STAT417 Group Project

Aiden Chang, Jack Chen, Vincent Cheon

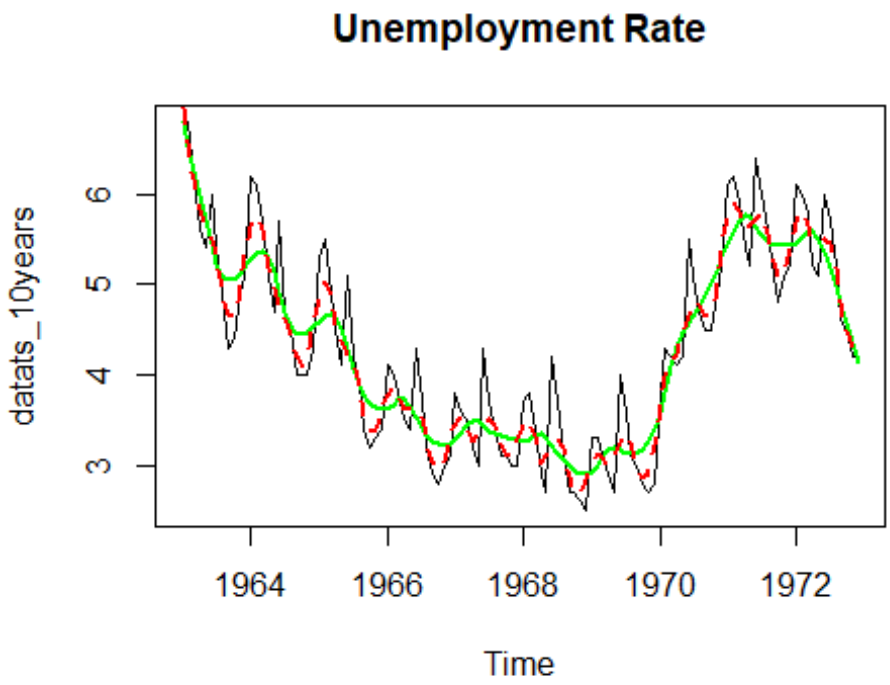
11/21/2019

Loading Data

```
ur <- read.csv("C:/Users/ac991/Documents/UMBC/Fall 2019/STAT417/Group
Project/Unemployment Rate Time series Data Group-1.csv")
```

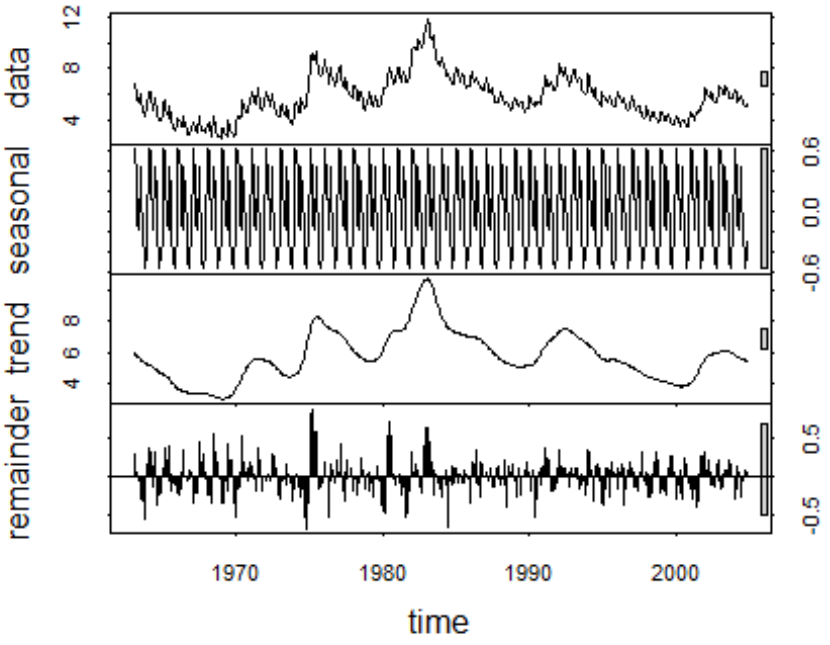
##plot the time-series data

```
data <- c(ur$Rate.., ur$Rate...1, ur$Rate...2, ur$Rate...3,
          ur$Rate...4, ur$Rate...5)

datats <- ts(data, start=1963, end=c(2004,12), frequency = 12)
length(datats)

## [1] 504

datats_10years <- ts(datats[1:120], start=1963, frequency = 12)
plot(datats_10years, main = "Unemployment Rate")
lines(lowess(datats_10years, f=0.10), lwd=2, lty=1, col="green")
lines(lowess(datats_10years, f=0.05), lwd=2, lty=2, col="red")
```



decomposing the data

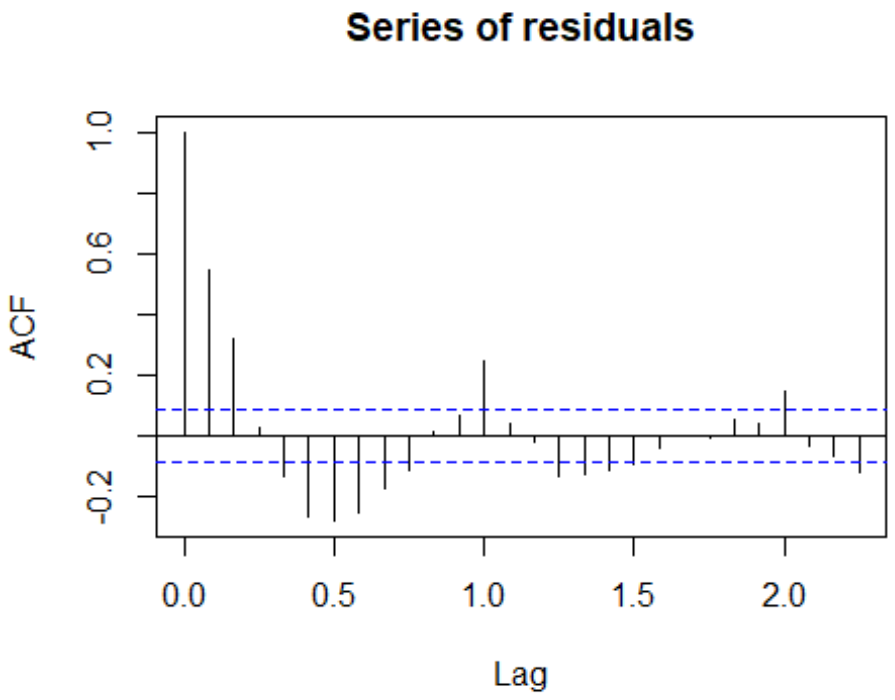
```
ss <- stl(datats, "per")
plot(ss)
```



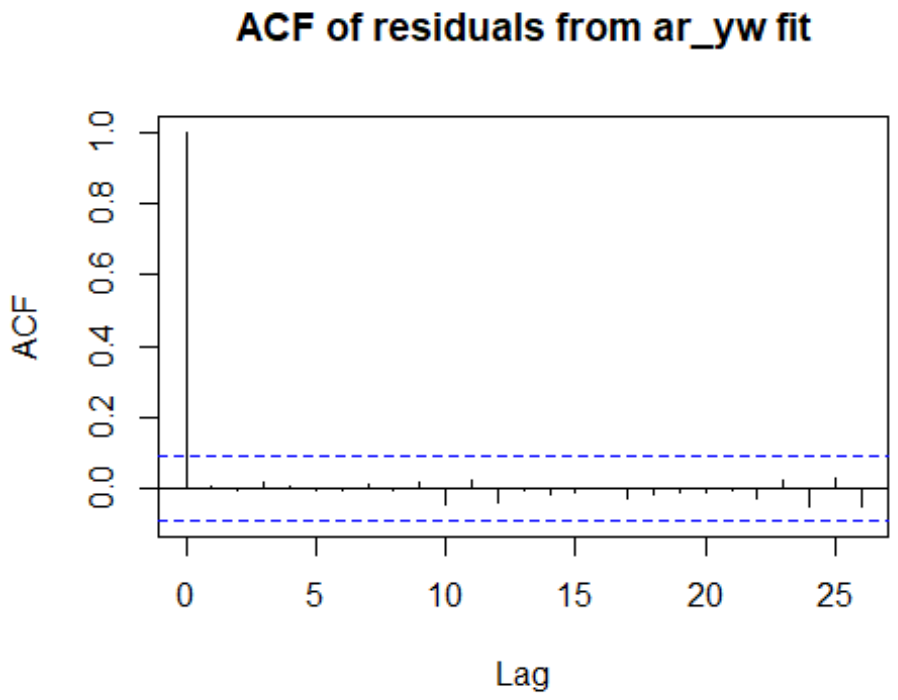

```
season <- ss$time.series[,1]
trend <- ss$time.series[,2]
res <- ss$time.series[,3]
```

check stationary

```
acf(res,main="Series of residuals")
```



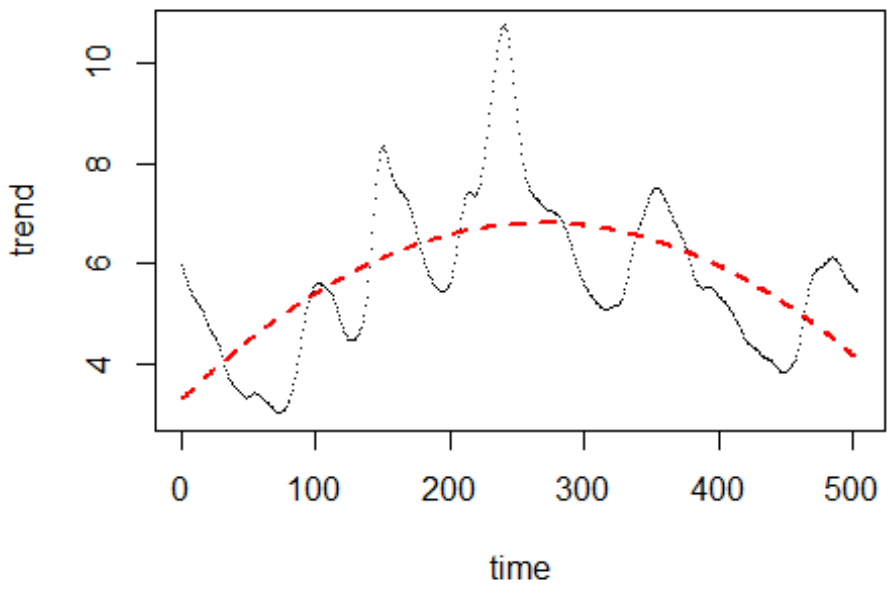

```
fit_ar <- ar.yw(res, order.max = NULL)
fit_ar

##
## Call:
## ar.yw.default(x = res, order.max = NULL)
##
## Coefficients:
##       1        2        3        4        5        6        7        8
##  0.5398   0.0919  -0.1252  -0.0636  -0.1541   0.0150  -0.0469  -0.0310
##       9       10       11       12       13       14       15       16
## -0.0523  -0.0671  -0.0505   0.2894  -0.2388  -0.0665  -0.0618  -0.0162
##      17       18       19       20       21       22       23       24
##  0.0802  -0.1179   0.0292  -0.0221  -0.0798   0.0267  -0.0541   0.1359
##      25
## -0.1667
##
## Order selected 25  sigma^2 estimated as  0.02592

# the fit suggests that the order is 25
ts_resid <- ts(fit_ar$resid)
acf(ts_resid[-c(1:25)], main="ACF of residuals from ar_yw fit")
```



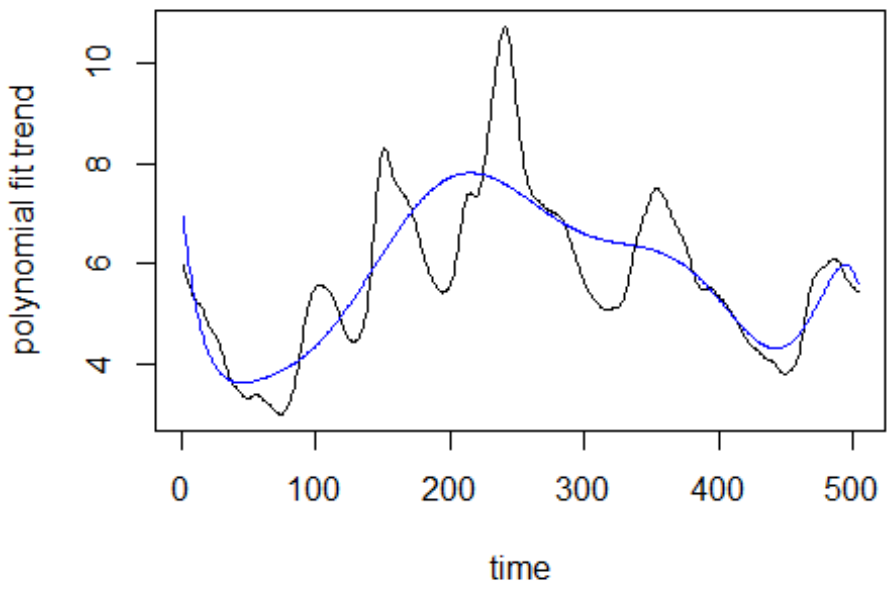
trend

$$trend = \alpha_1 cos(r * time/365) + \alpha_2 sin(r * time/365) + \epsilon$$

```
time <- 1:504
dat.trend <- as.data.frame(cbind(trend,time))
r <- 0.5
fit1<-lm(trend~I(cos(r*time/365)) + I(sin(r*time/365)), data=dat.trend)
coeff1 <- summary(fit1)$coef

# whole dataset
plot(time,trend, pch=".")
lines(time, coeff1[1]+ coeff1[2]*cos(r*time/365) + coeff1[3]*sin(r*time/365),
lwd=2, lty=2, col="red")
```




```r
# model using a POLYNOMIAL function
fit2 <- lm(trend ~ time + I(time^2) + I(time^3) + I(time^4) + I(time^5) +
I(time^6) + I(time^7) + I(time^8) + I(time^9), data = dat.trend)
coeff2 <- summary(fit2)$coef
# whole dataset
plot(time,trend, type = "l", ylab = "polynomial fit trend")
lines(time,coeff2[1] + coeff2[2]*time + coeff2[3]*time^2 + coeff2[4]*time^3 +
coeff2[5]*time^4 + coeff2[6]*time^5 + coeff2[7]*time^6 + coeff2[8] * time^7 +
coeff2[9]*time^8 + coeff2[10]*time^9, col = "blue")
```



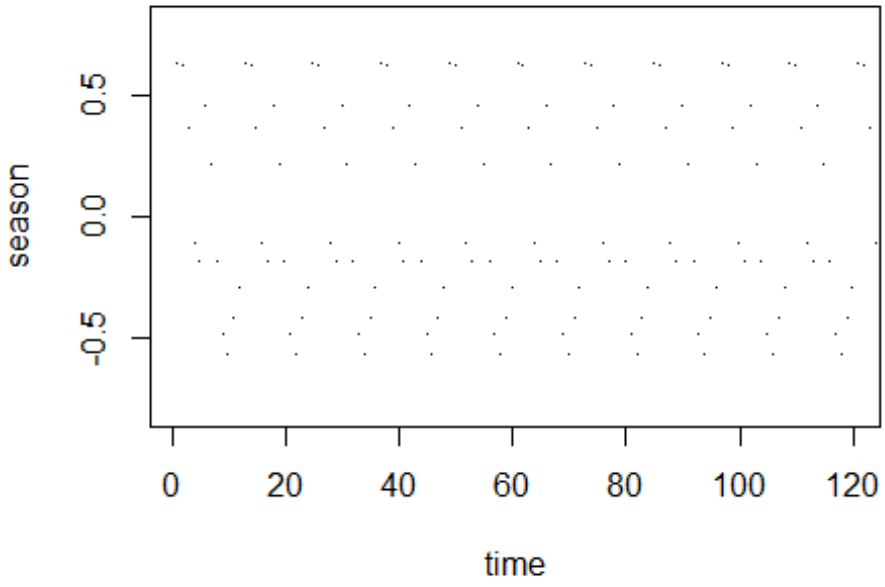
seasonal

$$season = \alpha_1 cos(a * time) + \alpha_2 sin(a * time) + \alpha_3 cos(2a * time) + \alpha_4 sin(2a * time) + \epsilon$$

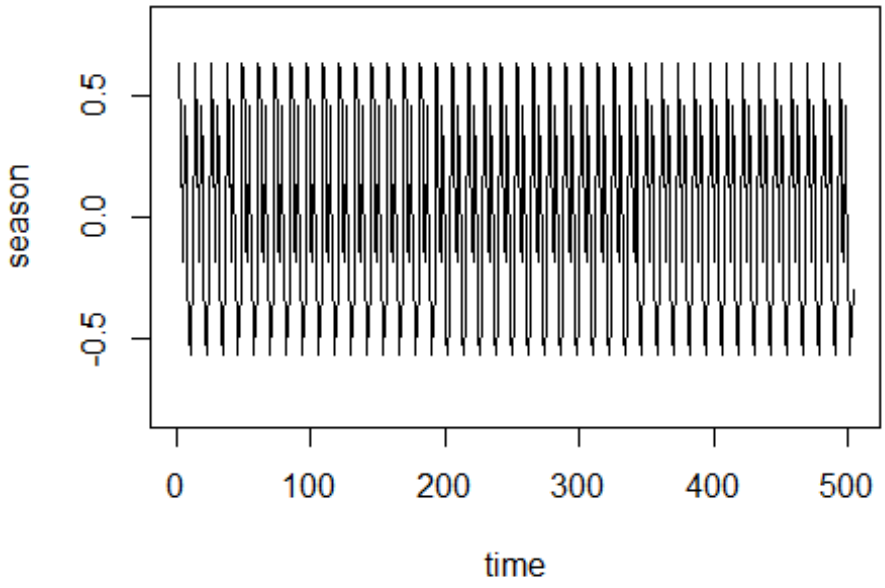
```
season <- ss$time.series[,1]
dd <- as.data.frame(cbind(season, time))
a <- 0.5237
fit.ss<-lm(season~I(cos(a*time))+I(sin(a*time)) + I(cos(2*a*time)) +
I(sin(2*a*time)), data=dd)
coefss <- summary(fit.ss)$coef
coefss
```

```
##                          Estimate  Std. Error      t value      Pr(>|t|)
## (Intercept)          -5.879719e-05 0.007291815 -0.008063451  9.935696e-01
## I(cos(a * time))     -8.046295e-02 0.010311308 -7.803369224  3.560977e-14
## I(sin(a * time))      3.778225e-01 0.010313058 36.635350313 1.436804e-143
## I(cos(2 * a * time))  6.957938e-02 0.010311692  6.747620239  4.178976e-11
## I(sin(2 * a * time))  3.740002e-01 0.010312675 36.266071235 5.684073e-142
```

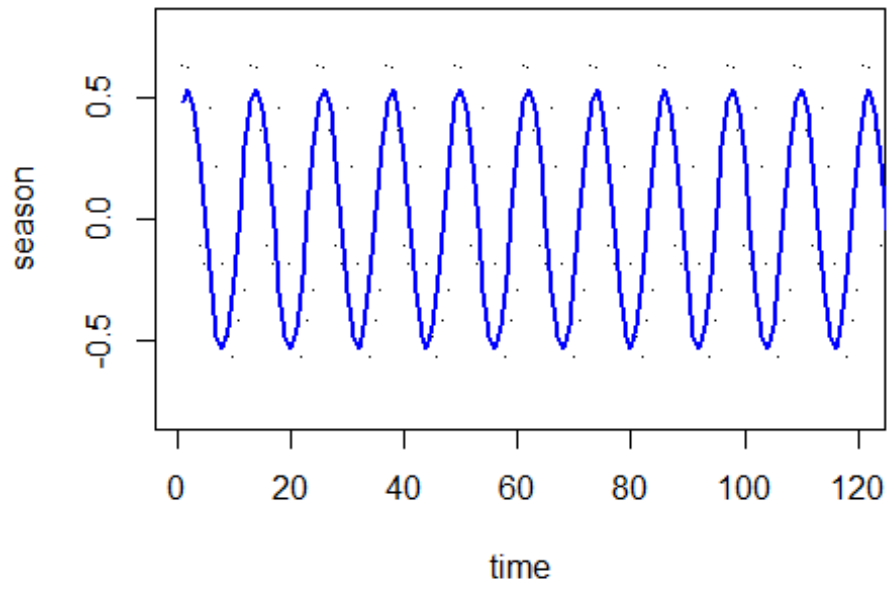
```
# first 10 years
plot(time,season,pch=".",ylim = c(-0.8,0.8), xlim = c(1,120))
```



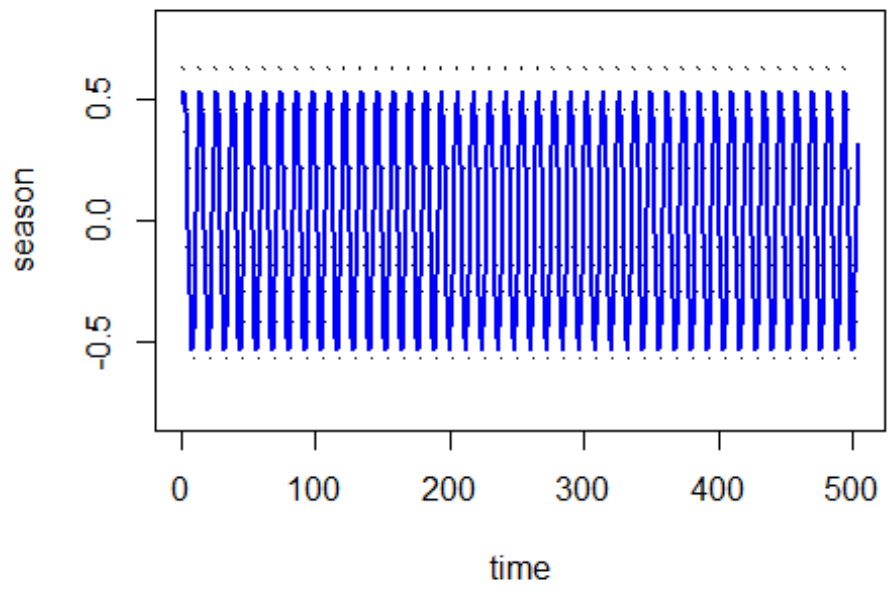

```r
# whole dataset
plot(time,season,pch=".",ylim = c(-0.8,0.8), type = "l")
```



```r
# Model with TRIGONOMETRIC functions
# first 10 years
plot(time,season,pch=".",ylim = c(-0.8,0.8), xlim = c(1,120))
lines(time, coefss[1]+ coefss[2]*cos(a*time)+ coefss[3]*sin(a*time) +
coefss[3]*cos(a*time)+ coefss[4]*sin(a*time), lwd=2, col="blue")
```




```r
# whole dataset
plot(time,season,pch=".",ylim = c(-0.8,0.8))
lines(time, coefss[1]+ coefss[2]*cos(a*time)+ coefss[3]*sin(a*time) +
coefss[3]*cos(a*time)+ coefss[4]*sin(a*time), lwd=2, col="blue")
```



model 1, 2, and 3

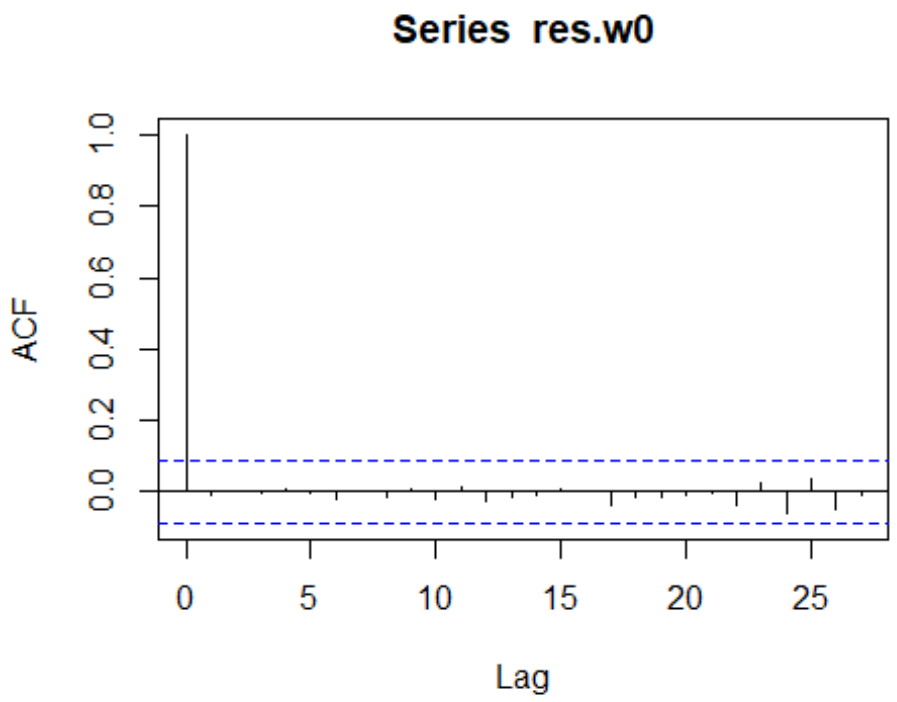
```
fit_ar <- ar.yw(res, order.max = NULL)
fit_ar
```

```
##
## Call:
## ar.yw.default(x = res, order.max = NULL)
##
## Coefficients:
##       1        2        3        4        5        6        7        8
##  0.5398   0.0919  -0.1252  -0.0636  -0.1541   0.0150  -0.0469  -0.0310
##       9       10       11       12       13       14       15       16
## -0.0523  -0.0671  -0.0505   0.2894  -0.2388  -0.0665  -0.0618  -0.0162
##      17       18       19       20       21       22       23       24
##  0.0802  -0.1179   0.0292  -0.0221  -0.0798   0.0267  -0.0541   0.1359
##      25
## -0.1667
##
## Order selected 25  sigma^2 estimated as  0.02592
```

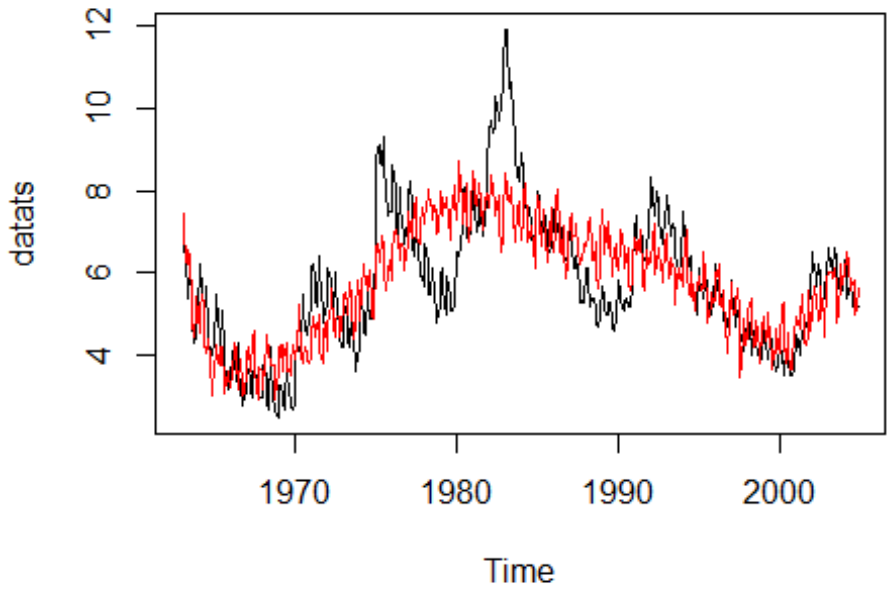
```
ts_resid <- ts(fit_ar$resid)

# fitting AR(25) with and without intercept
fit.w0 <- arima(res,order = c(25,0,0), include.mean = TRUE)
fit.w0$aic
```

```
## [1] -387.7817
```

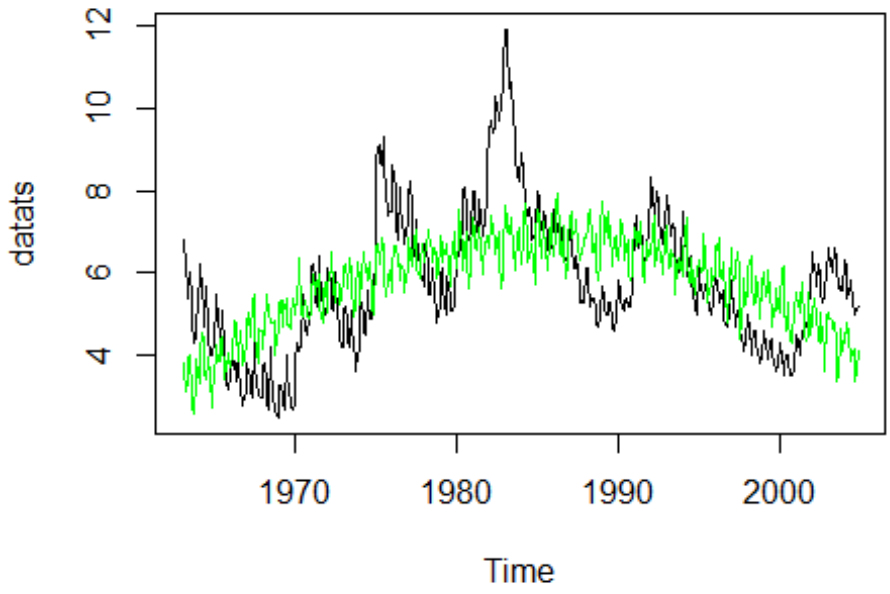
```r
fit.w1 <- arima(res,order = c(25,0,0), include.mean = FALSE)
fit.w1$aic

## [1] -389.7572

# Model with intercept is suggested

res.w0 <- ts(fit.w1$residuals)
acf(res.w0)
```



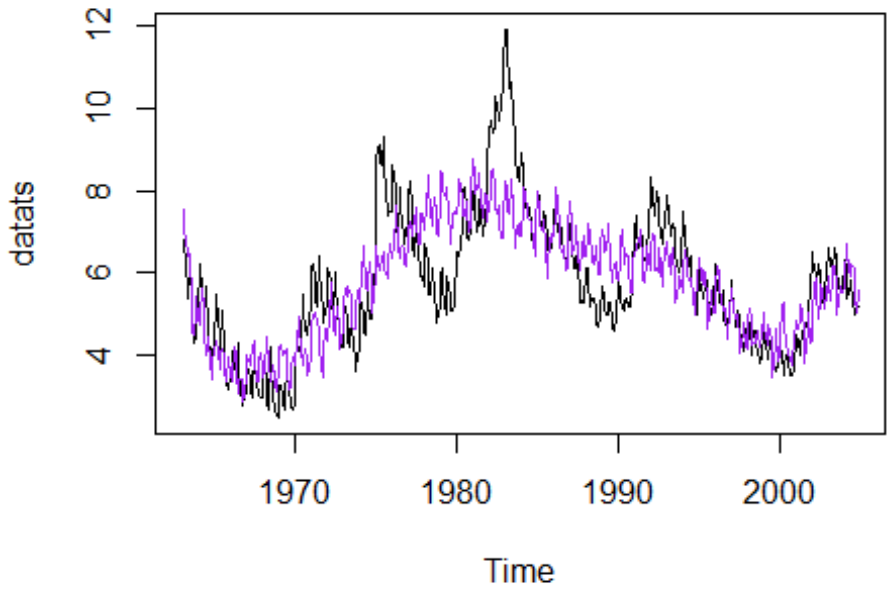

```r
coef.w1 <- coef(fit.w0)
w.arima <- arima.sim(list(ar = coef.w1), sd = sqrt(fit.w0$sigma2), n = 504)

set.seed(10)
#Random error term where W~WN(0, sd = sd(res))
W <- rnorm(length(time), mean = 0, sd = sd(res))

#Model 1: Polynomial Trend + trig seasonality + Random error
yhat1 <-coeff2[1] + coeff2[2]*time + coeff2[3]*time^2 + coeff2[4]*time^3 +
coeff2[5]*time^4 + coeff2[6]*time^5 + coeff2[7]*time^6 + coeff2[8] * time^7 +
coeff2[9]*time^8 + coeff2[10]*time^9 + coefss[1] + coefss[2] * cos(a*time) +
coefss[3] * sin(a*time) + coefss[4]*cos(2.5*a*time) + coefss[5] *
sin(2.5*a*time) + W

d.y1 <- ts(yhat1, start = 1963, end = c(2004,12), frequency = 12)
plot(datats)
lines(d.y1, col = "red")
```




```r
#Model 2: Trig Trend + Trig Seasonality + Ranodm error
yhat2 <- coeff1[1]+coeff1[2]*cos(r*time/365)+coeff1[3]*sin(r*time/365) +
coefss[1] + coefss[2] * cos(a*time) + coefss[3] * sin(a*time) +
coefss[4]*cos(2.5*a*time) + coefss[5] * sin(2.5*a*time) + W

d.y2 <- ts(yhat2, start = 1963, end = c(2004,12), frequency = 12)
plot(datats)
lines(d.y2, col = "green")
```




```r
#Model 3: Polynomial Trend + Trig Seasonality + Modeled residuals
yhat3 <- coeff2[1] + coeff2[2]*time + coeff2[3]*time^2 + coeff2[4]*time^3 +
coeff2[5]*time^4 + coeff2[6]*time^5 + coeff2[7]*time^6 + coeff2[8] * time^7 +
coeff2[9]*time^8 + coeff2[10]*time^9 + coefss[1] + coefss[2] * cos(a*time) +
coefss[3] * sin(a*time) + coefss[4]*cos(2.5*a*time) + coefss[5] *
sin(2.5*a*time) + w.arima

d.y3 <- ts(yhat3, start = 1963, end = c(2004,12), frequency = 12)
plot(datats)
lines(d.y3, col = "purple")
```




```
y <- as.numeric(datats)
#MSE of model 1
sum((y-yhat1)^2)/length(y)
```

```
## [1] 1.239045
```

```
#MSE of model 2
sum((y-yhat2)^2)/length(y)
```

```
## [1] 2.004632
```

```
#MSE of model 3
sum((y-yhat3)^2)/length(y)
```

```
## [1] 1.235543
```

since model 3 has the lowest MSE, we'll use model 3

forcast for future value

```
library(forecast)
```

```
## Registered S3 method overwritten by 'xts':
##   method     from
##   as.zoo.xts zoo
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
## Registered S3 methods overwritten by 'forecast':
##   method             from
##   fitted.fracdiff    fracdiff
##   residuals.fracdiff fracdiff

unemployment.forecast <- forecast(datats, h = 60, level = 95)


d.y3 <- ts(yhat3, start = 1963, end = c(2004,12), frequency = 12)


plot(unemployment.forecast)
lines(d.y3, col = "purple")
```