207, HW3

1

Invertible MA(1), iid WN process $\{Z_t\}$ w/ σ^2. Let $\boldsymbol{X} = \{X_n, X_{n-1}, ...\}$.

a

$$\tilde{X}_{n+1} = \mathbb{E}[X_{n+1}|\boldsymbol{X}]$$
$$= \mathbb{E}[Z_{n+1} + \theta Z_n|\boldsymbol{X}]$$
$$\stackrel{lin}{=} \mathbb{E}[Z_{n+1}|\boldsymbol{X}] + \theta\mathbb{E}[Z_n|\boldsymbol{X}]$$

$\mathbb{E}[Z_{n+1}|\boldsymbol{X}] \stackrel{\perp}{=} \mathbb{E}[Z_{n+1}] = 0$

$$= \theta\mathbb{E}[Z_n|\boldsymbol{X}]$$

And now using the invertibility of X_n, $Z_n = \sum_{j=0}^{\infty} \pi_j X_{n-j}$,

$$= \theta\mathbb{E}[\sum_{j=0}^{\infty} \pi_j X_{n-j}|\boldsymbol{X}]$$

Which is deterministic given the infinite previous values. Thus, $\{Z_n|\boldsymbol{X}\}$ is deterministic as well, and:

$$\theta\mathbb{E}[Z_n|\boldsymbol{X}] = \theta Z_n$$

b

Derive MSE, $\mathbb{E}[(\tilde{X}_{n+1} - X_{n+1})^2]$:

$$\mathbb{E}[(\tilde{X}_{n+1} - X_{n+1})^2] = E[(\tilde{X}_{n+1} - \mathbb{E}[X_{n+1}|\boldsymbol{X}] + \mathbb{E}[X_{n+1}|\boldsymbol{X}] - X_{n+1})^2]$$

$\tilde{X}_{n+1} - \mathbb{E}[X_{n+1}|\boldsymbol{X}] = 0$ from part (a):

$$= \mathbb{E}[(\mathbb{E}[X_{n+1}|\boldsymbol{X}] - X_{n+1})^2]$$

And using the result from (a) and definition of X_{n+1}:

$$= \mathbb{E}[(\theta Z_n - Z_{n+1} - \theta Z_n)^2]$$
$$= \mathbb{E}[Z_{n+1}^2] = \sigma^2$$

2

Invertible MA(q) for WN $\{Z_t\}$ w/ σ^2 and uncorrelated.

a

Using Theorem 5.28, best linear prediction of X_{n+m} based upon $\boldsymbol{X} = a_1 X_1 + ... + a_n X_n$ (0 means and finite second moments) is $\boldsymbol{a}^* = \Delta^{-1}\xi$. But $\xi_i = Cov[X_{n+m}, X_i], i \leq n$ and:

$$Cov[X_{n+m}, X_i] = Cov[\sum_{k=0}^{q} \theta_k Z_{n+m-k}, \sum_{j=0}^{q} \theta_j Z_{i-j}]$$

$$= \sum_{k=0}^{q} \sum_{j=0}^{q} \theta_k \theta_j Cov[Z_{n+m-k}, Z_{i-j}]$$

But $\forall i \leq n$ and $m > q$, $n + m - k \neq i - j, \forall j, k \in [0, q]$. Further, WN is uncorrelated yielding 0 for each summand. Thus,

$$Cov[X_{n+m}, X_i] = 0, \forall i \leq n, m > q$$

$$\implies \xi_i = 0 \forall i \leq n$$

$$\implies \boldsymbol{a}^* = 0$$

$\implies$ BLP of X_{n+m} based upon $\boldsymbol{X} = a_1 X_1 + ... + a_n X_n$ is $(\boldsymbol{a^*})^T \boldsymbol{X} = 0$

b

Now WN is iid. Using Theorem 5.27, best mean squared error prediction of X_{n+m} based upon $\boldsymbol{X} = \{X_n, ..., X_1\}$ is $\mathbb{E}[X_{n+m}|\boldsymbol{X}]$.

$$\mathbb{E}[X_{n+m}|\boldsymbol{X}] = \mathbb{E}[\theta(B) Z_{n+m}|\boldsymbol{X}]$$

$$= \mathbb{E}[Z_{n+m} + \theta Z_{n+m-1} + ... + \theta^q Z_{n+m-q}|\boldsymbol{X}]$$

Using $\mathbb{E}[X|U] = \mathbb{E}[\mathbb{E}[X|Y, U]|U]$,

$$= \mathbb{E}[\mathbb{E}[Z_{n+m} + \theta Z_{n+m-1} + ... + \theta^q Z_{n+m-q}|\boldsymbol{X}, \boldsymbol{Z}]|\boldsymbol{X}]$$

But $\boldsymbol{X} = \{X_n, ..., X_1\} = \{\theta(B) Z_n, ..., \theta(B) Z_1\} = \{Z_n + \theta Z_{n-1} + \theta^q Z_{n-q} ..., Z_0\}$
$\implies$ Fixing $\{\boldsymbol{X}, \boldsymbol{Z}\}$ is the same as fixing just $\{\boldsymbol{Z}\}$, for $\boldsymbol{Z} = \{Z_n, ..., Z_1\}$

$$\therefore \mathbb{E}[\mathbb{E}[Z_{n+m} + \theta Z_{n+m-1} + ... + \theta^q Z_{n+m-q}|\boldsymbol{X}, \boldsymbol{Z}]|\boldsymbol{X}] = \mathbb{E}[\mathbb{E}[Z_{n+m} + \theta Z_{n+m-1} + ... + \theta^q Z_{n+m-q}|\boldsymbol{Z}]|\boldsymbol{X}]$$

But since we assumed iid, for $m > q$, these $q + 1$ terms are $\perp$ of $\boldsymbol{Z}$, yielding the unconditional expectations:

$$\mathbb{E}[Z_{n+m} + \theta Z_{n+m-1} + ... + \theta^q Z_{n+m-q}|\boldsymbol{Z}] \overset{iid/lin.}{=} \mathbb{E}[Z_{n+m}] + \theta \mathbb{E}[Z_{n+m-1}] + ... + \theta^q \mathbb{E}[Z_{n+m-q}] = 0$$

Thus,

$$\mathbb{E}[\mathbb{E}[Z_{n+m} + \theta Z_{n+m-1} + ... + \theta^q Z_{n+m-q}|\boldsymbol{Z}]|\boldsymbol{X}] = \mathbb{E}[0|\boldsymbol{X}] = 0 \implies \mathbb{E}[X_{n+m}|\boldsymbol{X}] = 0$$

3

a

Casual, zero mean, AR(1) for WN $\{Z_t\}$ w/ σ^2 and uncorrelated. Using Theorem 5.28, best linear prediction of X_{n+m} based upon $a_1X_1 + ... + a_nX_n$ (0 means and finite second moments) is:

$$\boldsymbol{a}^* = \Delta^{-1}\xi$$

$$\xi_i = Cov[X_{n+m}, X_i], \Delta_{i,j} = Cov[X_i, X_j]$$

Rearranging terms to $\Delta\boldsymbol{a} = \xi$ and expressing in Toeplitz form we get:

$$\begin{bmatrix} \gamma(0) & \gamma(1) & \gamma(2) & \gamma(n-1) \\ \gamma(1) & \gamma(0) & \gamma(1) & \vdots \\ \gamma(2) & \gamma(1) & \gamma(0) & \vdots \\ \vdots & \vdots & \vdots & \gamma(1) \\ \gamma(n-1) & \cdots & \gamma(1) & \gamma(0) \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} \gamma(n+m-1) \\ \gamma(n+m-2) \\ \vdots \\ \gamma(m+1) \\ \gamma(m) \end{bmatrix}$$

Dividing through by $\gamma(0)$ gives,

$$\begin{bmatrix} 1 & \rho(1) & \rho(2) & \rho(n-1) \\ \rho(1) & 1 & \rho(1) & \vdots \\ \rho(2) & \rho(1) & 1 & \vdots \\ \vdots & \vdots & \vdots & \rho(1) \\ \rho(n-1) & \cdots & \rho(1) & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} \rho(n+m-1) \\ \rho(n+m-2) \\ \vdots \\ \rho(m+1) \\ \rho(m) \end{bmatrix}$$

And specifically for an AR(1) casual process, $\rho(h) = \phi^h$, thus:

$$\begin{bmatrix} 1 & \phi^1 & \phi^2 & \phi^{n-1} \\ \phi^1 & 1 & \phi^1 & \vdots \\ \phi^2 & \phi^1 & 1 & \vdots \\ \vdots & \vdots & \vdots & \phi^1 \\ \phi^{n-1} & \cdots & \phi^1 & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} \phi^{n+m-1} \\ \phi^{n+m-2} \\ \vdots \\ \phi^{m+1} \\ \phi^m \end{bmatrix}$$

Which has a solution at $a_n = \phi^m$ and $a_i = 0, \forall i \neq n$

$$\implies \tilde{X}_{n+m} = \phi^m X_n$$

b

$$\mathbb{E}[(X_{n+m} - \tilde{X}_{n+m})^2] = \mathbb{E}[(X_{n+m} - \phi^m X_n)^2]$$
$$= \mathbb{E}[X_{n+m}^2] - 2\phi^m Cov[X_{n+m}, X_n] + \phi^{2m}\mathbb{E}[X_n^2]$$
$$= \gamma(0) - 2\phi^m\gamma(m) + \phi^{2m}\gamma(0)$$

And using that for a casual AR(1) process, $\gamma(h) = \frac{\phi^h\sigma^2}{1-\phi^2}, h \geq 0$,

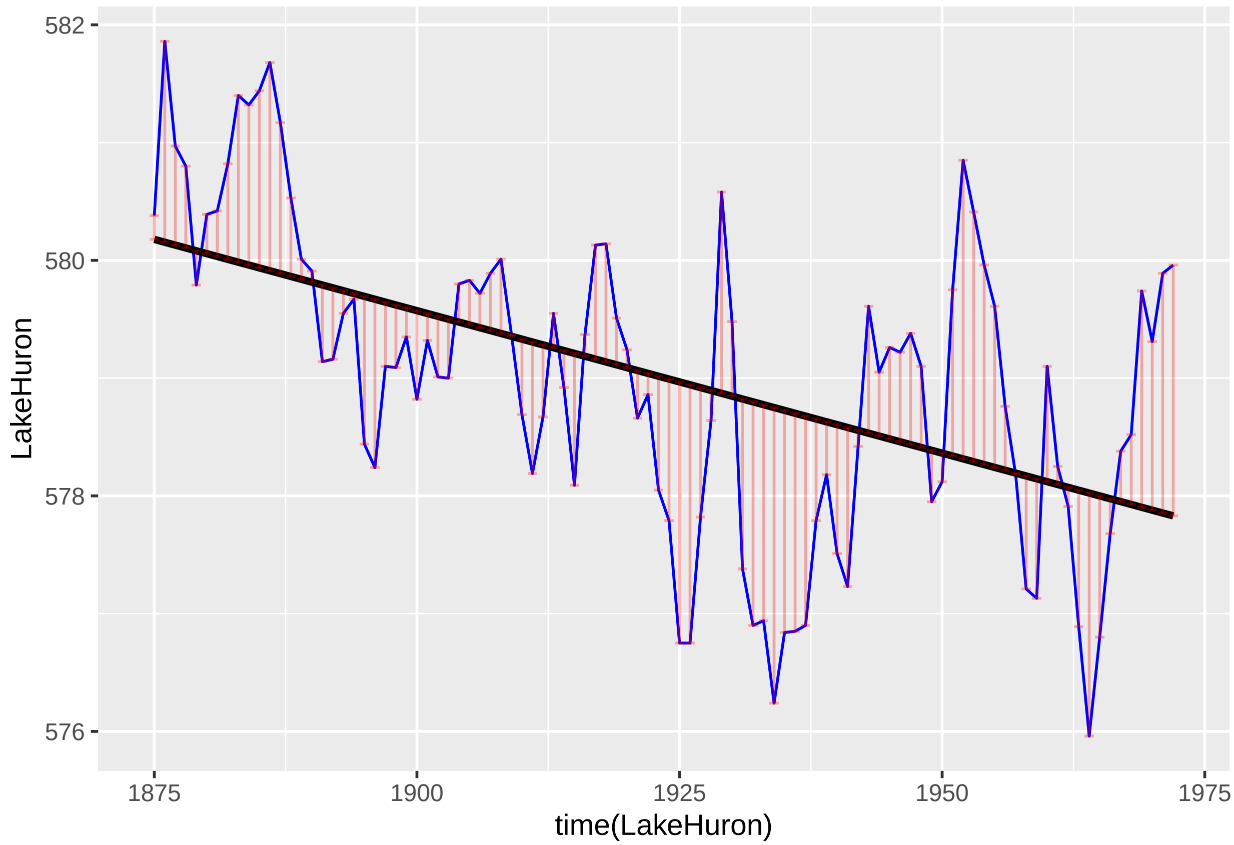
$$= \frac{\sigma^2 - 2\sigma^2\phi^{2m} + \sigma^2\phi^{2m}}{1-\phi^2}$$

$$= \sigma^2\frac{(1-\phi^{2m})}{1-\phi^2}$$

1, Prediction

a

```
library(datasets)
library(ggplot2)

trend = lm(LakeHuron~c(1:length(LakeHuron)))
ggplot(data.frame(LakeHuron, trend = trend$fitted.values, residuals = trend$residuals), aes(x = time(La
  geom_line(col = "blue")+ geom_line(aes(y = trend), size= 1.3) + geom_errorbar(aes(ymax = trend + resid
```



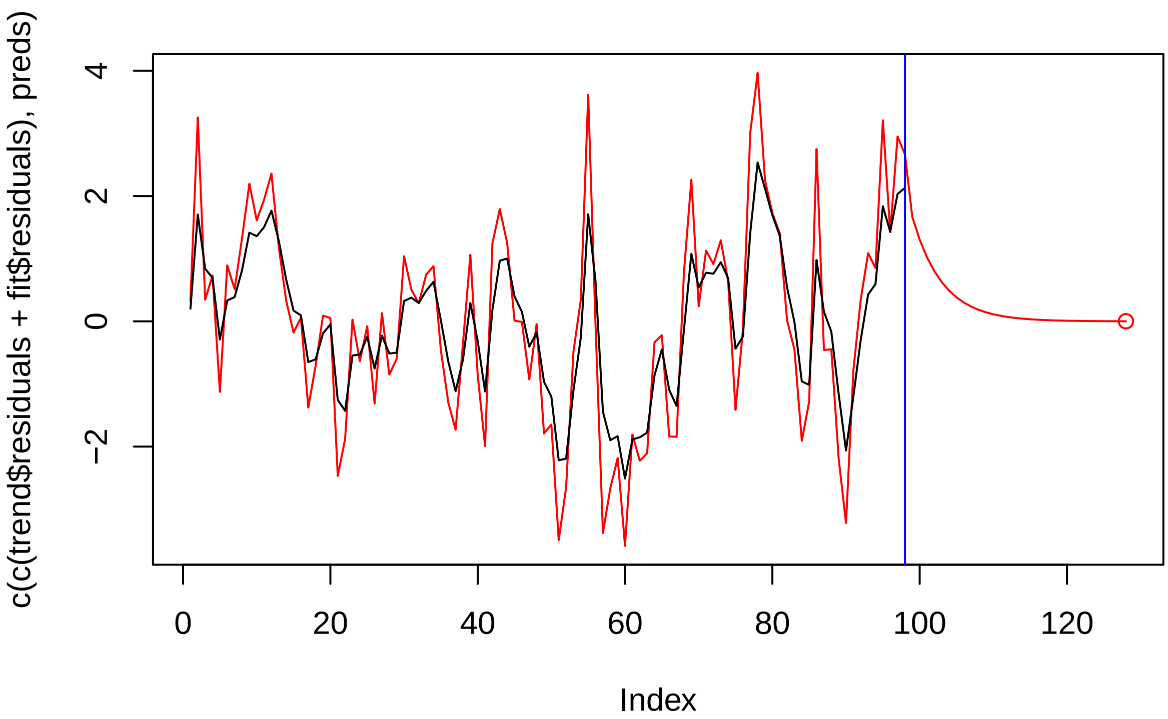
b

$$X_t = \rho X_{t-1} + Z_t$$

c

$$\tilde{X}_{n+m} = \phi^m X_n$$

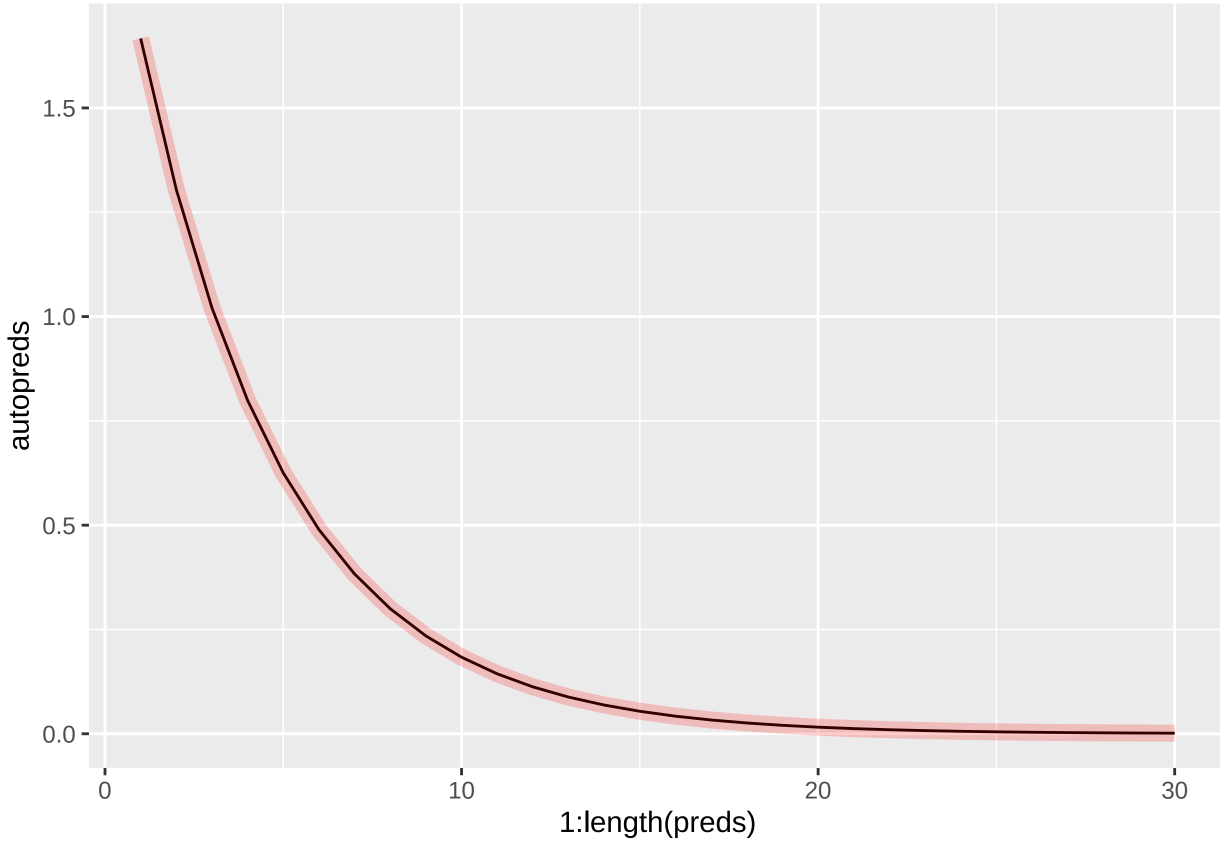
```
fit = arima(trend$residuals, order = c(1,0,0), include.mean = FALSE)
phi = fit$coef[[1]]
preds = sapply(1:30, function(x) trend$residuals[length(trend$residuals)] * phi^x)
plot(c(c(trend$residuals + fit$residuals),preds), type = 'l', col = 'red')
lines(trend$residuals, type = 'l')
abline(v = length(trend$residuals), col = 'blue')
points(length(trend$residuals) + length(preds), tail(preds, n =1), col = 'red')
```



d

```
ggplot(data.frame(autopreds = predict(fit, n.ahead = 30)$pred, manual = preds), aes(x = 1:length(preds)
  geom_line(aes(y = autopreds)) +
  geom_line(aes(y = manual), alpha = .2, col = 'red', size = 3)
```

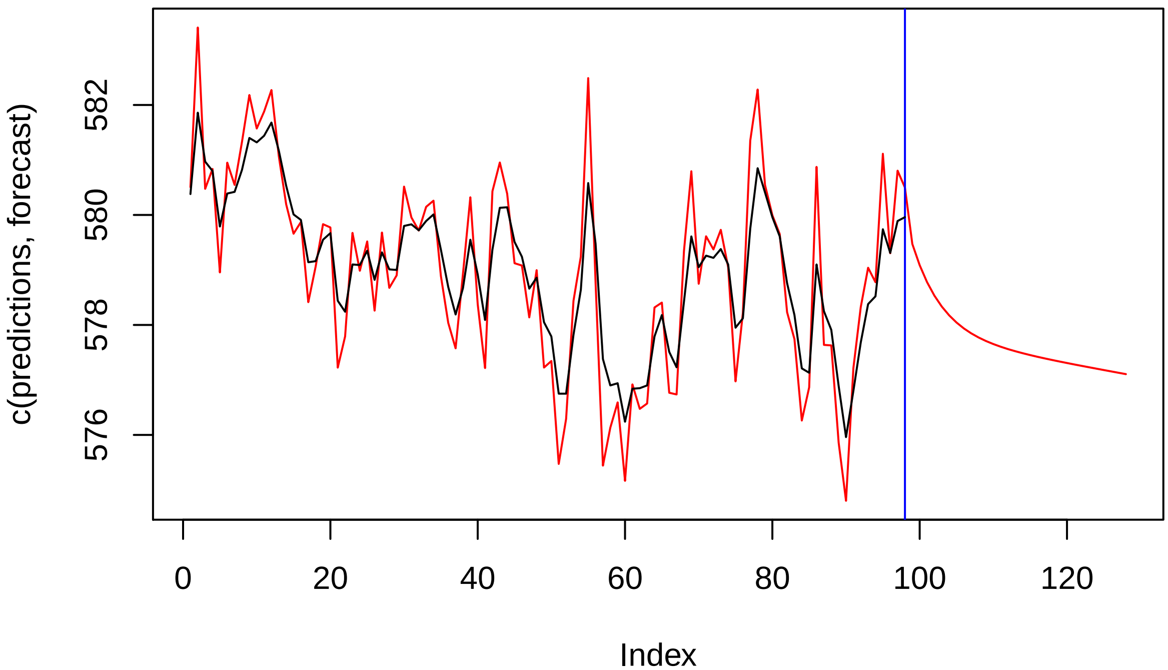
```
## Don't know how to automatically pick scale for object of type ts. Defaulting to continuous.
```



The predictions are identical from R compared to the manually calculated predictions.

e

```
predictions = trend$fitted.values + trend$residuals + fit$residuals
forecast = sapply(length(LakeHuron)+ 1:30, function(x) x*trend$coefficients[[2]] + trend$coefficients[[
plot(c(predictions, forecast), type = "l", col = "red")
abline(v = length(trend$residuals), col = 'blue')
lines(1:length(LakeHuron), LakeHuron)
```



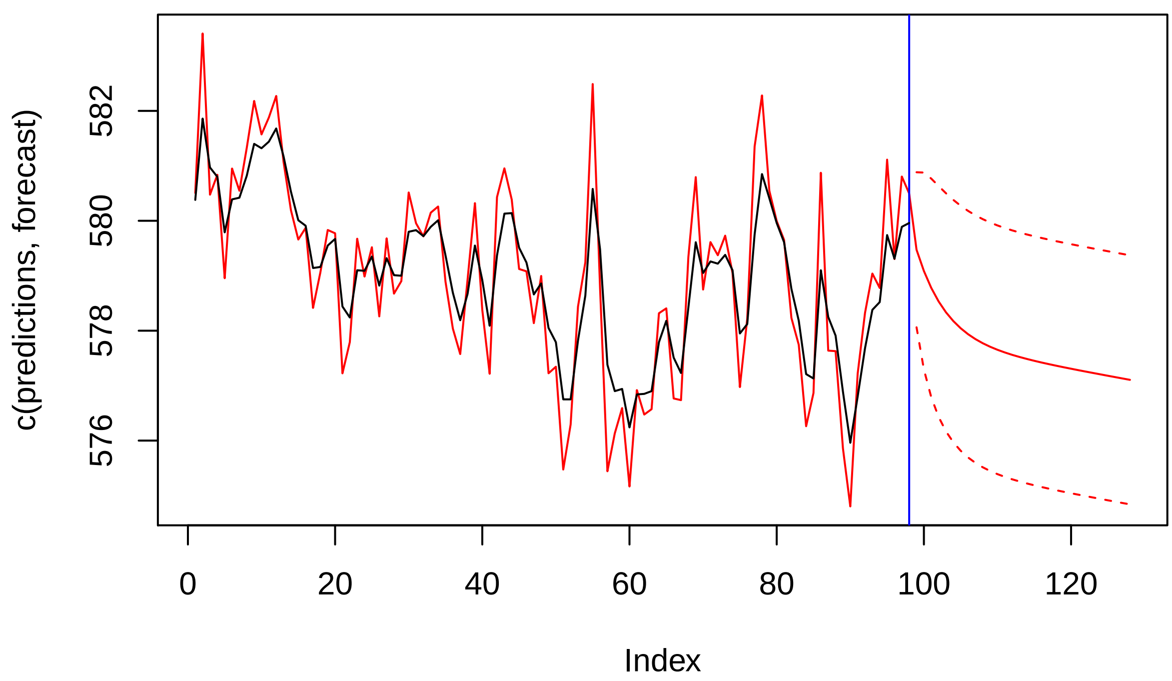
f

$$\mathbb{E}[(X_{n+m} - \tilde{X}_{n+m})^2] = \sigma^2 \frac{(1-\phi^{2m})}{1-\phi^2}$$

```
manerrors = sapply(1:30, function(x) sqrt(0.4975*(1-phi^(2*x))/(1-phi^2)))
manerrors
```

```
##  [1] 0.7053368 0.8956614 0.9944034 1.0503070 1.0831230 1.1027401 1.1145847
##  [8] 1.1217775 1.1261603 1.1288363 1.1304722 1.1314729 1.1320854 1.1324604
## [15] 1.1326900 1.1328306 1.1329168 1.1329695 1.1330018 1.1330216 1.1330337
## [22] 1.1330411 1.1330457 1.1330484 1.1330501 1.1330512 1.1330518 1.1330522
## [29] 1.1330525 1.1330526
```

```
manpredsupper = forecast + 2 * manerrors
manpredslower = forecast - 2 * manerrors
plot(c(predictions, forecast), type = "l", col = "red")
abline(v = length(trend$residuals), col = 'blue')
lines(1:length(LakeHuron), LakeHuron)
lines(length(LakeHuron)+ 1:30,manpredsupper ,type="l",lty=2,col=2) ## upper bound
lines(length(LakeHuron)+ 1:30,manpredslower ,type="l",lty=2,col=2) ## lower bound
```

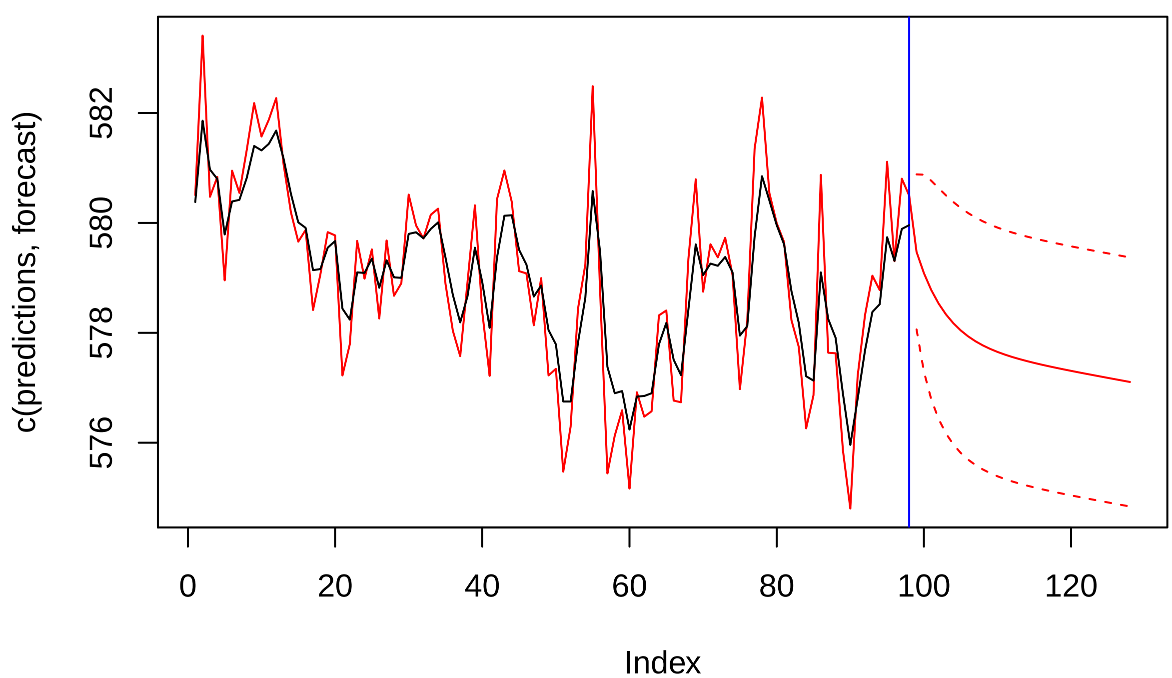


g

```r
predict(fit, n.ahead = 30)$se
```

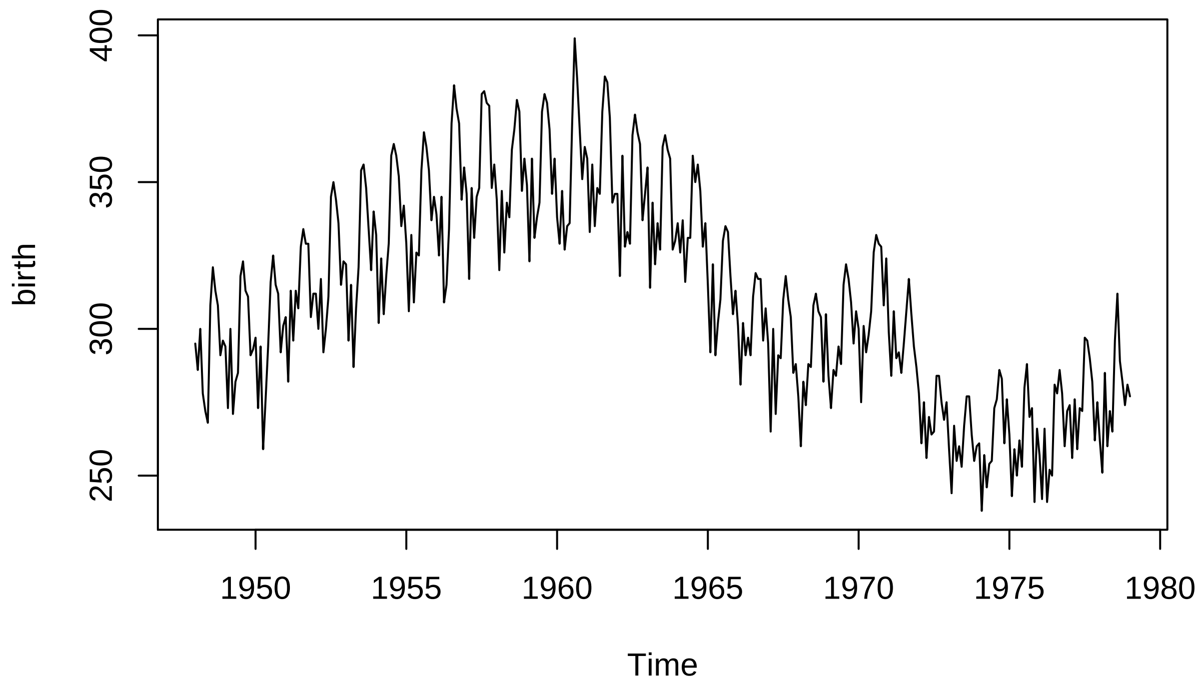
```
## Time Series:
## Start = 99
## End = 128
## Frequency = 1
##  [1] 0.7053615 0.8956928 0.9944382 1.0503438 1.0831609 1.1027787 1.1146237
##  [8] 1.1218168 1.1261997 1.1288758 1.1305117 1.1315125 1.1321250 1.1325001
## [15] 1.1327297 1.1328703 1.1329564 1.1330091 1.1330414 1.1330612 1.1330733
## [22] 1.1330808 1.1330853 1.1330881 1.1330898 1.1330908 1.1330915 1.1330919
## [29] 1.1330921 1.1330923
```

```r
predsupper = forecast + 2 * predict(fit, n.ahead = 30)$se
predslower = forecast - 2 * predict(fit, n.ahead = 30)$se
plot(c(predictions, forecast), type = "l", col = "red")
abline(v = length(trend$residuals), col = 'blue')
lines(1:length(LakeHuron), LakeHuron)
lines(predsupper ,type="l",lty=2,col=2) ## upper bound
lines(predslower ,type="l",lty=2,col=2) ## lower bound
```



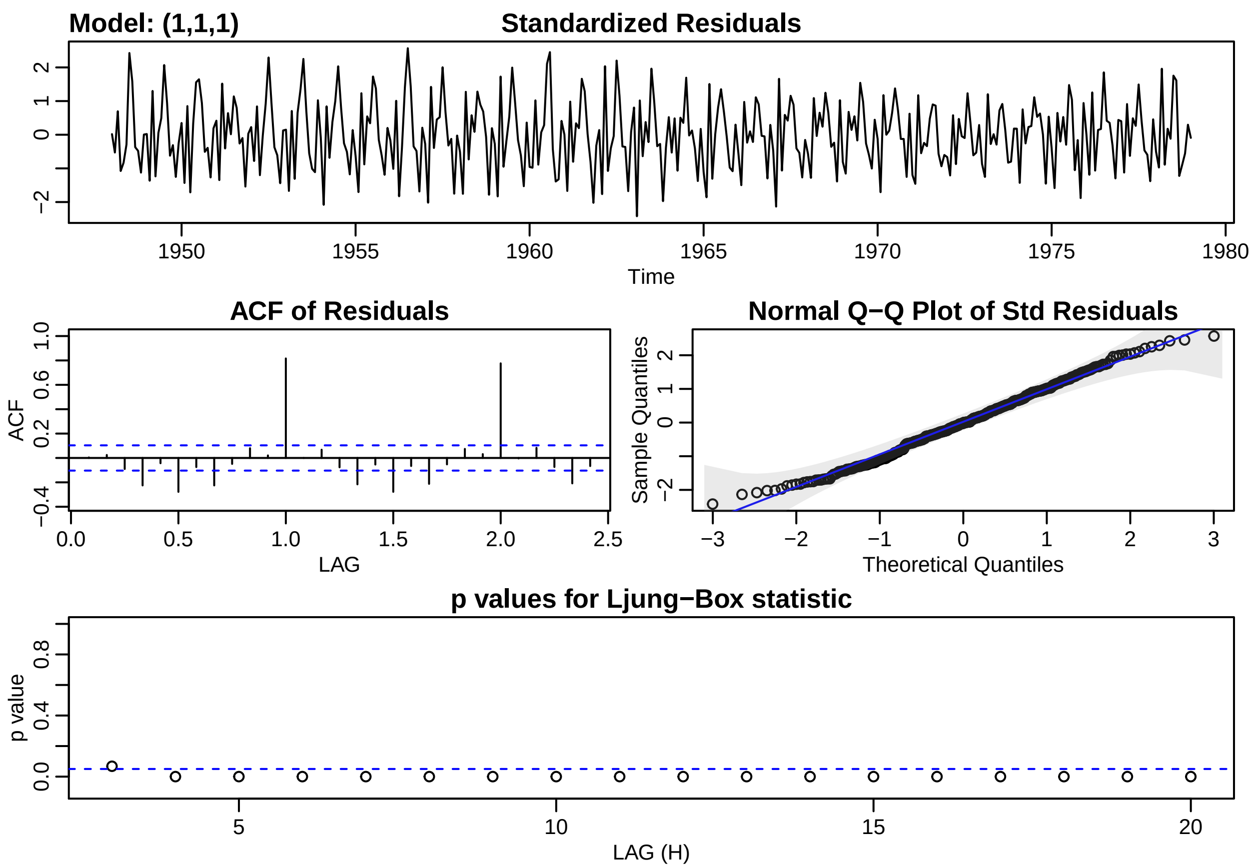
The predict function's forecasted SE appears to be identical to the theoretical values.

2 Model Fitting and Diagnostics



$$X_t = (I - B)^d Y_t, \ \phi(B)(X_t - \mu) = \theta(B) Z_t$$

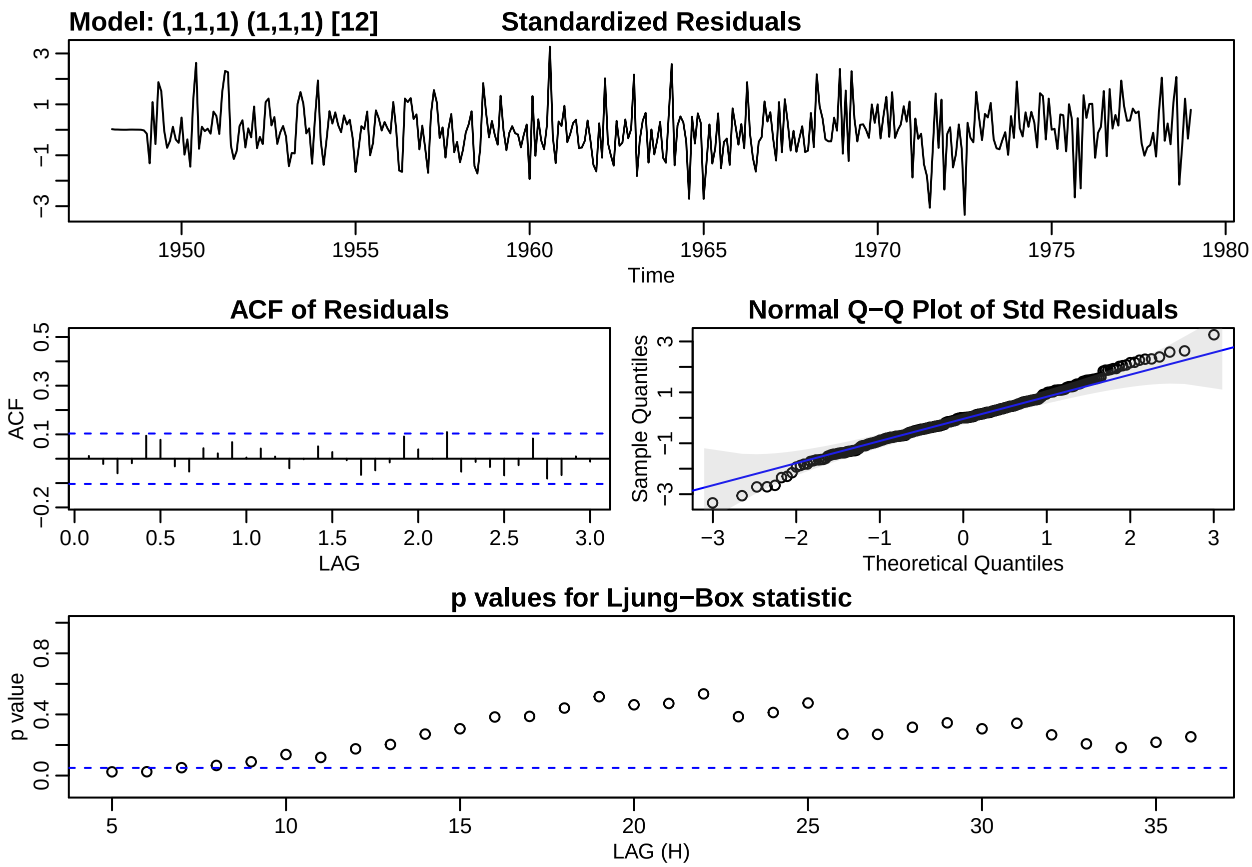
```
sarima1 = sarima(xdata = birth, 1,1,1)
```



The standardized residuals appear to have been demeaned and have homoscedastic errors. The ACF of the residuals indicate that a large amount of the serial correaltion still has not been explained. For example, lags 1 and 2 have extreme peaks as well as significant measures at several other lags. The ends of the Q-Q plot indicate that the sample tails differ from that of gaussianity. Finally, the p values for the Ljung-Box statistic show that at almost all lags, the p values are < 0.05. Thus, we can conclude at a 95% confidence interval that the sample residuals differ from that of the hypothesized distribution (χ^2_{k-p-q})

$$\Phi(B^s)\phi(B)\nabla_s^D\nabla^d Y_t = \delta + \Theta(B^s)\theta(B)Z_t$$

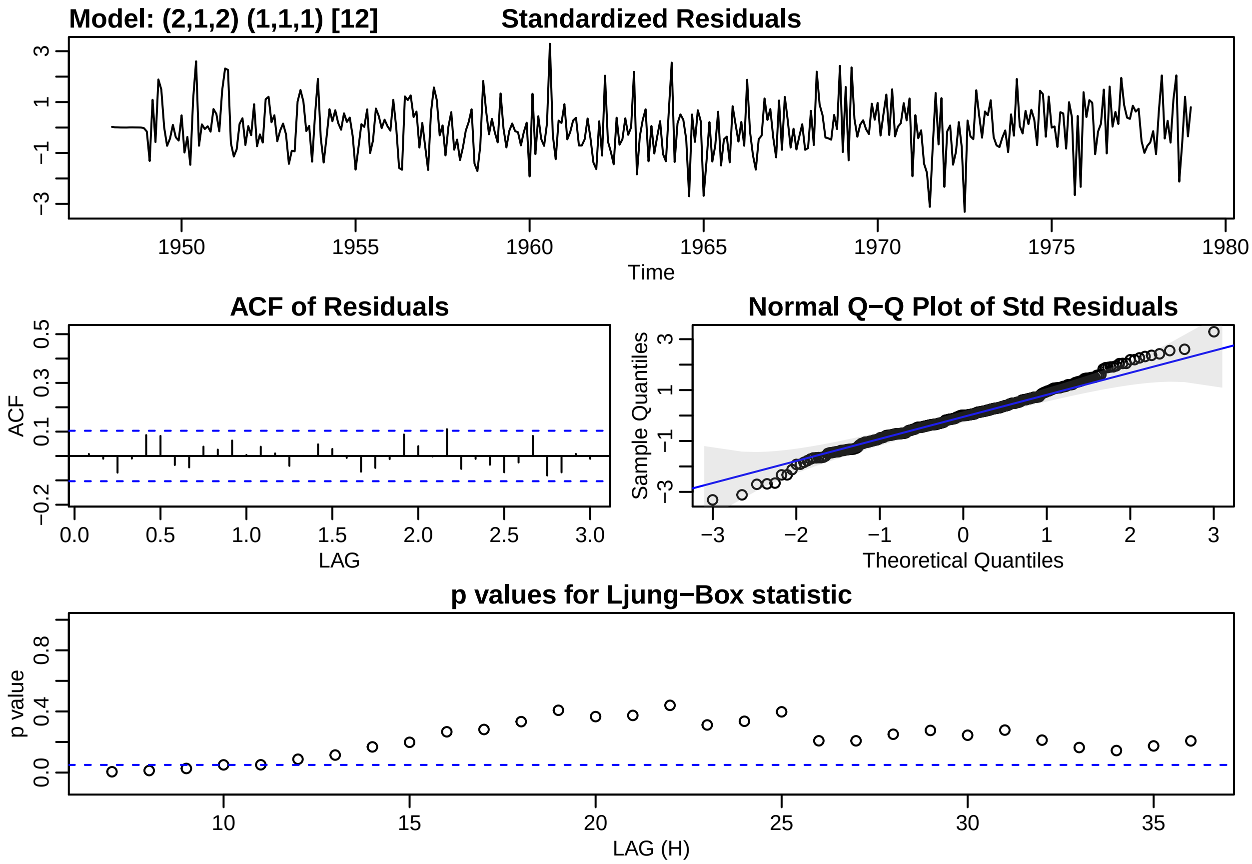
```
sarima2 = sarima(xdata = birth, 1,1,1, 1,1,1,12)
```



The standardized residuals also appear demeaned and homoscedastic. The ACF of the residuals indicate that alot more correlation has been explained by the model. The ends of the Q-Q plot still indicate the tails diverge from normality, but the majority of the plot matches the theoretical quantiles. The p values are primarily insignificant. There are some significant residuals at lags 5 and 6.

$$\Phi(B^s)\phi(B)\nabla_s^D\nabla^d Y_t = \delta + \Theta(B^s)\theta(B)Z_t$$

```
sarima3 = sarima(xdata = birth, 2,1,2, 1,1,1,12)
```

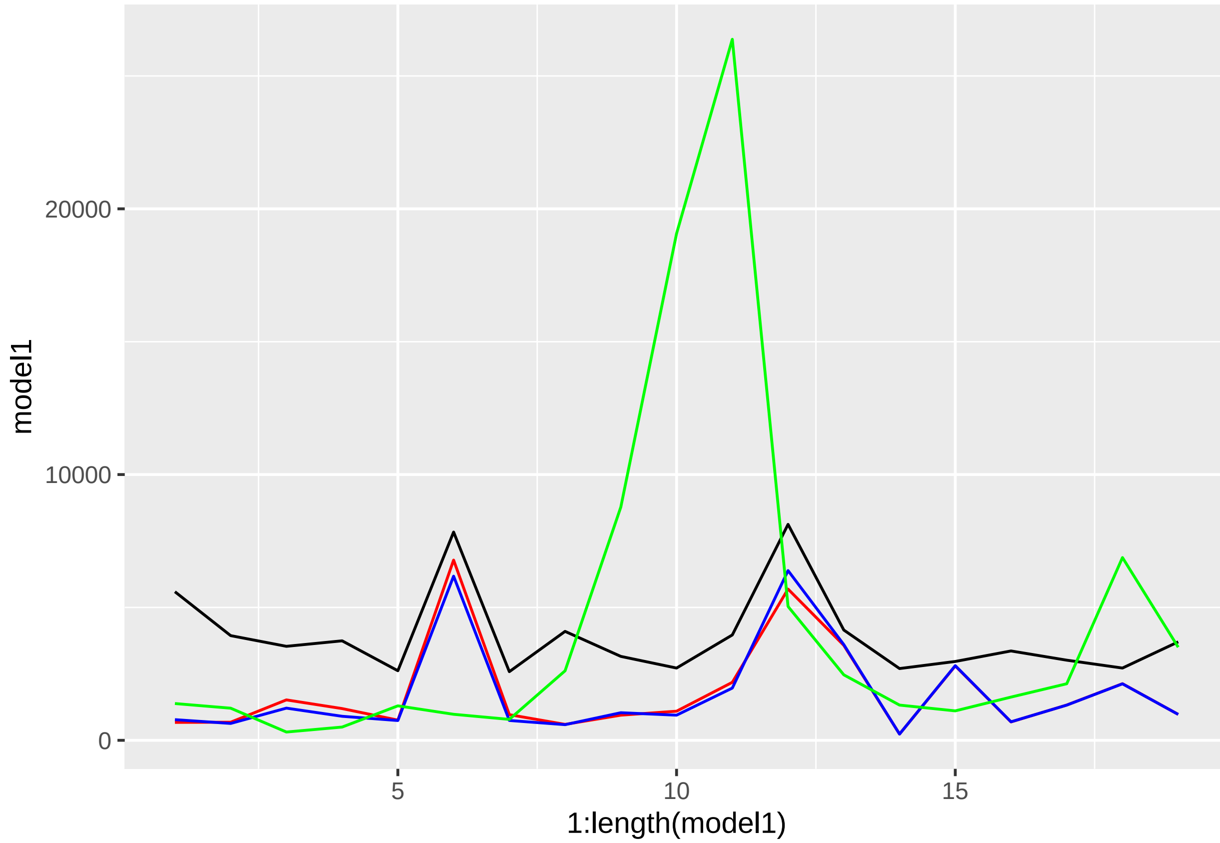


The standardized residuals do not differ from the previous plots. The ACF appears to explain the same correlation in the previous model. The Q-Q plot does not differ from before. There are slightly more signficant p values for the residuals, indicating that this model isn't an improvement in fit.

3 Model Selection

```
library(ggplot2)
start = 1948; end = 1959

model1 = c(); model2 = c(); model3 = c(); model4 = c()

for(i in 1959:1977){
  end = i
  train = window(birth, start,c(end, 12))
  newobs = window(birth, end + 1, c(end + 1, 12))

  sarima1 = sarima.for(xdata = train, n.ahead = 12, 1,1,1, plot.all = FALSE)
  model1 = c(model1,(c(sarima1$pred) - newobs)^2)

  sarima2 = sarima.for(xdata = train, n.ahead = 12, 1,1,1, P = 1, D = 1, Q = 1, S = 12, plot.all = FALSE
  model2 = c(model2,(c(sarima2$pred) - newobs)^2)

  sarima3 = sarima.for(xdata = train, n.ahead = 12, 2,1,2, P = 1, D = 1,Q = 1, S = 12, plot.all = FALSE
  model3 = c(model3,(c(sarima3$pred) - newobs)^2)

  df=data.frame(y = train,
              t = 1:length(train),
              month = as.factor(1:length(train) %% 12))
  fit = lm(y~poly(t,3) + month, df)

  df2 = data.frame(y = newobs,
              t = length(train)+ 1:length(newobs),
              month = as.factor(1:length(newobs)%% 12))
  model4 = c(model4, (df2$y - predict(fit, newdata = df2))^2)
}
```

```r
library(ggplot2)
ggplot(data.frame(model1, model2, model3, model4), aes(x = 1:length(model1))) +
  geom_line(aes(y= model1)) +
  geom_line(aes(y = model2), col = "red") +
  geom_line(aes(y = model3), col = "blue") +
  geom_line(aes(y = model4), col = "green")
```




```r
apply(data.frame(model1, model2, model3, model4),2,mean)
```

```
##   model1   model2   model3   model4
## 3922.875 1832.787 1783.665 4598.687
```

Model 1 has the best crossvalidation error. The other two SARIMA models also perform better than the linear regression model.