W271 - Applied Regression and Time Series Analysis - HW8

Subhashini R., Lei Yang, Ron Cordell

April 7, 2016

Build an univariate linear time series model (i.e AR, MA, and ARMA models) using the series in hw08_series.csv.

Use all the techniques that have been taught so far to build the model, including date examination, data visualization, etc. All the steps to support your final model need to be shown clearly. Show that the assumptions underlying the model are valid. Which model seems most reasonable in terms of satisfying the model's underling assumption? Evaluate the model performance (both in- and out-of-sample) Pick your "best" models and conduct a 12-step ahead forecast. Discuss your results. Discuss the choice of your metrics to measure "best".

```
library(forecast)
```

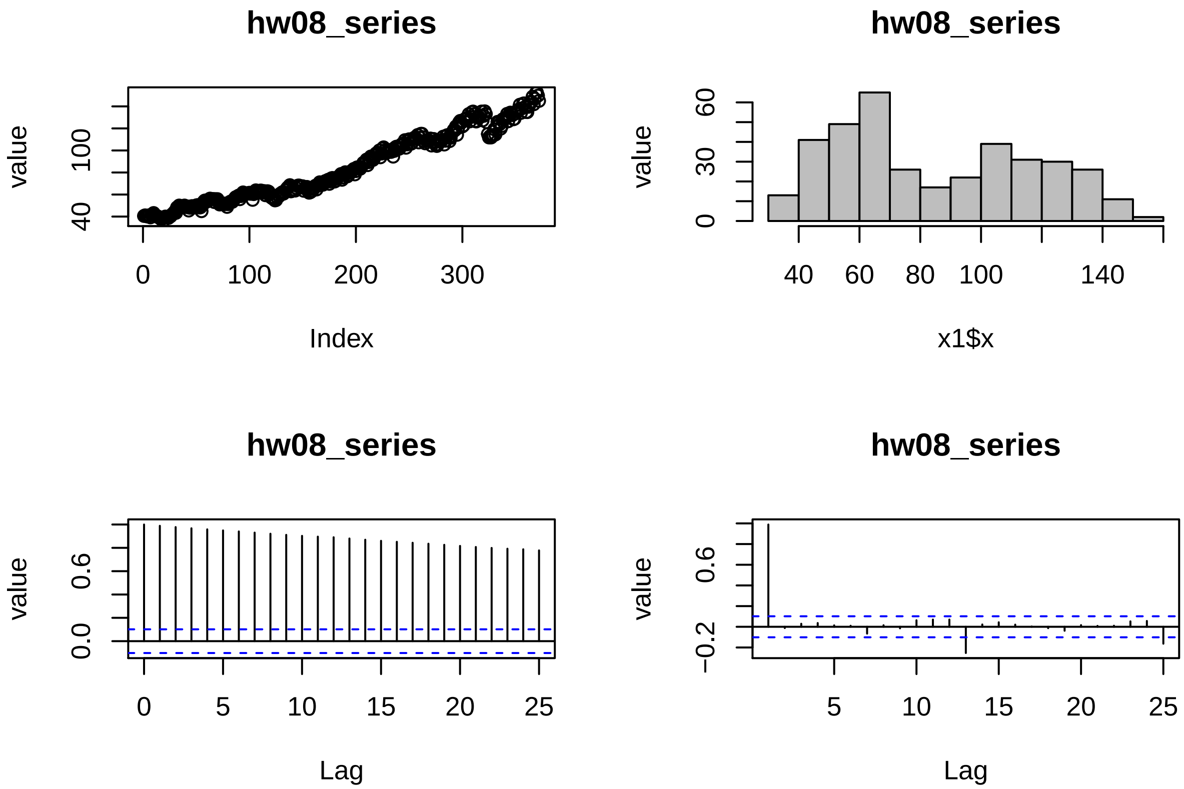
```
## Warning: package 'forecast' was built under R version 3.1.3
```

```
## Loading required package: zoo
##
## Attaching package: 'zoo'
##
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
##
## Loading required package: timeDate
## This is forecast 6.2
```

```
# load data
setwd("~/Desktop/W271Data")
x1 <- read.csv('hw08_series.csv', header = T)
```

Let's evaluate some plots

```
par(mfrow=c(2,2))
plot(x1$x, ylab="value", main="hw08_series")
hist(x1$x, col="grey", ylab="value", main="hw08_series")
acf(x1$x, ylab="value", main="hw08_series")
pacf(x1$x, ylab="value", main="hw08_series")
```



according to the plot of ACF and PACF, it is intriguing to build a AR(1) model.

```
ar1.fit <- arima(x1$x, order=c(1,0,0))
  ar1.fit
```

```
##
## Call:
## arima(x = x1$x, order = c(1, 0, 0))
##
## Coefficients:
##          ar1  intercept
##       0.9982    90.6882
## s.e.  0.0021    39.1616
##
## sigma^2 estimated as 7.145:  log likelihood = -896.41,  aic = 1798.83
```

```
  summary(ar1.fit)
```

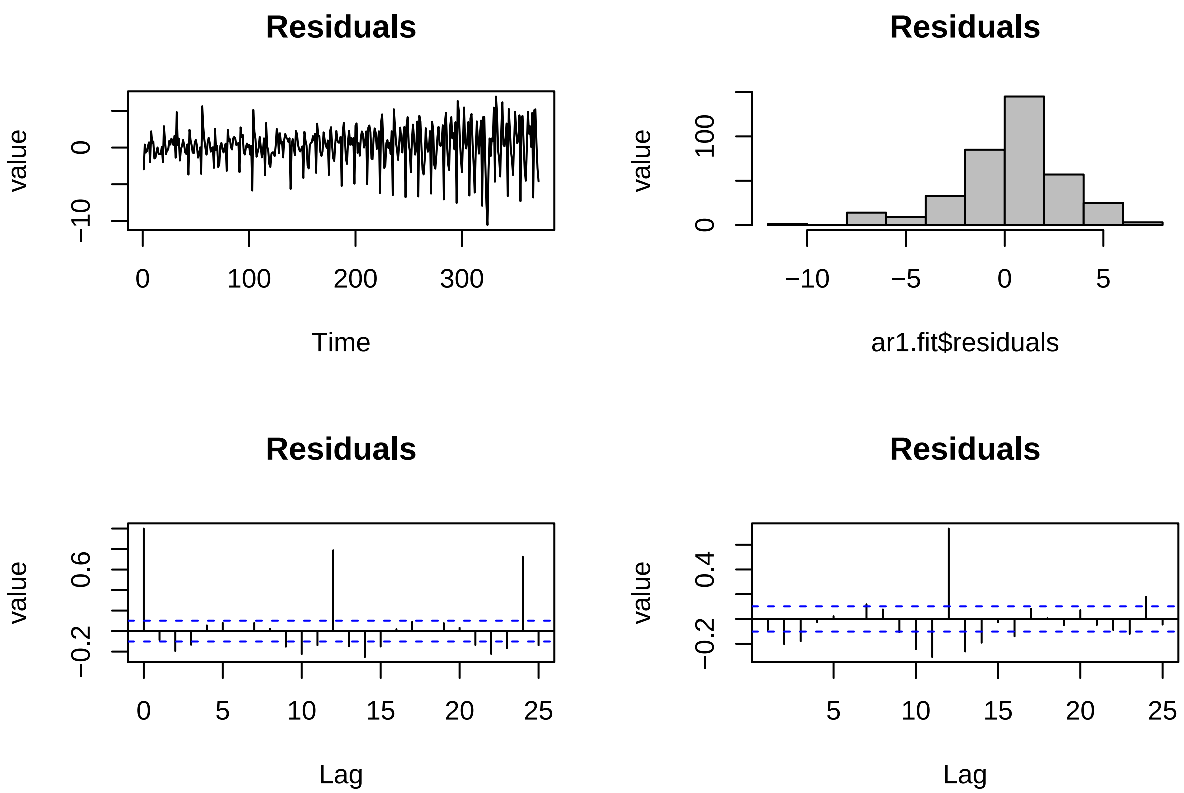
```
##
## Call:
## arima(x = x1$x, order = c(1, 0, 0))
##
## Coefficients:
##          ar1  intercept
##       0.9982    90.6882
## s.e.  0.0021    39.1616
##
## sigma^2 estimated as 7.145:  log likelihood = -896.41,  aic = 1798.83
```

```
##
## Training set error measures:
##                     ME     RMSE      MAE       MPE     MAPE      MASE
## Training set 0.2620849 2.672941 1.966946 0.2340277 2.328949 0.9999134
##                     ACF1
## Training set -0.08763521
```

from the coefficient, it's almost a random walk.

Let's check residue for the stationality assumption

```r
par(mfrow=c(2,2))
plot(ar1.fit$residuals, ylab="value", main="Residuals")
hist(ar1.fit$residuals, col="grey", ylab="value", main="Residuals")
acf(ar1.fit$residuals, ylab="value", main="Residuals")
pacf(ar1.fit$residuals, ylab="value", main="Residuals")
```



obviously it's not random noise, as the magnitude is increasing in the second half. Although the Box test indicate non-significant results:

```r
Box.test(ar1.fit$residuals, type="Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  ar1.fit$residuals
## X-squared = 2.88, df = 1, p-value = 0.08968
```

Let's increase the AR order:

```r
ar.fit <- ar(x1$x, method="mle")
  ar.fit
```

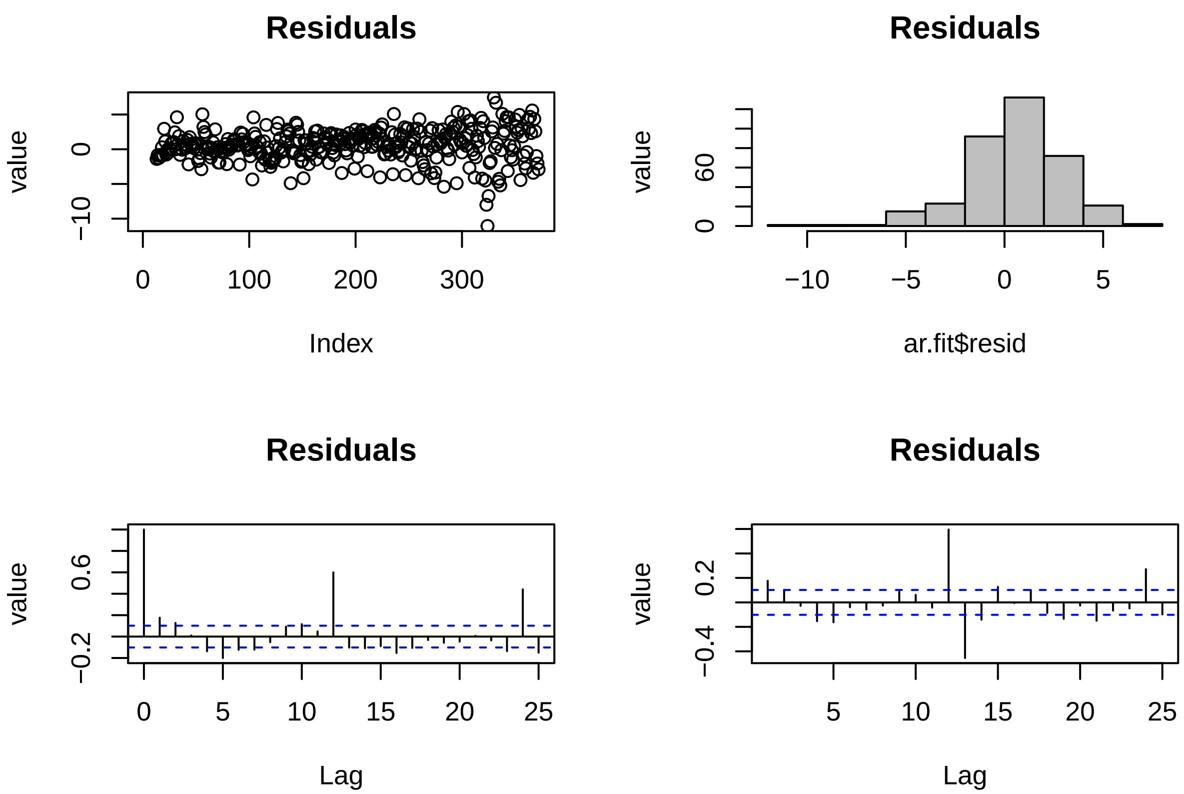
```
##
## Call:
## ar(x = x1$x, method = "mle")
##
## Coefficients:
##       1        2        3        4        5        6        7        8
##  0.7795   0.0091   0.1034   0.1859   0.0578  -0.0459   0.0271  -0.1118
##       9       10       11       12
## -0.1795  -0.1012  -0.0123   0.2872
##
## Order selected 12  sigma^2 estimated as  5.548
```

```r
  summary(ar.fit)
```

```
##              Length Class  Mode
## order          1    -none- numeric
## ar            12    -none- numeric
## var.pred       1    -none- numeric
## x.mean         1    -none- numeric
## aic           13    -none- numeric
## n.used         1    -none- numeric
## order.max      1    -none- numeric
## partialacf     0    -none- NULL
## resid        372    -none- numeric
## method         1    -none- character
## series         1    -none- character
## frequency      1    -none- numeric
## call           3    -none- call
## asy.var.coef 144    -none- numeric
```

Let's check residue for the stationality assumption for the new AR model

```r
par(mfrow=c(2,2))
plot(ar.fit$resid, ylab="value", main="Residuals")
hist(ar.fit$resid, col="grey", ylab="value", main="Residuals")
acf(ar.fit$resid, ylab="value", main="Residuals", na.action=na.pass)
pacf(ar.fit$resid, ylab="value", main="Residuals", na.action=na.pass)
```



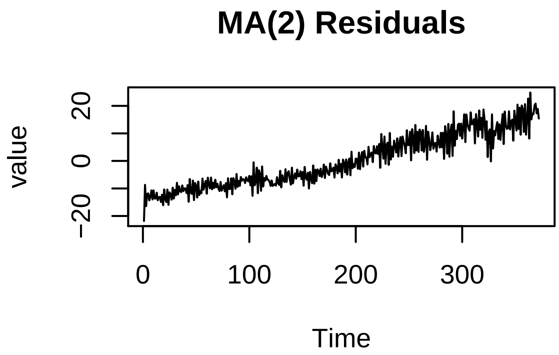
from the plot, it's still not quite random, and the Box test indicates significant results as well.

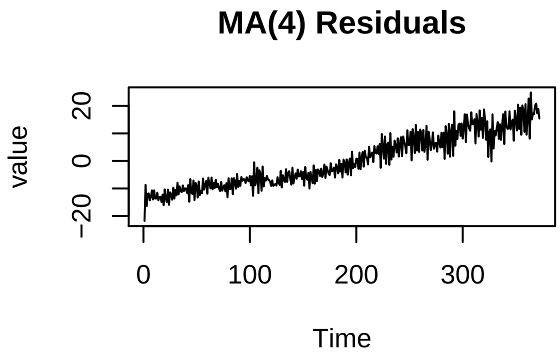
Let's check MA model:

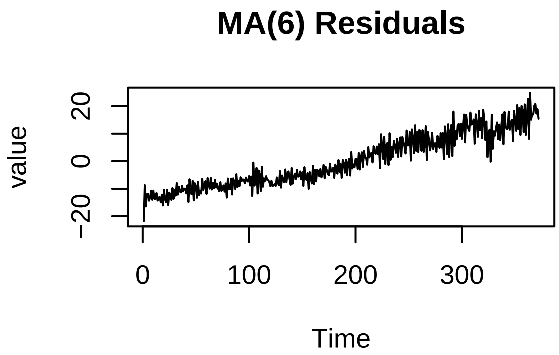
```
ma2.fit <- arima(x1$x, order=c(0,0,2))
ma4.fit <- arima(x1$x, order=c(0,0,4))
ma6.fit <- arima(x1$x, order=c(0,0,6))
ma8.fit <- arima(x1$x, order=c(0,0,8))
```

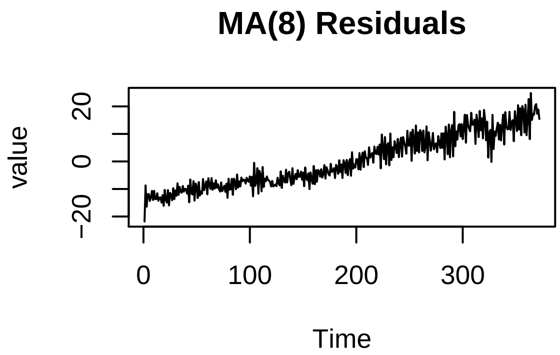
and check the residuals:

```
par(mfrow=c(2,2))
plot(ma2.fit$resid, ylab="value", main="MA(2) Residuals")
plot(ma2.fit$resid, ylab="value", main="MA(4) Residuals")
plot(ma2.fit$resid, ylab="value", main="MA(6) Residuals")
plot(ma2.fit$resid, ylab="value", main="MA(8) Residuals")
```









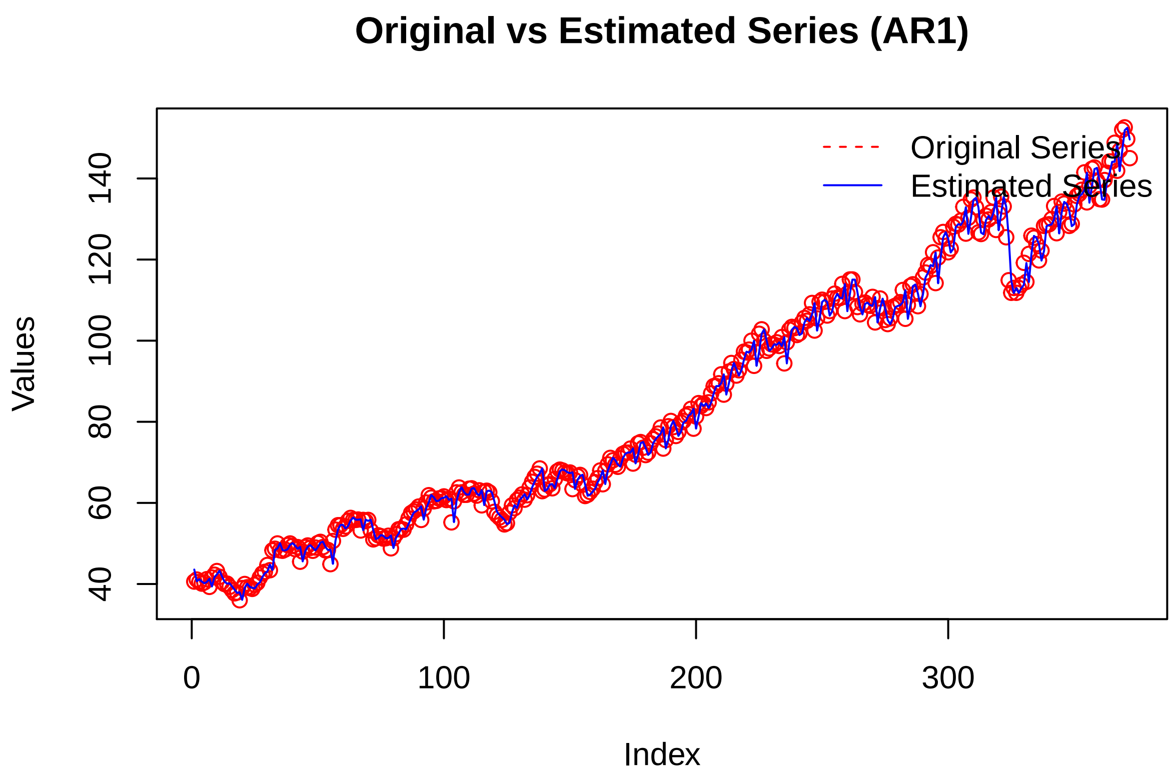
obviously MA model is not adquate for the series.

Let's turn to ARMA model, and it turns out we are not able to fit ARMA model as the series is not stationary, obviously.

```
#arma11 <- arima(x1$x, order=c(1,0,1))
#arma21 <- arima(x1$x, order=c(2,0,1))
#arma22 <- arima(x1$x, order=c(2,0,2))
#arma23 <- arima(x1$x, order=c(2,0,3))
#arma33 <- arima(x1$x, order=c(3,0,3))
```

Based on the principle of parsimonious, we will use AR(1) as the model, since both the series and the residuals are non-stationary.

```
par(mfrow=c(1,1))
  plot(x1$x, col="red",
       main="Original vs Estimated Series (AR1)",
       ylab="Values", lty=2)
  lines(fitted(ar1.fit),col="blue")
  leg.txt <- c("Original Series", "Estimated Series")
  legend("topright", legend=leg.txt, lty=c(2,1),
          col=c("red","blue"), bty='n', cex=1)
```



in-sample prediction fits the original data pretty well.

12-step prediction:

```
ar1.fit.fcast <- forecast.Arima(ar1.fit, 12)
  summary(ar1.fit.fcast)
```

```
##
## Forecast method: ARIMA(1,0,0) with non-zero mean
##
## Model Information:
##
## Call:
## arima(x = x1$x, order = c(1, 0, 0))
##
## Coefficients:
##          ar1  intercept
##       0.9982    90.6882
## s.e.  0.0021    39.1616
##
## sigma^2 estimated as 7.145:  log likelihood = -896.41,  aic = 1798.83
##
## Error measures:
##                     ME     RMSE      MAE       MPE     MAPE      MASE
## Training set 0.2620849 2.672941 1.966946 0.2340277 2.328949 0.9999134
##                     ACF1
## Training set -0.08763521
##
## Forecasts:
##     Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
```

```
## 373        144.9044 141.4789 148.3299 139.6655 150.1433
## 374        144.8090 139.9688 149.6491 137.4066 152.2113
## 375        144.7137 138.7910 150.6365 135.6557 153.7718
## 376        144.6186 137.7857 151.4516 134.1685 155.0688
## 377        144.5237 136.8909 152.1565 132.8504 156.1971
## 378        144.4290 136.0750 152.7829 131.6527 157.2052
## 379        144.3344 135.3190 153.3498 130.5465 158.1222
## 380        144.2400 134.6105 153.8694 129.5130 158.9669
## 381        144.1457 133.9411 154.3503 128.5391 159.7523
## 382        144.0516 133.3045 154.7988 127.6153 160.4880
## 383        143.9577 132.6958 155.2196 126.7342 161.1812
## 384        143.8639 132.1116 155.6163 125.8903 161.8376
```

```
plot(ar1.fit.fcast, main="12-Step Ahead Forecast and Original & Estimated Series",
    xlab="Simulated Time Period", ylab="Original, Estimated, and Forecasted Values",
    xlim=c(), lty=2, col="navy")
lines(fitted(ar1.fit),col="blue")
```