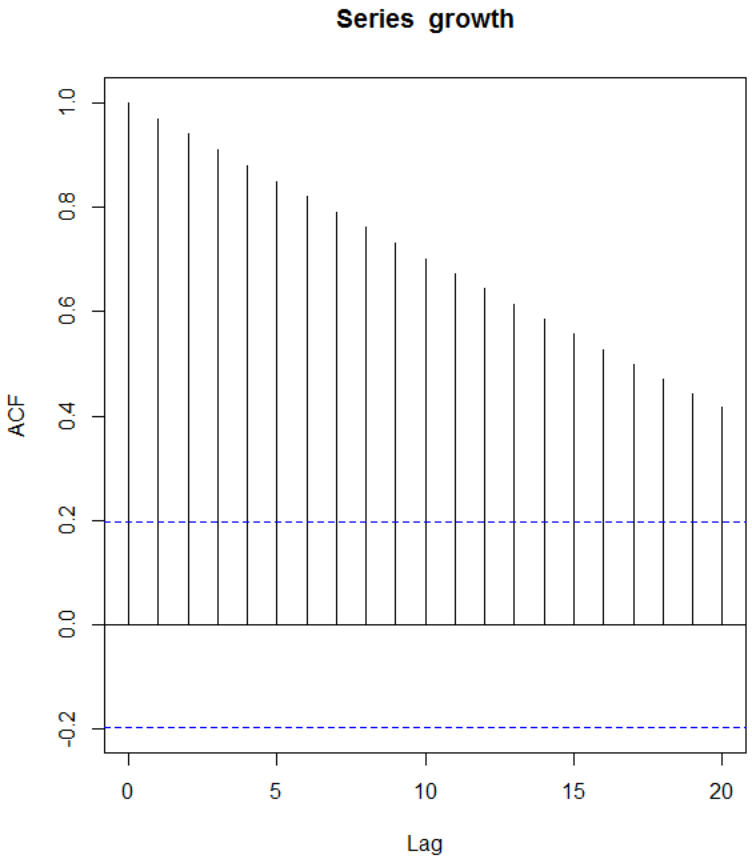


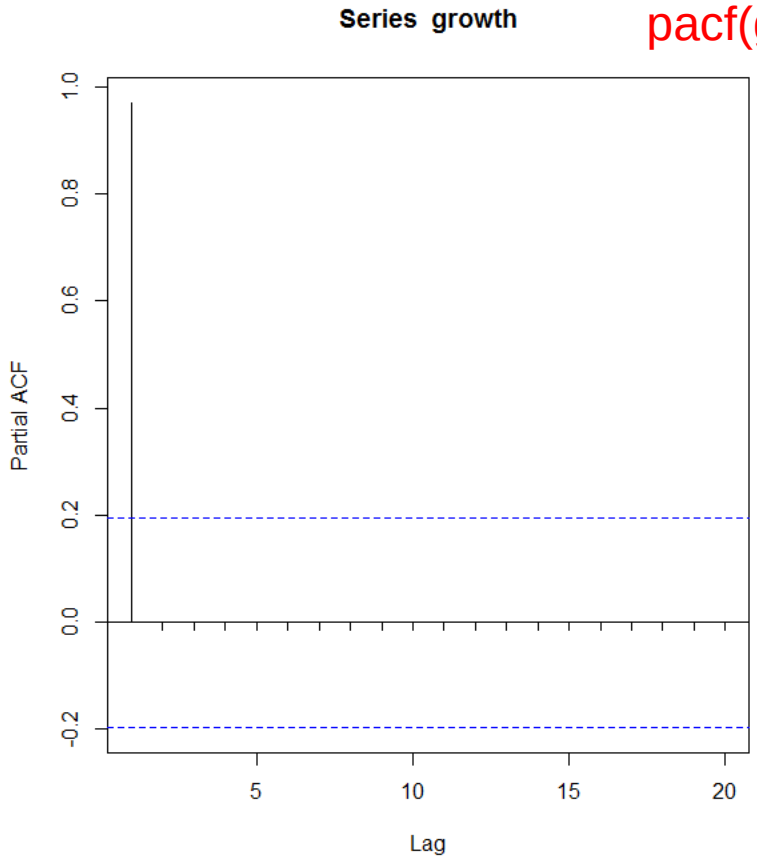
ACF and PACF – Idealized Trend



ACF and PACF – Idealized Trend

R code: growthT <- ts(growthT)
acf(growthT)
pacf(growthT)

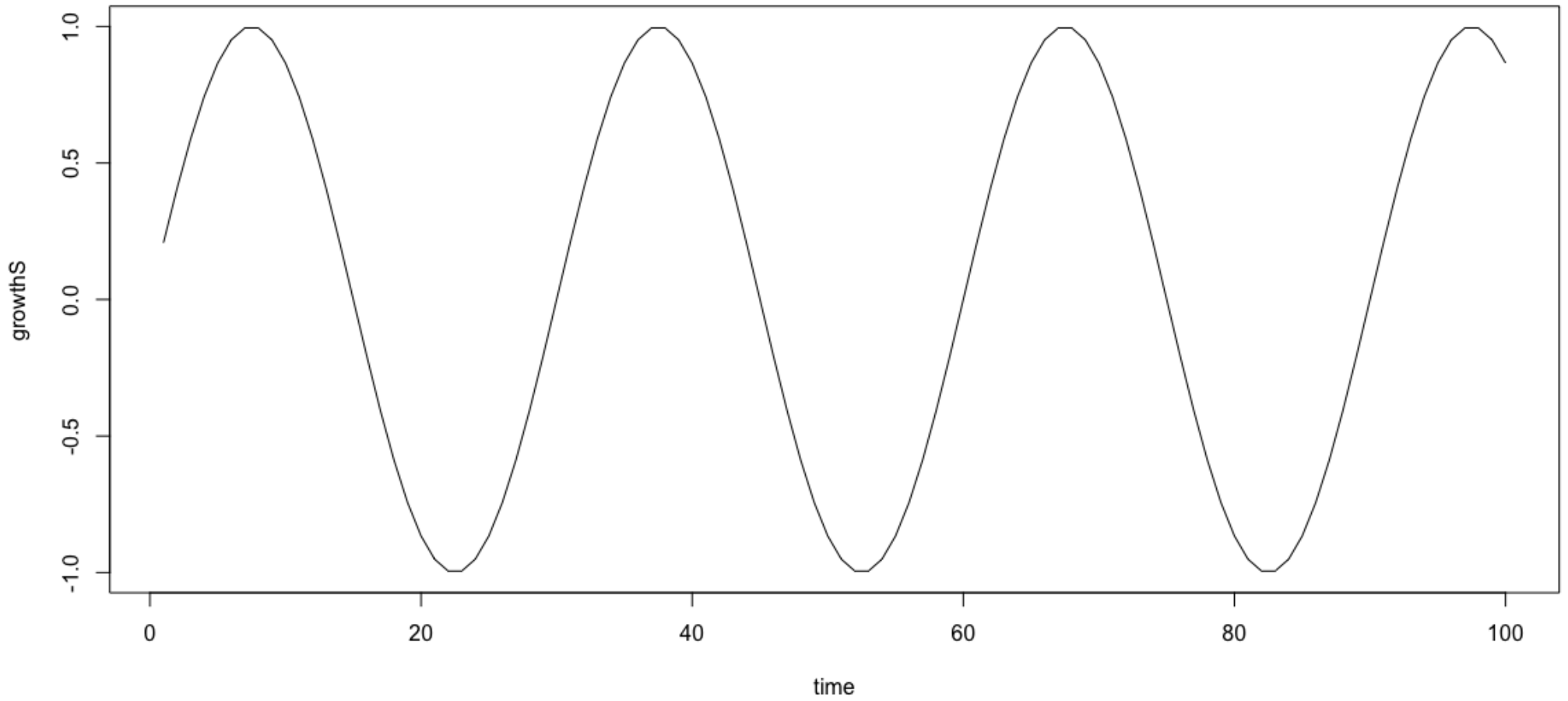




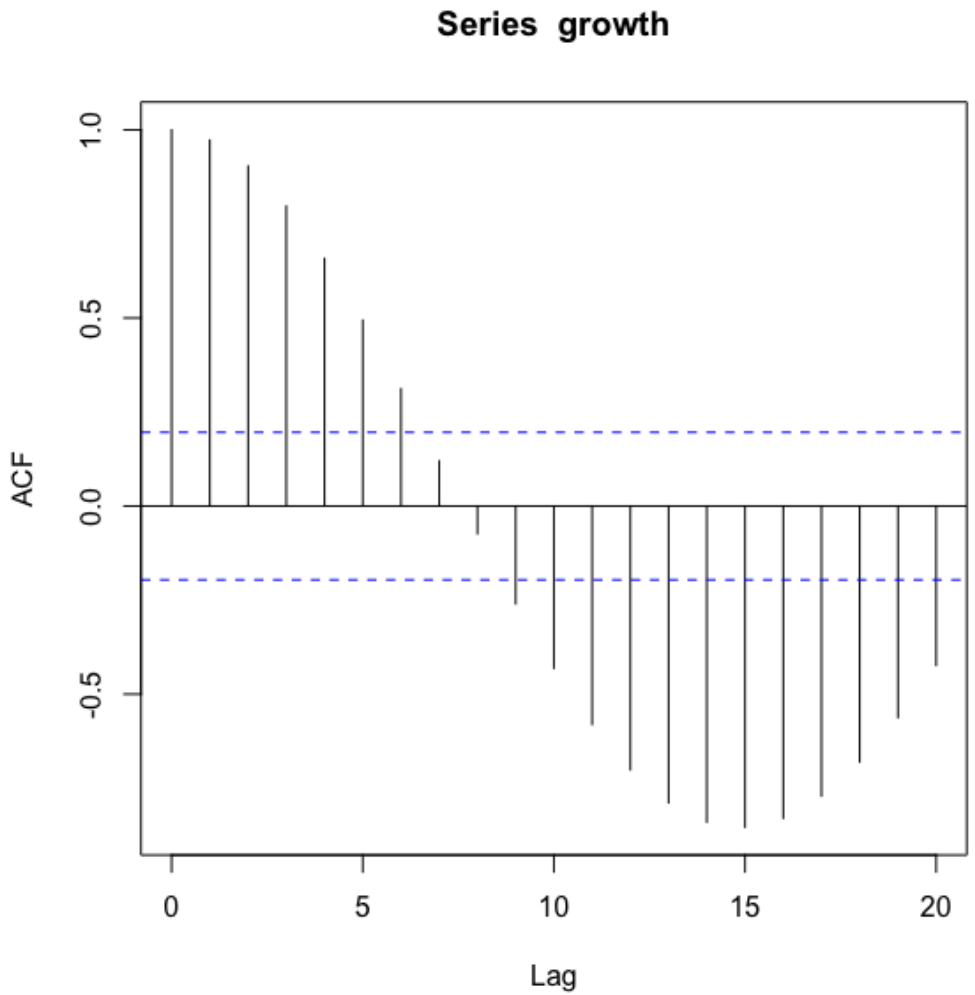
95% CI: $0 \pm \frac{1.96}{\sqrt{n}}$

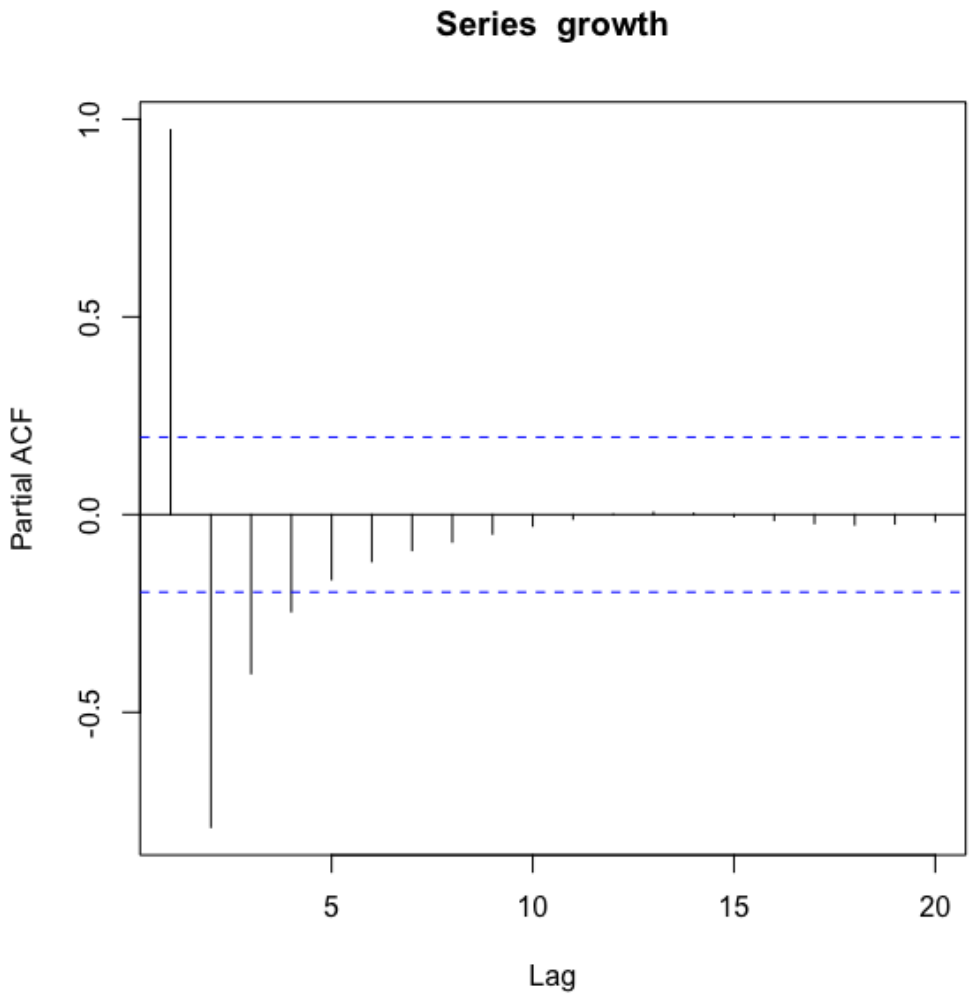
- ACF is a bar chart of correlation coefficients of the time series and its lags.
- PACF is a plot of the partial correlation coefficients of the time series and its lags.

ACF and PACF – Idealized Seasonality



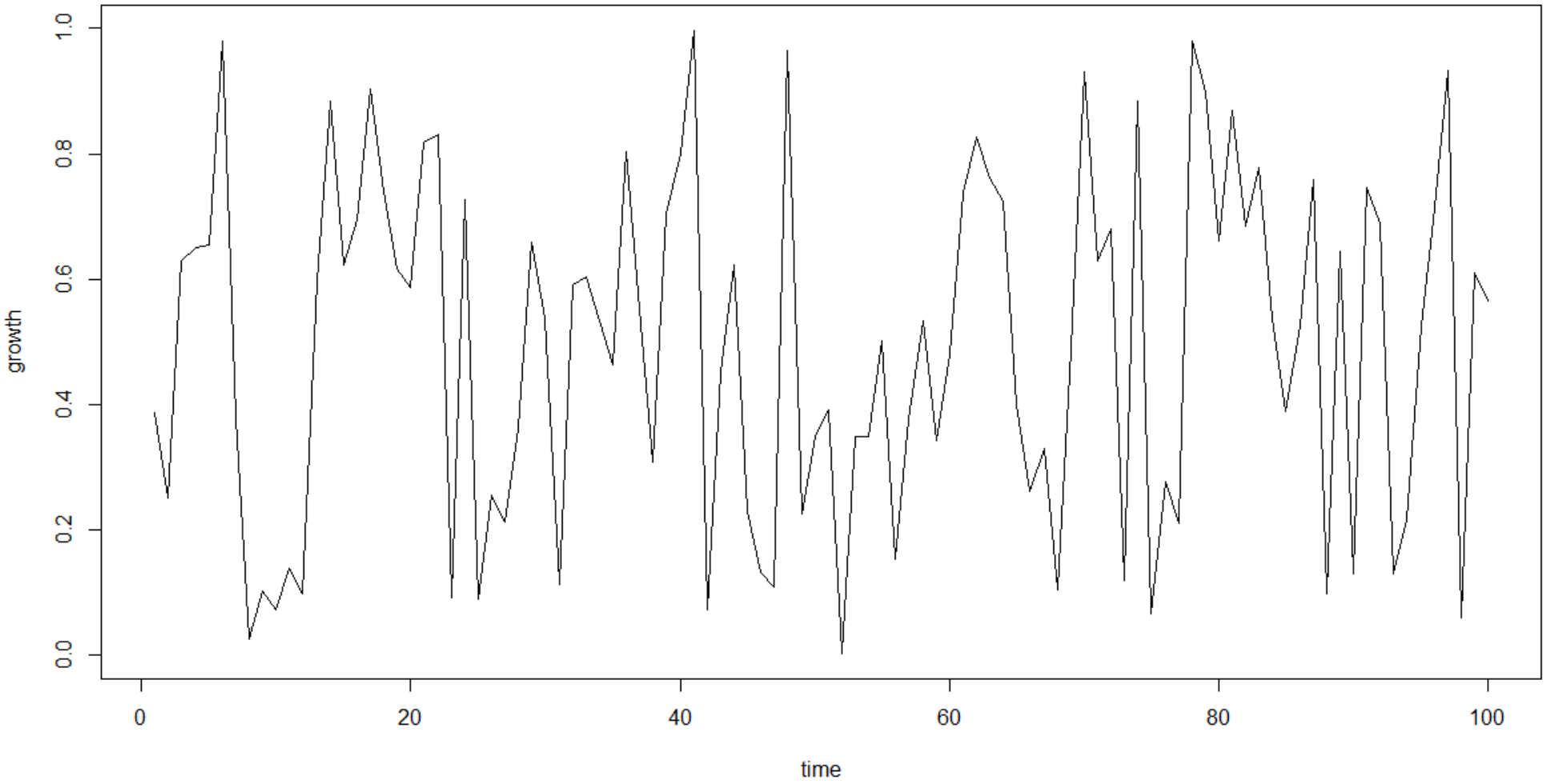
ACF and PACF – Idealized Seasonality



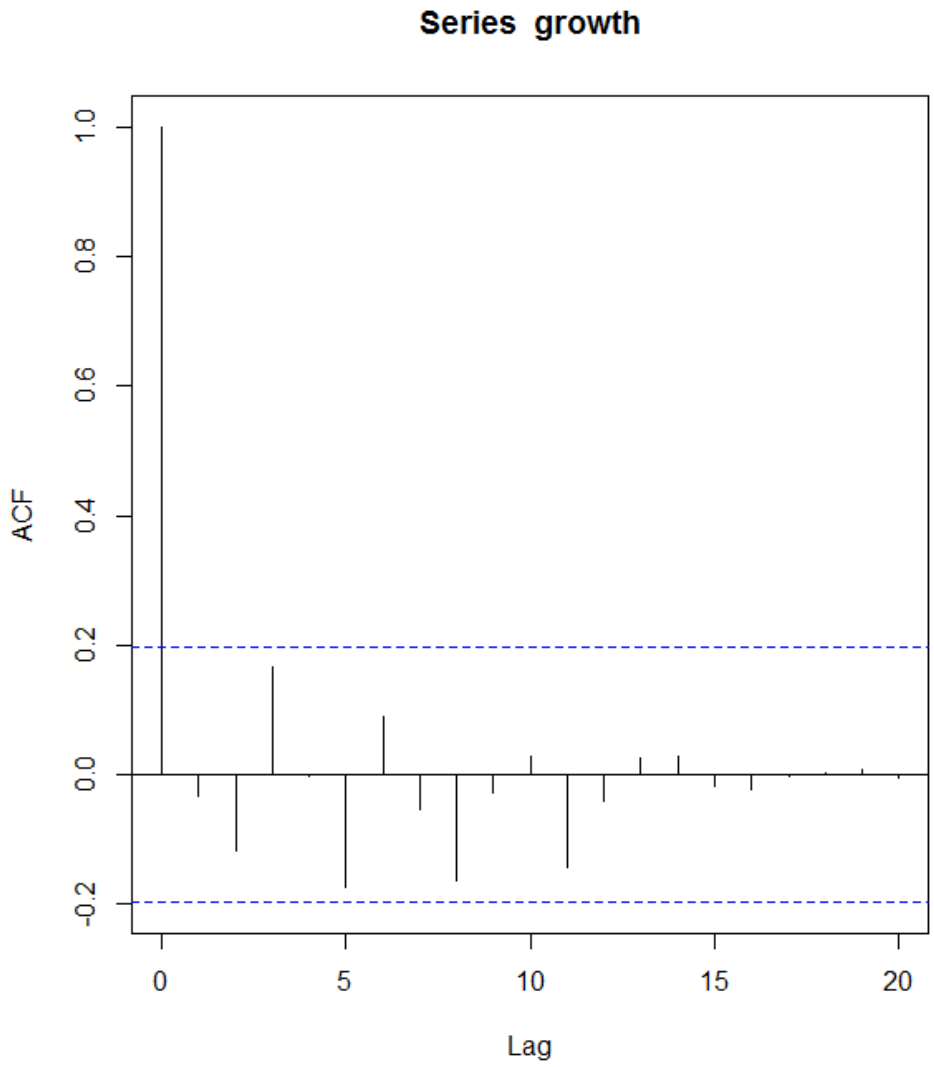


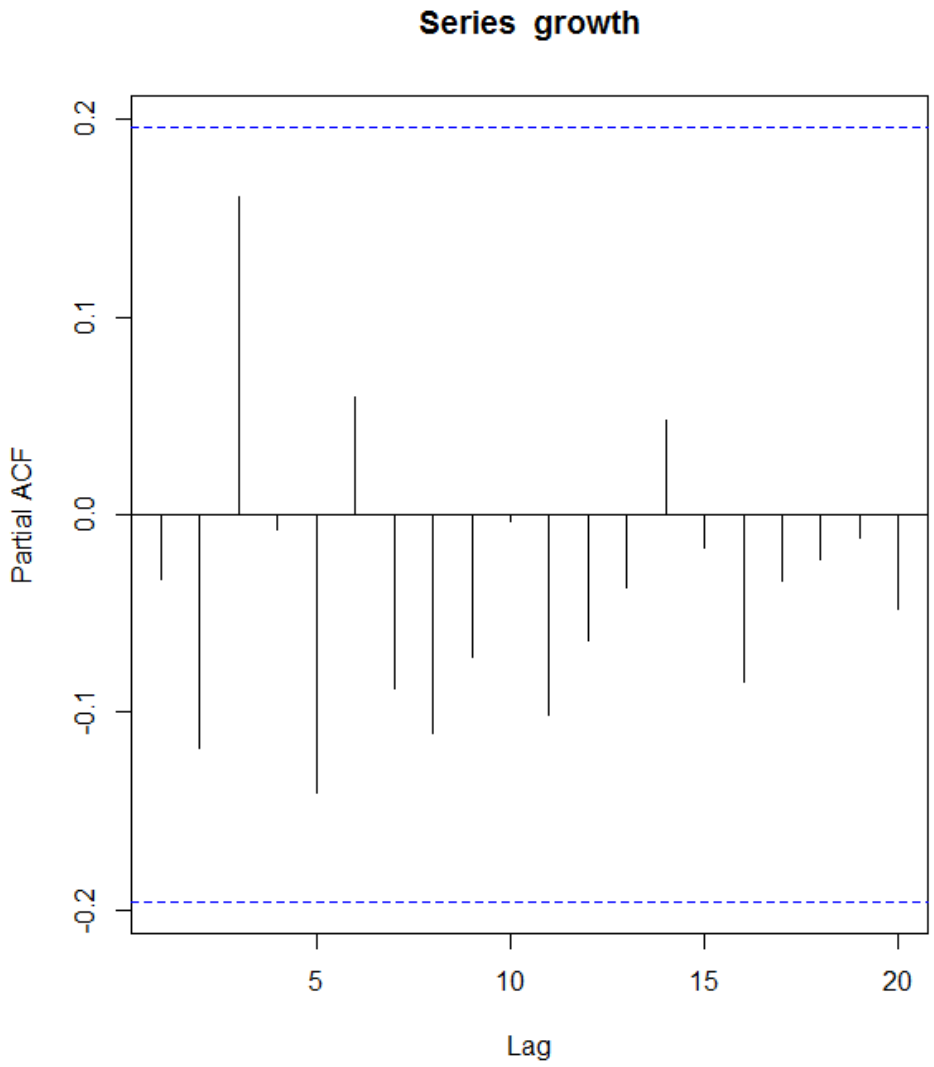
CSE 7302c

ACF and PACF – Idealized Randomness



ACF and PACF – Idealized Randomness



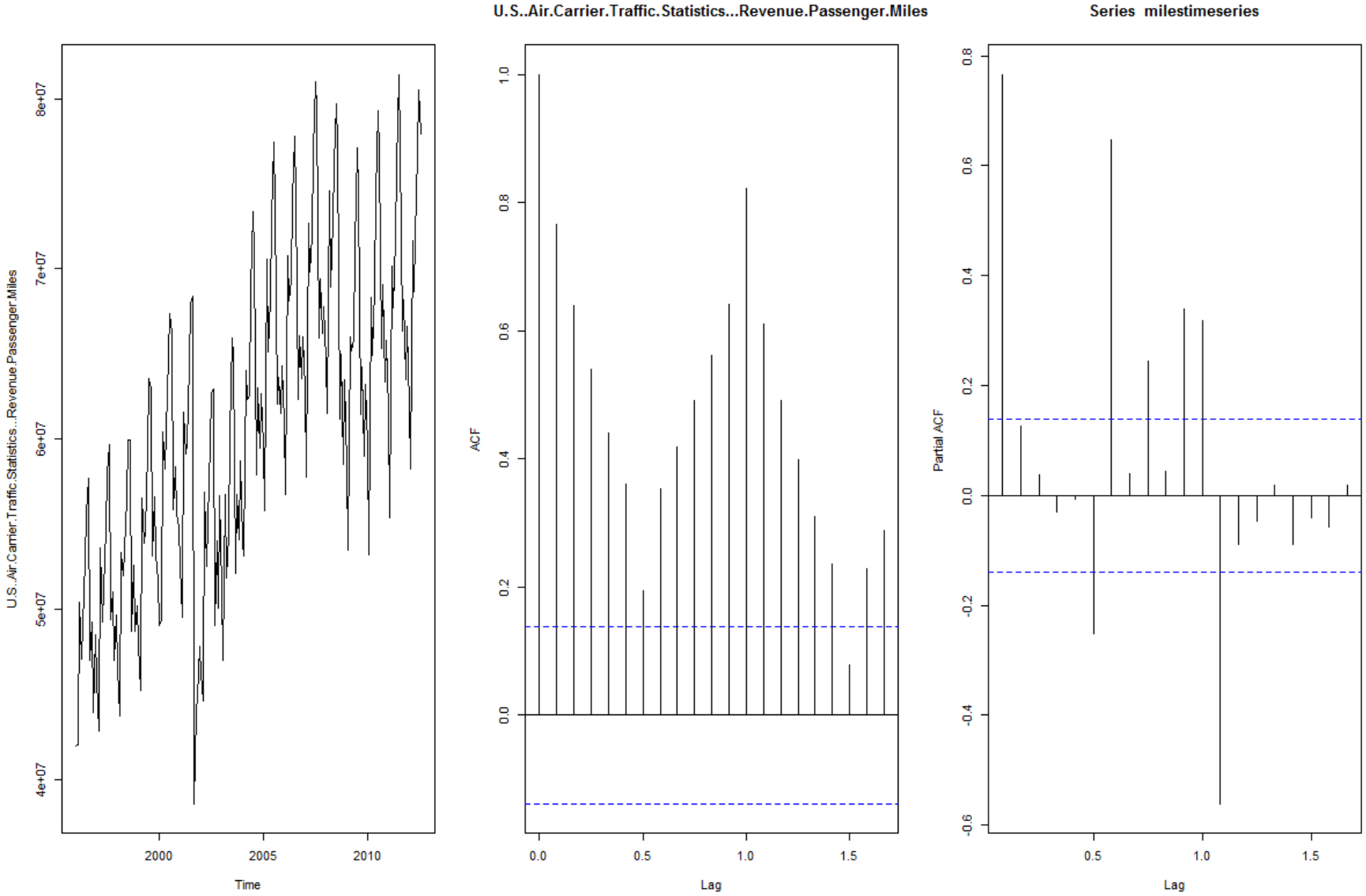


CSE 7302c

ACF and PACF – Idealized Trend, Seasonality and Randomness

- Ideal Trend: **Decay**ing ACF and **Cutoff** after 1 or 2 lags of PACF
- Ideal Seasonality: **Decay**ing Cyclicality in ACF and **Cutoff** after a few lags of PACF with some positive and some negative
- Ideal Random: A spike may or may not be present; even if present, magnitude will be small

ACF and PACF (Real-world): Decomposing Time Series into the 3 Components – RPM

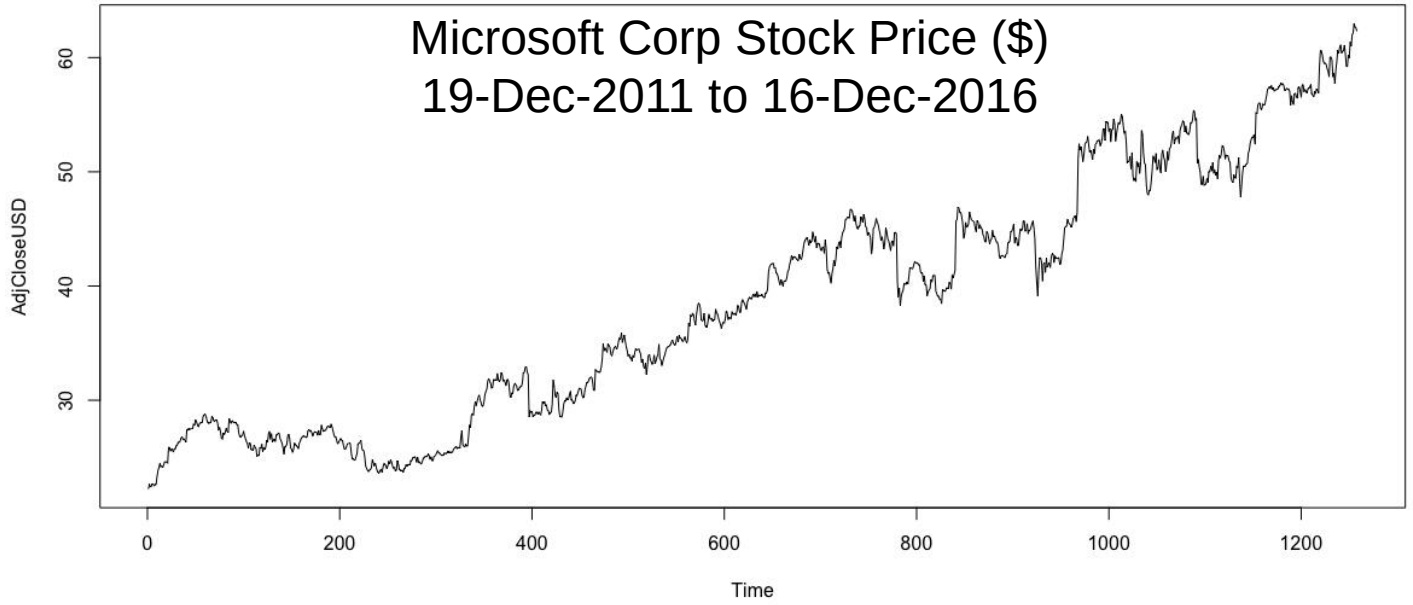


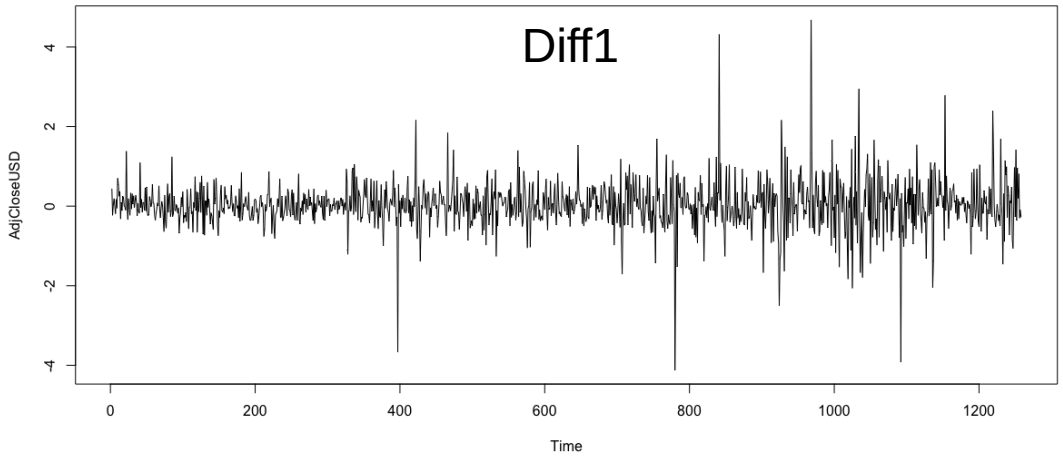
Stationary and Non-Stationary

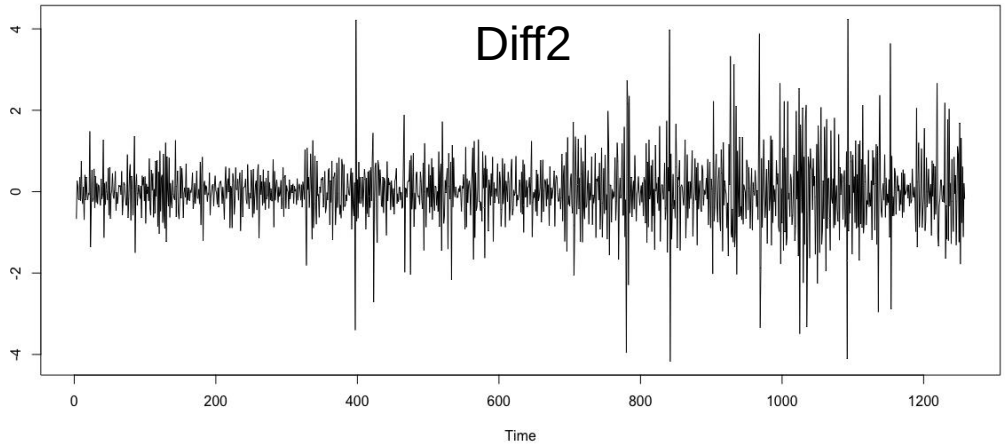
- Stationary data has constant statistical properties – mean, variance, autocorrelation, etc. – over time
- If the data is stationary, forecasting is easier!
- Differencing to convert non-stationary to stationary

EXCEL ACTIVITY

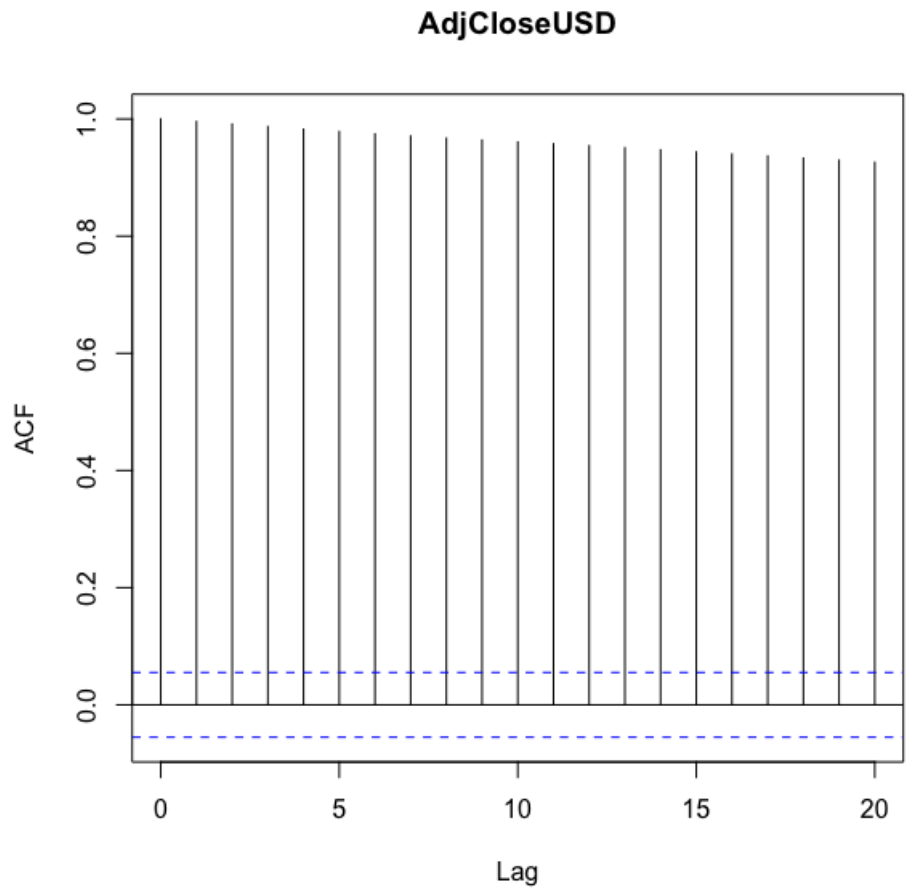
Removing Trend from Data

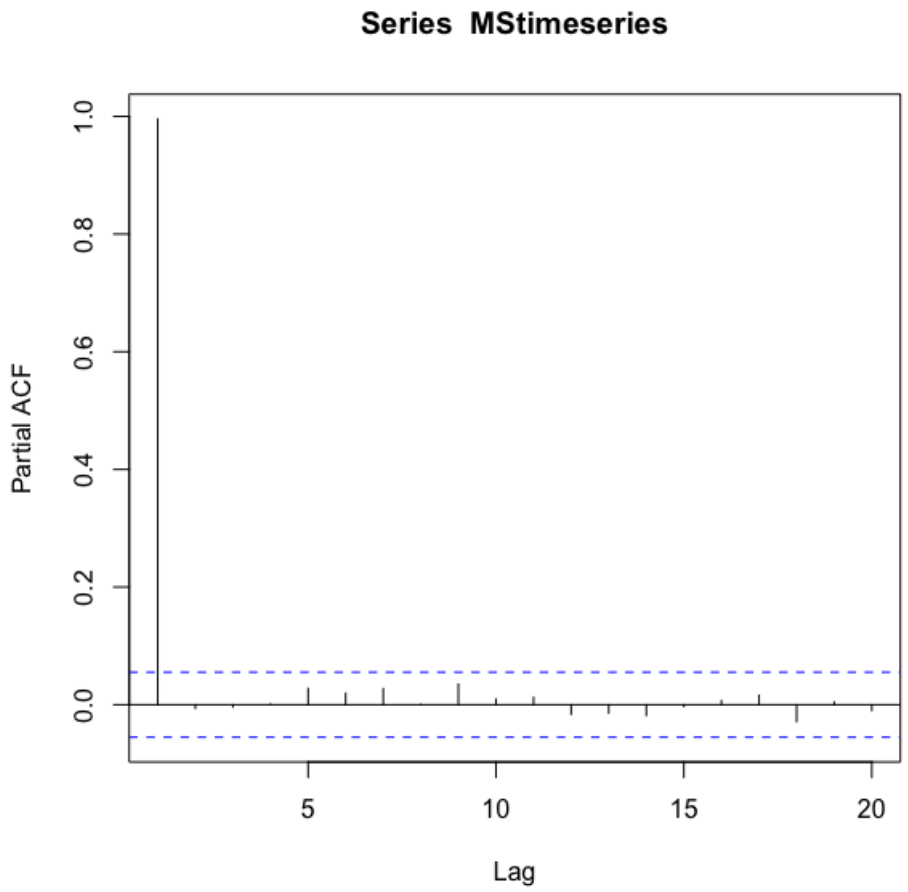






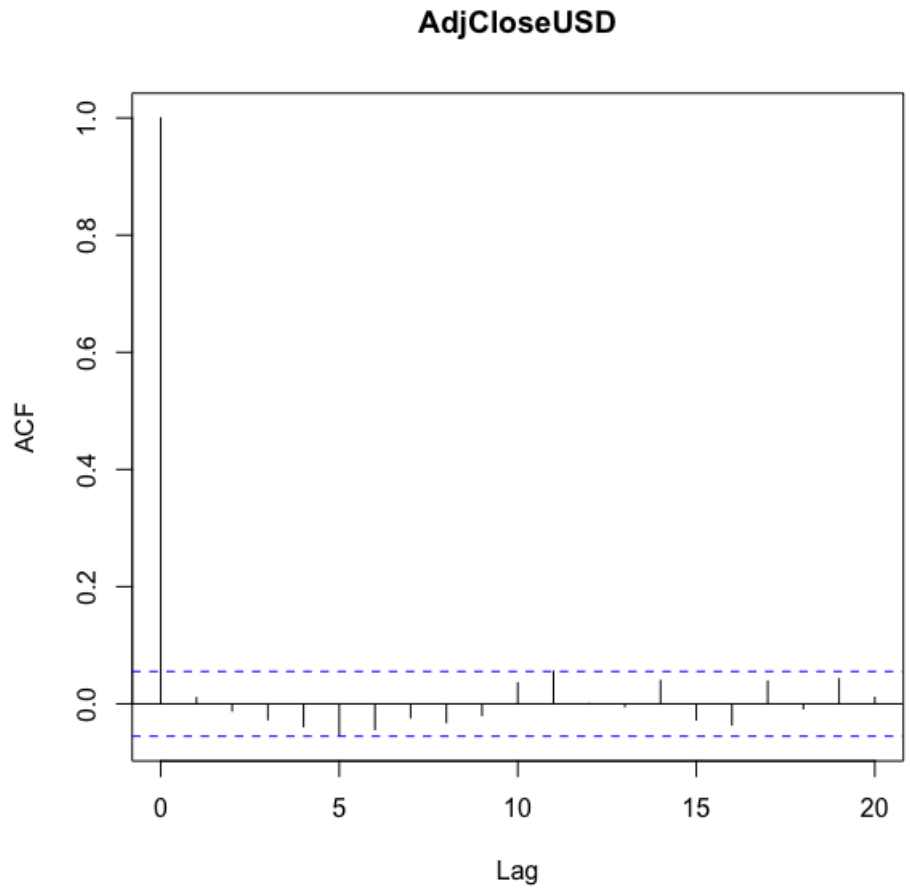
ACF and PACF of Stationary and Non-Stationary

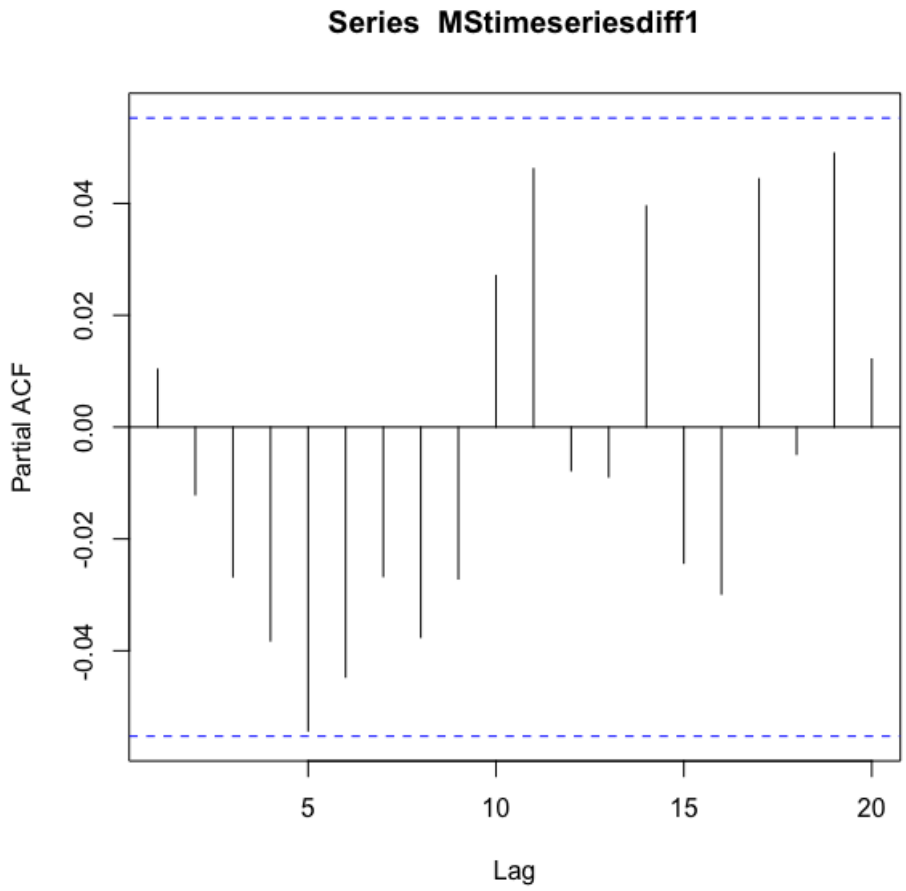




Price of Microsoft stock is highly correlated with the previous day's value.

ACF and PACF of Stationary and Non-Stationary





Daily changes in Microsoft stock price are essentially random.

ACF and PACF of Stationary and Non-Stationary

- Non-stationary series have an ACF that remains significant for half a dozen or more lags, rather than quickly declining to zero.

- You must difference such a series until it is stationary before you can identify the process.

ARIMA

MODEL BUILDING

Box–Jenkins Methodology

- Model identification and model selection
  - Make sure variables are stationary. Difference as necessary to get a constant mean and transformations to get constant variance.
  - Check for seasonality: Decays and spikes at regular intervals in ACF plots.
- Parameter estimation
  - Compute coefficients that best fit the selected model.
- Model checking
  - Check if residuals are independent of each other and constant in mean and variance over time (white noise).

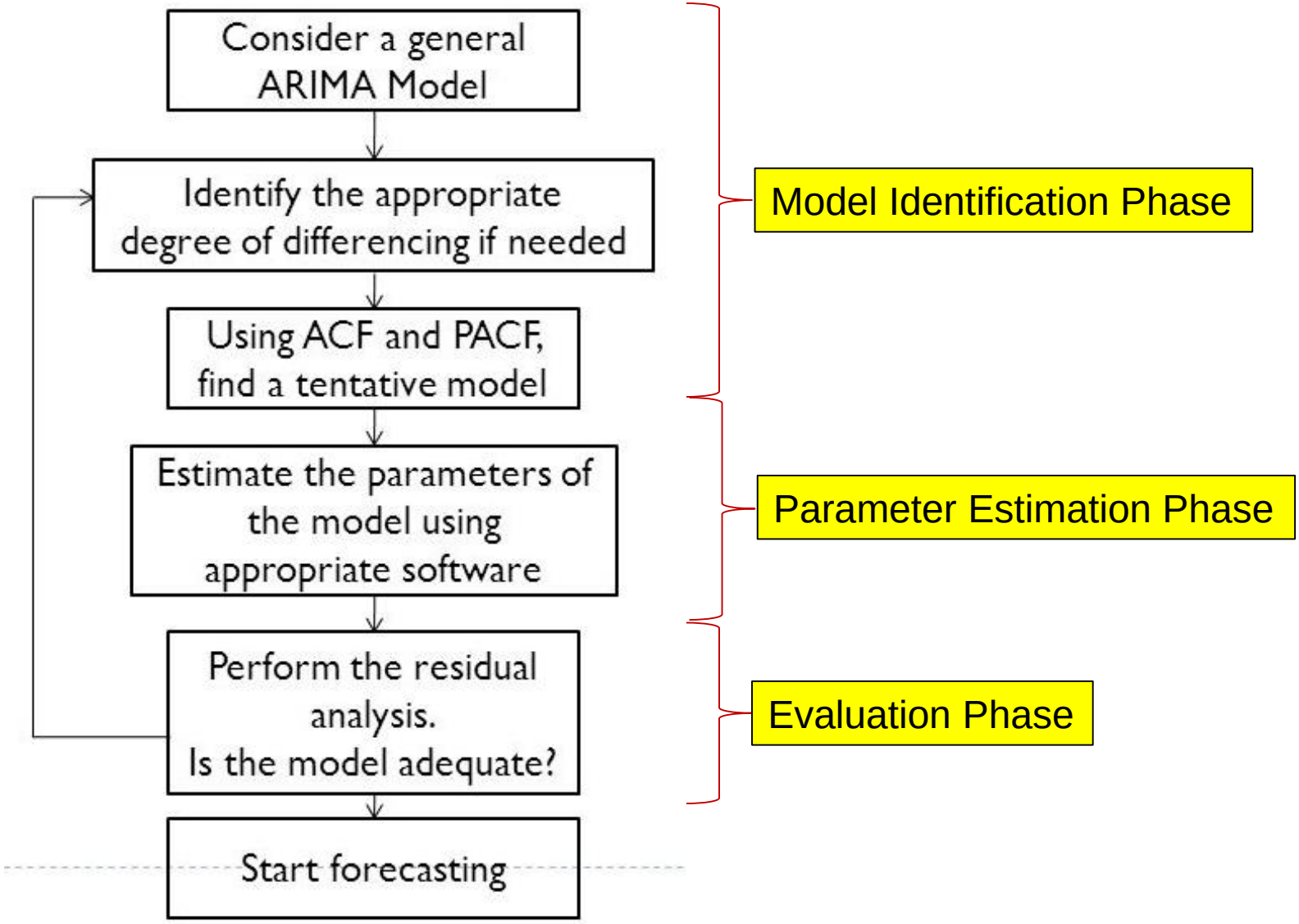
http://www.ncss.com/wp-content/themes/ncss/pdf/Procedures/NCSS/The_Box-Jenkins_Method.pdf

AutoRegressive Integrated Moving Average - ARIMA(p,d,q) Model

- *p* is the number of autoregressive [AR(p)] terms (a linear regression of the current value of the series against one or more prior values of the series)
  - Maximum lag beyond which PACF is 0
- *d* is the number of non-seasonal differences (order of the differencing) used to make the time series stationary [I(d)]
- *q* is the order of the moving average [MA(q)] model
  - Maximum lag beyond which the ACF is 0

- Non-seasonal ARIMA models are denoted ARIMA(p,d,q)
- Seasonal ARIMA (SARIMA) models are denoted ARIMA(p,d,q)$(P,D,Q)_m$, where m refers to the number of periods in each season and (P,D,Q) refer to the autoregressive, differencing and moving average terms of the seasonal part of the ARIMA model.

Time Series Model Building Using ARIMA

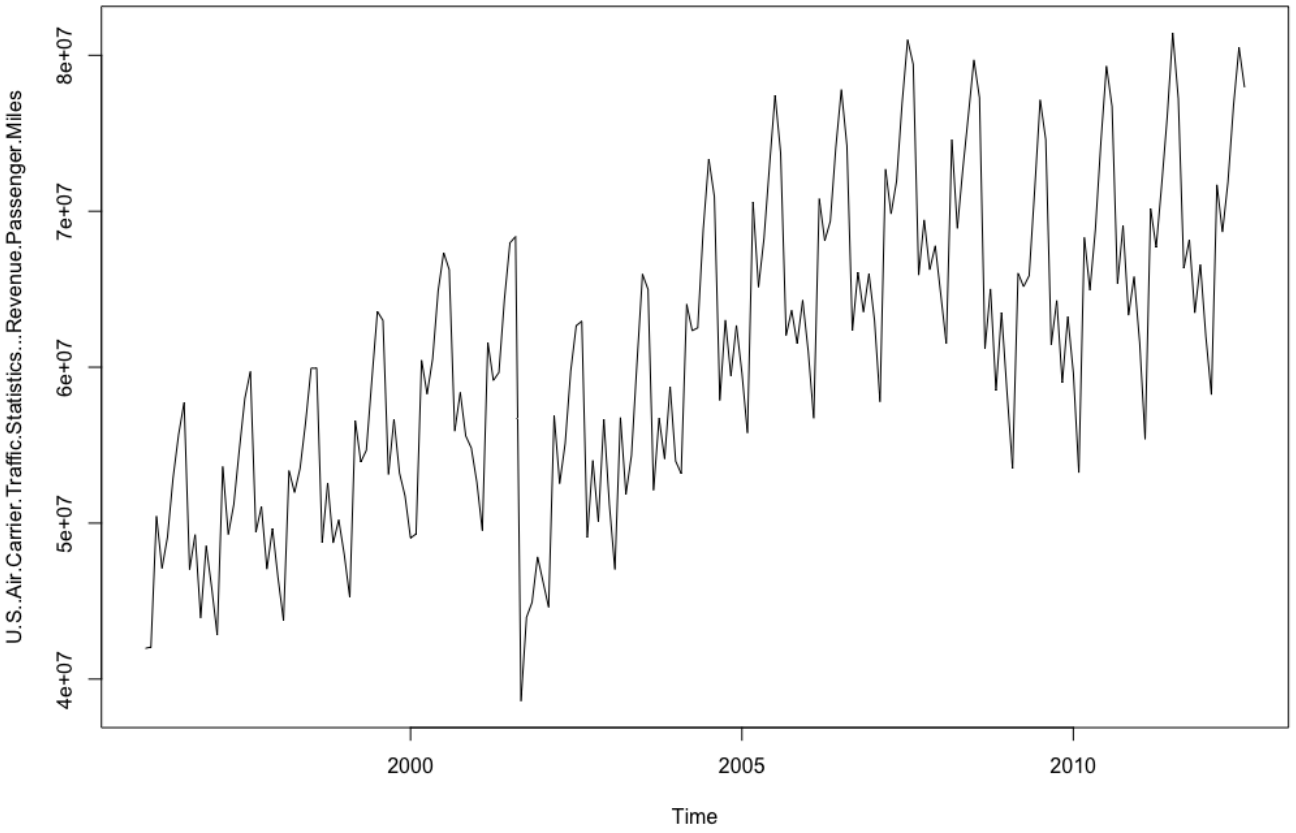


Time Series Model Building Using ARIMA

Identification Phase

Step 1: Plot the data (transform data to stabilize variance, if required)

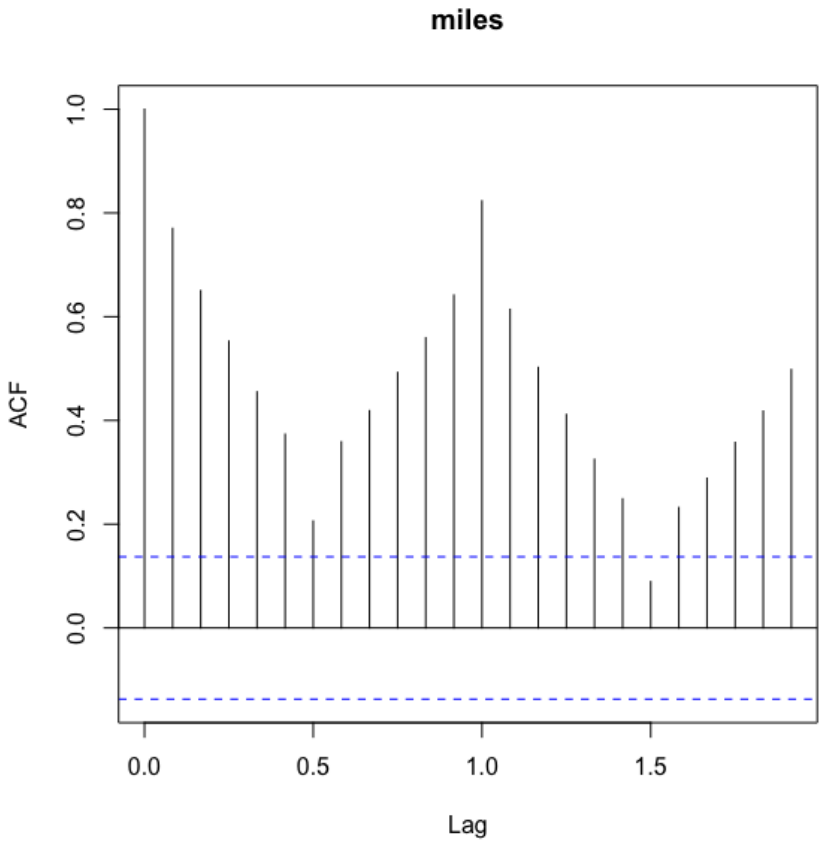
R code: plot(milestimeseries)

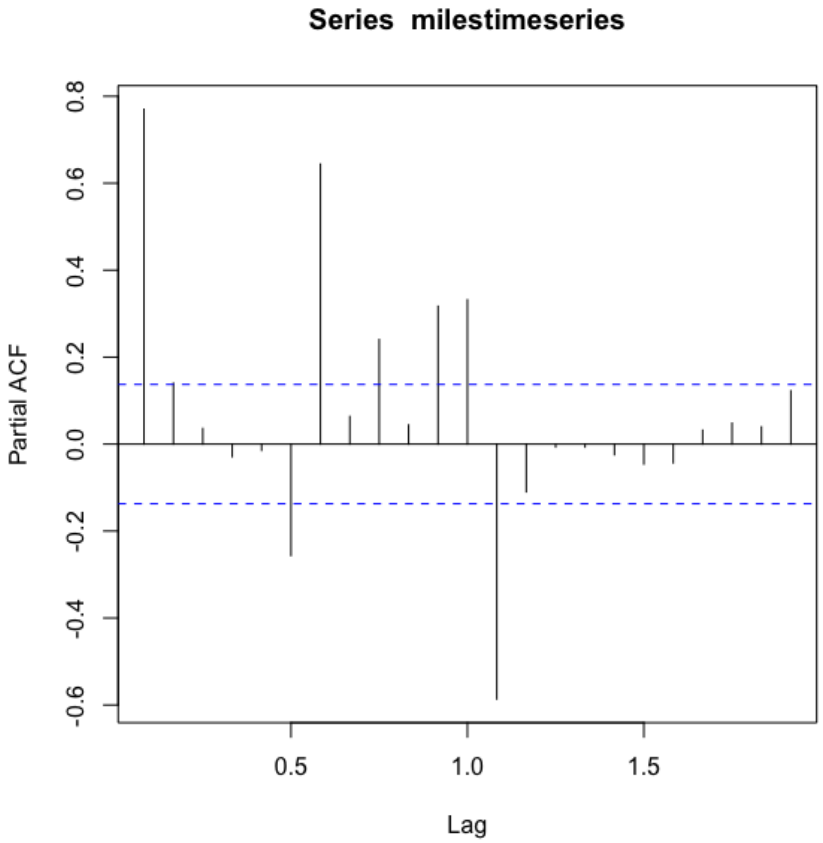


Time Series Model Building Using ARIMA

Identification Phase

Step 2: Plot ACF and PACF to get preliminary understanding of the processes involved. R code: acf(milestimeseries) pacf(milestimeseries)

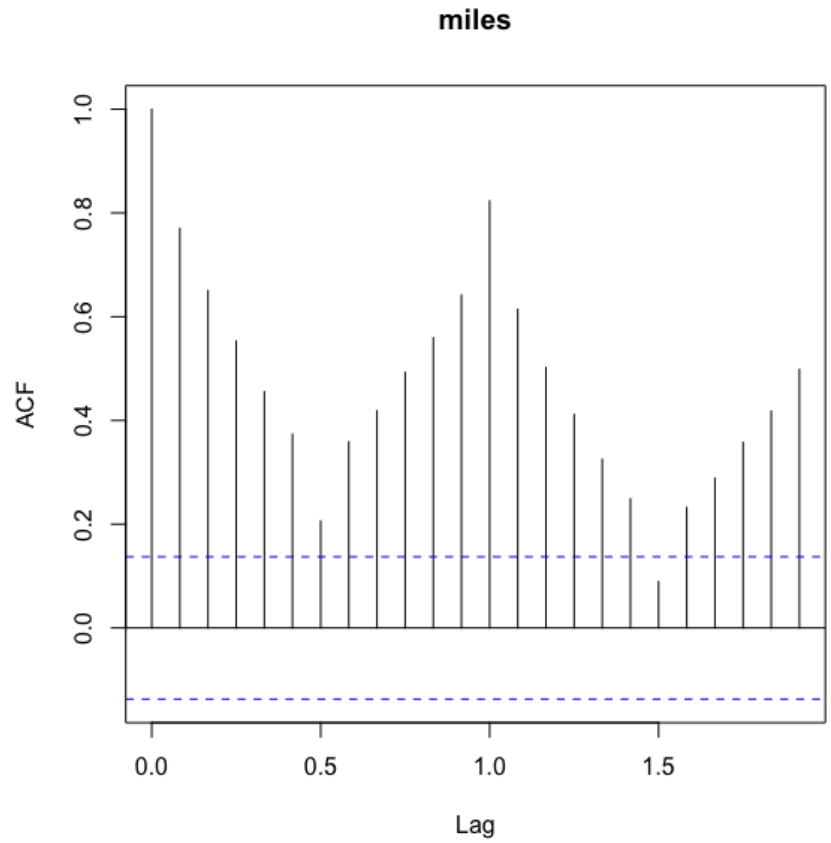




Time Series Model Building Using ARIMA

Identification Phase

Step 2: Plot ACF and PACF to get preliminary understanding of the processes involved.

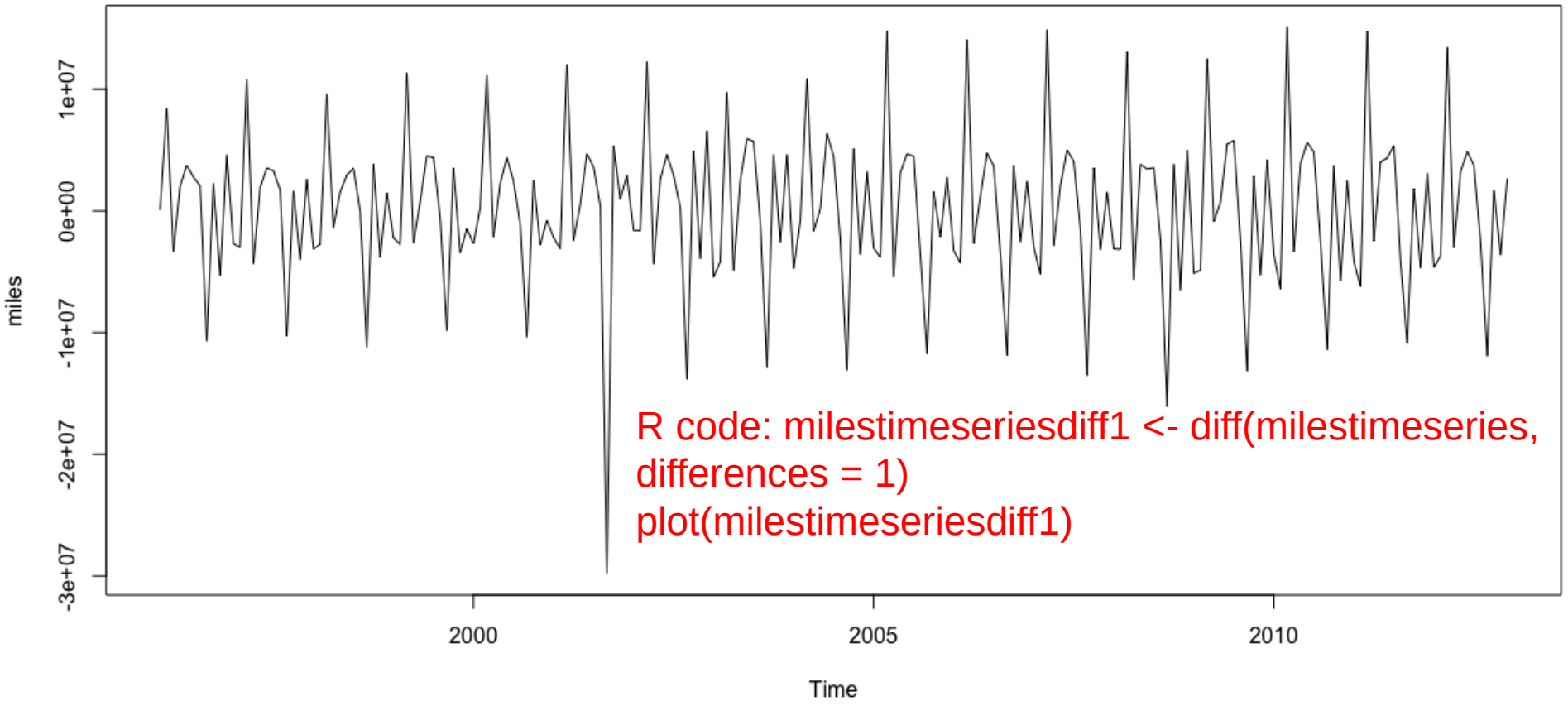


The suspension bridge pattern in ACF (also, positive and negative spikes in PACF) suggests non-stationarity and strong seasonality.

Time Series Model Building Using ARIMA

Identification Phase

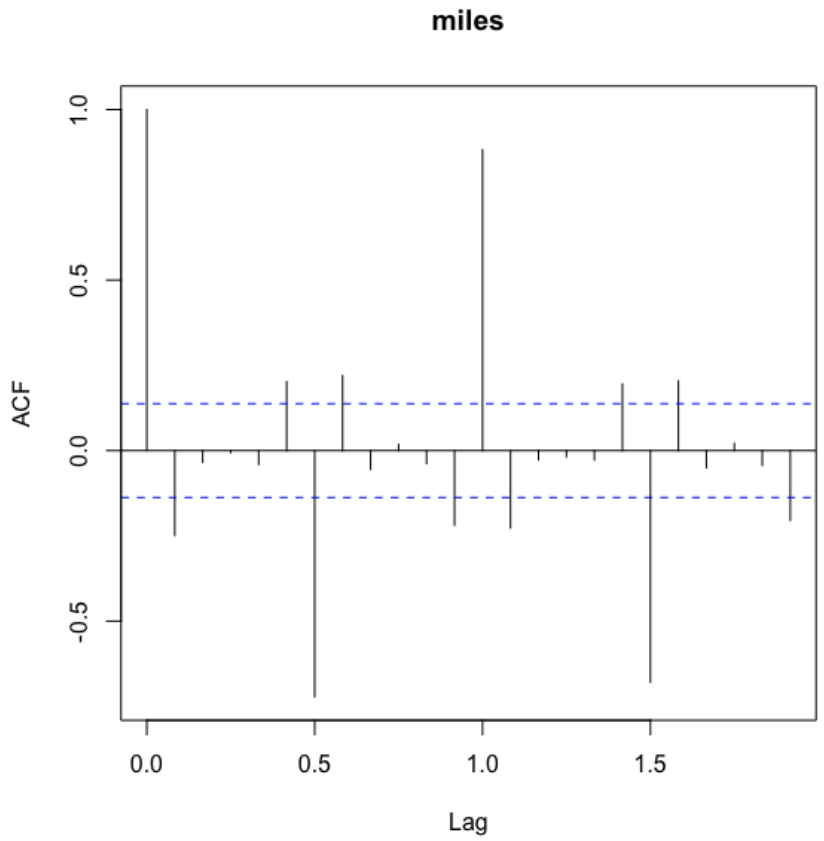
Step 3: Perform a non-seasonal difference. It is the same as an ARIMA(0,1,0) model.

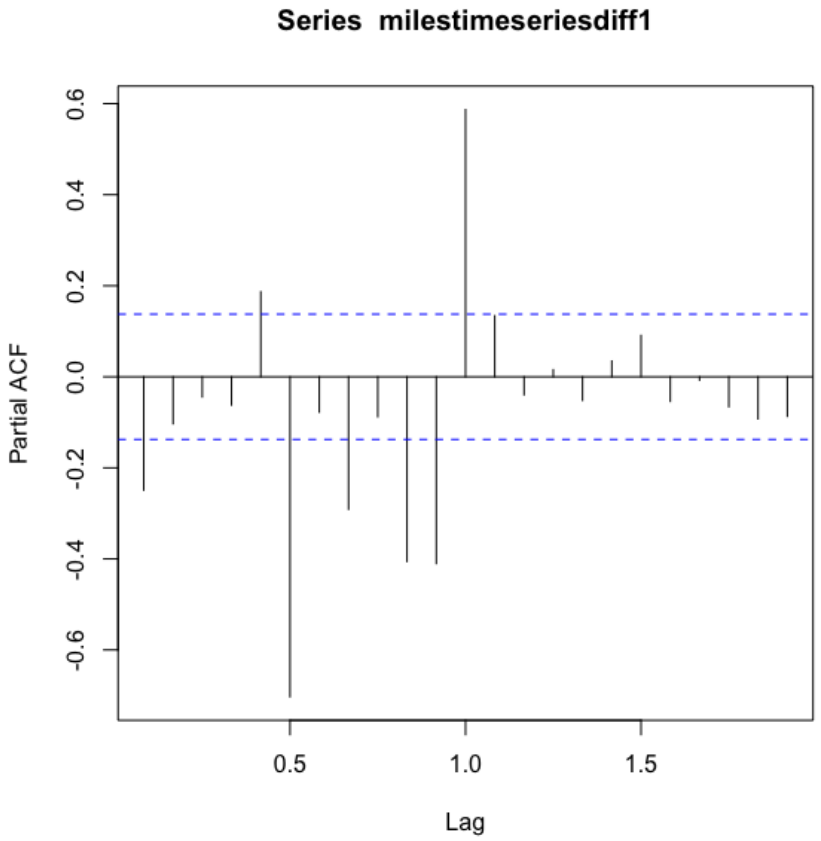


Time Series Model Building Using ARIMA

Identification Phase

Step 4: Check ACF and PACF of differenced data to explore remaining dependencies.





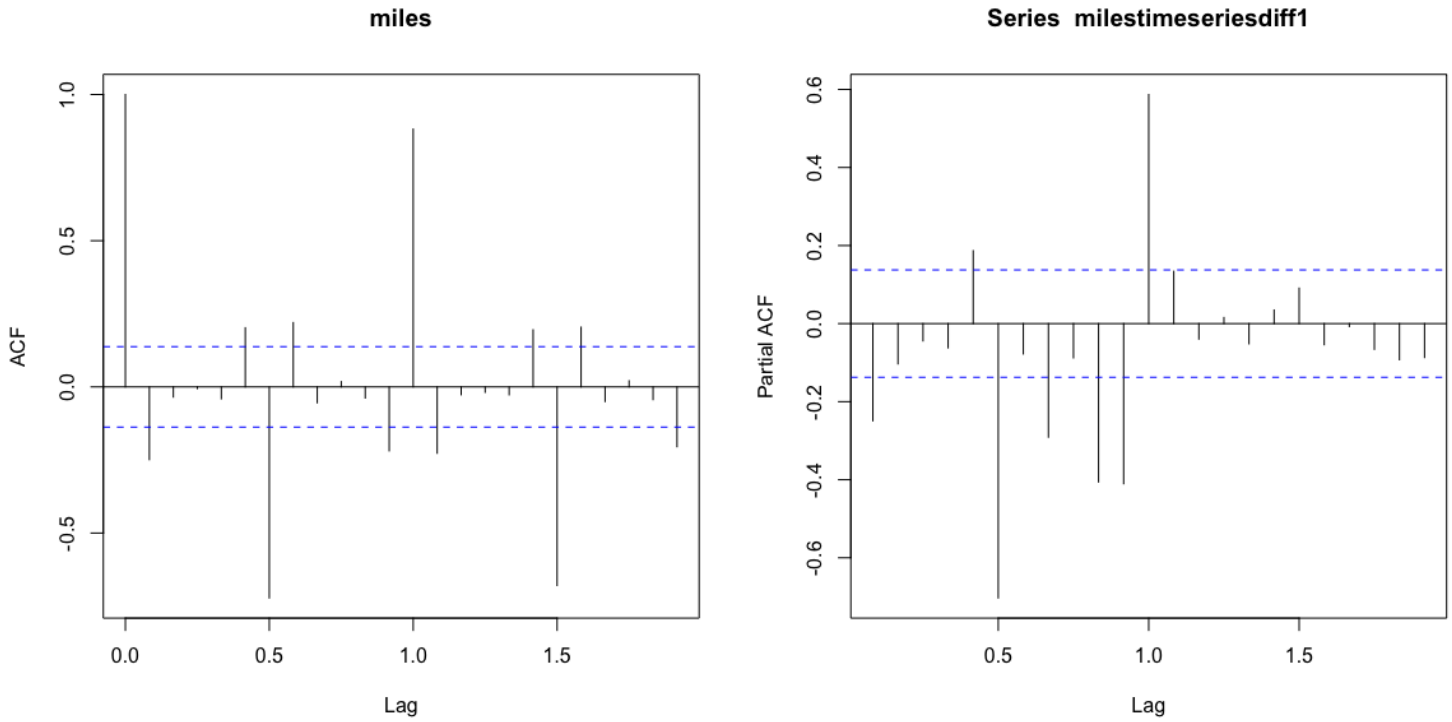
CSE 7302c

Time Series Model Building Using ARIMA

Identification Phase

Step 4: Check ACF and PACF of differenced data to explore remaining dependencies.

The differenced series looks somewhat stationary but has strong seasonal lags.

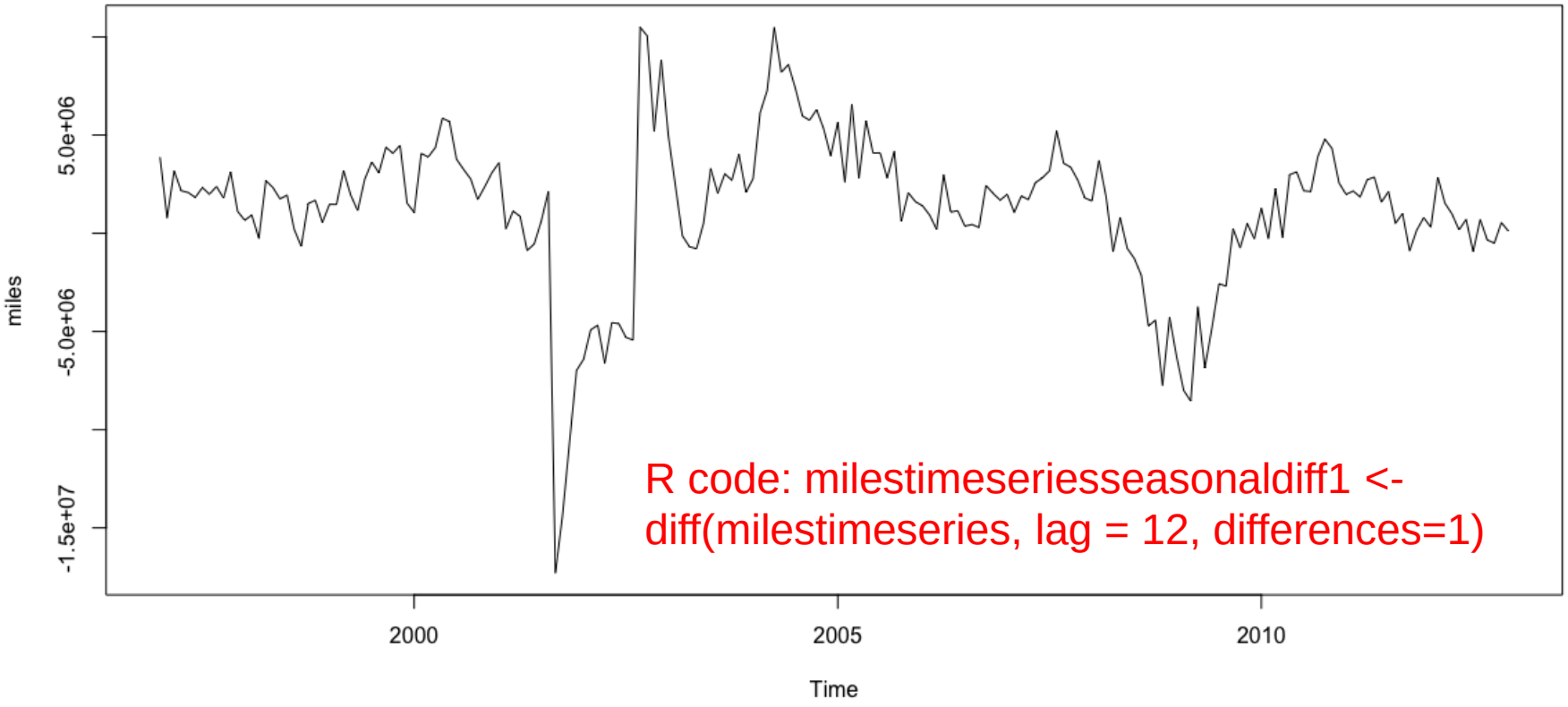


CSE 7302c

Time Series Model Building Using ARIMA

Identification Phase

Step 5: Perform seasonal differencing (t_0-t_{12}, t_1-t_{13}, etc.) on the **original** time series to get seasonal stationarity. This is the same as an ARIMA(0,0,0)(0,1,0)$_{12}$ model.

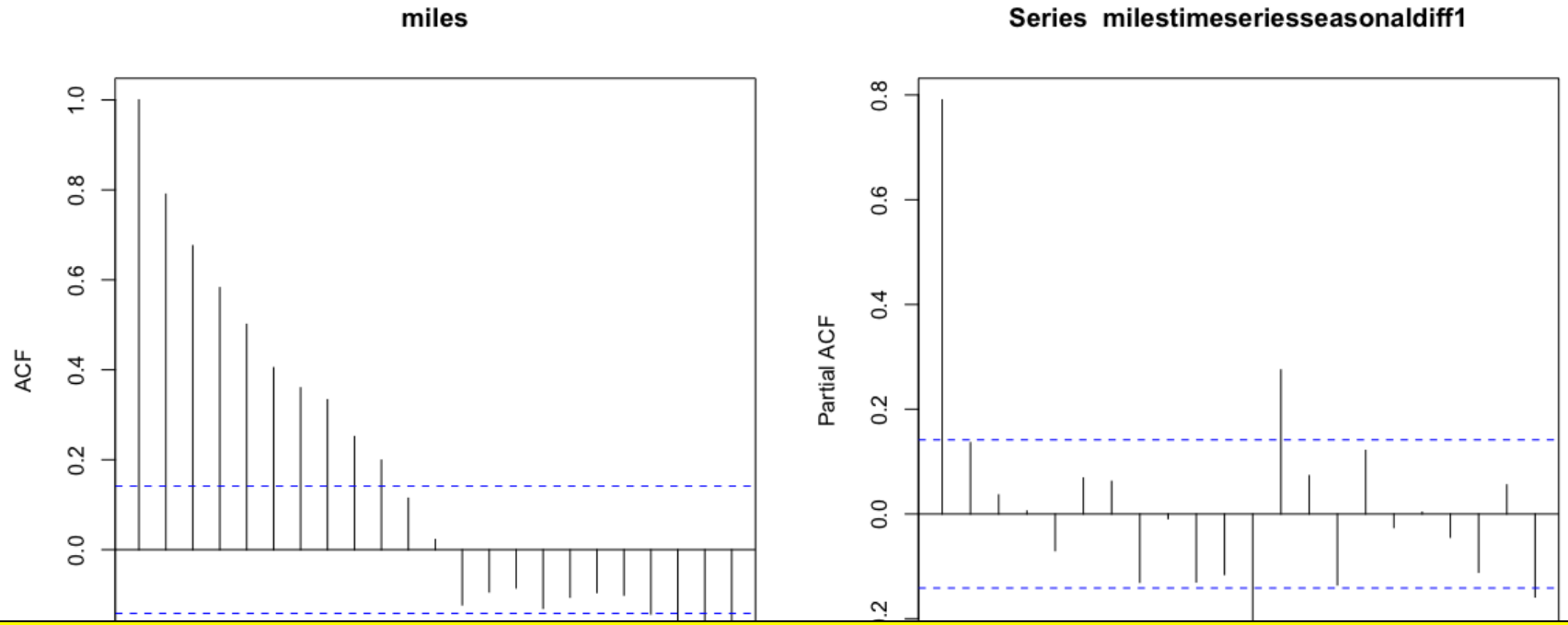


R code: milestimeseriesseasonaldiff1 <-
diff(milestimeseries, lag = 12, differences=1)

Time Series Model Building Using ARIMA

Identification Phase

Step 6: Check ACF and PACF of seasonally differenced data to explore remaining dependencies and identify model(s).

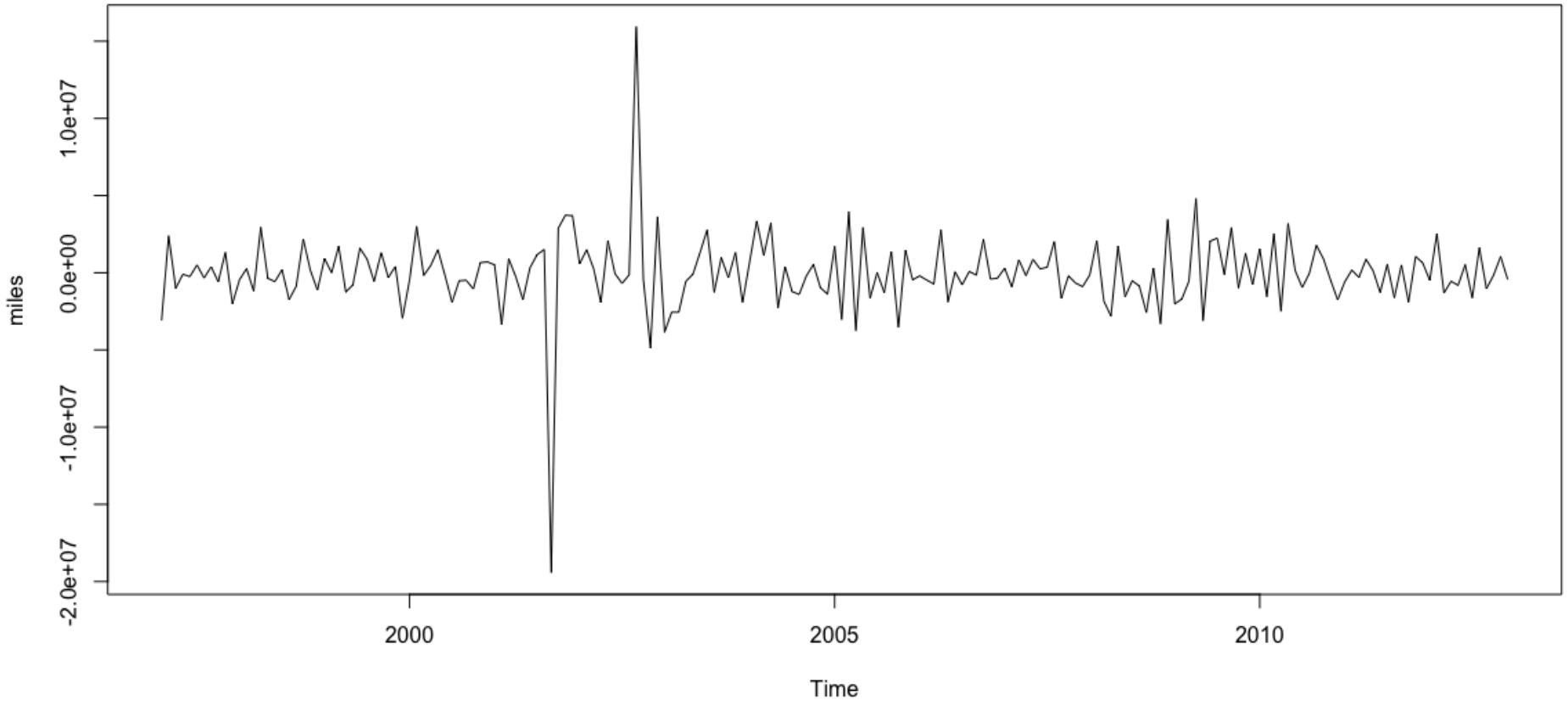


Positive autocorrelation indicates *underdifferencing* requiring either an AR term or a non-seasonal differencing to fix it.

Time Series Model Building Using ARIMA

Identification Phase

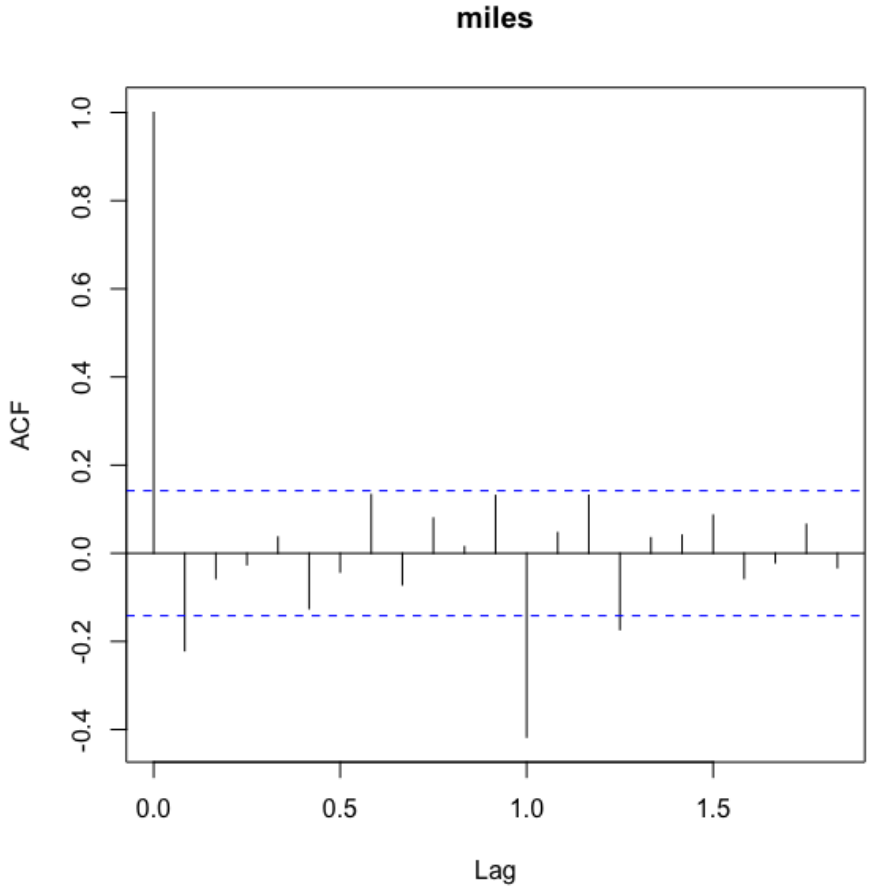
Step 7: Perform a non-seasonal differencing on seasonally differenced data. This is like an ARIMA(0,1,0)(0,1,0)$_{12}$ model.

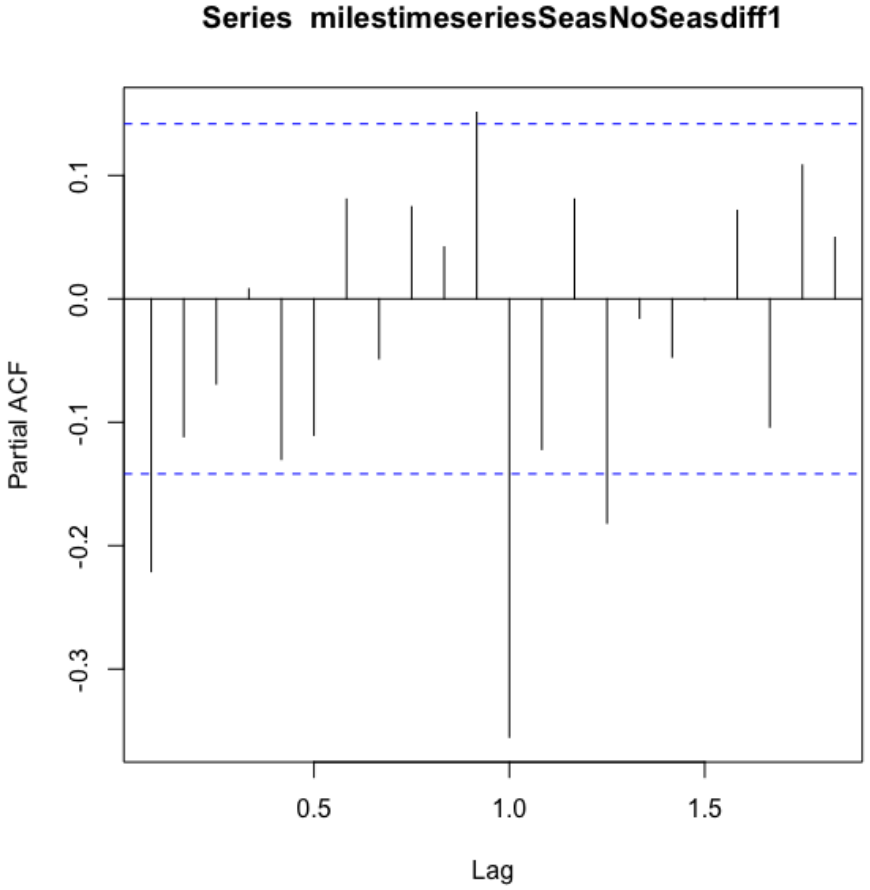


Time Series Model Building Using ARIMA

Identification Phase

Step 8: Check ACF and PACF to explore remaining dependencies.





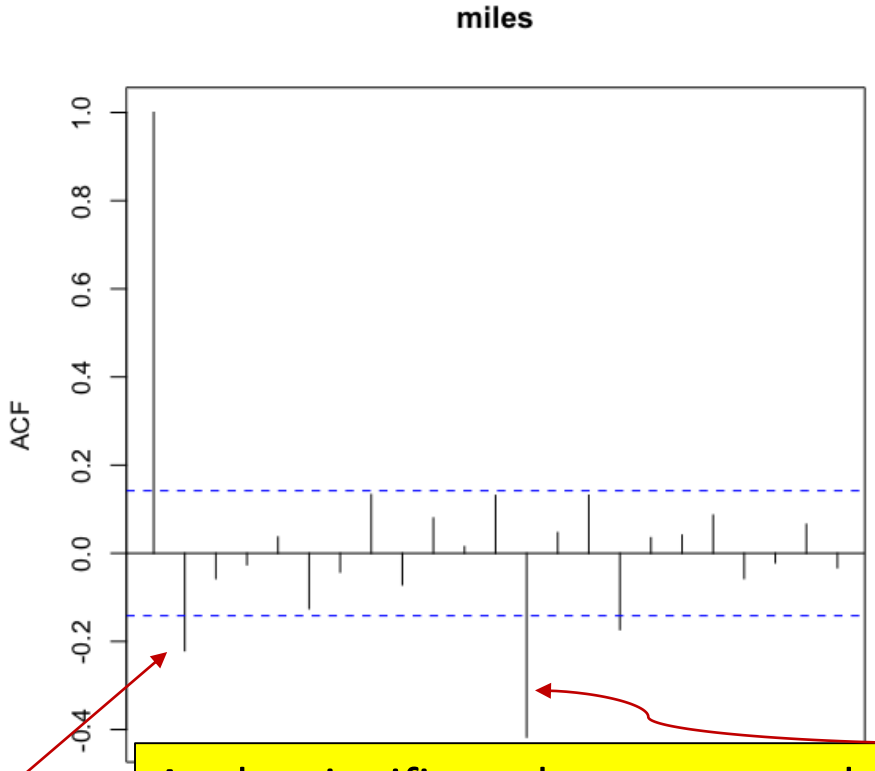
CSE 7302c

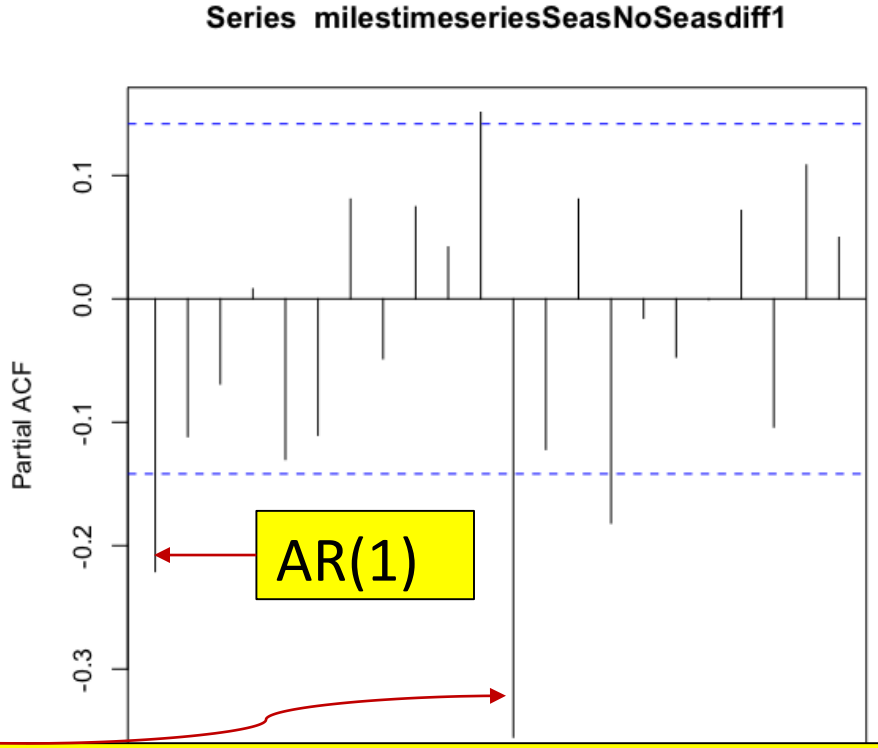
Time Series Model Building Using ARIMA

Identification Phase

Step 8: This indicates an ARIMA(1,1,1)(0,1,1)$_{12}$ model.

R code: Arima(milestimeseries, order = c(1,1,1), seasonal = c(0,1,1), include.drift = FALSE)





MA(1)

As the significant lag at seasonal period is **negative**, include a Seasonal MA(1) term. Negative autocorrelation indicates *overdifferencing* requiring MA term to fix it.

Time Series Model Building Using ARIMA

Parameter Estimation Phase

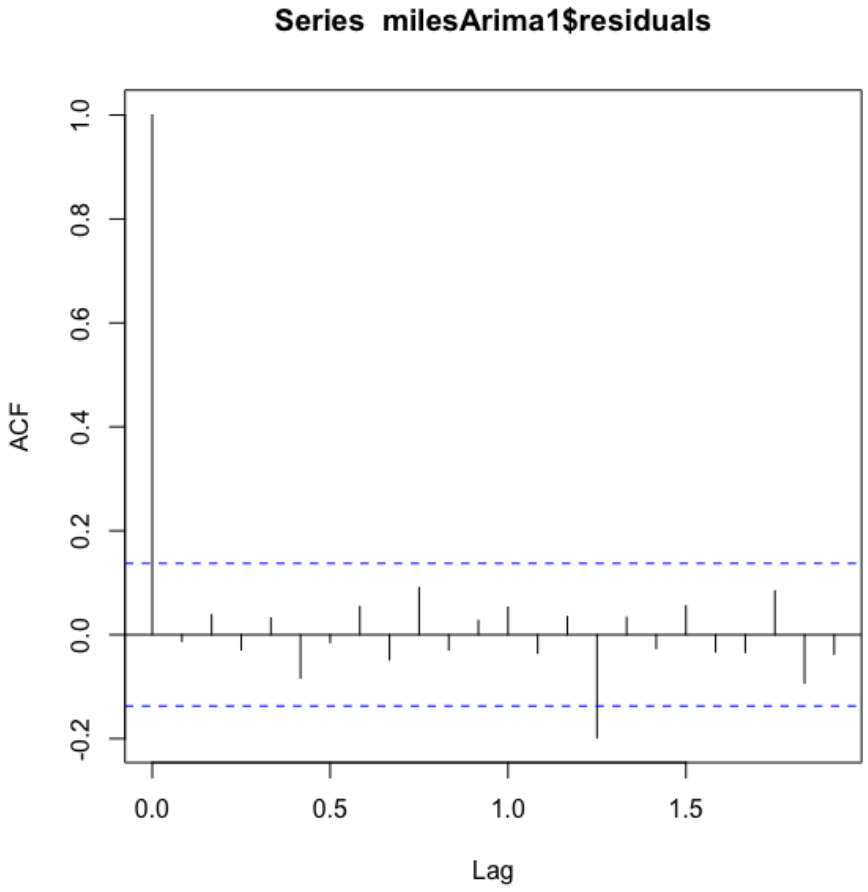
Step 9: Calculate parameters using the identified model(s). Use AIC to pick the best model.

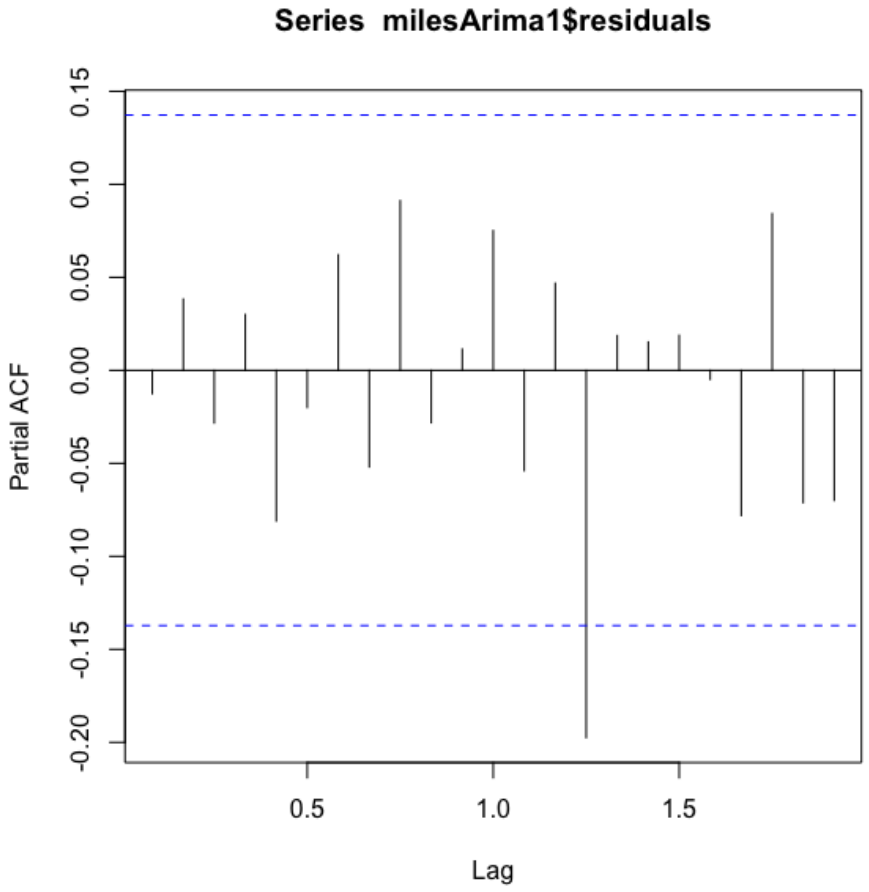
```
Series: milestimeseries
ARIMA(1,1,1)(0,1,1)[12]

Coefficients:
          ar1      ma1     sma1
      0.4501  -0.7035  -0.7393
s.e.  0.1755   0.1407   0.0641

sigma^2 estimated as 3.917e+12:  log likelihood=-3043.49
AIC=6094.99   AICc=6095.2   BIC=6107.99
```

Time Series Model Building Using ARIMA

Evaluation Phase

Step 10: Check ACF and PACF of the residuals to evaluate model.



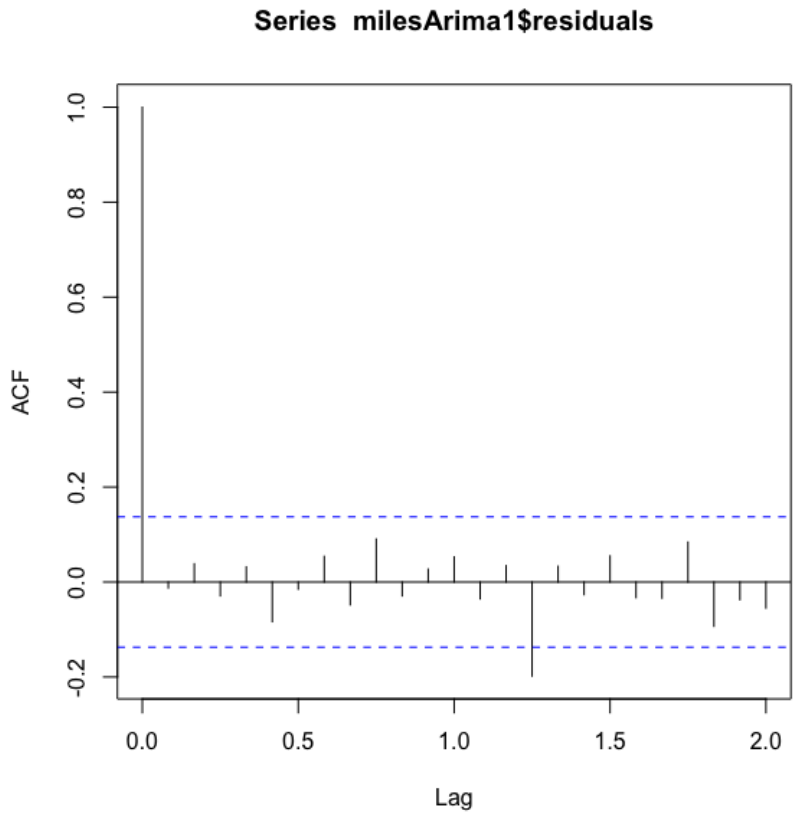


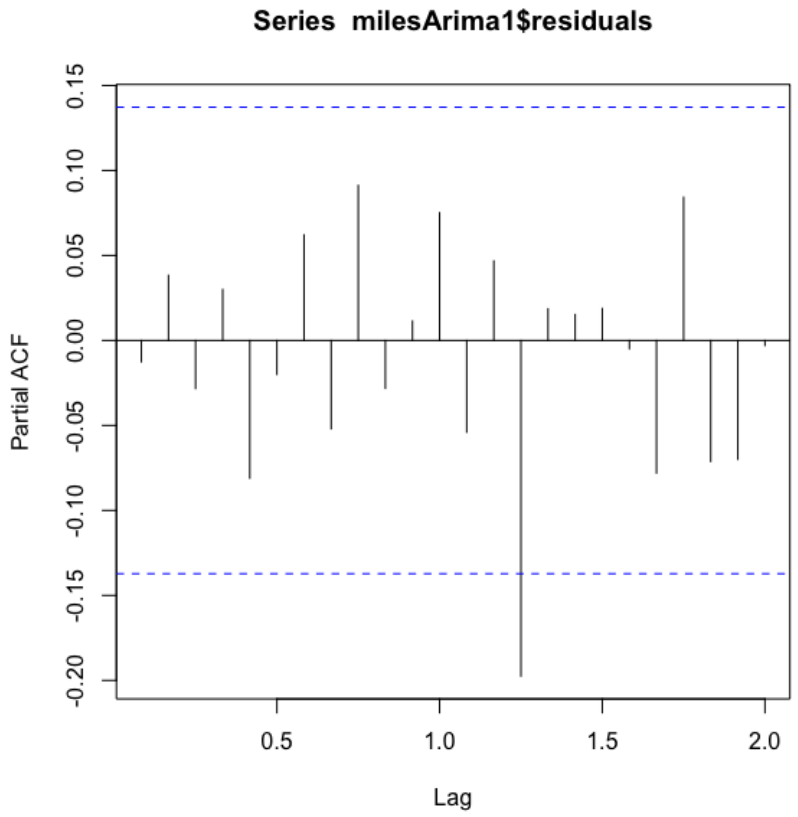
CSE 7302c

Time Series Model Building Using ARIMA

Evaluation Phase

Step 10: The residuals indicate white noise. Indicates a good model that can be used for forecasting.





Time Series Model Building Using ARIMA

Evaluation Phase

Step 10: The residuals indicate white noise. Can be checked using Ljung-Box test. R code: Box.test(milesArima1$residuals, lag=24, type="Ljung-Box")

$$Q^* = n(n+2)\sum_{k=1}^{h}\frac{{r_k}^2}{n-k}$$

h is the maximum lag being considered
n is the # of observations (length of the time series)
r_k is the autocorrelation

* For non-seasonal time series, use *h = min(10, n/5)*
For seasonal time series, use *h = min(2m, n/5)*, where m is the seasonal period

```
        Box-Ljung test

data:  milesArima1$residuals
X-squared = 21.65, df = 24, p-value = 0.6002
```

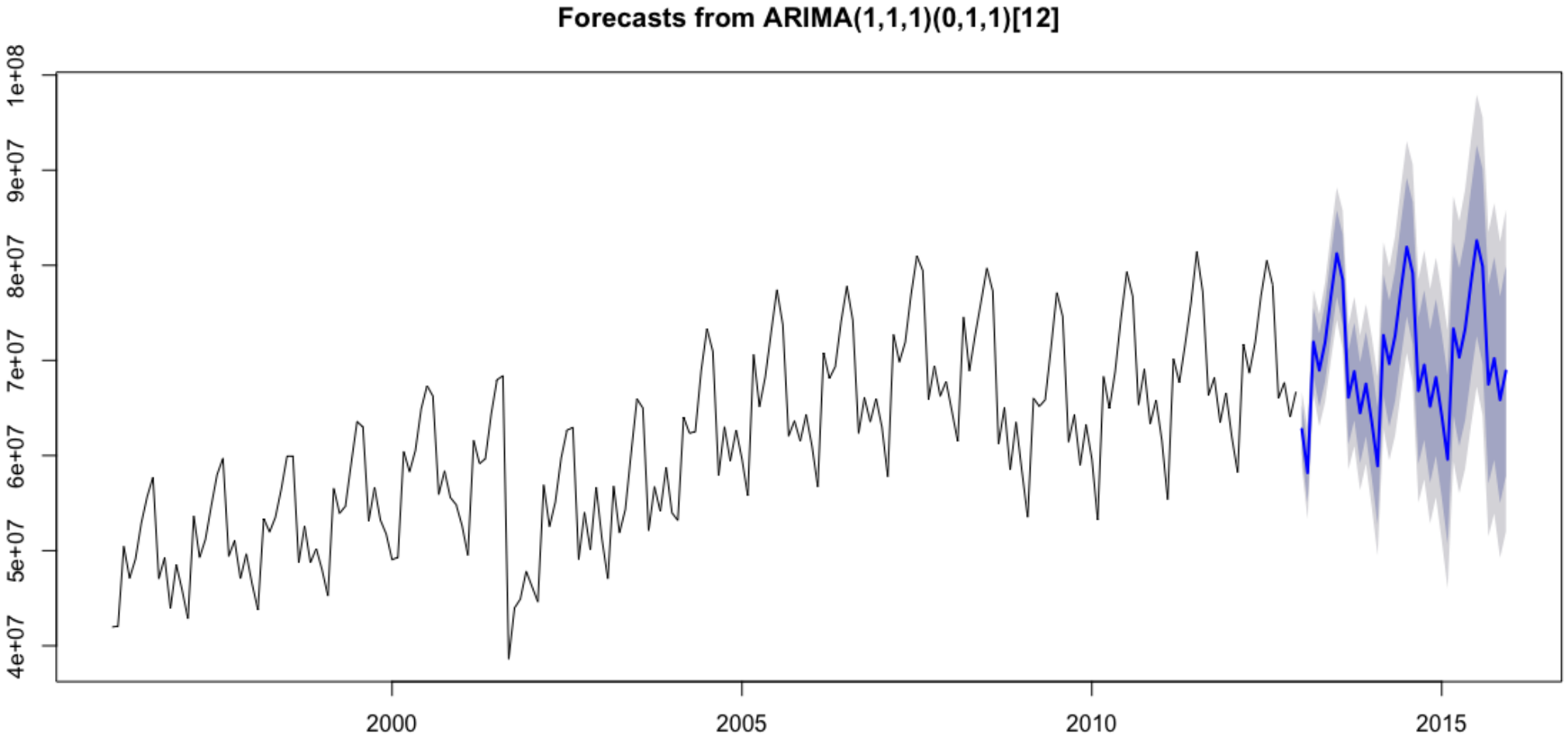
If residuals are white noise (purely random), then *Q* has a χ^2 distribution with *h-p* degrees of freedom, where *p* is the number of parameters estimated in the model

* http://robjhyndman.com/hyndsight/ljung-box-test/

Time Series Model Building Using ARIMA

Forecasting Phase

Step 11: Start forecasting. R code: forecast.Arima(milesArima1, h=36)



Time Series Model Building Using ARIMA

Alternate Steps 7-11 from previous model

Step 7: Add an AR term to seasonally differenced data. This is like an ARIMA(1,0,0)(0,1,0)$_{12}$ model.

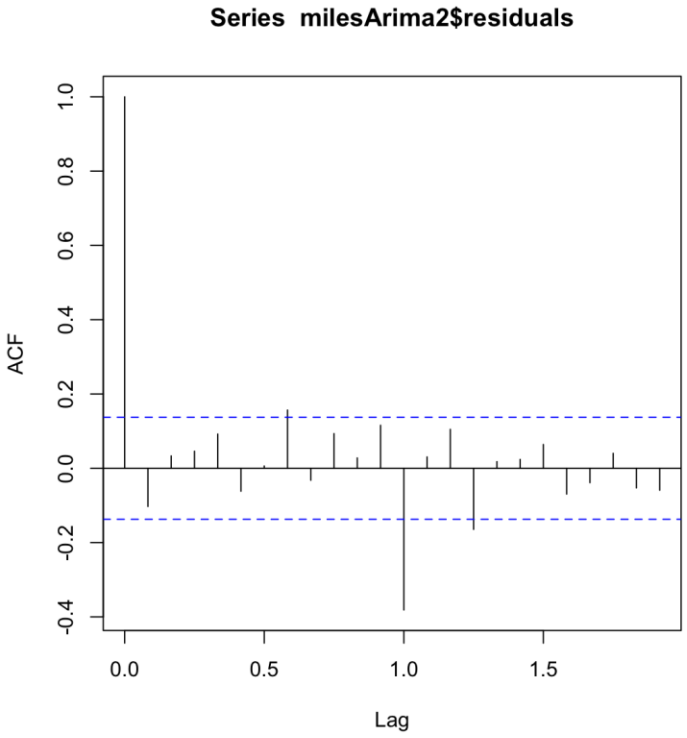
```
Series: milestimeseries
ARIMA(1,0,0)(0,1,0)[12] with drift

Coefficients:
          ar1      drift
      0.7886  109198.14
s.e.  0.0436   65385.94

sigma^2 estimated as 5.541e+12:  log likelihood=-3088.87
AIC=6183.73   AICc=6183.86   BIC=6193.5
```

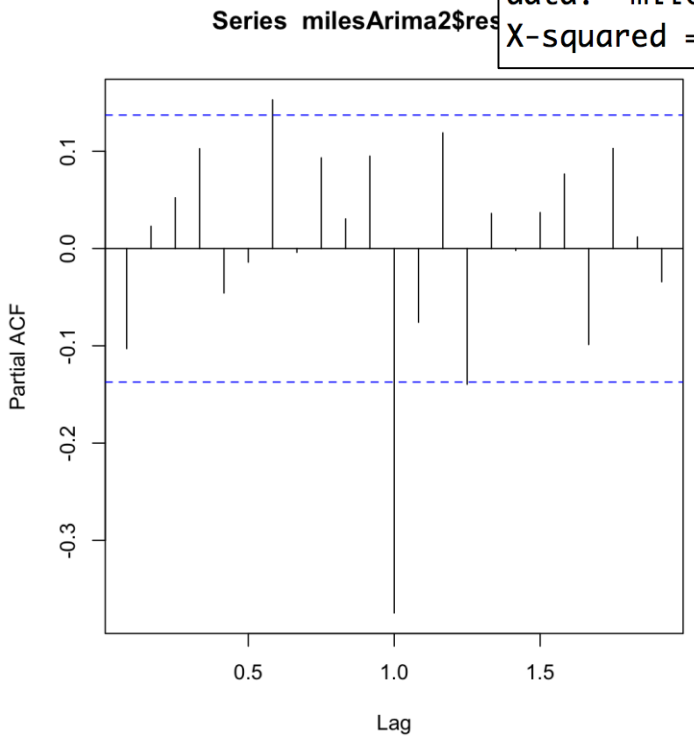
Time Series Model Building Using ARIMA

Alternate Steps 7-11 from previous model

Step 8: Check ACF and PACF to identify remaining dependencies. Strong negative autocorrelation at the seasonal period indicates need for a seasonal MA term for an ARIMA(1,0,0)(0,1,1)$_{12}$ model.

```
        Box-Ljung test

data:  milesArima2$residuals
X-squared = 61.725, df = 24, p-value = 3.632e-05
```





CSE 7302c

Time Series Model Building Using ARIMA

Alternate Steps 7-11 from previous model

Step 9: ARIMA(1,0,0)(0,1,1)$_{12}$ model.

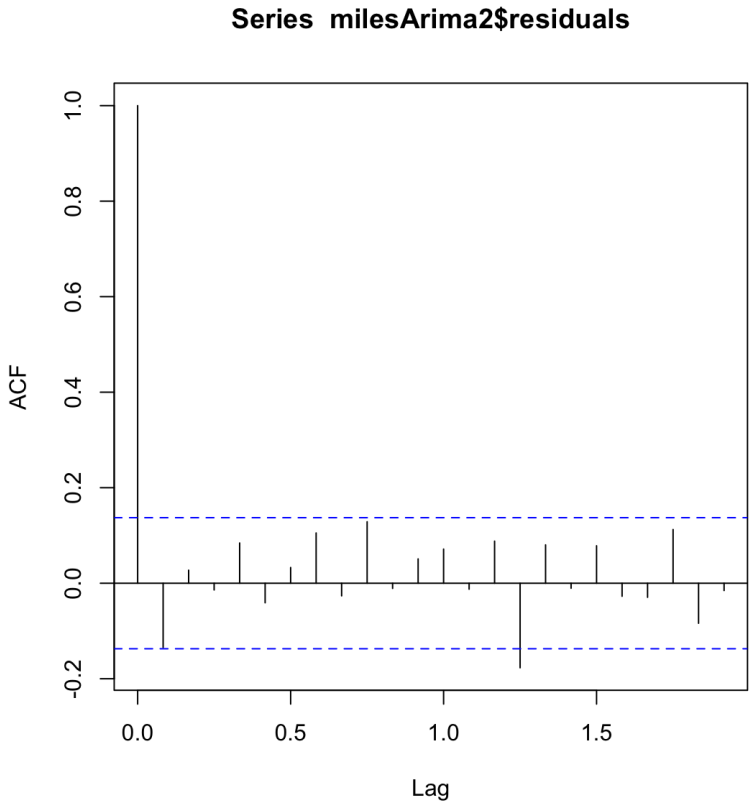
```
Series: milestimeseries
ARIMA(1,0,0)(0,1,1)[12] with drift

Coefficients:
          ar1     sma1      drift
      0.8527  -0.7254  109302.05
s.e.  0.0378   0.0678   25762.84

sigma^2 estimated as 3.923e+12:  log likelihood=-3059.73
AIC=6127.46   AICc=6127.67   BIC=6140.49
```

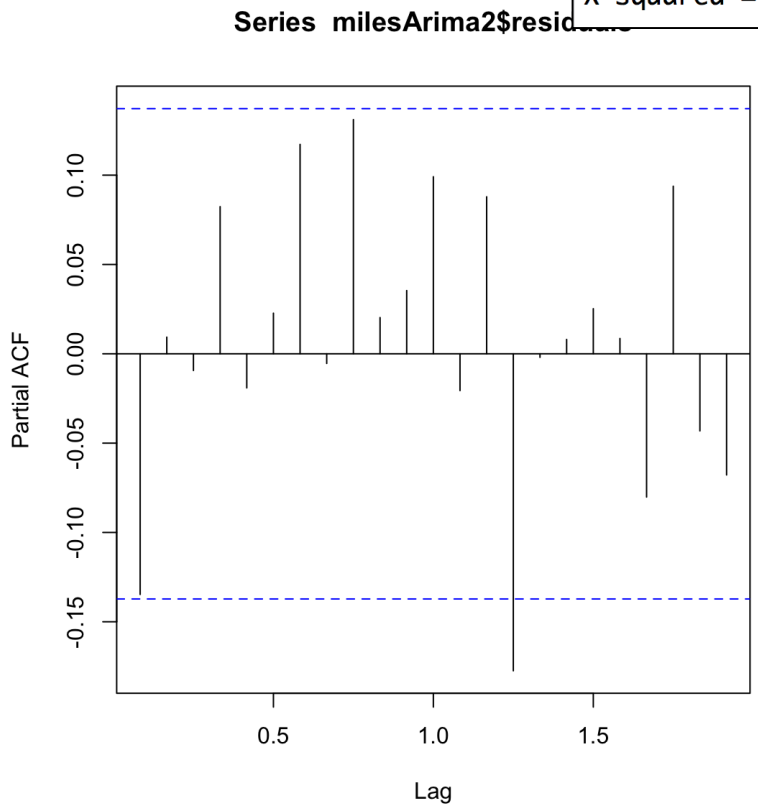
Time Series Model Building Using ARIMA

Alternate Steps 7-11 from previous model

Step 10: The residuals indicate white noise. Indicates a good model that can be used for forecasting.

```
        Box-Ljung test

data:  milesArima2$residuals
X-squared = 30.776, df = 24, p-value = 0.1603
```



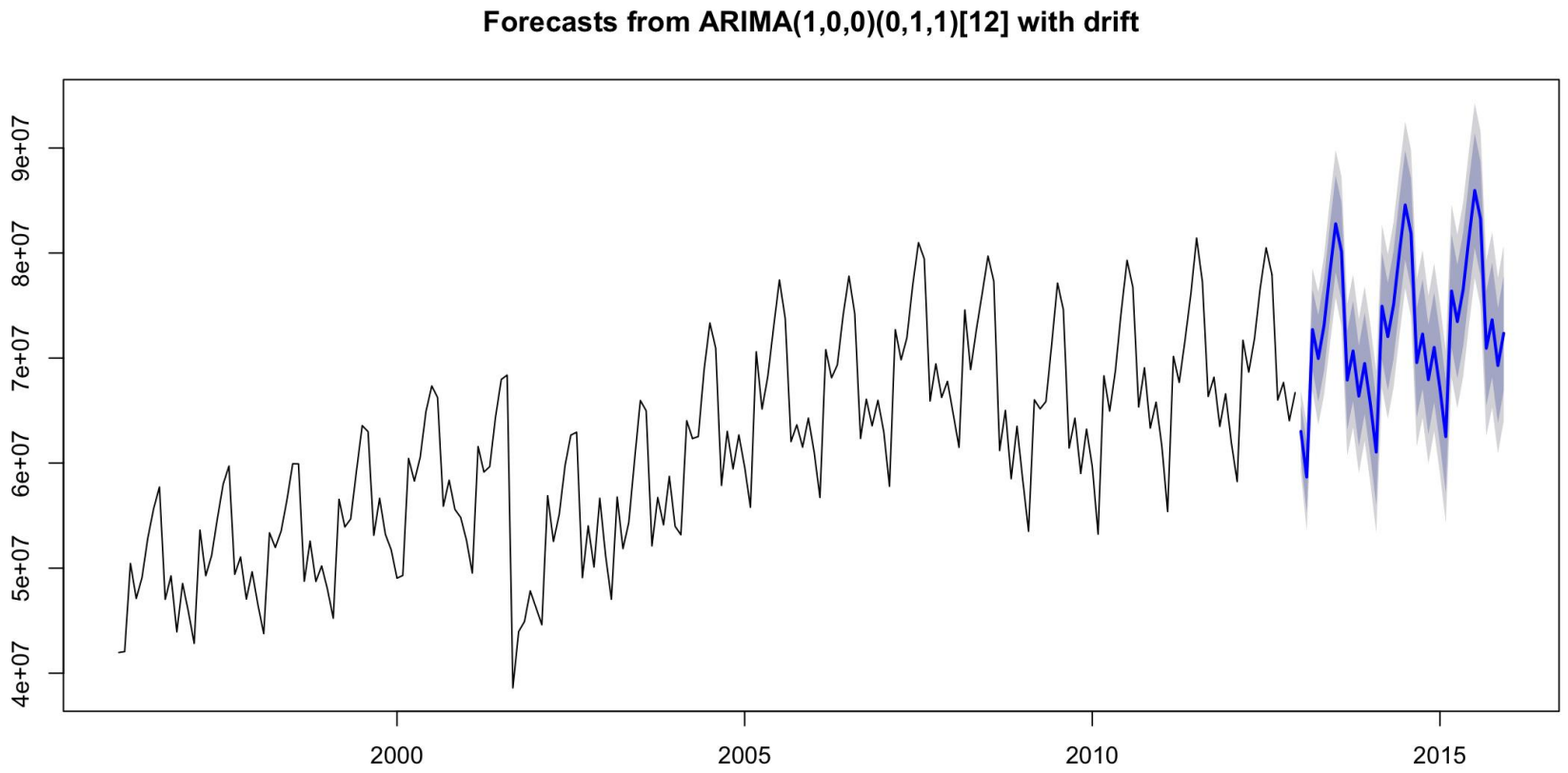


CSE 7302c

Time Series Model Building Using ARIMA

Alternate Steps 7-11 from previous model

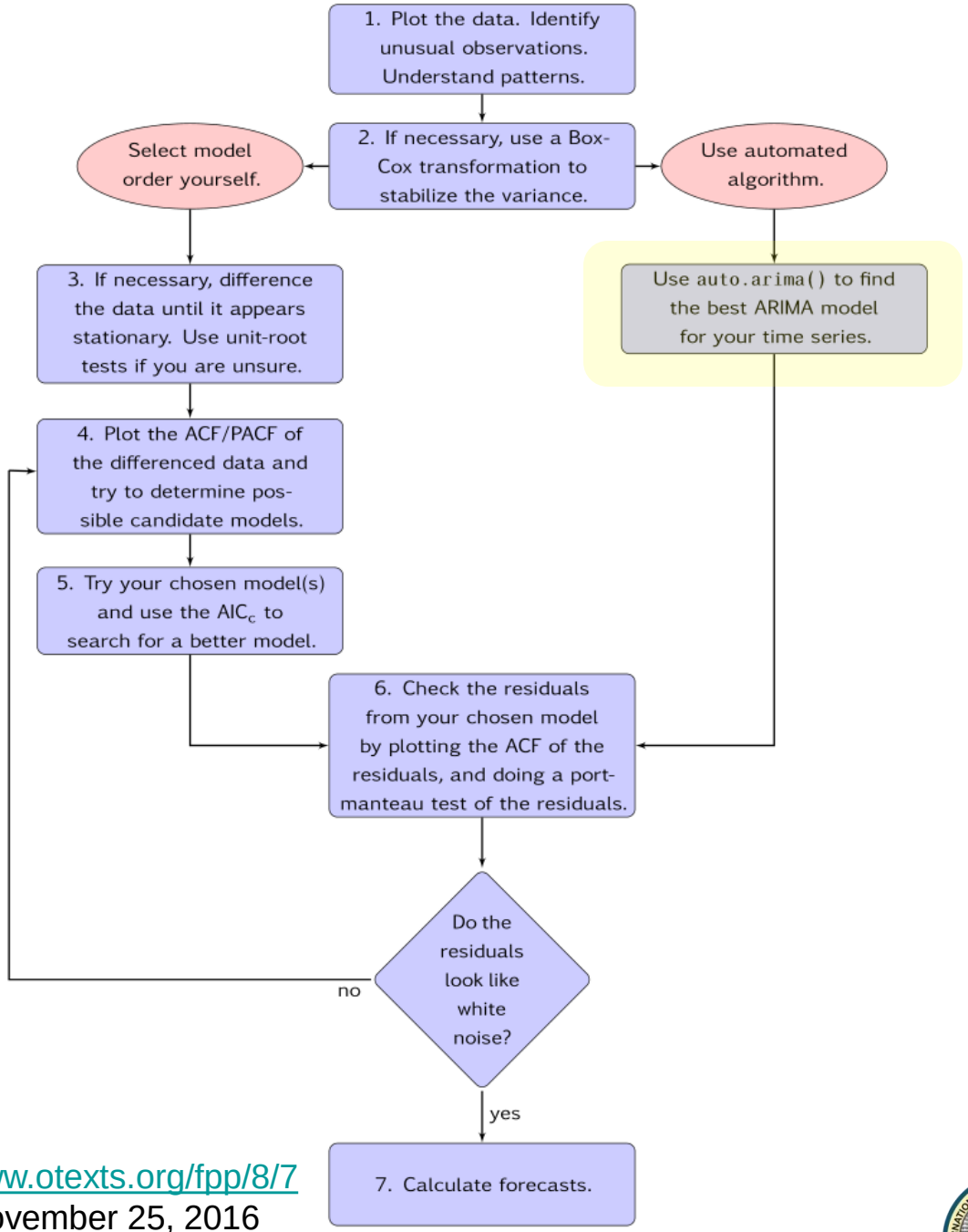
Step 11: Start forecasting.



CSE 7302c

Model Selection in Practice

There are techniques that automate model selection



Source: https://www.otexts.org/fpp/8/7
Last accessed: November 25, 2016

Model Selection in Practice

auto.arima() in R using Hyndman-Khandakar algorithm.

1. Find *d* for stationarity.
2. *p* and *q* are selected using a stepwise search to minimize AICc on the differenced data.
   a. The best model among the following is selected as the current model: ARIMA(2,*d*,2), ARIMA(0,*d*,0), ARIMA(1,*d*,0), ARIMA(0,*d*,1). If *d*=0, constant *c* is included; if $d \geq 1$, *c* is set to 0.
   b. *p* and *q* are varied by ± 1.
   c. *c* is included/excluded.
   d. 2(b) and (c) are repeated till AICc doesn't reduce further.

Source: https://www.otexts.org/fpp/8/7
Last accessed: November 25, 2016

Forecast using Auto ARIMA

R code: auto.arima(milestimeseries,ic='aic')

```
Series: milestimeseries
ARIMA(1,0,1)(0,1,1)[12] with drift

Coefficients:
         ar1      ma1     sma1      drift
      0.9092  -0.2128  -0.7257  108456.69
s.e.  0.0358   0.0873   0.0673   31546.77

sigma^2 estimated as 3.834e+12:  log likelihood=-3056.9
AIC=6123.81   AICc=6124.13   BIC=6140.09
```

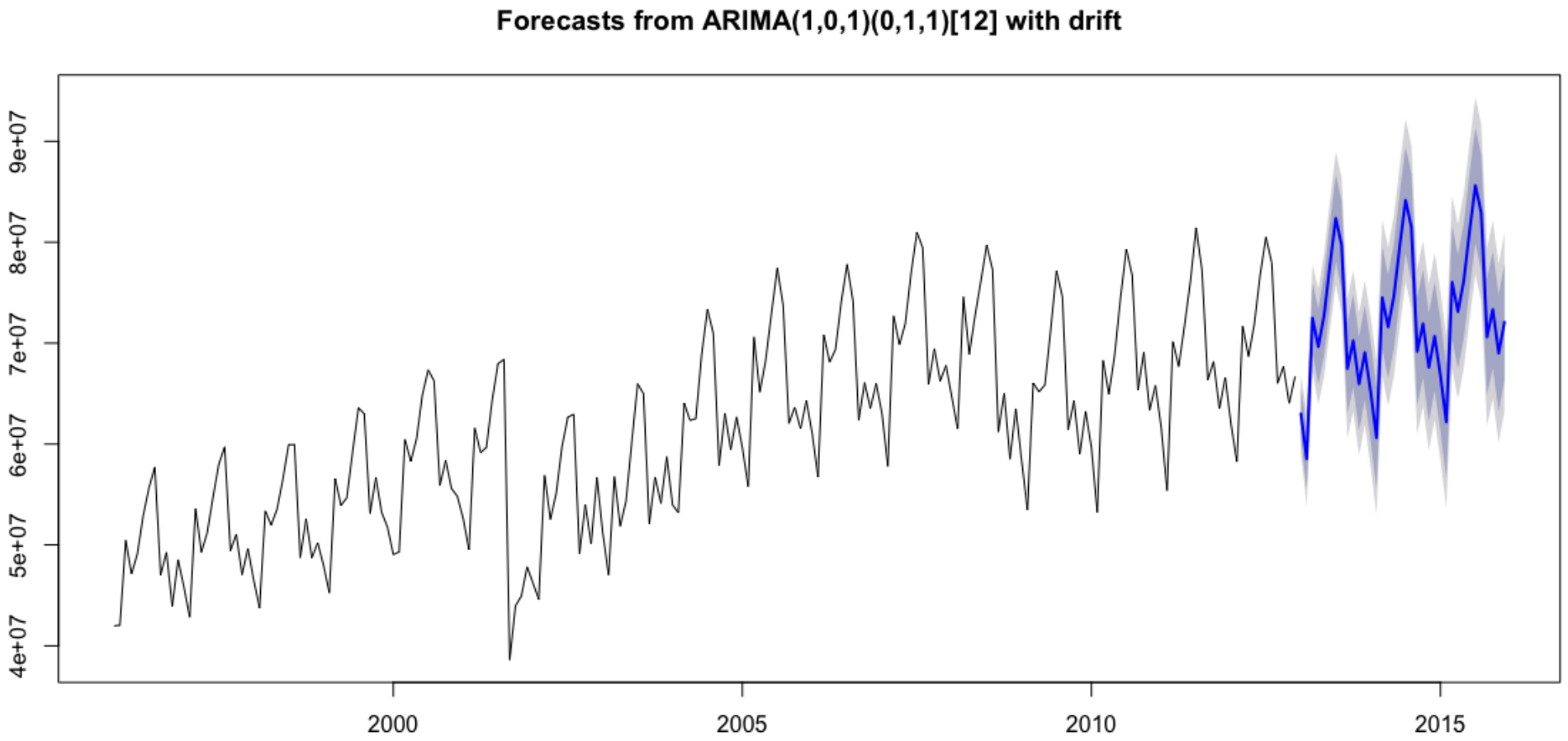
Auto ARIMA

```
Series: milestimeseries
ARIMA(1,0,0)(0,1,1)[12] with drift

Coefficients:
         ar1     sma1      drift
      0.8527  -0.7254  109302.05
s.e.  0.0378   0.0678   25762.84

sigma^2 estimated as 3.923e+12:  log likelihood=-3059.73
AIC=6127.46   AICc=6127.67   BIC=6140.49
```

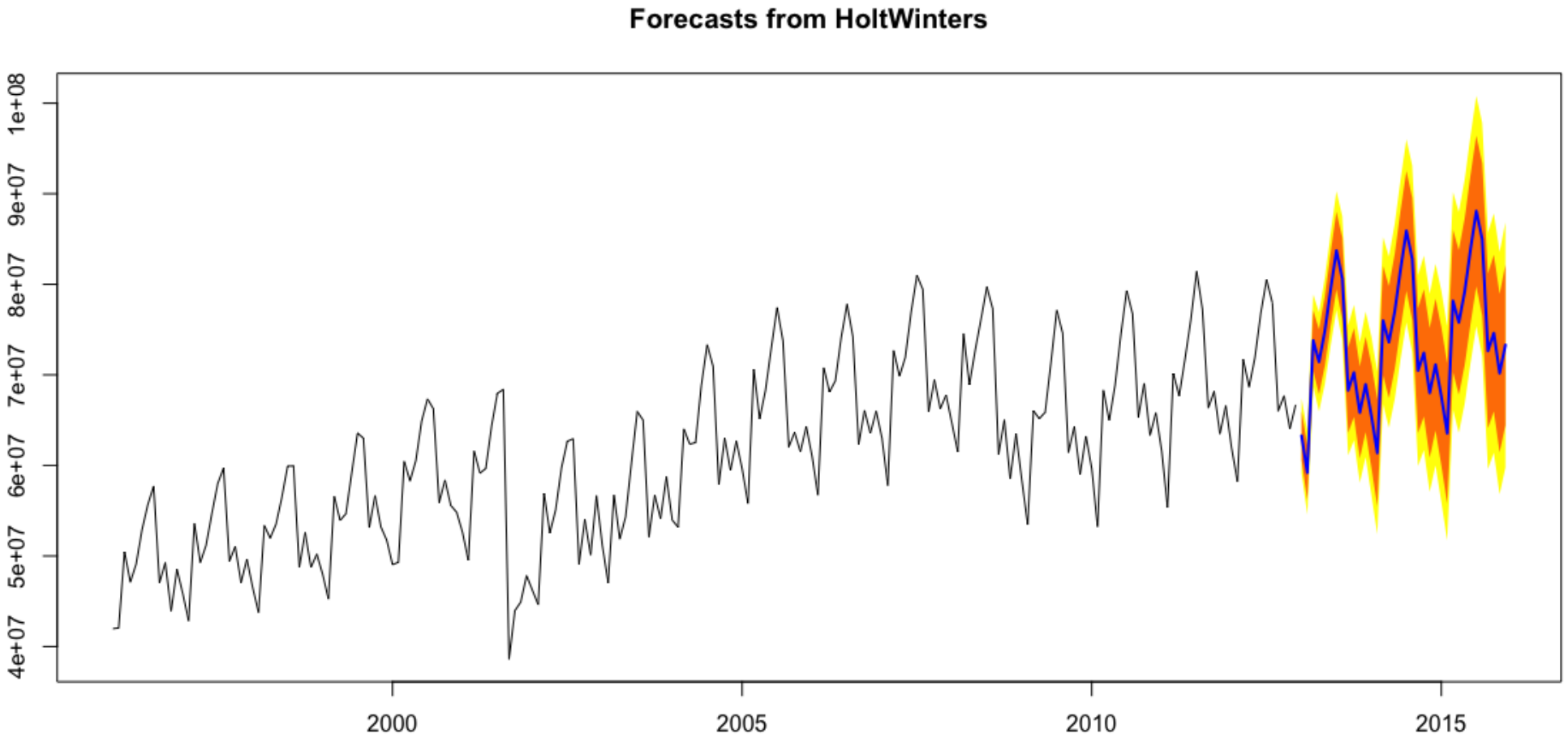
Manual ARIMA2

Forecast using Auto-ARIMA - RPM

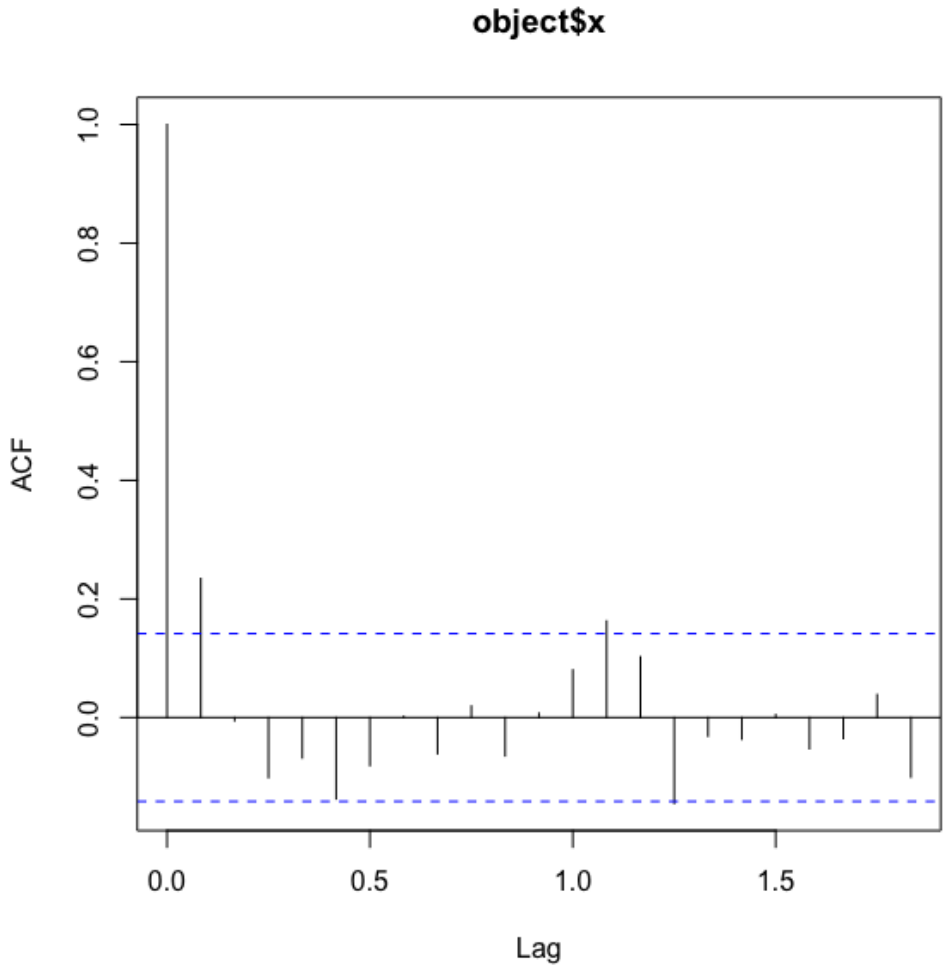
R code: forecast.Arima(milesAutoArima, h=36)

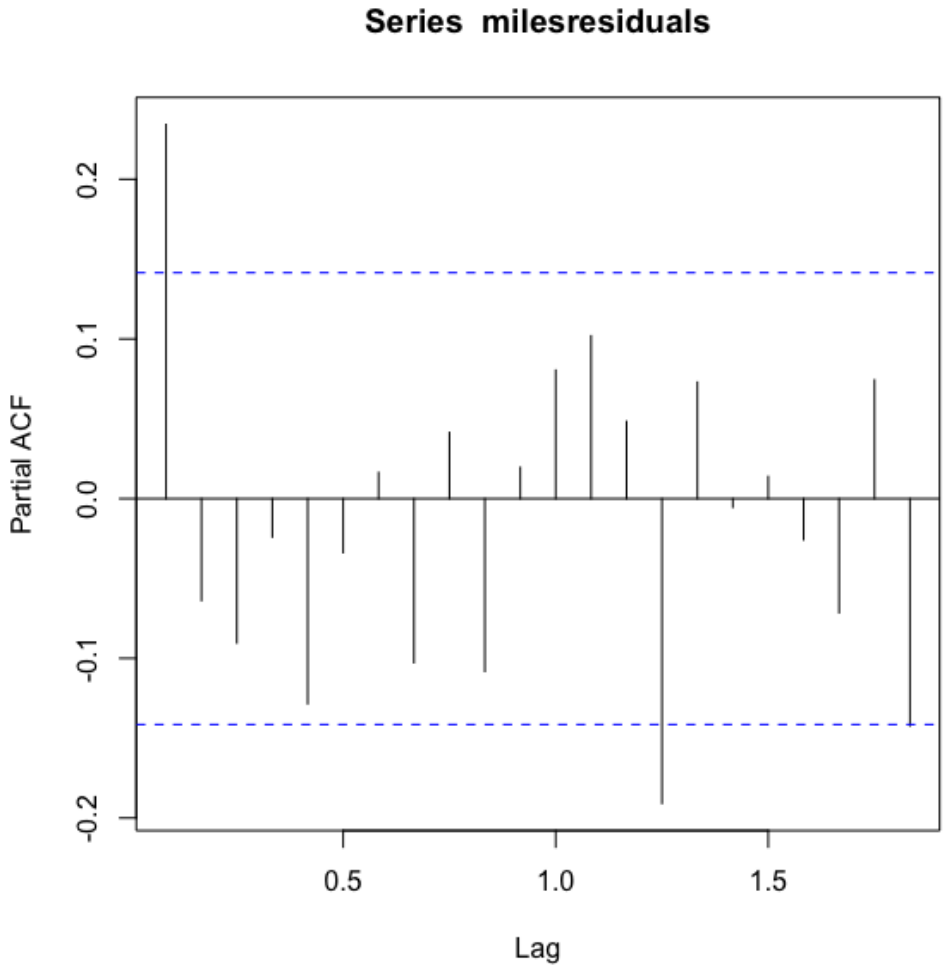


Forecast using Holt-Winters - RPM



Holt-Winters Method: Residuals





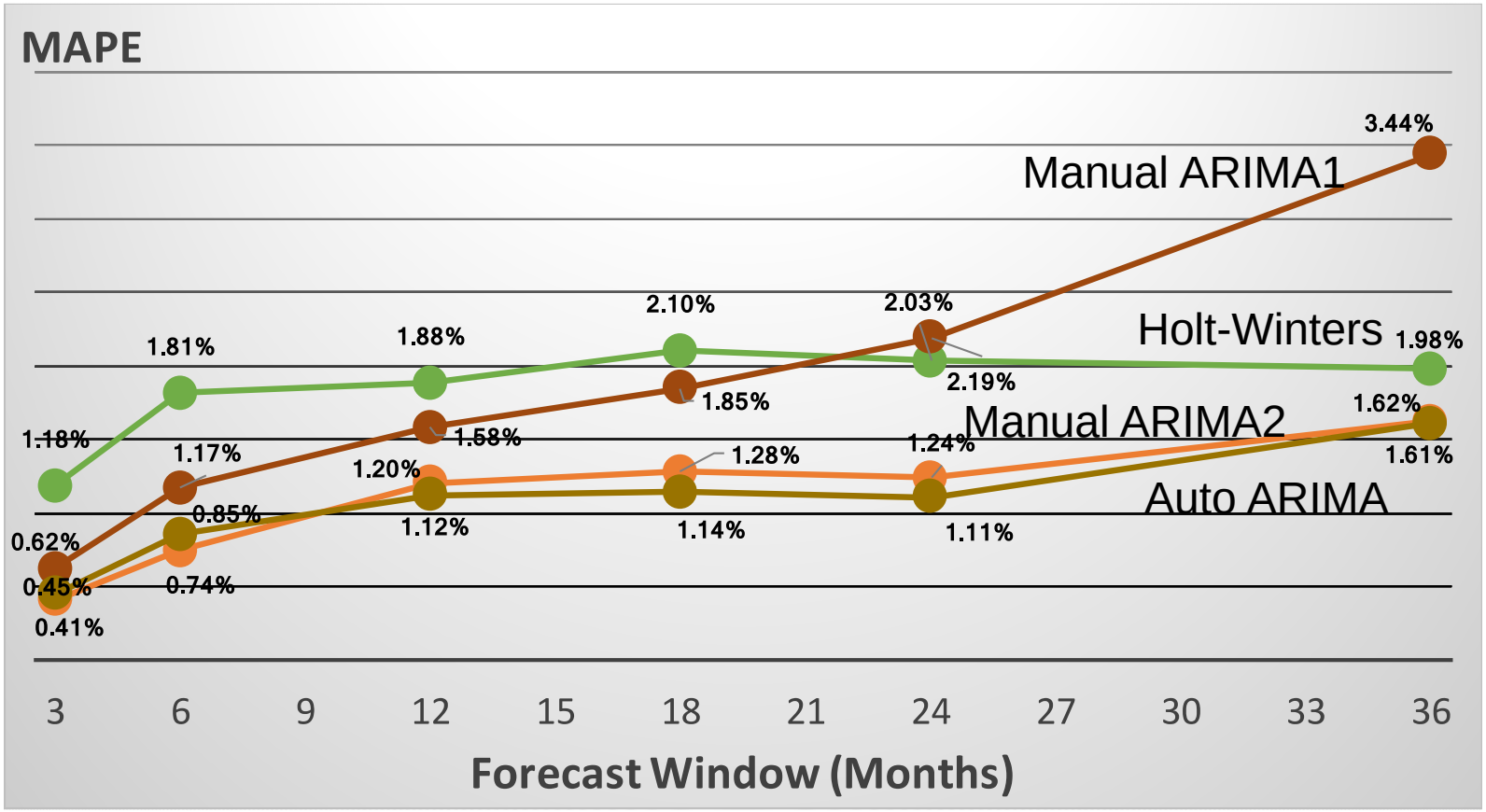
CSE 7302c

Goodness of Fit

- MAE (Mean absolute error) $\frac{\sum|y_i-\hat{y}_i|}{n}$
- MSE (Mean square error) $\frac{\sum(y_i-\hat{y}_i)^2}{n}$
- RMSE (Root mean square error) $\sqrt{\frac{\sum(y_i-\hat{y}_i)^2}{n}}$
- MAPE (Mean absolute percent error) $\frac{1}{n}\left(\frac{\sum|y_i-\hat{y}_i|}{y_i}\right)*100$

Comparing Models - MAPE

Forecast window	Holt-Winters	Manual ARIMA1	Manual ARIMA2	Auto ARIMA
3-month	1.18%	0.62%	0.41%	0.45%
6-month	1.81%	1.17%	0.74%	0.85%
12-month	1.88%	1.58%	1.20%	1.12%
18-month	2.10%	1.85%	1.28%	1.14%
24-month	2.03%	2.19%	1.24%	1.11%
36-month	1.98%	3.44%	1.62%	1.61%

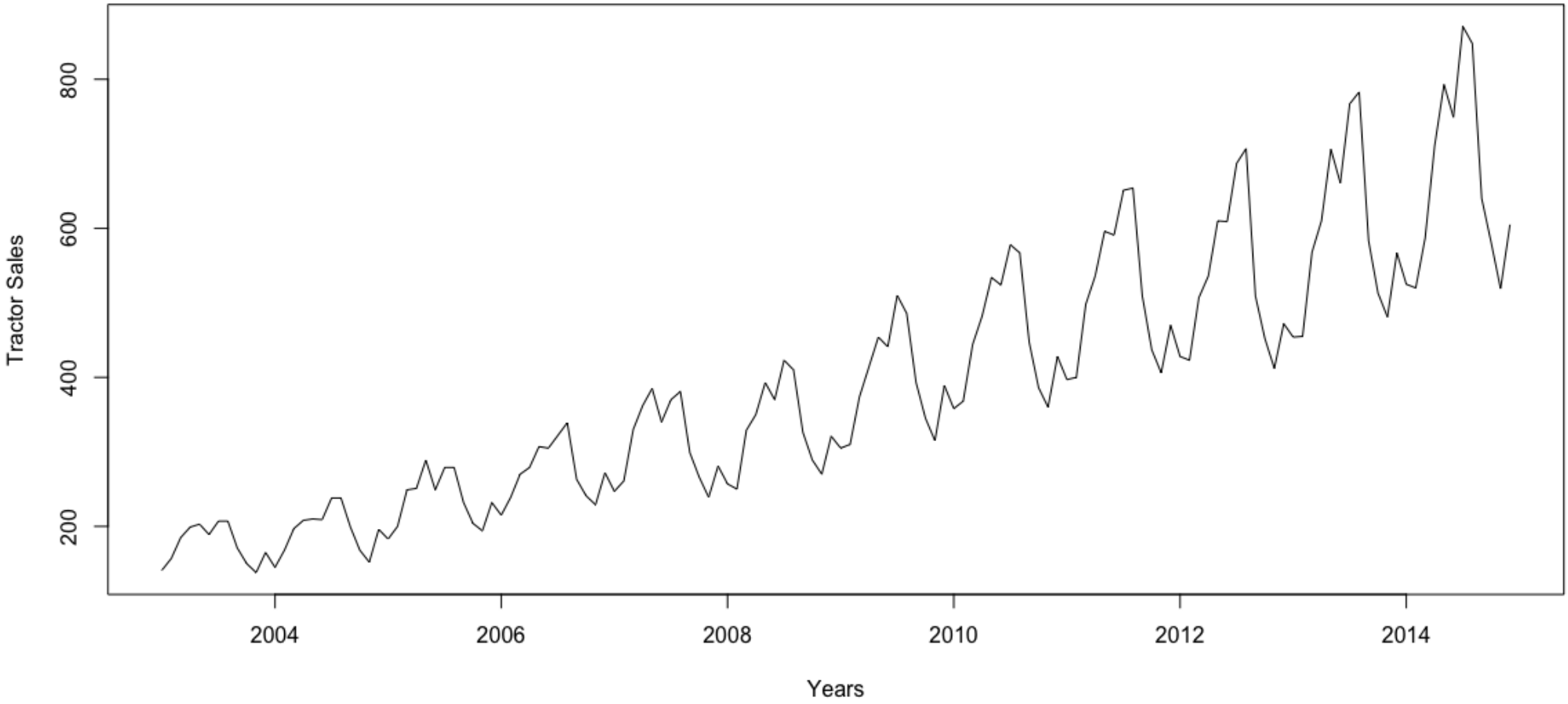


Manufacturing Case Study

FORECASTING TRACTOR SALES

Tractor Sales Case Study

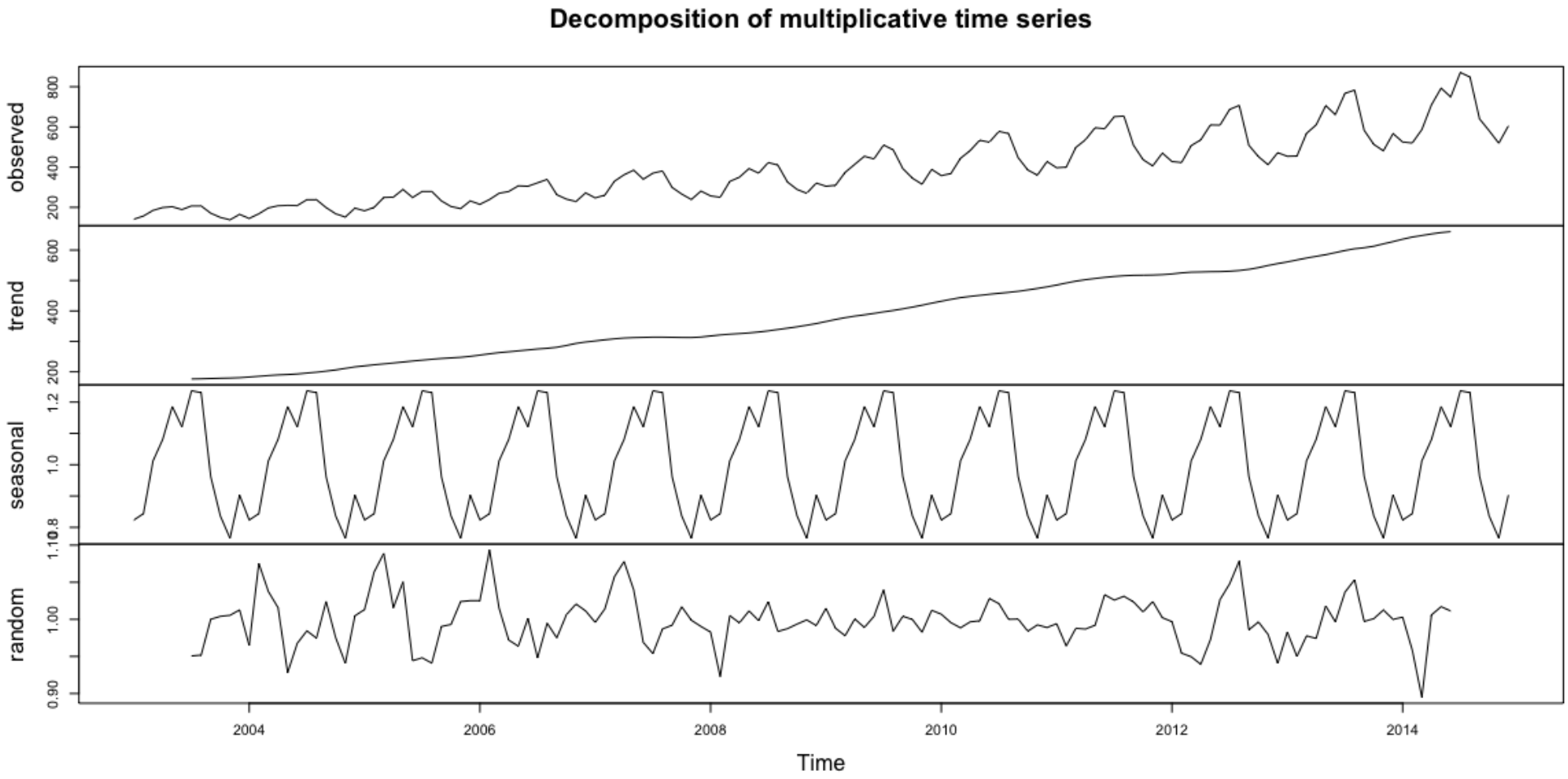
Step 1: Convert data into time series, and plot and decompose into components to understand it.



CSE 7302c

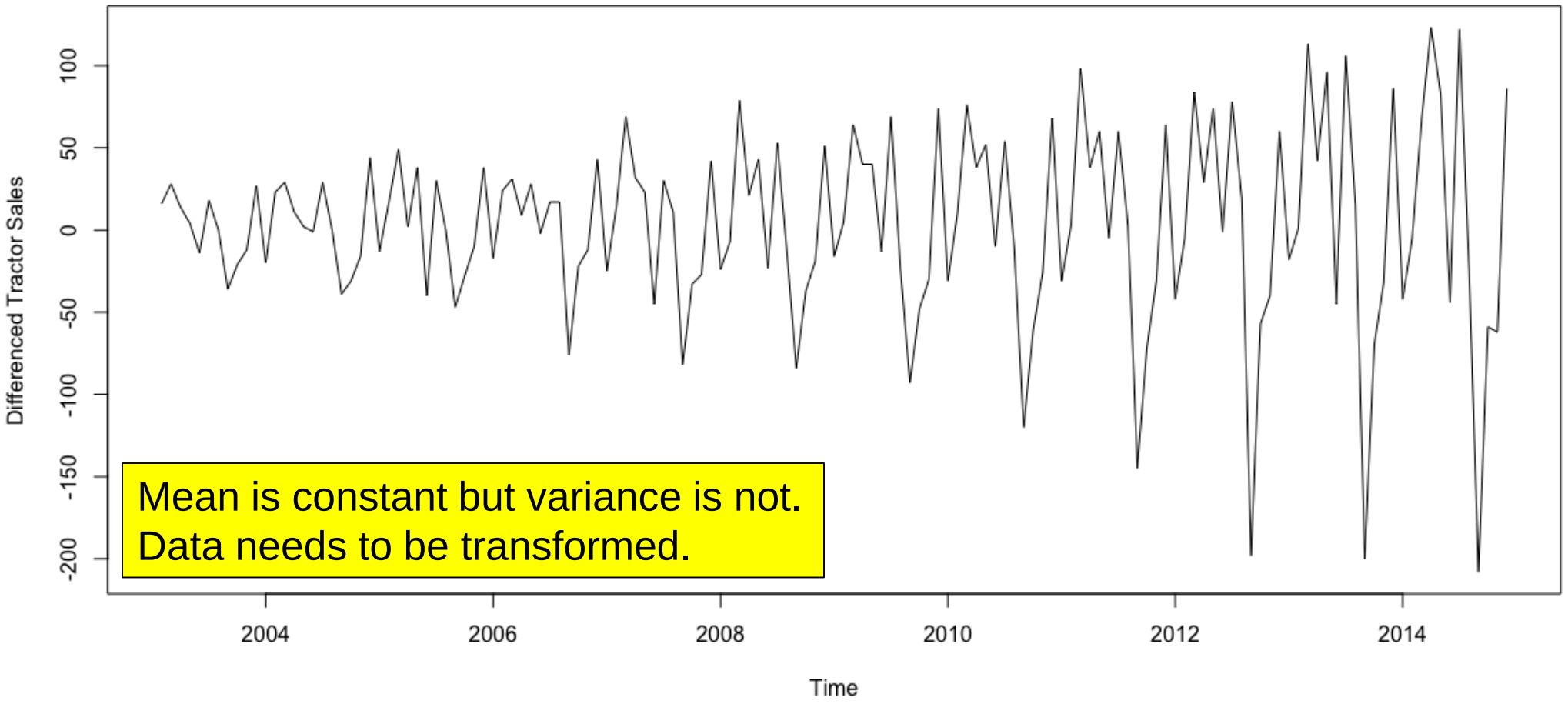
Tractor Sales Case Study

Step 1: Convert data into time series, and plot and decompose into components to understand it.



Tractor Sales Case Study

Step 2: Difference the data to make it stationary.

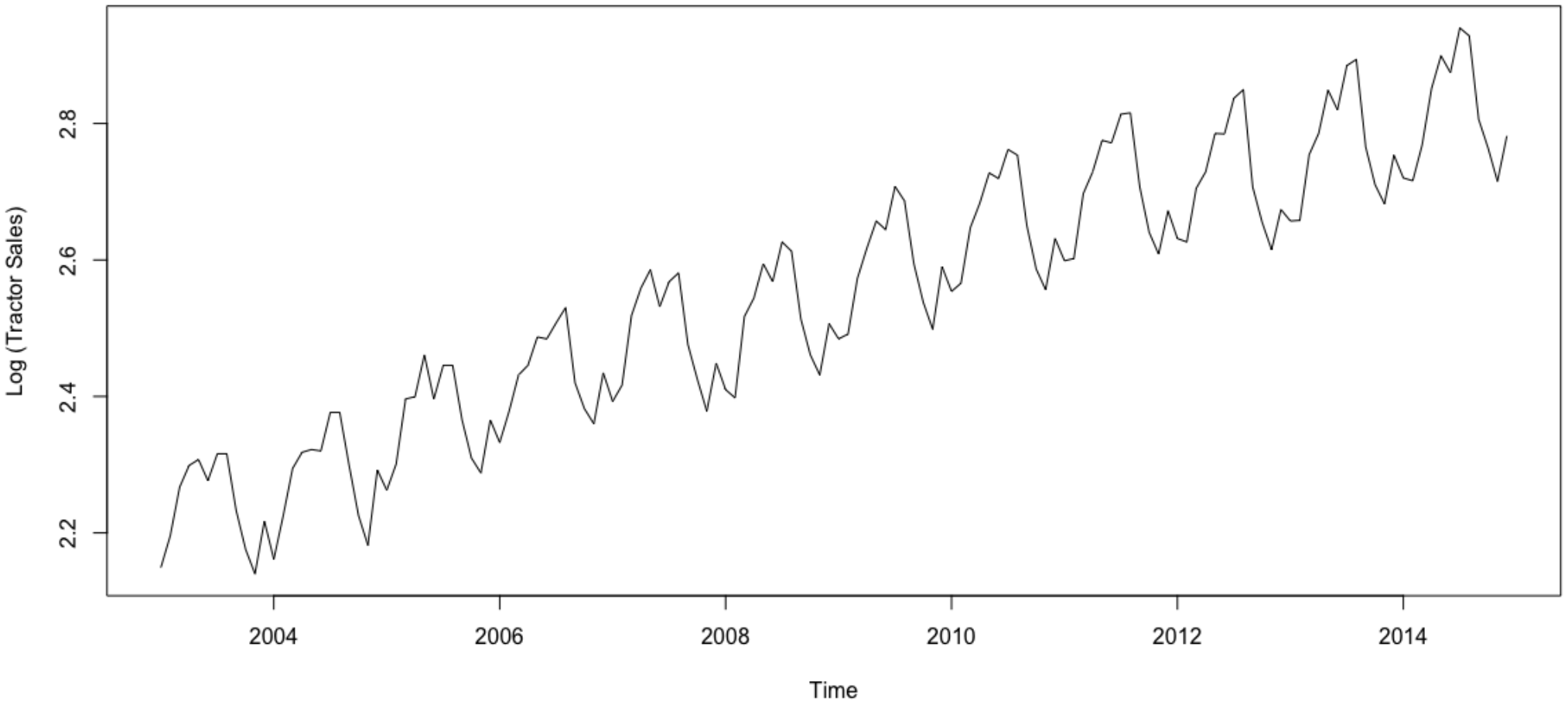


Mean is constant but variance is not.
Data needs to be transformed.

CSE 7302c

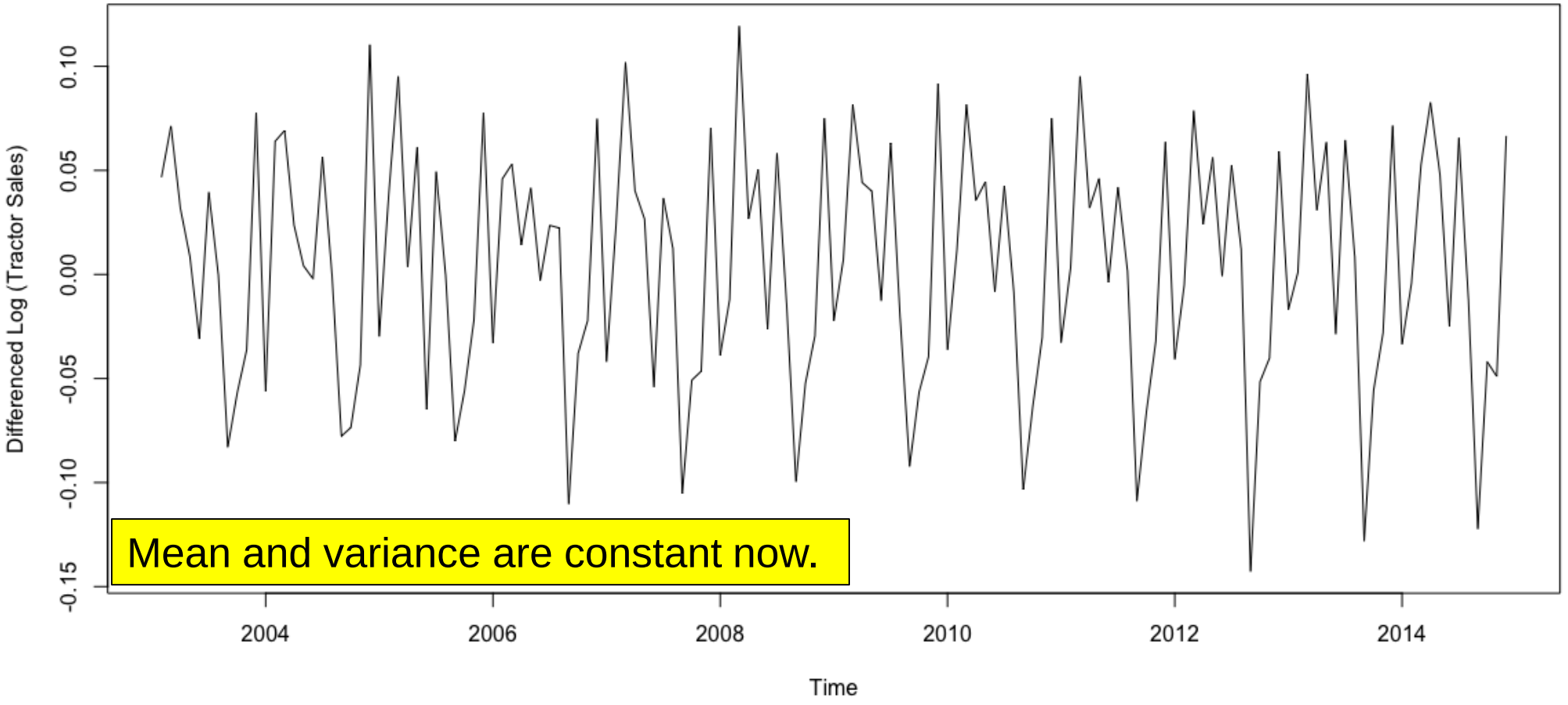
Tractor Sales Case Study

Step 3: Log transform the data.



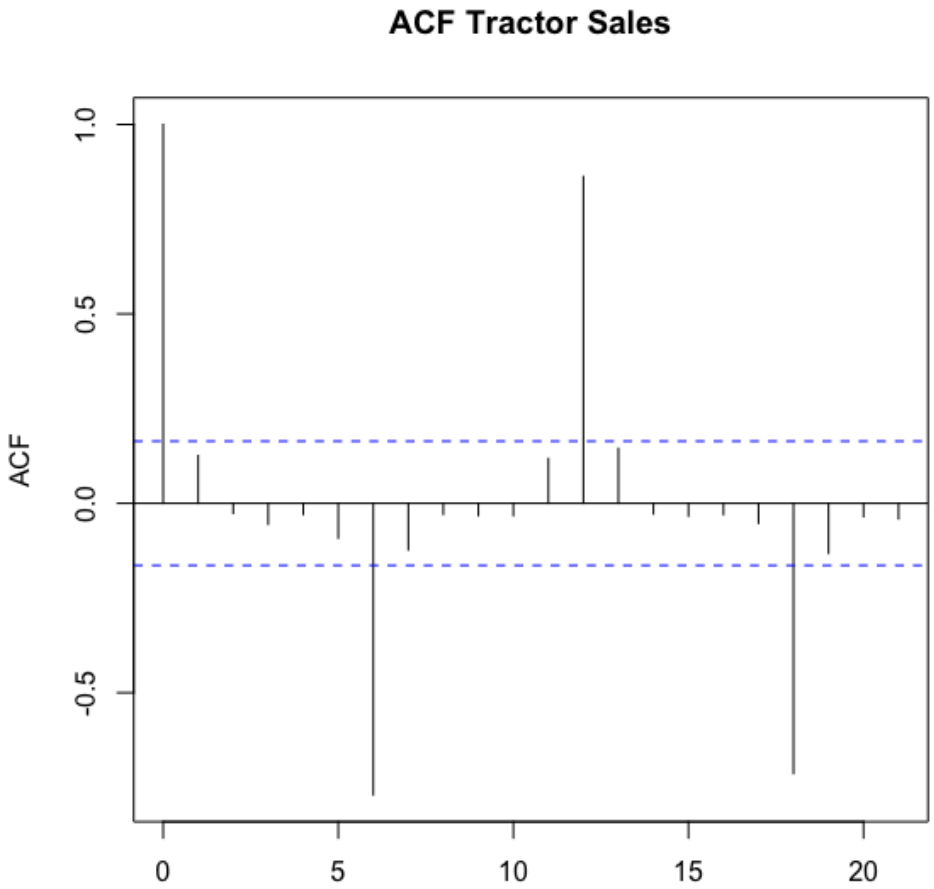
Tractor Sales Case Study

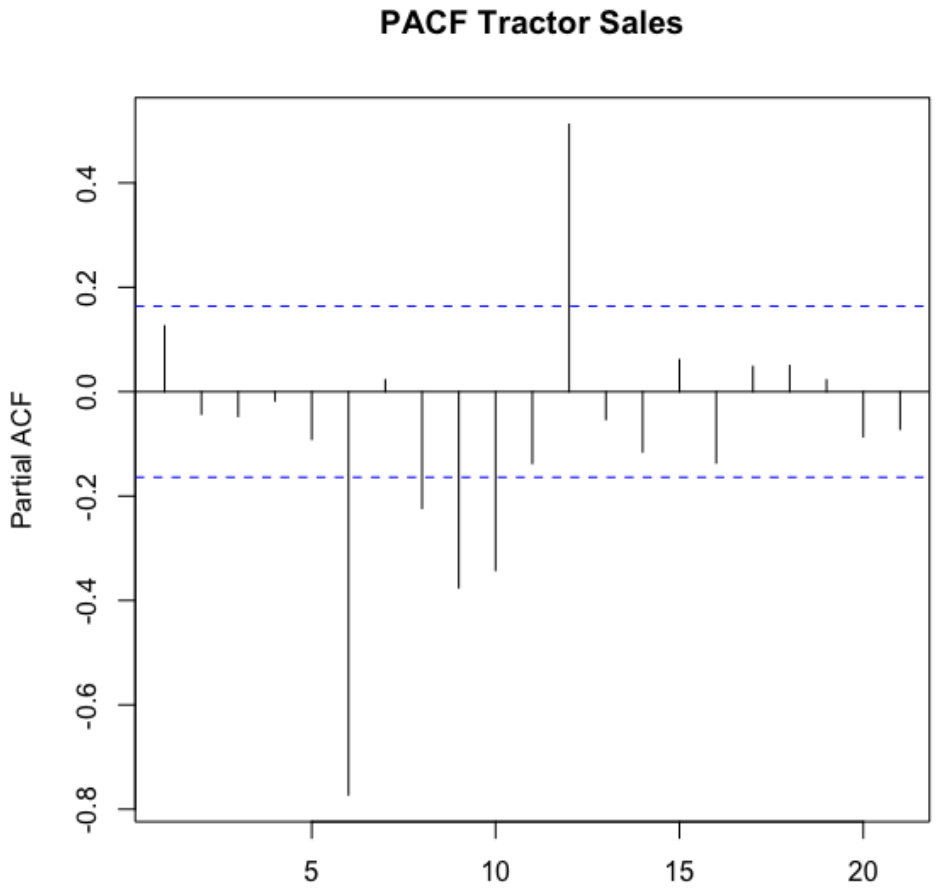
Step 4: Difference the log transformed data to check for stationarity.



Tractor Sales Case Study

Step 5: Check ACF and PACF to explore remaining dependencies.





ACF has cutoff but PACF shows some decay. This indicates an MA process. Also, there is strong seasonality.

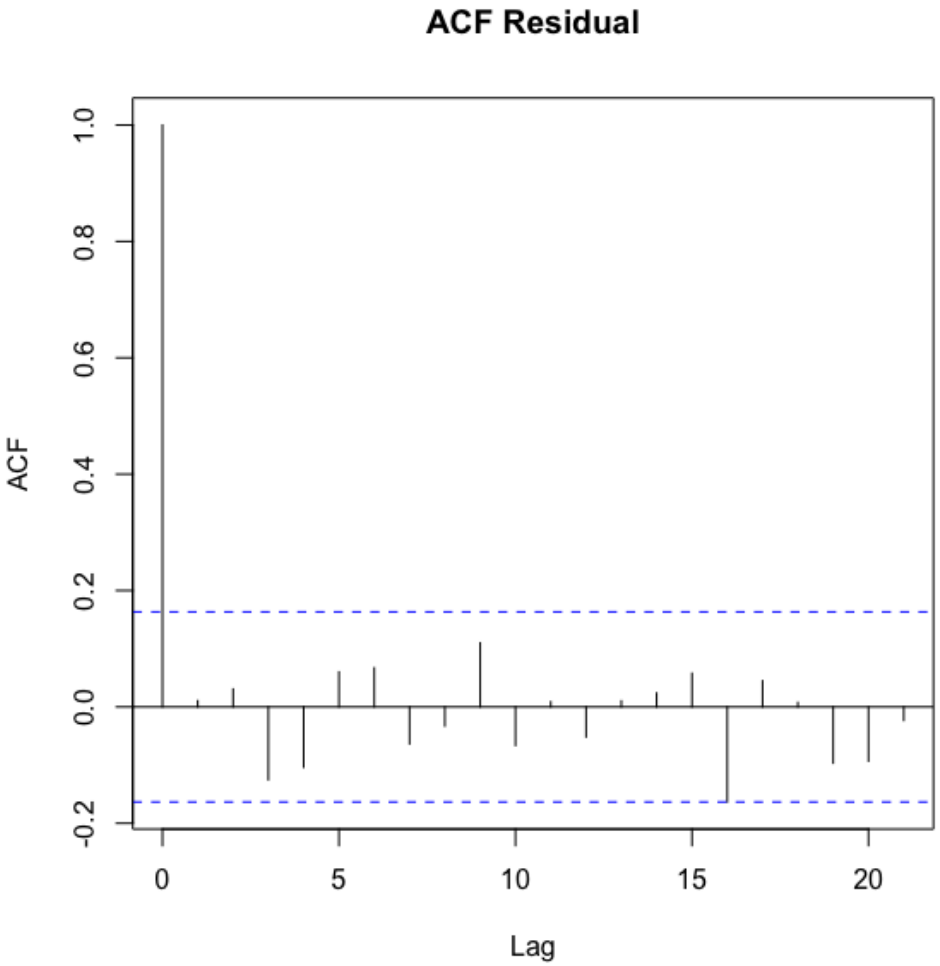
Tractor Sales Case Study

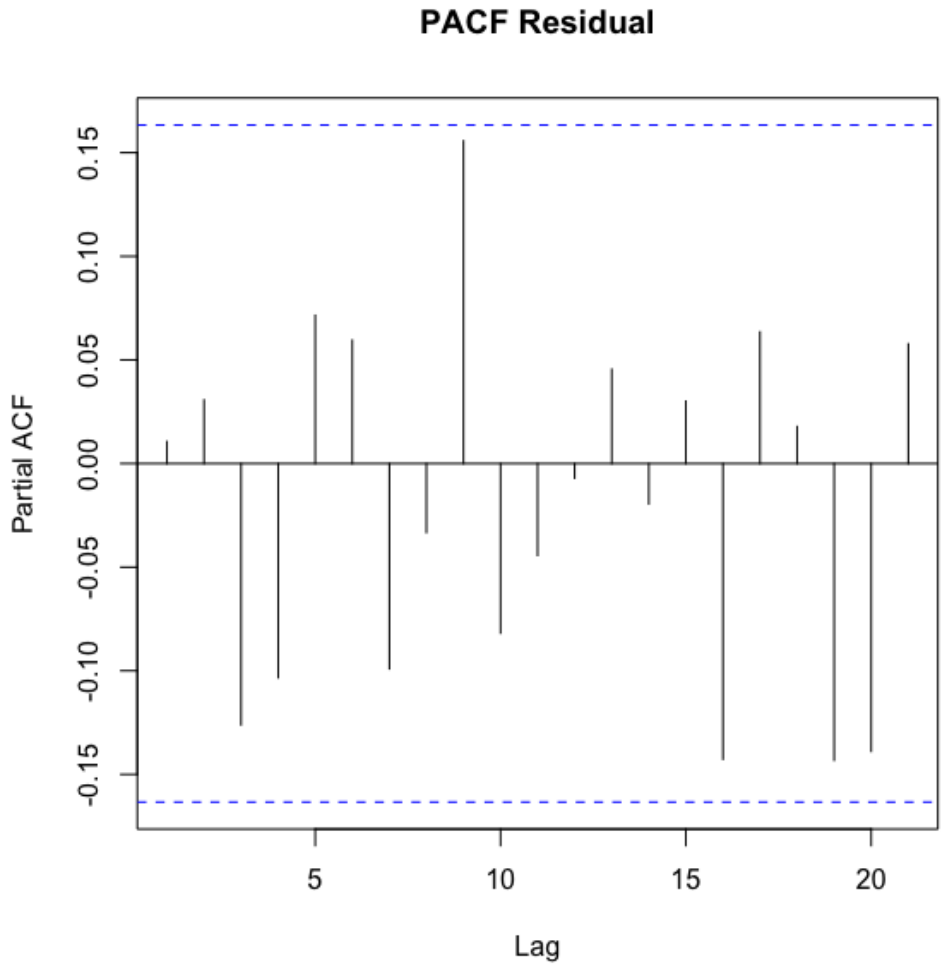
Step 6: Run Auto ARIMA.

```
Series: log10(TractorSalesTS)
ARIMA(0,1,1)(0,1,1)[12]

Coefficients:
          ma1     sma1
      -0.4047  -0.5529
s.e.   0.0885   0.0734

sigma^2 estimated as 0.0002571:  log likelihood=354.4
AIC=-702.79   AICc=-702.6   BIC=-694.17
```

Tractor Sales Case Study

Step 7: Check ACF and PACF of residuals to ensure they are white noise.





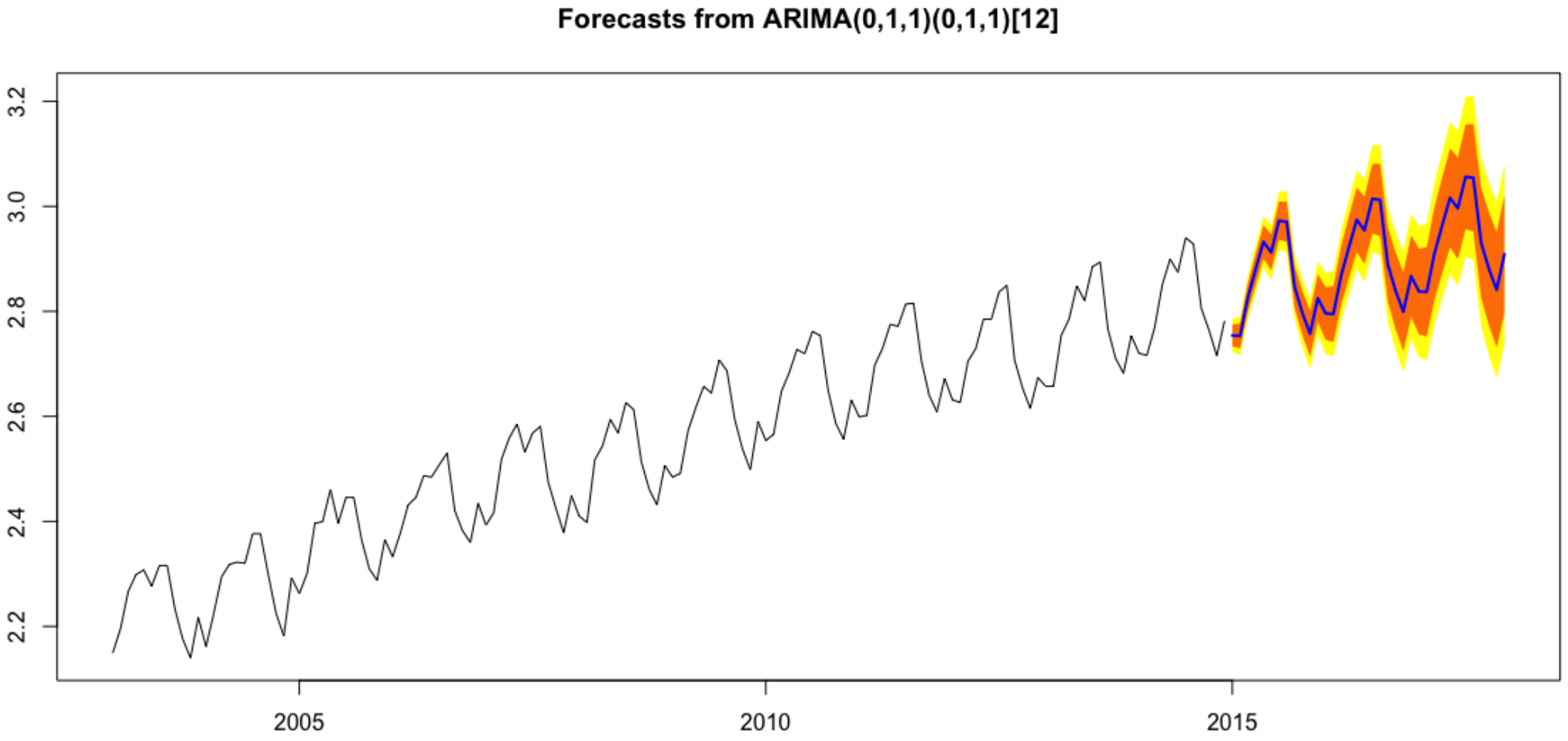
Tractor Sales Case Study

Step 7: Use Box-Ljung test to verify residuals are white noise. *It is good enough to do the verification visually on ACF and PACF.*

```
        Box-Ljung test

data:  LogTractorSalesARIMA$residuals
X-squared = 26.219, df = 24, p-value = 0.3422
```

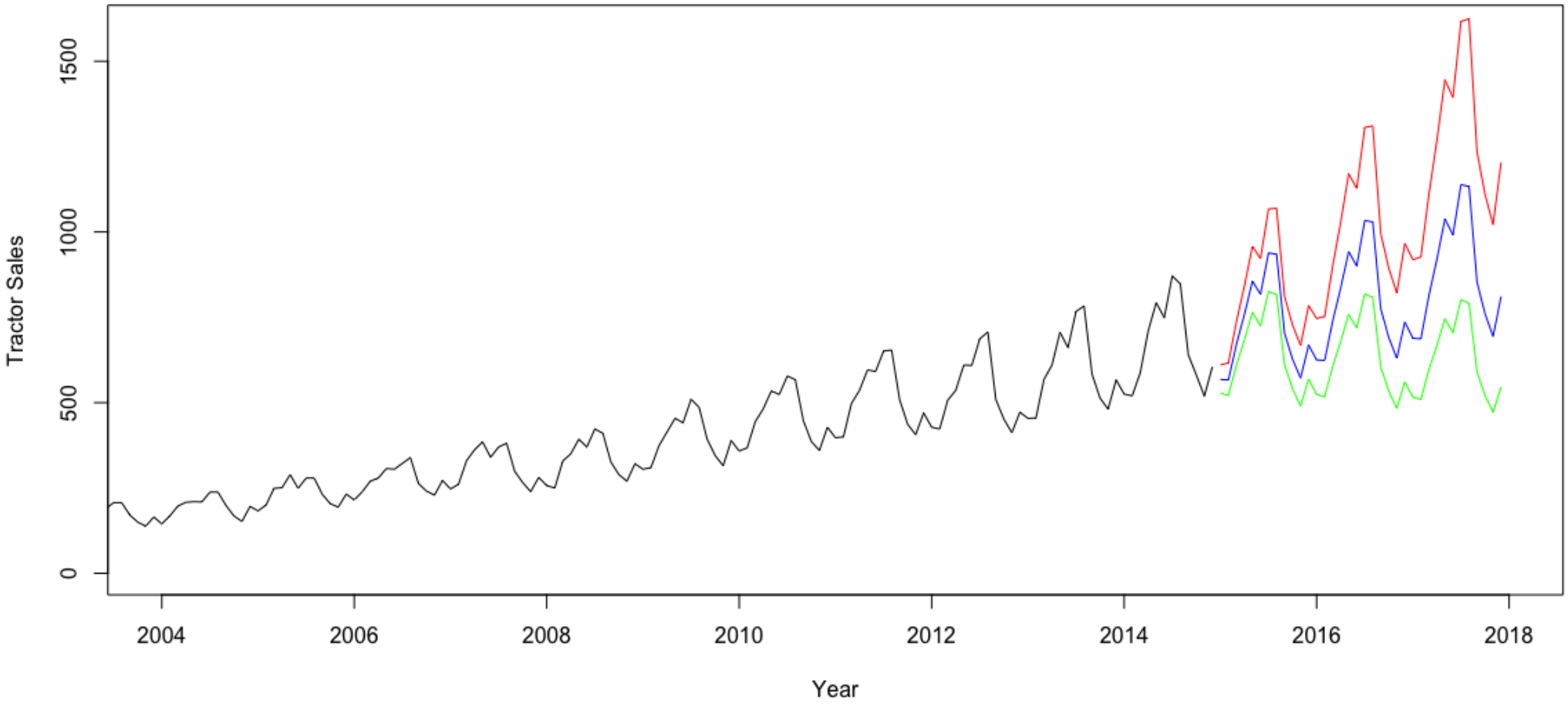
Tractor Sales Case Study

Step 8: Forecast.



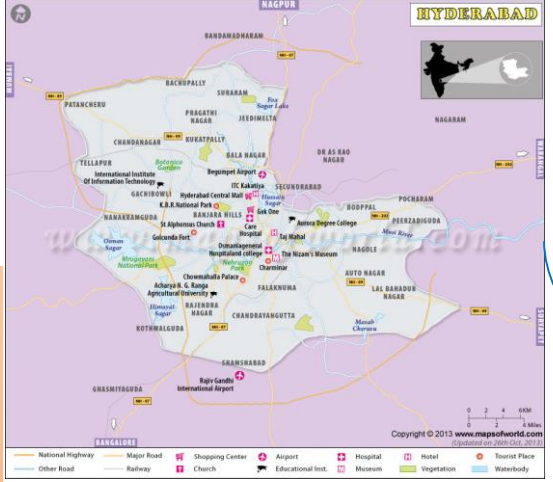
Tractor Sales Case Study

Step 8: Forecast on the original scale and not the logged data.

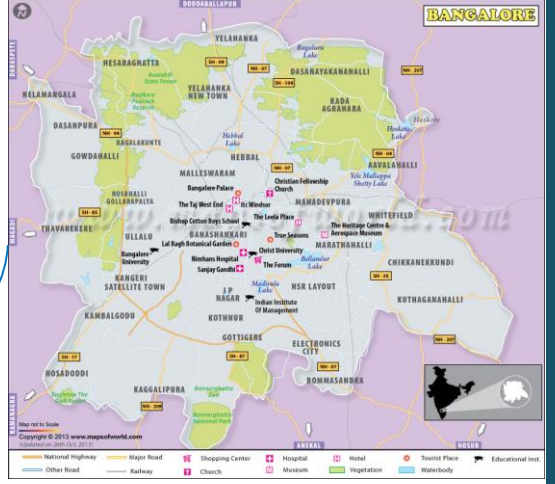


Resources

- https://www.otexts.org/fpp2/
- http://people.duke.edu/~rnau/411home.htm
- https://onlinecourses.science.psu.edu/stat510/node/41



Inspire...Educate...Transform.



HYDERABAD

2nd Floor, Jyothi Imperial, Vamsiram Builders, Old Mumbai Highway, Gachibowli, Hyderabad - 500 032
+91-9701685511 (Individuals)
+91-9618483483 (Corporates)

BENGALURU

Floors 1-3, L77, 15th Cross Road, 3A Main Road, Sector 6, HSR Layout, Bengaluru – 560 102
+91-9502334561 (Individuals)
+91-9502799088 (Corporates)

Social Media

Web:	http://www.insofe.edu.in
Facebook:	https://www.facebook.com/insofe
Twitter:	https://twitter.com/Insofeedu
YouTube:	http://www.youtube.com/InsofeVideos
SlideShare:	http://www.slideshare.net/INSOFE
LinkedIn:	http://www.linkedin.com/company/international-school-of-engineering

This presentation may contain references to findings of various reports available in the public domain. INSOFE makes no representation as to their accuracy or that the organization subscribes to those findings.