

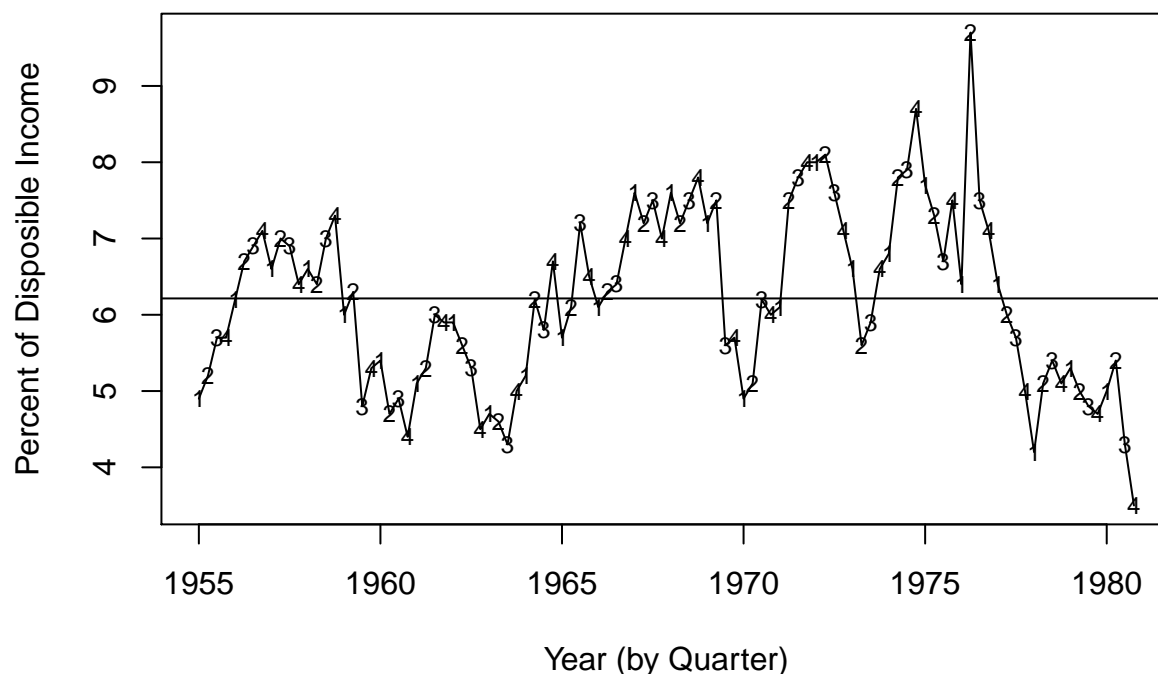
# Personal Savings Analysis - Rough Draft

Andrew Brown, Melissa Hooke, Frances Hung, Mai Nguyen, Brenner Ryan

11/29/2018

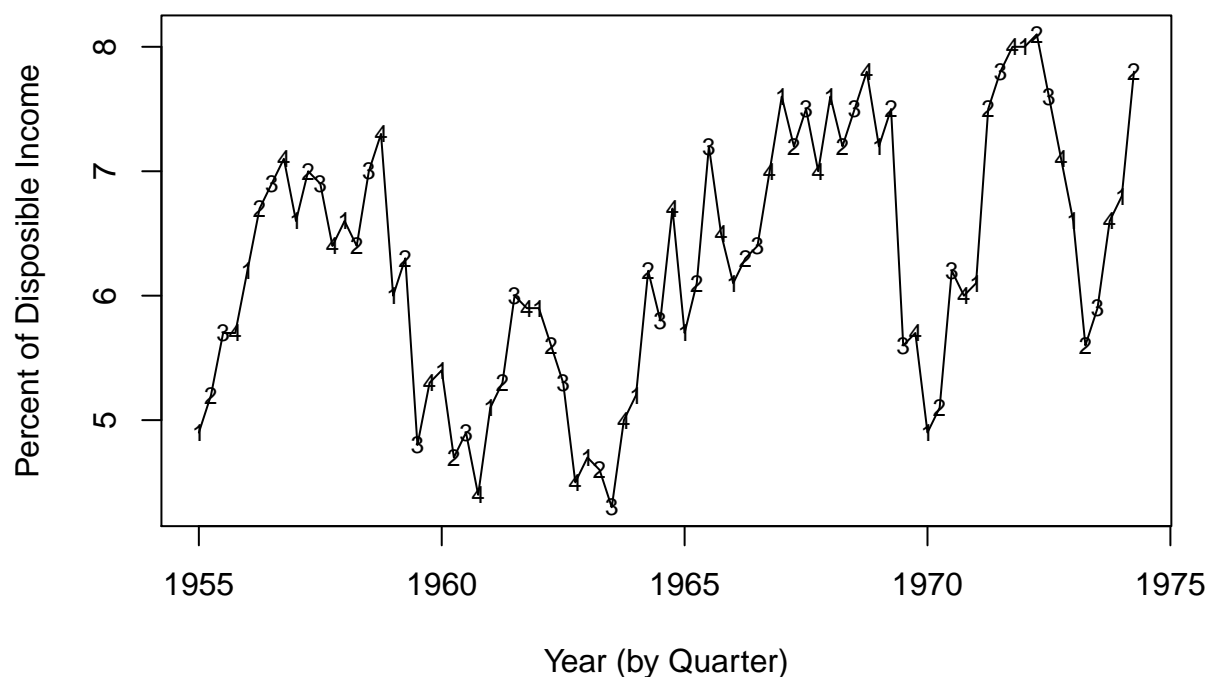
## Time Series Exploration

### Time Series of Personal Savings in the US (1955–1980)



The original time series spans over 26 years of personal savings data with each year divided into financial quarters, amounting to 104 total observations. From our original time series plot, we do not see any *obvious* seasonal trends. However, there seems to be a positive correlation between terms close to one another. The general increasing trend and volatility in the 70s may be linked to the economic crash in the early 70s and oil energy crisis in 1979. The 1970's were marked by high inflation and growing expenses due to rising interest rates. We took off the last 5 years because there was a huge drop due to a recession. Then we took off the last 6 observations in order to use them as test points at the end. The resulting series is below:

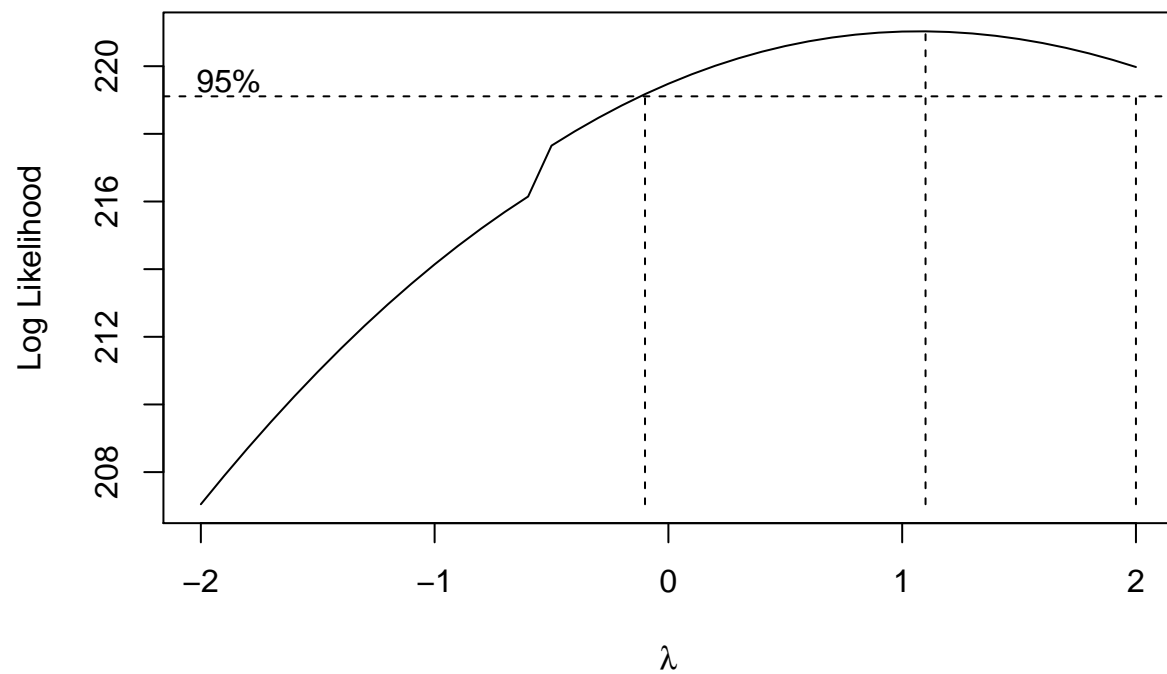
## Time Series of Personal Savings in the US (1955–1980)



There is a general upward trend in the data, as well as a dependence on previous quarters' savings to determine current personal savings percentages. We may want to take a difference to see if that helps with stationarity, but first let's explore without taking the difference.

```
# should we do a transformation?
```

```
boxcox = BoxCox.ar(savings)
```

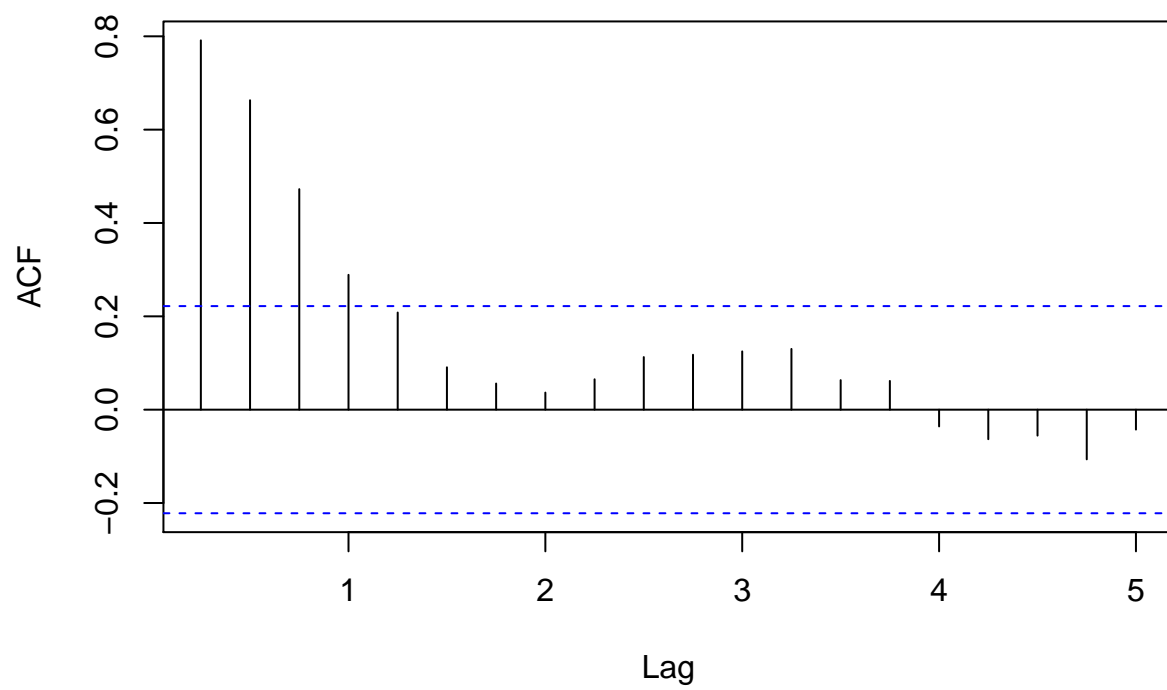


```
boxcox$mle
```

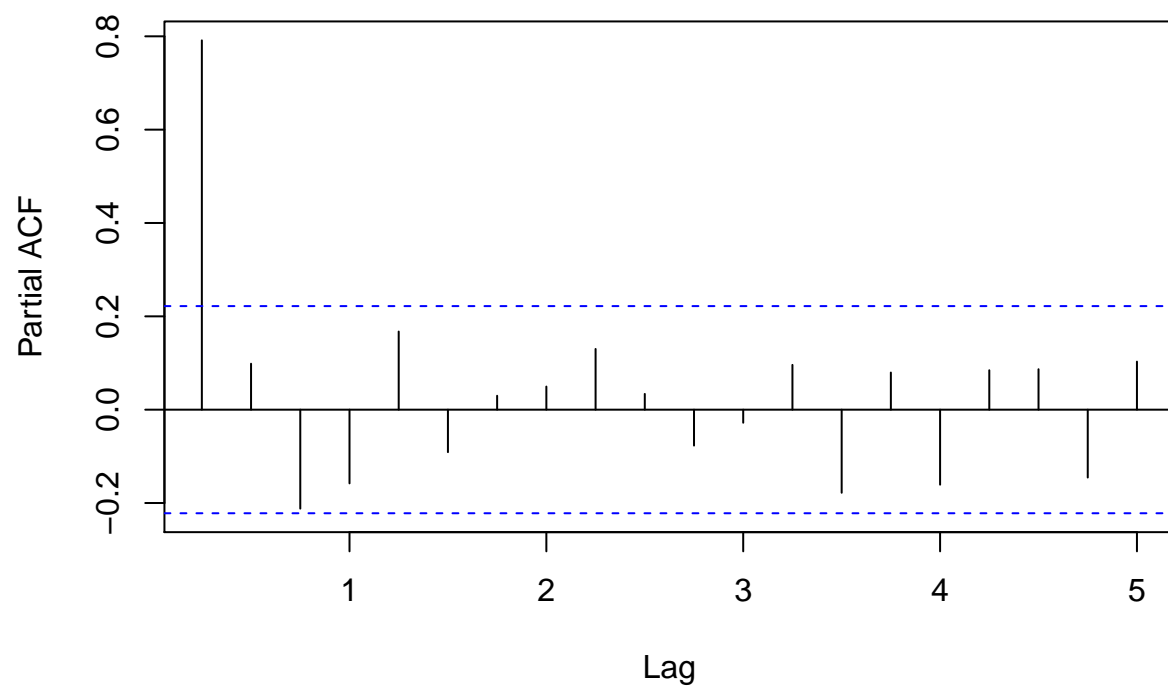
```
## [1] 1.1
```

The Boxcox output indicates that a transformation is not necessary in order to stabilize the variance since  $\lambda$  is about equal to 1.

### Series savings



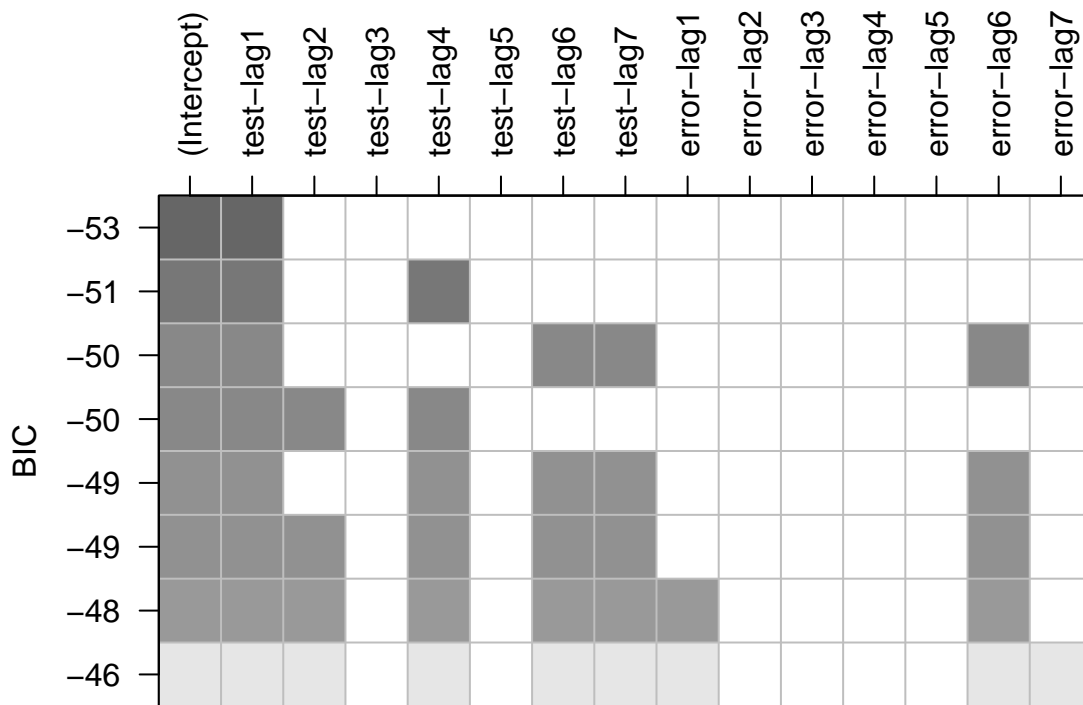
## Series savings



The PACF seems to indicate that an AR(1) process may be a good candidate model.

## Model Specification

```
## AR/MA
##   0 1 2 3 4 5 6 7 8 9 10 11 12 13
## 0 x x x x o o o o o o o o o
## 1 o x o o o o o o o o o o o
## 2 x x o o o o o o o o o o o
## 3 x x o o o o o o o o o o o
## 4 x o o o o o o o o o o o o
## 5 x x o o o o o x o o o o o
## 6 x o o o o o o o o o o o o
## 7 x o o o o o o o o o o o o
```

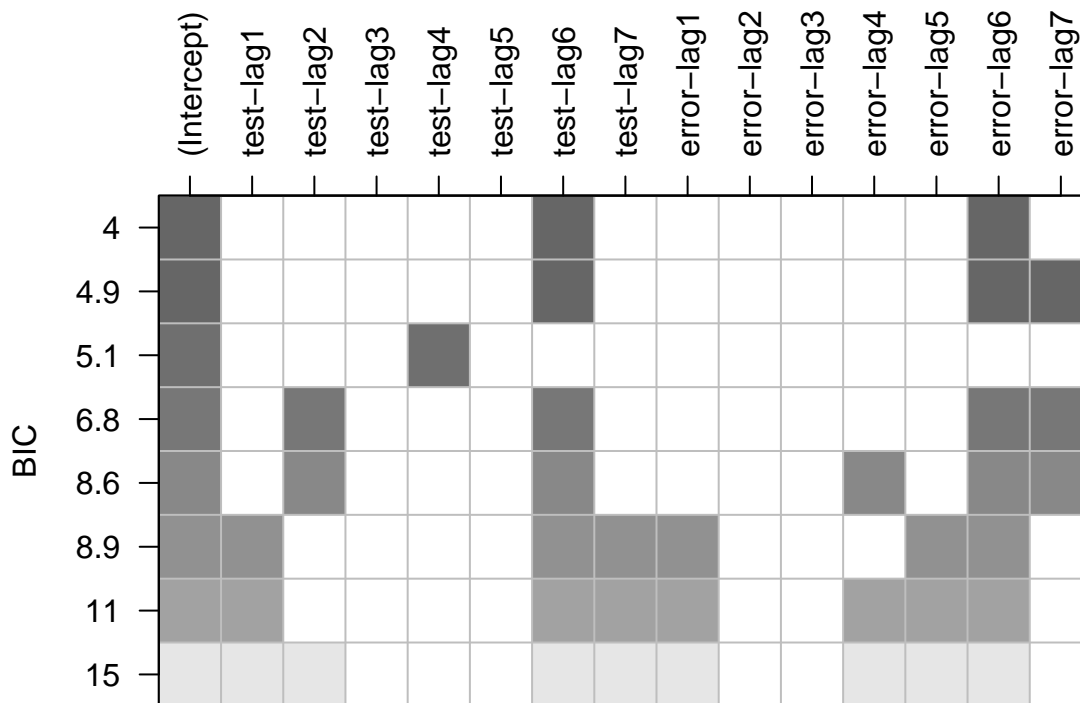


While the EACF is inconclusive, the best subsets method indicates that, an  $AR(1)$  process or a multiplicative  $AR(1) \times AR(1)_4$  with a seasonal period of 4 are good candidate models.

```
diffs = (savings-zlag(savings))[2:78]
eacf(diffs)
```

```
## AR/MA
##   0 1 2 3 4 5 6 7 8 9 10 11 12 13
## 0 o o o x o x o o o o o o x o
## 1 x o o x o o o o o o o o o o
## 2 x x o x o o o o o o o o o o
## 3 x x o o o o o o o o o o o o
## 4 o x o o o o o x o o o o o o
## 5 o x o o o o o o o o o o o o
## 6 x x o o o o o o o o o o o o
## 7 x o x o o o o o o o o o o o
```

```
sub = armasubsets(y=diffs,nar=7,nma=7, y.name='test', ar.method='ols')
plot(sub)
```



For the differenced series, a seasonal period of 6 is being suggested, which doesn't seem to make much sense.

## Model Fitting

```
# fit an AR(1) process
AR1model = arima(savings, order = c(1, 0, 0), seasonal = list(order = c(0, 0, 0)), method=c('ML'))
AR1model
```

```
##
## Call:
## arima(x = savings, order = c(1, 0, 0), seasonal = list(order = c(0, 0, 0)),
##      method = c("ML"))
##
## Coefficients:
##      ar1  intercept
##      0.8250    6.2768
## s.e.  0.0652    0.3543
##
## sigma^2 estimated as 0.336:  log likelihood = -68.71,  aic = 141.43
```

Here are our parameters for the AR(1) model. Given these parameters, the AR(1) model is  $Y_t - 6.28 = .825(Y_{t-1} - 6.28) + e_t$ .

```
# fit the seasonal model
SAR1model = arima(savings, order = c(1, 0, 0), seasonal = list(order = c(1, 0, 0), period = 4), method=
```

```
SAR1model
```

```
##
## Call:
## arima(x = savings, order = c(1, 0, 0), seasonal = list(order = c(1, 0, 0), period = 4),
##       method = c("ML"))
##
## Coefficients:
##          ar1      sar1  intercept
##       0.8610  -0.2276    6.2710
## s.e.  0.0596   0.1157    0.3514
##
## sigma^2 estimated as 0.3198:  log likelihood = -66.87,  aic = 139.75
```

```
diffs = savings - zlag(savings)
```

```
diffs = diffs[1:78]
```

```
SAR6model = arima(diffs, order = c(0, 0, 0), seasonal = list(order = c(1, 0, 1), period = 6), method=c("ML"))
SAR6model
```

```
##
## Call:
## arima(x = diffs, order = c(0, 0, 0), seasonal = list(order = c(1, 0, 1), period = 6),
##       method = c("ML"))
##
## Coefficients:
##          sar1      sma1  intercept
##       -0.1761  -0.0902    0.0311
## s.e.   0.3871   0.3868    0.0524
##
## sigma^2 estimated as 0.3386:  log likelihood = -67.78,  aic = 141.56
```

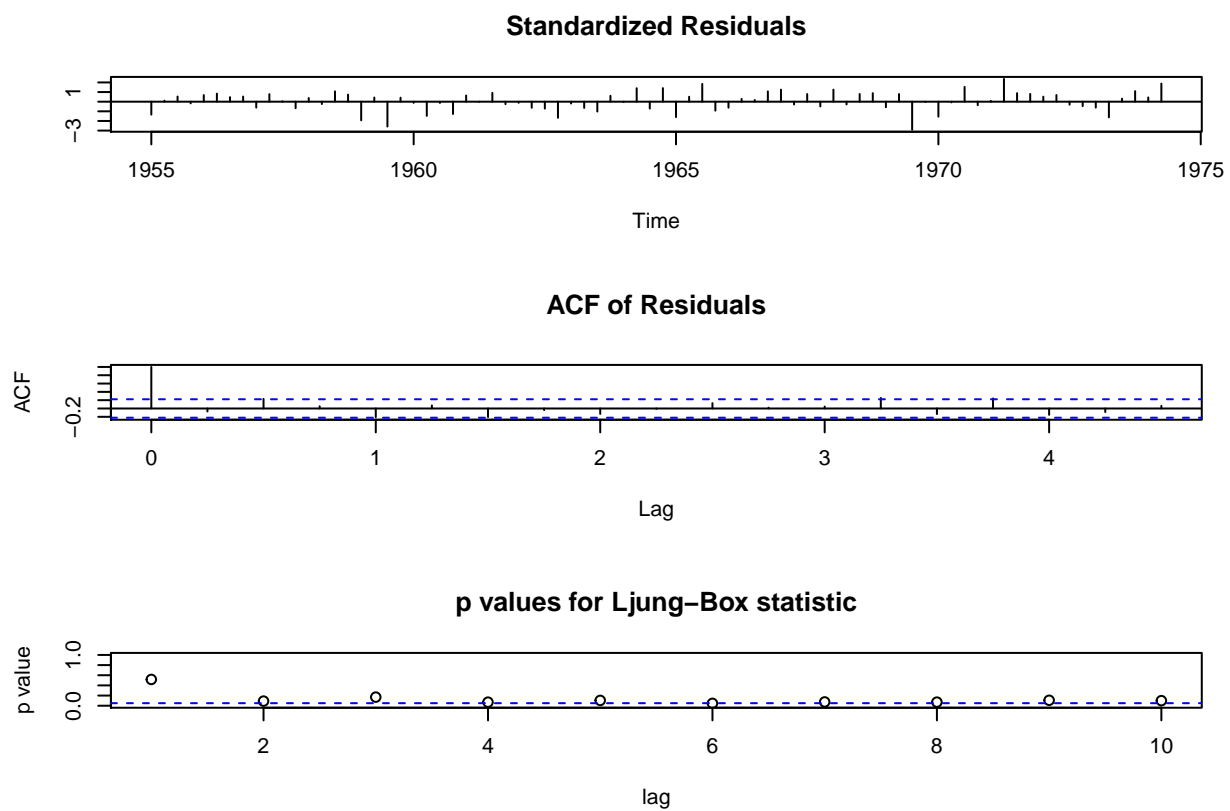
And here are the parameters for the multiplicative seasonal model. Given these parameters, the seasonal model is given by  $(Y_t - 6.27)(1 - .861(B - 6.27))(1 + .228(B - 6.27)^4) = e_t$

## Diagnostics

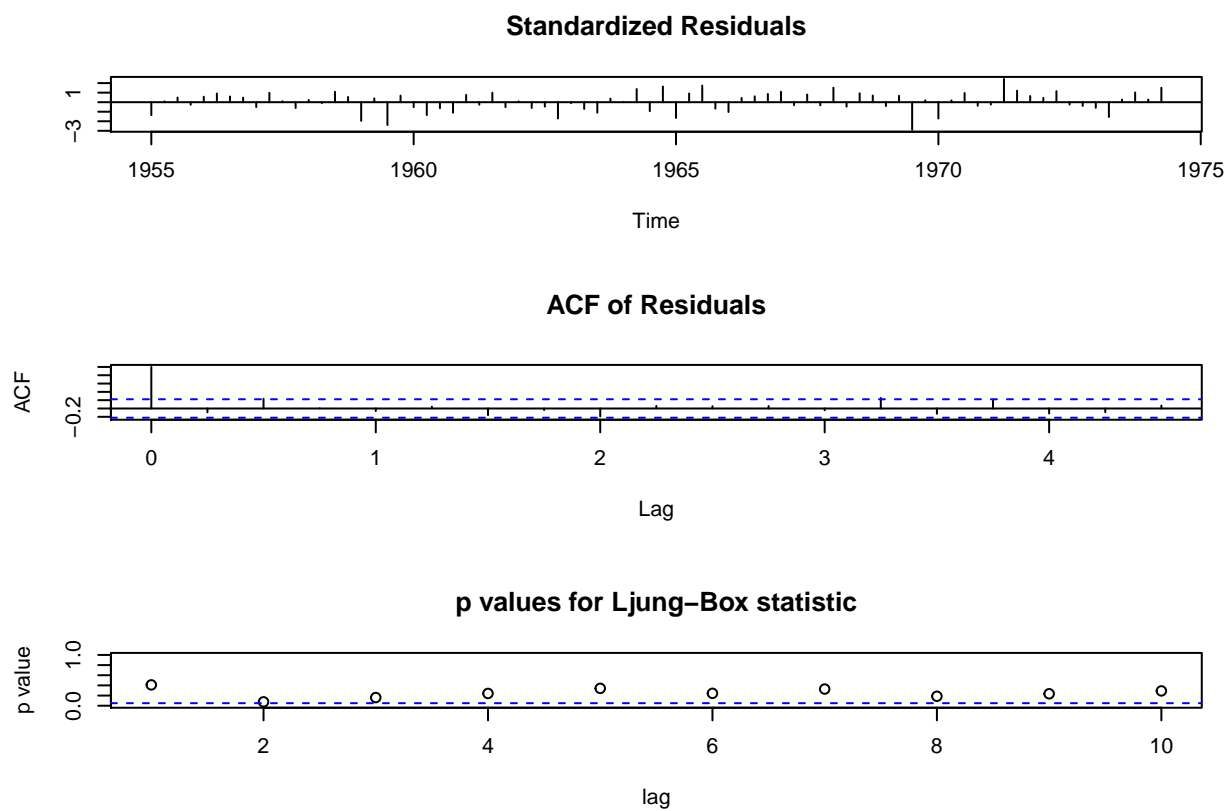
These two models are almost indistinguishable in terms of error. The AR(1) model has a higher AIC and BIC, but the seasonal model has a standard error, and none of the differences between the two models are substantial. Our error measurements don't give us much insight into which model is best, so we should turn to forecasting to see which model is more accurate in predicting future terms.

```
tsdiag(AR1model)
```

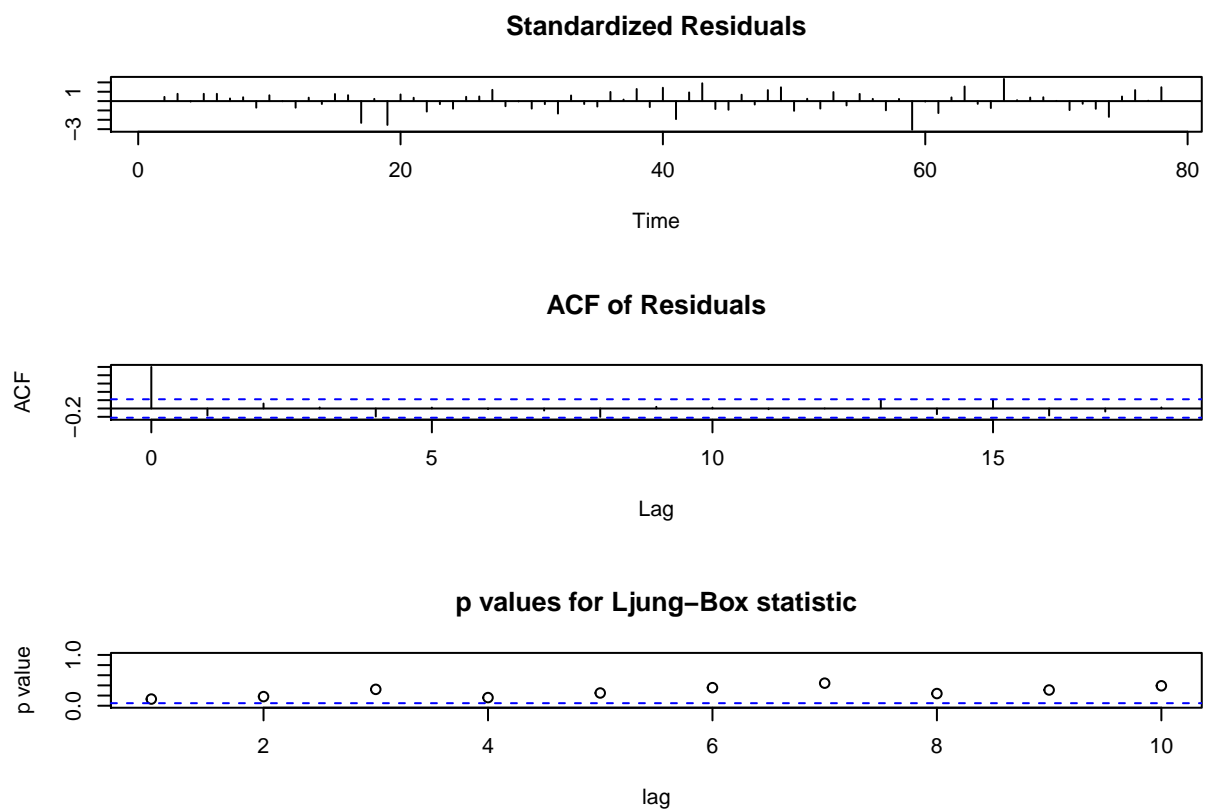




```
tsdiag(SAR1model)
```



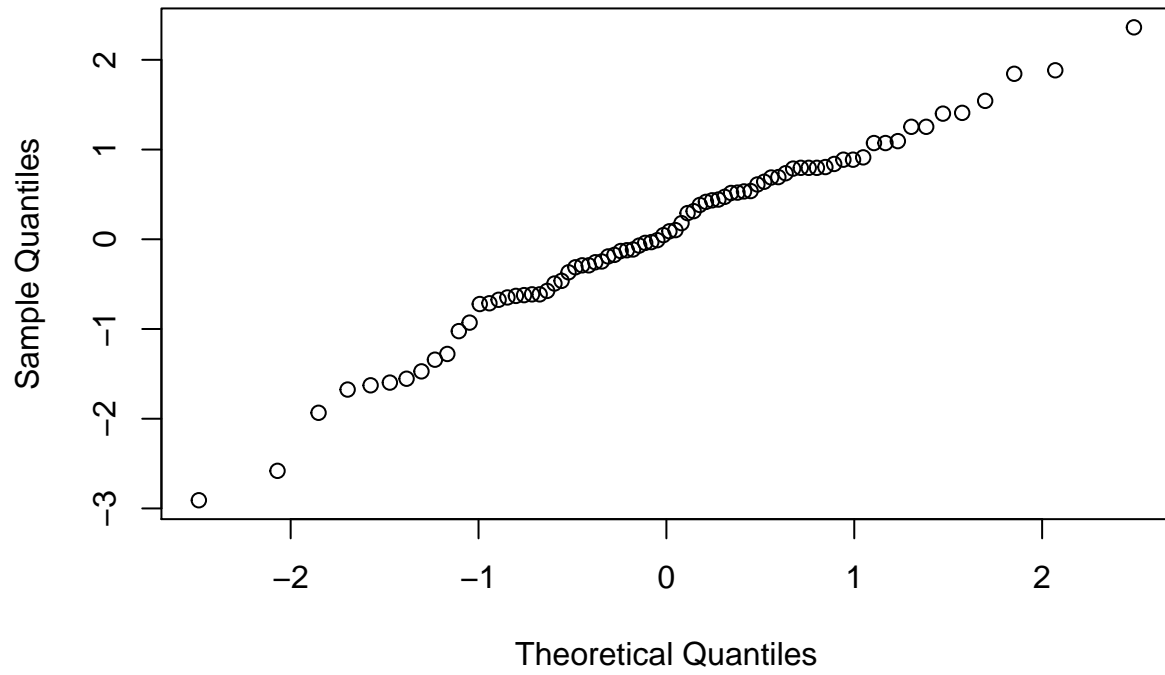
```
tsdiag(SAR6model)
```



```
ARresids = rstandard(AR1model)
SARresids = rstandard(SAR1model)
SAR6resids = rstandard(SAR6model)

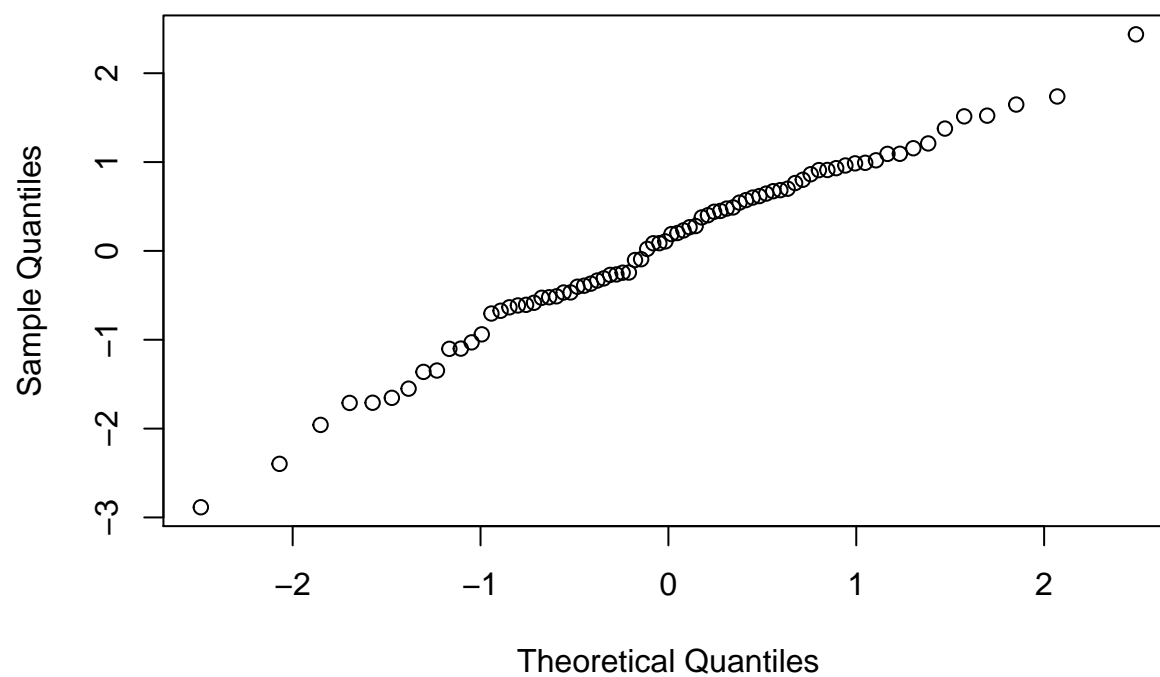
qqnorm(ARresids)
```

Normal Q-Q Plot



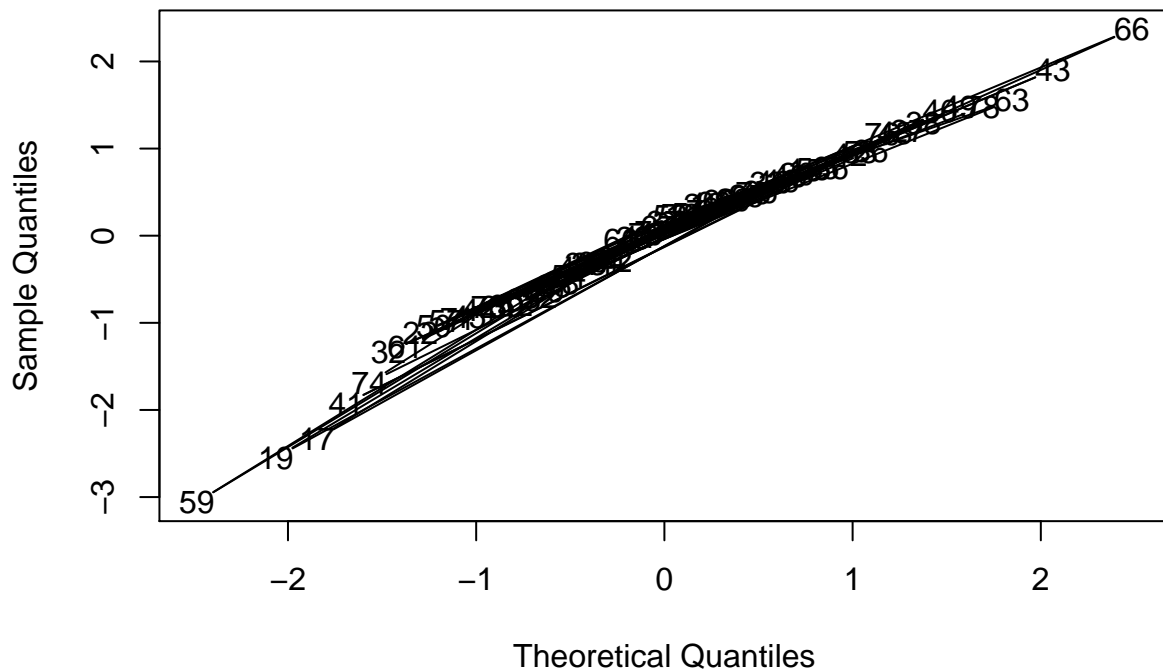
```
qqnorm(SARresids)
```

Normal Q-Q Plot



```
qqnorm(SAR6resids)
```

## Normal Q-Q Plot



```
shapiro.test(ARresids)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  ARresids
## W = 0.98097, p-value = 0.2951
```

```
shapiro.test(SARresids)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  SARresids
## W = 0.98267, p-value = 0.3691
```

```
shapiro.test(SAR6resids)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  SAR6resids
## W = 0.97932, p-value = 0.2427
```

```
runs(ARresids)
```

```
## $pvalue
## [1] 0.729
##
```

```
## $observed.runs
## [1] 42
##
## $expected.runs
## [1] 39.97436
##
## $n1
## [1] 38
##
## $n2
## [1] 40
##
## $k
## [1] 0
```

```
runs(SARresids)
```

```
## $pvalue
## [1] 0.368
##
## $observed.runs
## [1] 44
##
## $expected.runs
## [1] 39.58974
##
## $n1
## [1] 35
##
## $n2
## [1] 43
##
## $k
## [1] 0
```

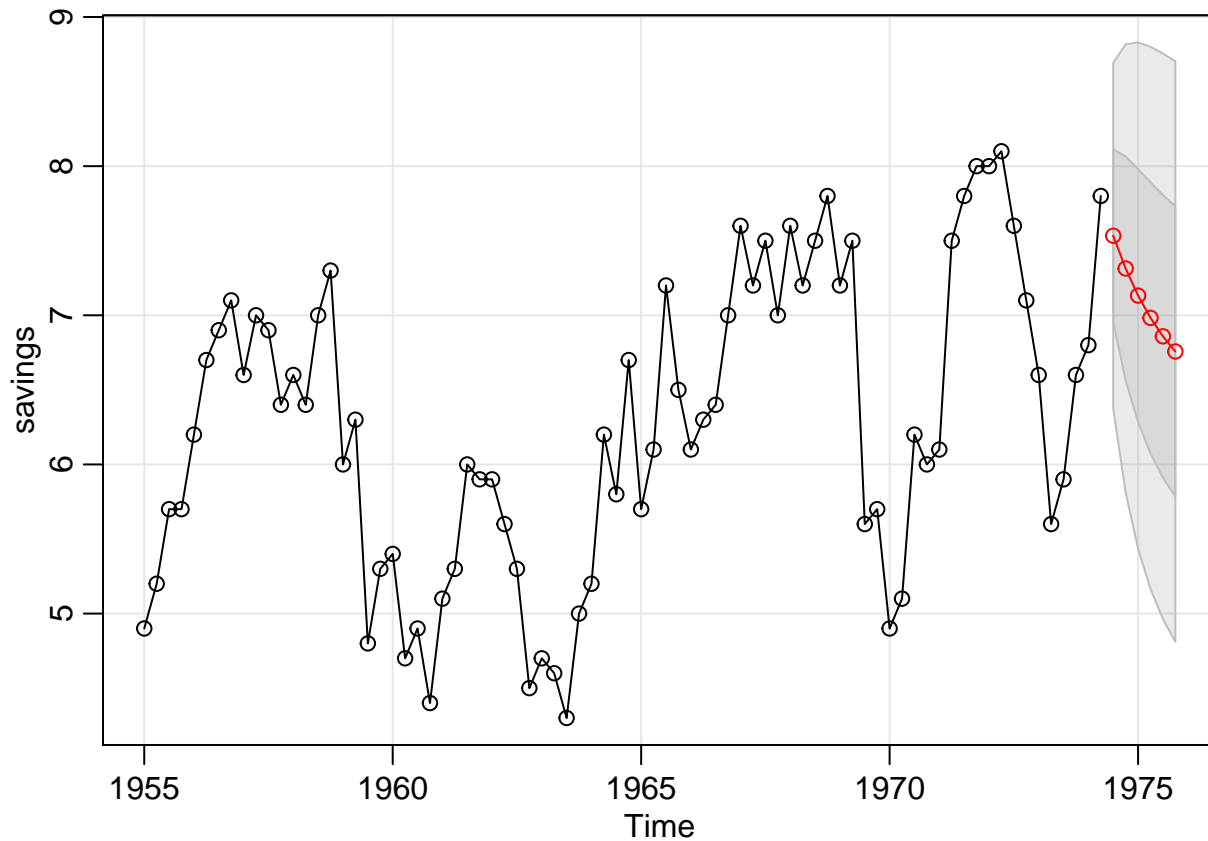
```
#runs(SAR6resids)
```

## Forecasting

```
sarima.for(savings,n.ahead = 6, 1, 0, 0, P = 0, D = 0, Q = 0, S = 0)
```

```
## $pred
##      Qtr1      Qtr2      Qtr3      Qtr4
## 1974              7.533497 7.313623
## 1975 7.132218 6.982552 6.859072 6.757196
##
## $se
##      Qtr1      Qtr2      Qtr3      Qtr4
## 1974              0.5796561 0.7514738
## 1975 0.8487602 0.9090457 0.9478906 0.9734455
```

```
points(79:84,savings.test, col="blue")
```

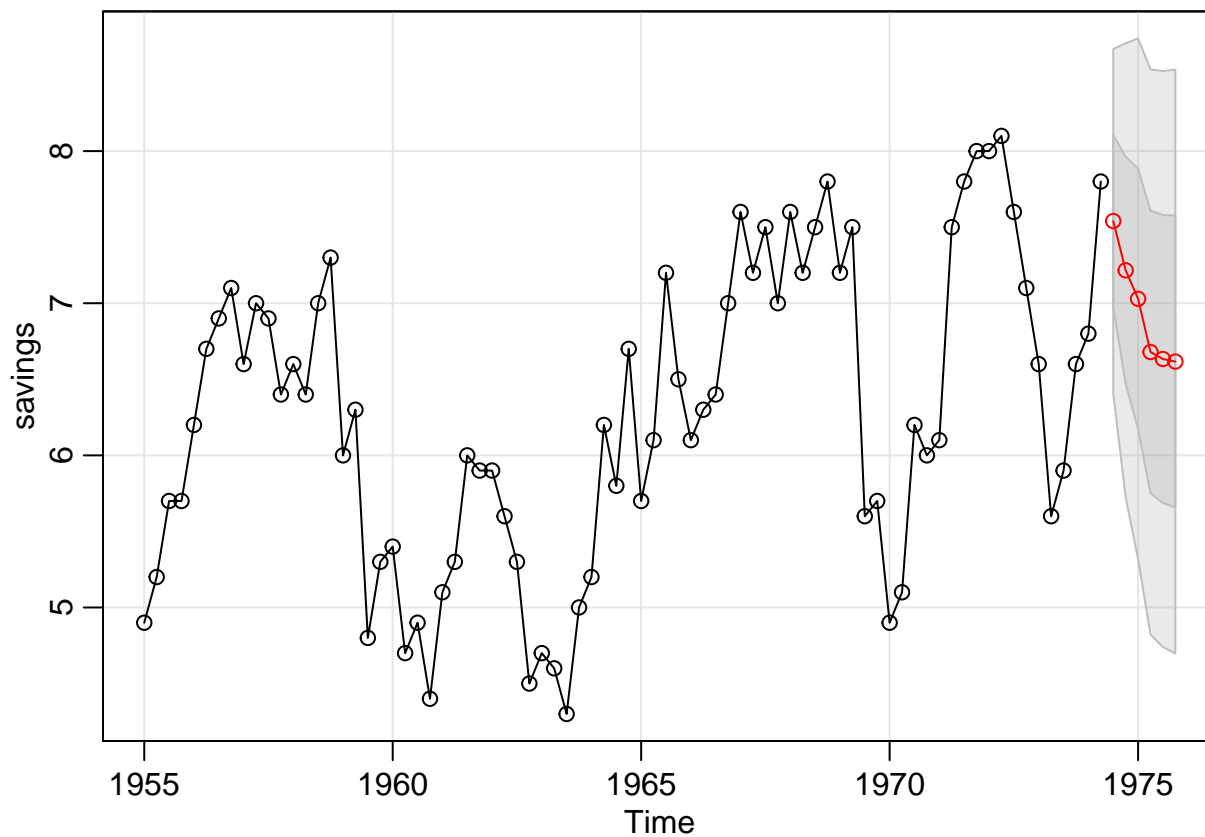


```
sarima.for(savings,n.ahead = 6, 1, 0, 0, P = 1, D = 0, Q = 0, S = 4)
```

```
## $pred
##      Qtr1      Qtr2      Qtr3      Qtr4
## 1974              7.540401 7.216360
## 1975 7.029013 6.679309 6.633252 6.616476
##
## $se
##      Qtr1      Qtr2      Qtr3      Qtr4
## 1974              0.5654719 0.7461892
## 1975 0.8558728 0.9288601 0.9465329 0.9594238
```

```
points(79:84,savings.test, col="blue")
```





## Appendix: R Code

```
# load the time series
savings = read.csv("savings.csv",header=TRUE, nrows=104)
savings = savings %>% select(2)
savings.entire = savings[1:84,]
savings<-ts(savings, start=c(1955),frequency=4)

# plot the original time series
plot(savings, xlab="Year (by Quarter)",
      ylab= "Percent of Disposable Income",
      main= "Time Series of Personal Savings in the US (1955-1980)")
points(y=savings,x=as.vector(time(savings)),pch=as.vector(season(savings)), cex=.75)
abline(mean(savings),0)

# set aside points to validate our forecasts
savings.test = savings[79:84,]

# remove the last 5 years because of the huge dip due to recession
savings = savings[1:78,]
savings<-ts(savings, start=c(1955),frequency=4)

# plot the shortened time series with labels for quarters
plot(savings, xlab="Year (by Quarter)",
```

```

    ylab= "Percent of Disposable Income",
    main= "Time Series of Personal Savings in the US (1955-1980)")
points(y=savings,x=as.vector(time(savings)),pch=as.vector(season(savings)), cex=.75)

# should we do a transformation?
boxcox = BoxCox.ar(savings)
boxcox
boxcox$mle

# plot the acf and pacf of the original series
acf(savings, lag.max = 20)
pacf(savings, lag.max = 20)

# use the eacf and best subsets to find a candidate model
eacf(savings)
sub = armasubsets(y=savings,nar=7,nma=7, y.name='test', ar.method='ols')
plot(sub)

```