ECON403B Project 2

Mohammed Ibraaz Syed, Yating Zhang, Minxuan Wang

February 8, 2018

NOTE TO GRADER: The data is in months, so a 1.0 time lag indicates a 12 month lag, a 2.0 time lag indicates a 24 month lag, and a lag that plots to about 2.1 indicates a 25 month lag.

GROUP MEMBERS: Mohammed Ibraaz Syed, Minxuan Wang, Yating Zhang

I. Introduction

The data we are using is data on New Privately Owned Housing Units Started. The date range we are using is January 1st, 1987, to December 1st, 2017 - a period of more than 30 years. The data is in thousands of units and is monthly.

```
data = Quandl("FRED/HOUSTNSA", start_date = "1987-01-01")
```

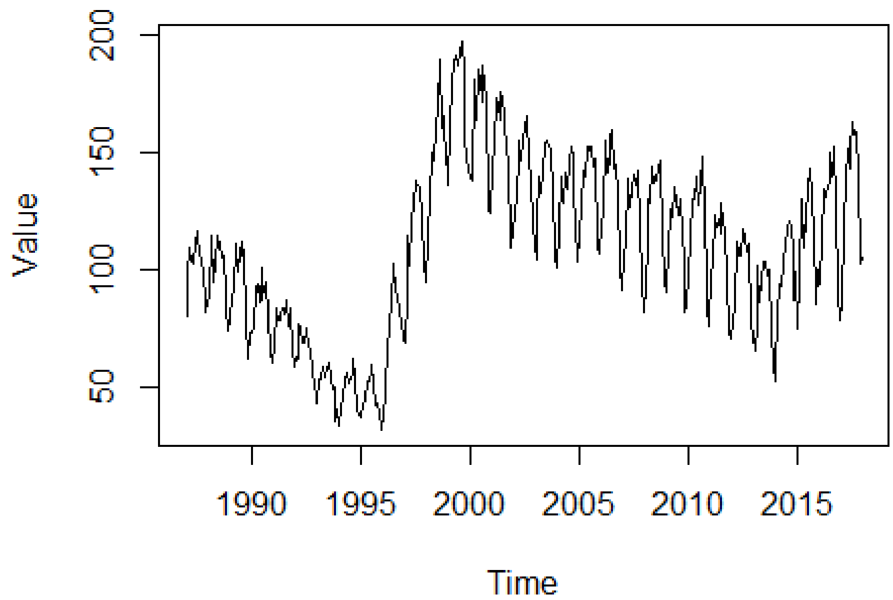
II. Results

1. Modeling and Forecasting Trend

Part (a)

Time series plot of data.

```
Value<-ts(data$Value, start = c(1987, 1), frequency = 12)
plot(Value)
```



Part (b)

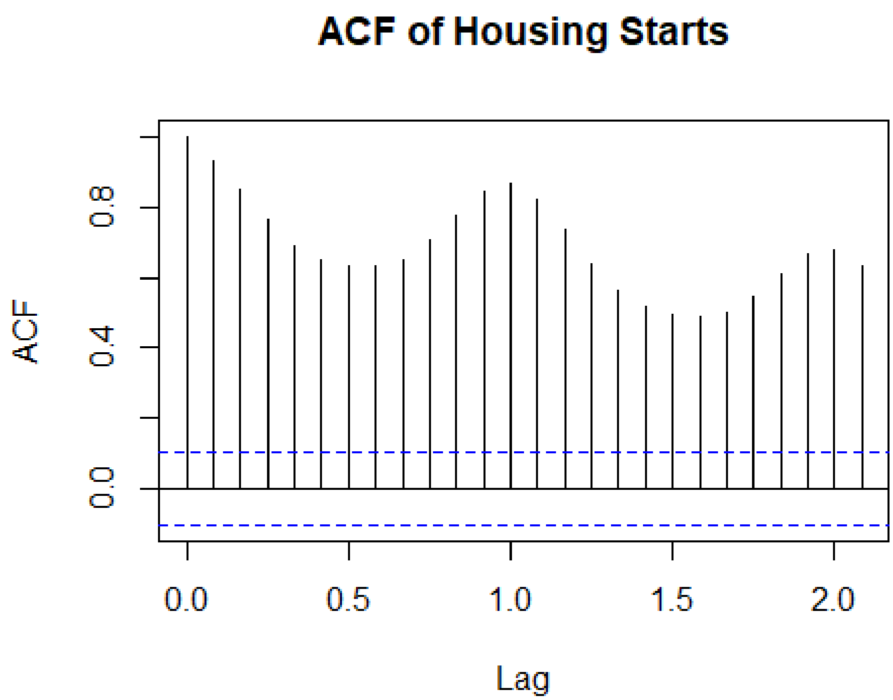
The plot in part (a) suggests that the data is not covariance stationary. Neither the mean nor the variance of the data point is constant over time.

In particular, the variance between 1996 and 1999 is much greater than the variance between 2004 and 2010. Also, the data does not appear to have a constant mean invariant under time. The mean is much higher in the period between 2000 and 2014 than in the period between 1987 and 1994. Therefore, the data generating process between 1987 and 1994 appears to be different from the data generating process between 1994 and 2000 as well as from the data generating process between 2000 and 2014 - all suggesting that the data is not covariance stationary.

Part (c)

ACF of Housing Starts

```
acf(Value,main="ACF of Housing Starts")
```



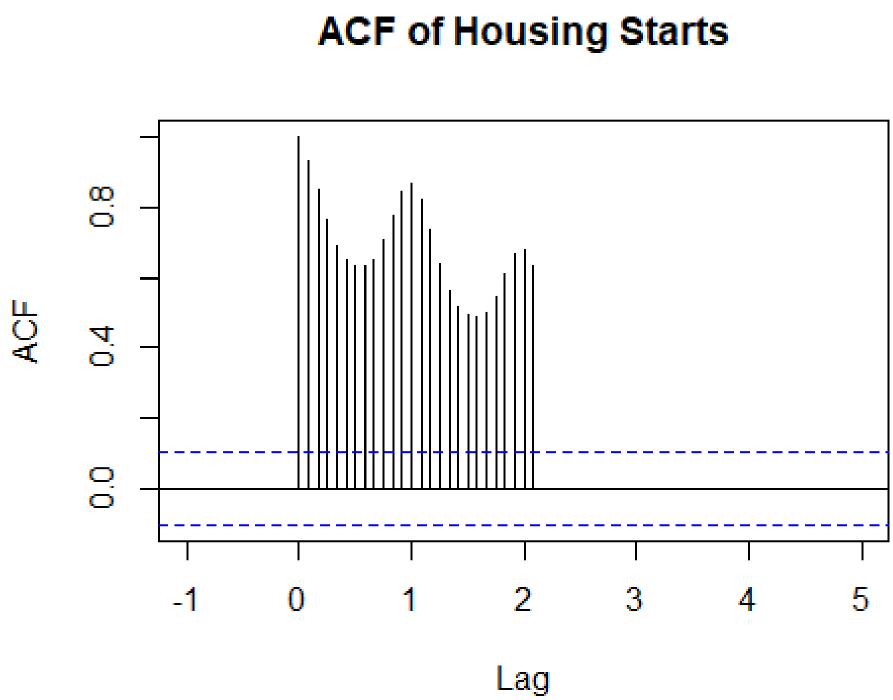

```
#acf(diff(Value),main="ACF of the Housing Starts")
```

The ACF suggests that our data can be estimated using an MA(2) model as the ACF is zero after around lag 2.

We can certainly say that an AR(1) model would not be appropriate for the data as the ACF does not exhibit an exponential decay, nor does it exhibit two alternating exponential decays.

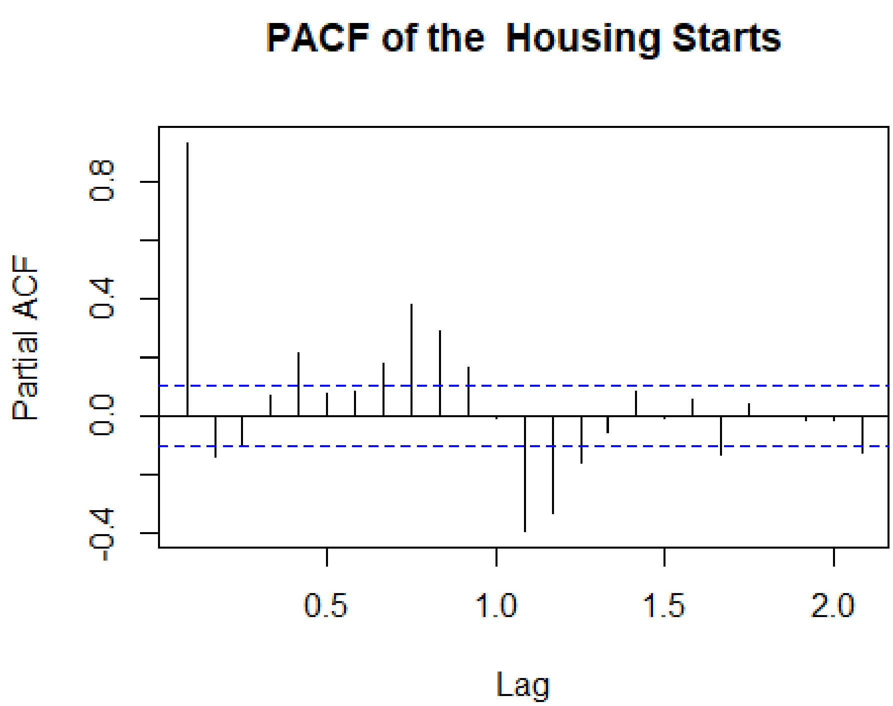
For an AR(2) model to be appropriate, we would need either a mixture of two exponential decays (for real-valued characteristic roots) or a damping sine and cosine waves (for complex-valued characteristic roots). However, the ACF plot does not exhibit such a decay pattern, but instead has a sharp cutoff around lag 2. This strongly indicates that an MA(2) model would best fit the data.

```
# Verification that ACF cuts off after around lag 2:
acf(Value,main="ACF of Housing Starts", xlim = c(-1, 5))
```



PACF of Housing Starts

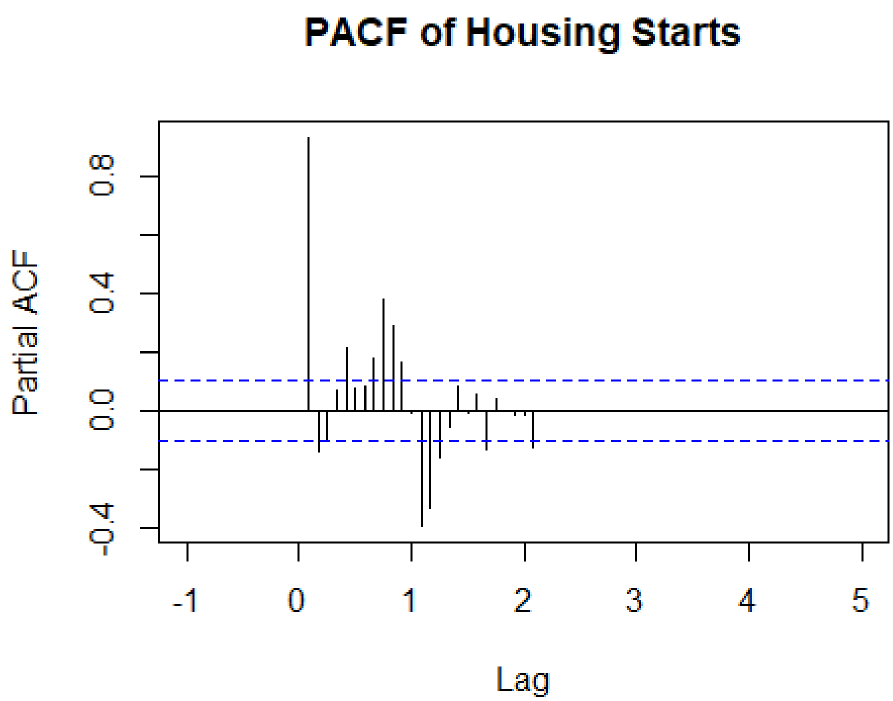
```
pacf(Value,main="PACF of the  Housing Starts")
```



```r
#pacf(diff(Value),main="PACF of the  Housing Starts")
```

The PACF is generally useful for determining the order of an AR model. The structure of the PACF indicates that an AR(1) model or even an AR(2) model could be potential fits as the PACF cuts off after around lag 2, however we noted that due to the structure of the ACF, an MA(2) model would likely be the best fit.

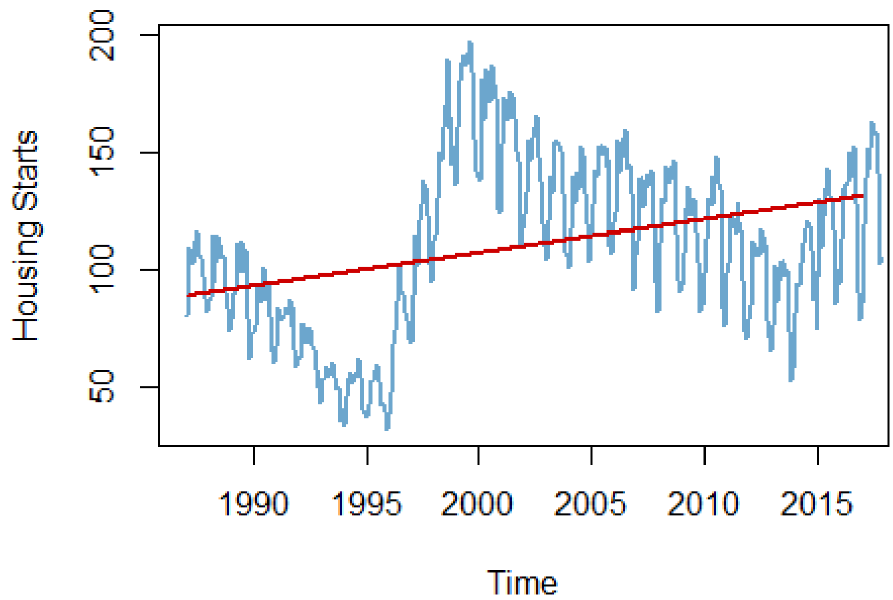
```r
# Verification that PACF cuts off after around lag 2:
pacf(Value,main="PACF of Housing Starts", xlim = c(-1, 5))
```



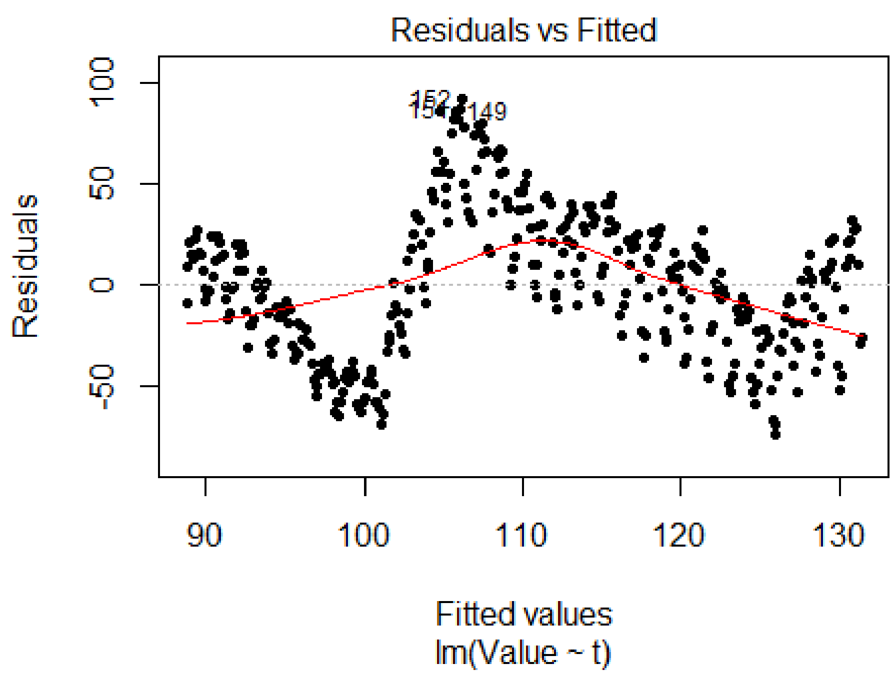
Parts (d) and (e)

Model 1: Linear

```r
# linear
t<-seq(1987, 2017,length=length(Value))
m1=lm(Value~t)
plot(Value,ylab="Housing Starts", xlab="Time", lwd=2, col='skyblue3', x
lim=c(1987,2017))
lines(t,m1$fit,col="red3",lwd=2)
```



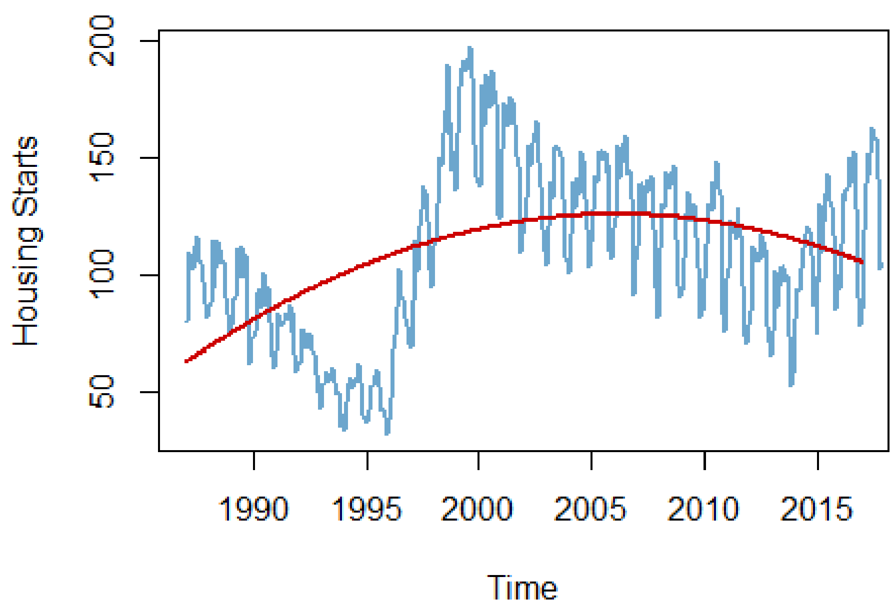

```
plot(m1,pch=20,which = c(1))
```



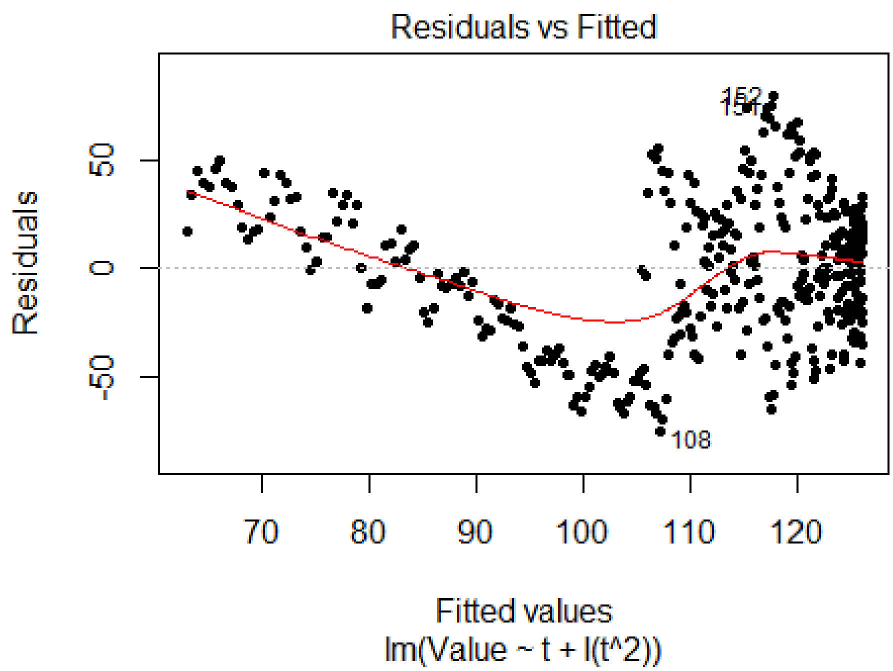
The residuals have an oscillating pattern, indicating that the linear model did not capture all the signal in the data.

Model 2: Quadratic

```
m2=lm(Value~t+I(t^2))
plot(Value,ylab="Housing Starts", xlab="Time", lwd=2, col='skyblue3', x
lim=c(1987,2017))
lines(t,m2$fit,col="red3",lwd=2)
```



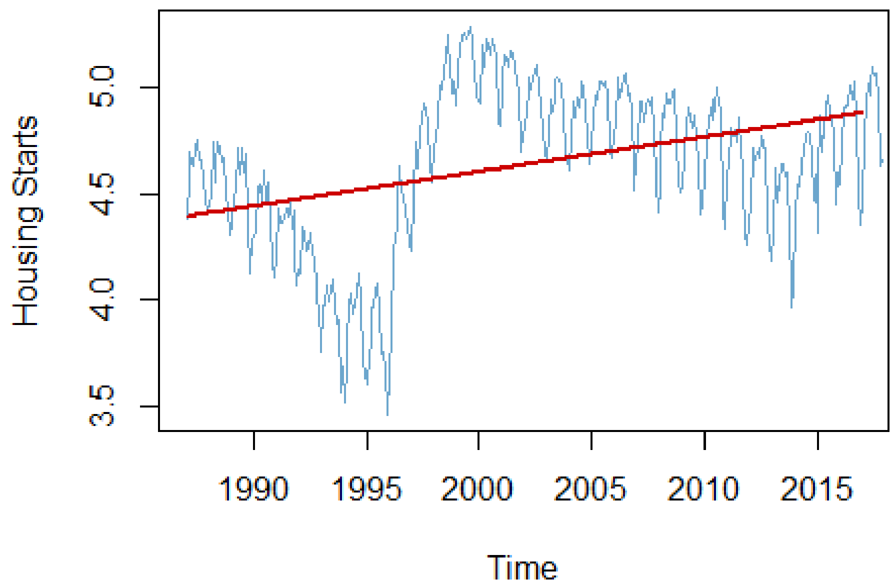

```
plot(m2,pch=20,which = c(1))
```



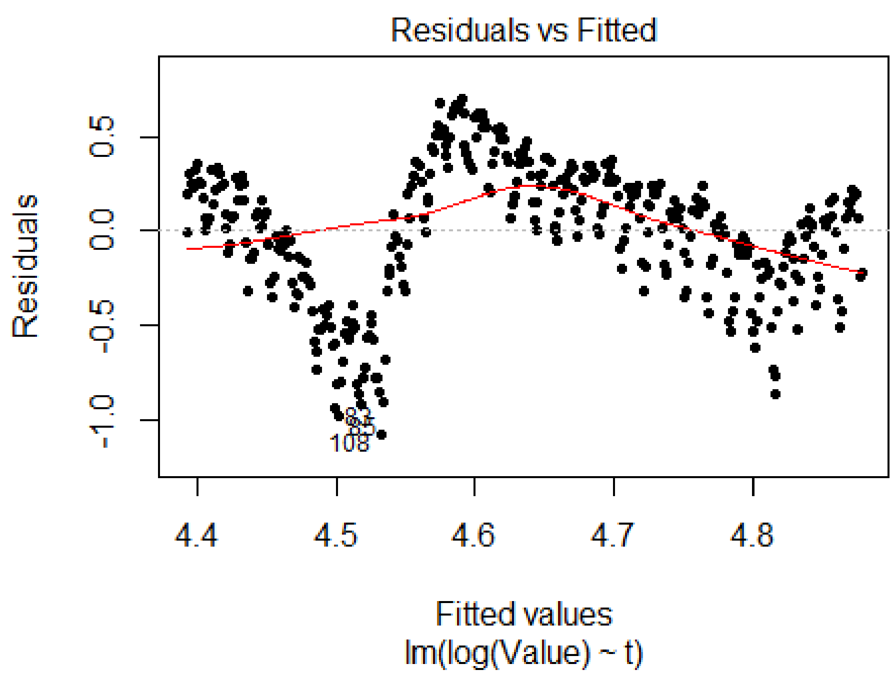
The residuals have a linear pattern between the fitted values of about 60 to 105, indicating that the quadratic model did not capture all the signal in the data.

Model 3: Log-Linear

```
m3=lm(log(Value)~t)
plot(log(Value),ylab="Housing Starts", xlab="Time", col='skyblue3', xli
m=c(1987,2017))
lines(t,m3$fit,col="red3",type='l',lwd=2, ylim = c(-2,7))
```



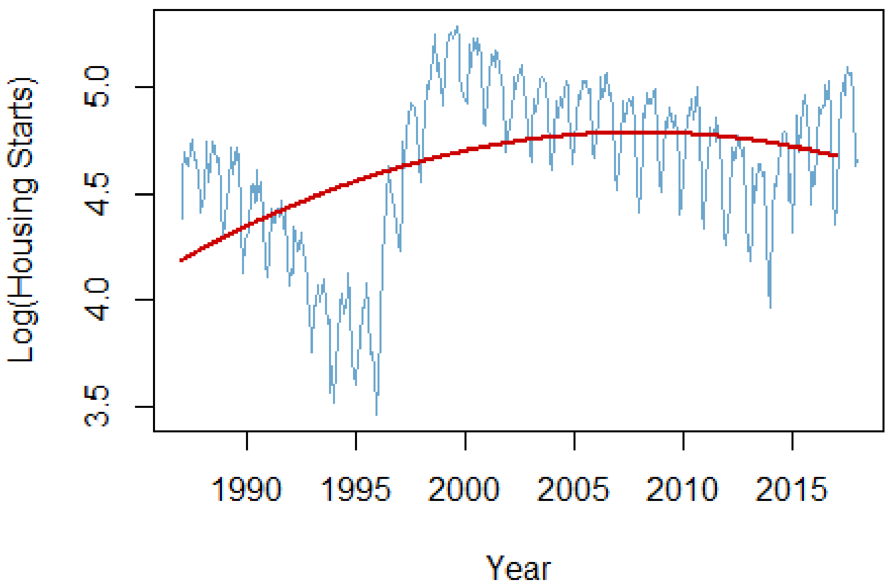

```
plot(m3,pch=20,which = c(1))
```



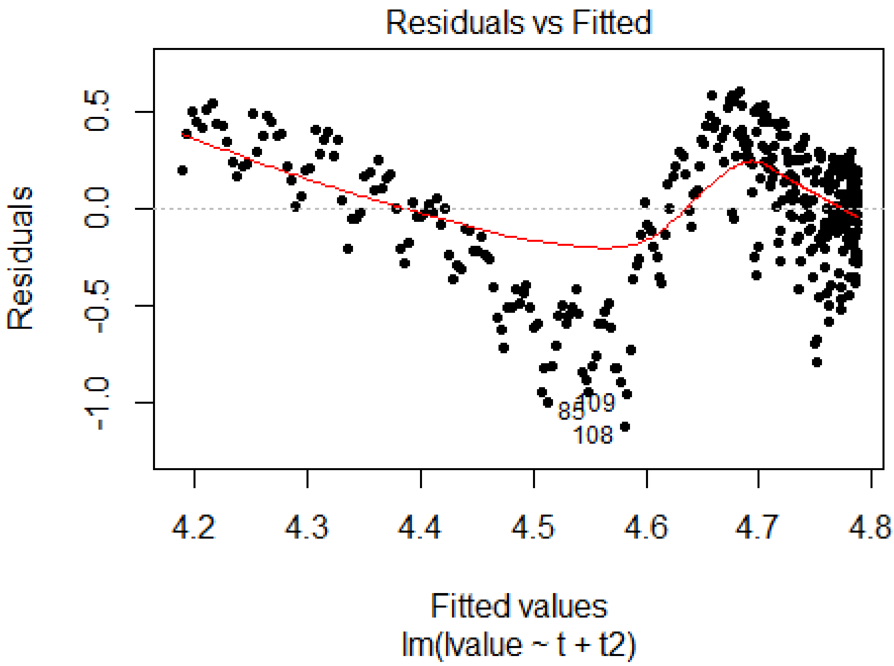
The pattern in the residuals indicates that the log-linear nodel did not capture all the signal in the data.

Model 4: Log-quadratic model

```
lvalue<-log(Value)
t2<-t^2
plot(lvalue,xlab="Year", ylab="Log(Housing Starts)", col='skyblue3')
m4=lm(lvalue~t+t2)
lines(t,m4$fit,col="red3",lwd=2)
```



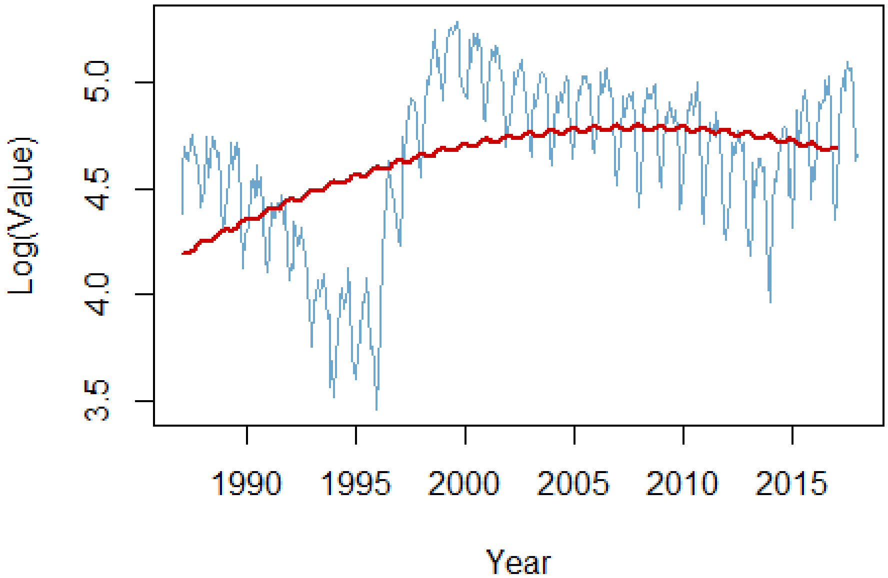

```
plot(m4,pch=20,which = c(1))
```



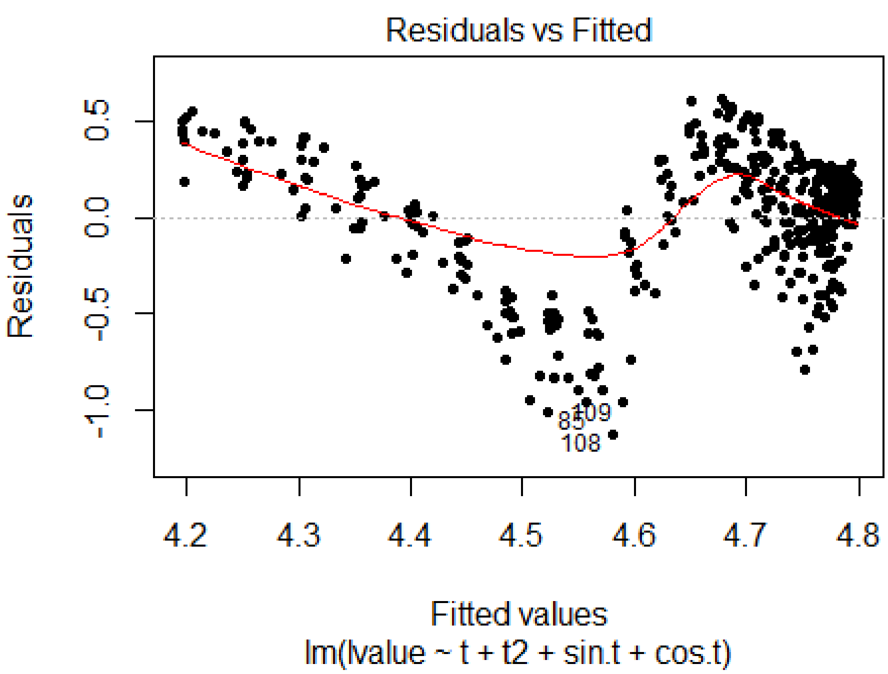
The log-quadratic model has a linear pattern in its residuals between fitted values of 4.2 and 4.5 in the log space. This pattern indicates that the model does not capture all the signal in the data.

Model 5: Log-quadratic-periodic

```
sin.t<-sin(2*pi*t)
cos.t<-cos(2*pi*t)
plot(lvalue,xlab="Year", ylab="Log(Value)", col='skyblue3')
m5=lm(lvalue~t+t2+sin.t+cos.t)
lines(t, m5$fit,col="red3",lwd=2)
```




```r
plot(m5,pch=20,which = c(1))
```

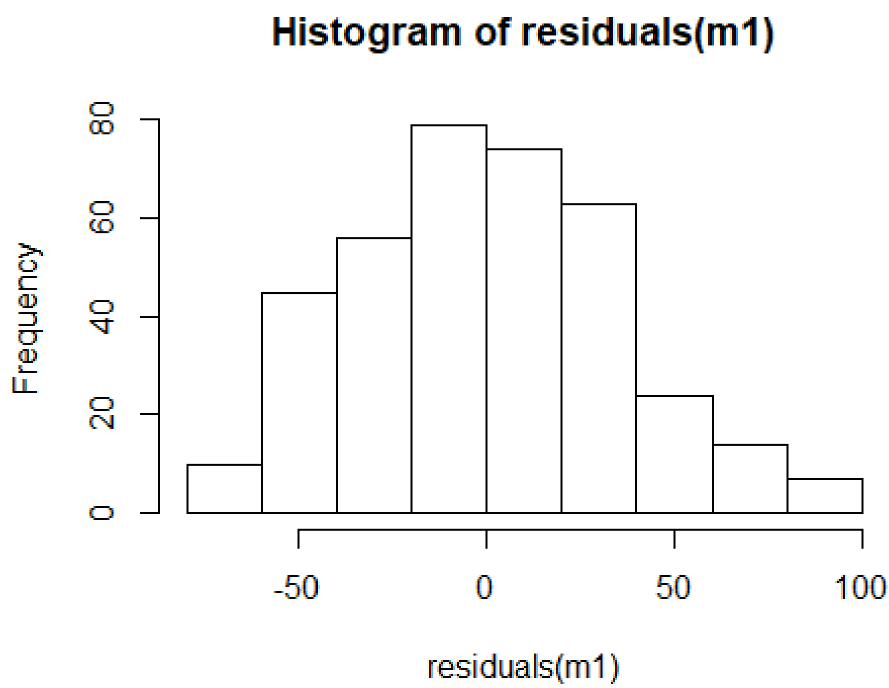


The log-quadratic periodic model has a pattern in the residuals similar to the log-quadratic model, which indicates that the model also did not capture all the signal in the data.

Part (f)

Histogram of Residuals - Model 1: Linear

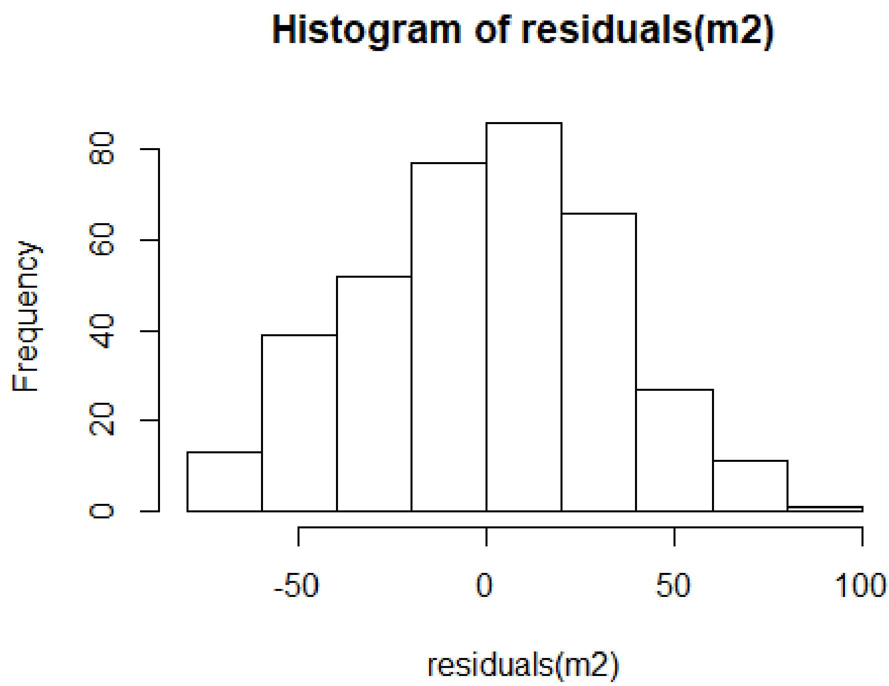
```
hist(residuals(m1))
```



The residuals cluster around zero, but there are some high-valued residuals.

Histogram of Residuals - Model 2: Quadratic

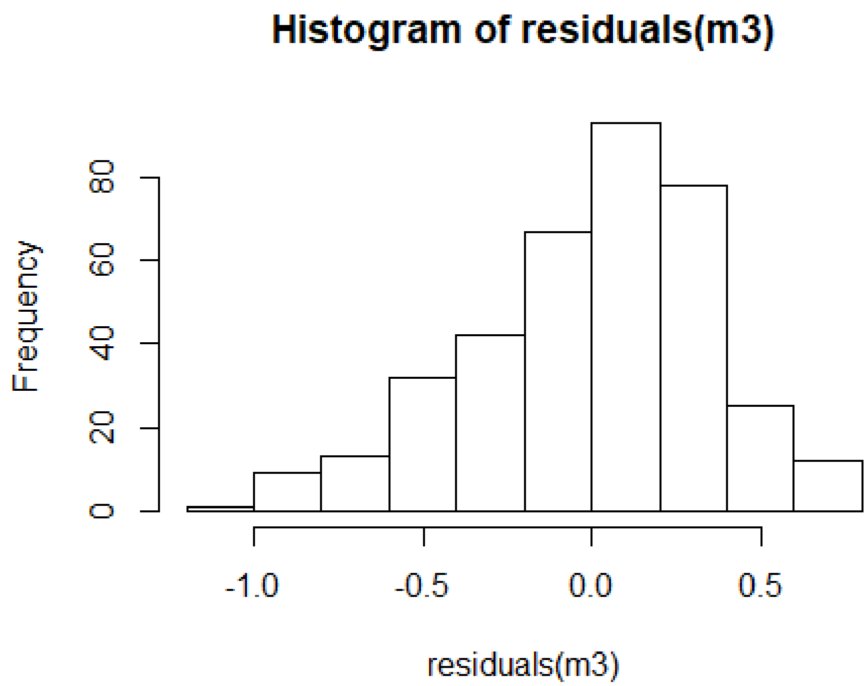
```
hist(residuals(m2))
```



The residuals generally appear to fit close to a normal distribution. There are fewer high-valued residuals than in the linear case.

Histogram of Residuals - Model 3: Log-Linear

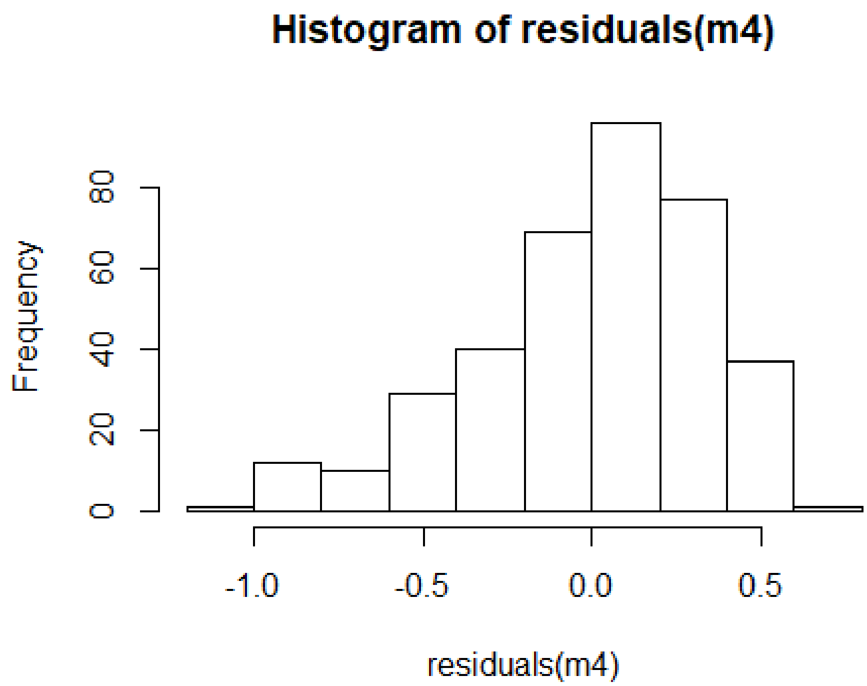
```
hist(residuals(m3))
```



The residuals are clustered around zero but have a negative skew.

Histogram of Residuals - Model 4: Log-quadratic model

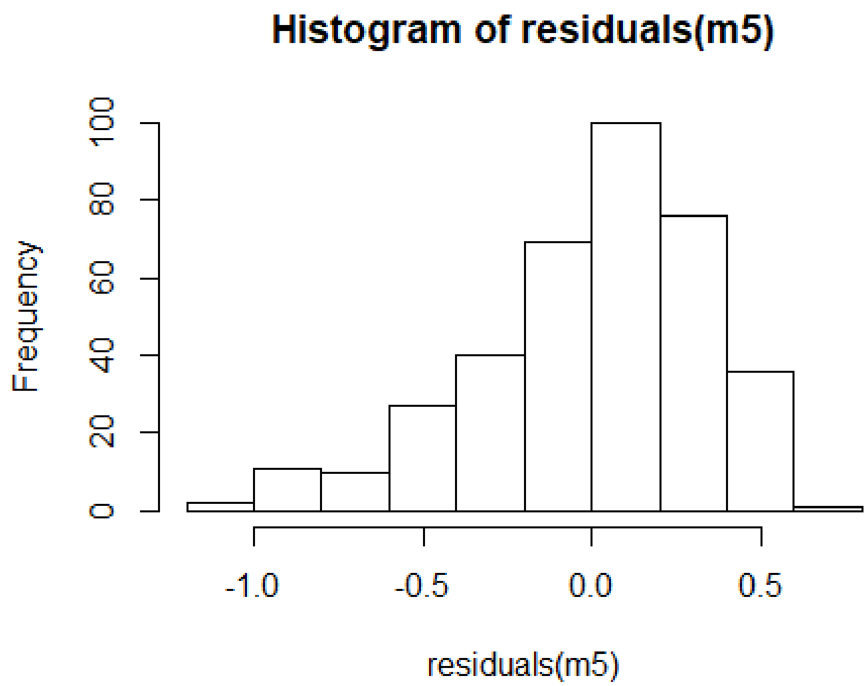
```
hist(residuals(m4))
```



The residuals appear to have a negative skew.

Histogram of Residuals - Model 5: Log-quadratic-periodic

```
hist(residuals(m5))
```



Like in the previous histogram, the residuals again appear to have a negative skew.

Part (g)

```
library(tseries)
# Model 1: Linear
jarque.bera.test(residuals(m1))

##
##  Jarque Bera Test
##
## data:  residuals(m1)
## X-squared = 5.7374, df = 2, p-value = 0.05677

# Model 2: Quadratic
jarque.bera.test(residuals(m2))

##
##  Jarque Bera Test
##
## data:  residuals(m2)
## X-squared = 5.1698, df = 2, p-value = 0.0754

# Model 3: Log-Linear
jarque.bera.test(residuals(m3))

##
##  Jarque Bera Test
##
```

```
## data:  residuals(m3)
## X-squared = 17.818, df = 2, p-value = 0.0001351
```

```
# Model 4: Log-quadratic model
jarque.bera.test(residuals(m4))
```

```
##
##  Jarque Bera Test
##
## data:  residuals(m4)
## X-squared = 36.312, df = 2, p-value = 1.303e-08
```

```
# Model 5: Log-quadratic-periodic
jarque.bera.test(residuals(m5))
```

```
##
##  Jarque Bera Test
##
## data:  residuals(m5)
## X-squared = 37.248, df = 2, p-value = 8.161e-09
```

We know that the Jarque Bera test is used to determine normality.

We reject normality for the Log-linear, Log-quadratic, and Log-quadratic-periodic models at the 5% significance level.

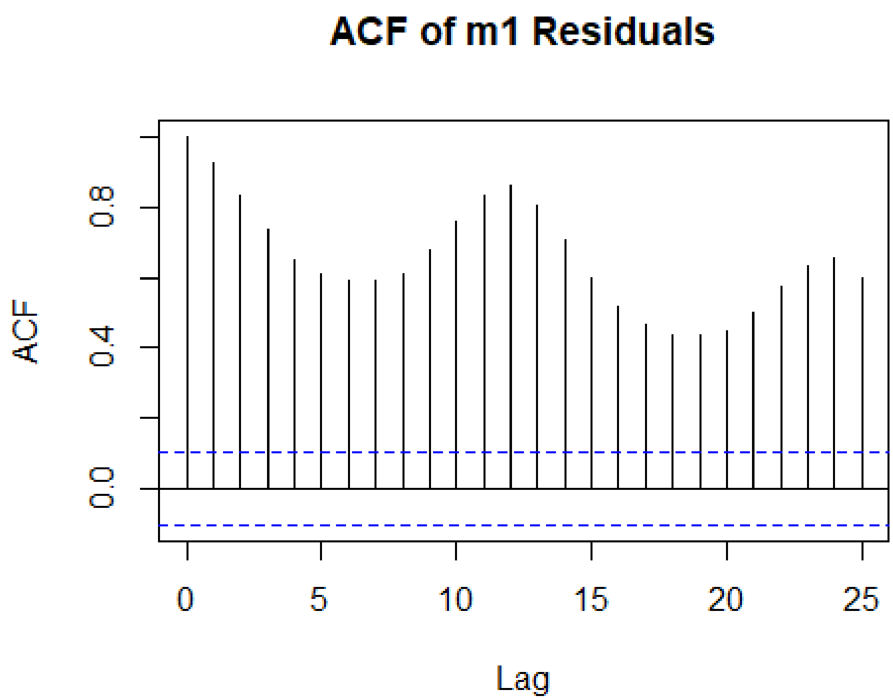
For the Linear and Quadratic models, we note that the p-values are very close to 0.05 for both models, and so normality for these models would be rejected at the 10% significance level.

In all, due to none of the models having residuals that appear to follow a normal distribution at the 10% significance level, we can claim that all the models are misspecified as there is signal in the data that all do not capture.

Part (h)

Model 1: Linear

```
acf(residuals(m1),main="ACF of m1 Residuals")
```

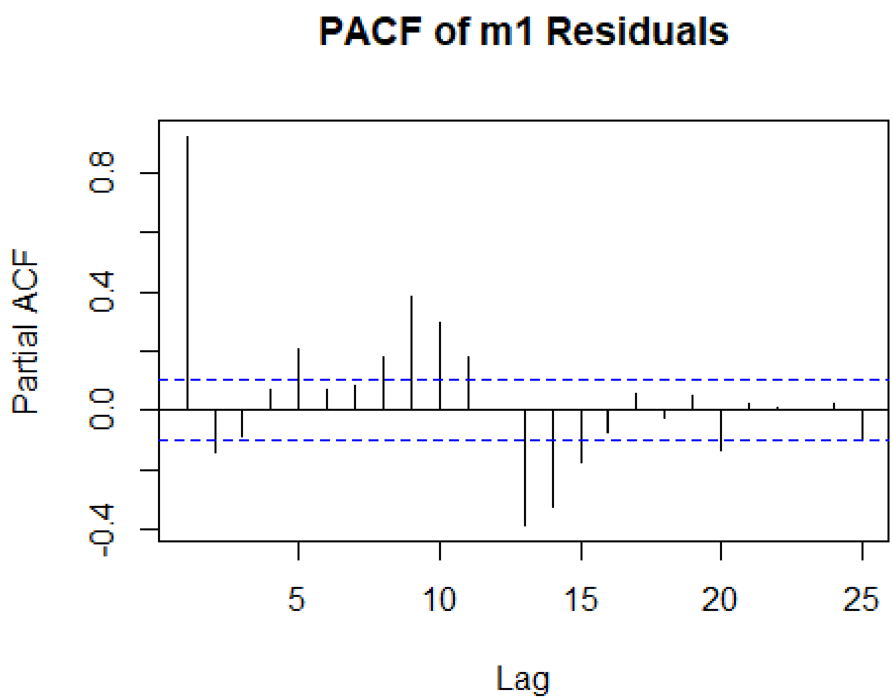



```r
#acf(diff(residuals(m1)),main="ACF of m1 Residuals")
```

The ACF suggests that the data can be estimated using an MA(25) model as the ACF is zero after lag 25. Since all the lags up to lag 25 are significant, the residuals have a structure that can be modeled, indicating that our model can be improved.

The ACF is not consistent with the signature which would lead us to use an AR model.

```r
pacf(residuals(m1),main="PACF of m1 Residuals")
```

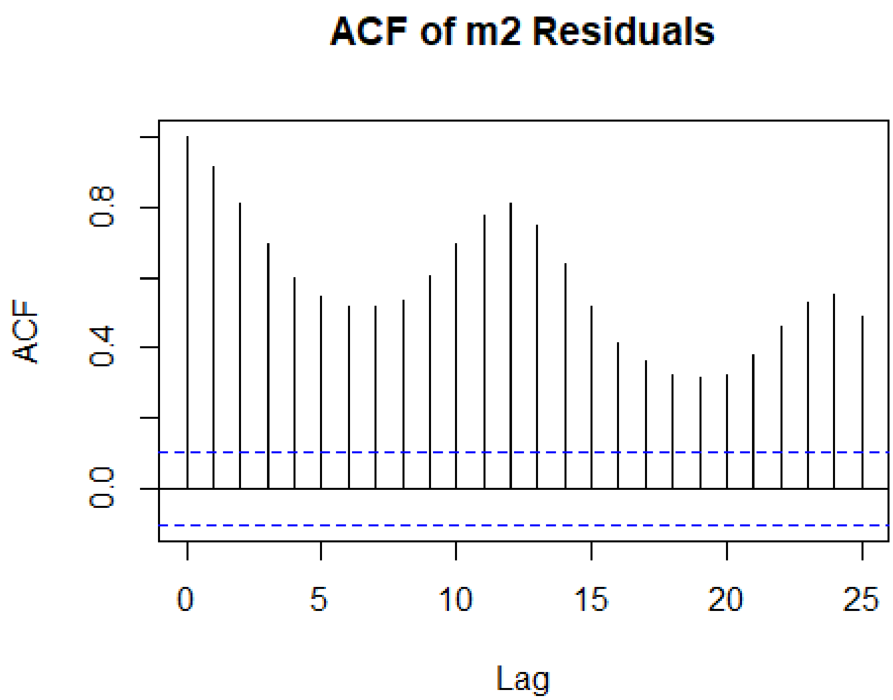



```
#pacf(diff(residuals(m1)),main="PACF of m1 Residuals")
```

The PACF has significant time lags up to 15 lags and one significant lag at 20.

Model 2: Quadratic

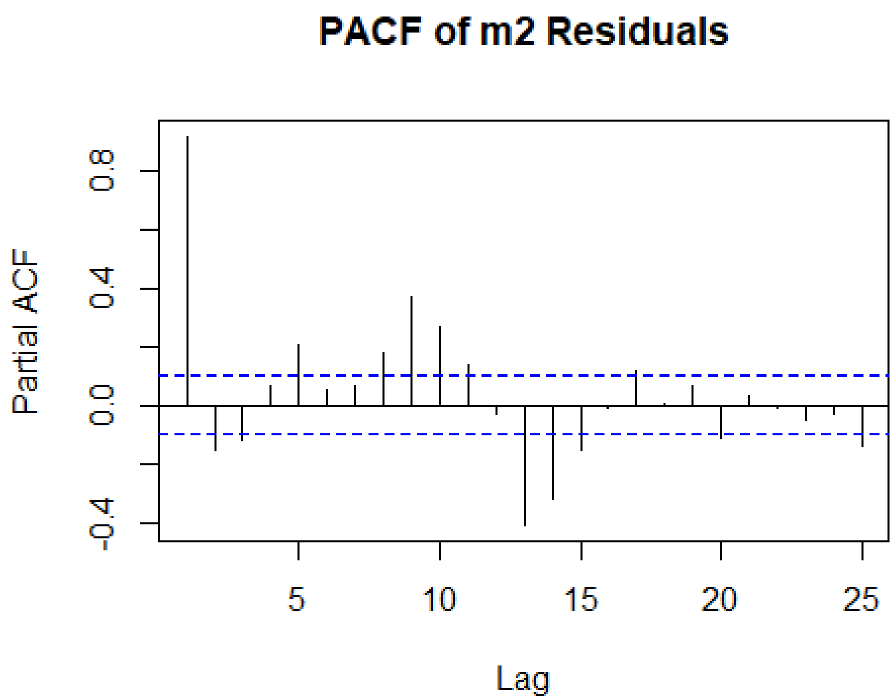
```
acf(residuals(m2),main="ACF of m2 Residuals")
```




```
#acf(diff(residuals(m2)),main="ACF of m2 Residuals")
```

The ACF indicates that an MA(25) model would be appropriate for the residuals of this model. The residuals have a structure that can be modeled, indicating that our model can be improved.

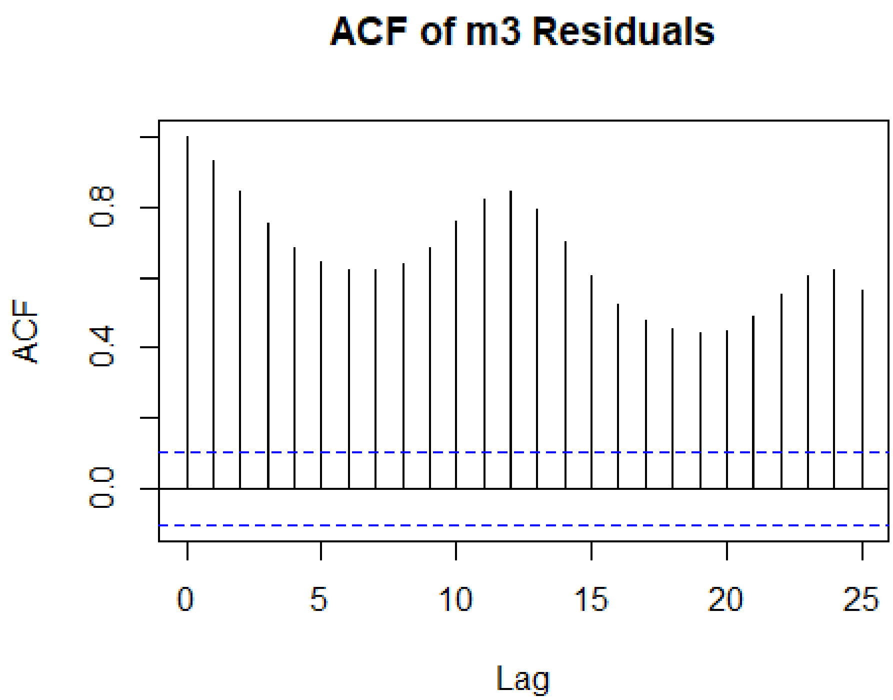
```
pacf(residuals(m2),main="PACF of m2 Residuals")
```




```
#pacf(diff(residuals(m2)),main="PACF of m2 Residuals")
```

The PACF has the highest significant lags at 14, 20, and 25.

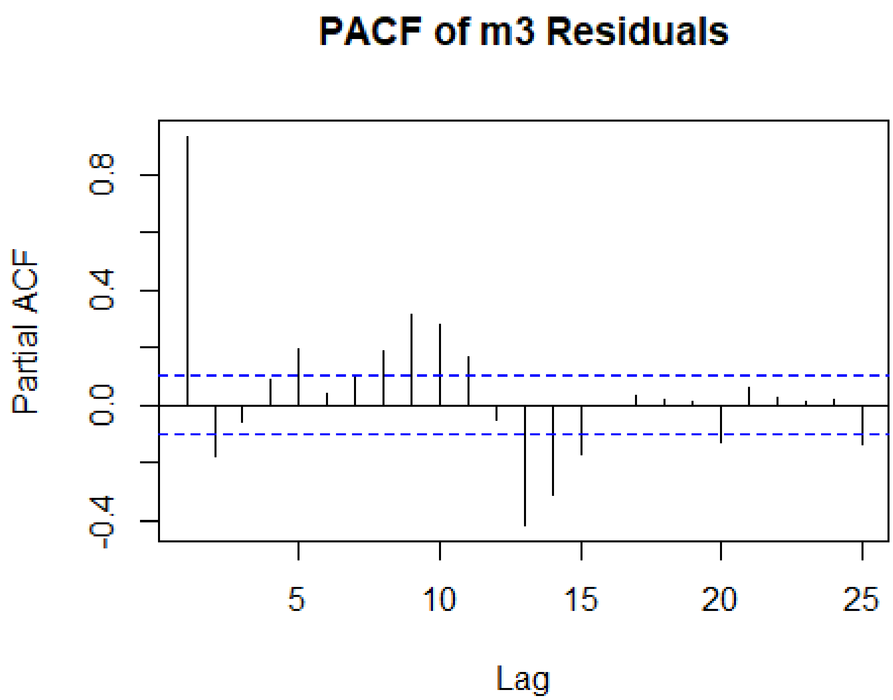
Model 3: Log-Linear
```
acf(residuals(m3),main="ACF of m3 Residuals")
```




```
#acf(diff(residuals(m3)),main="ACF of m3 Residuals")
```

The ACF indicates that an MA(25) model would be appropriate for the residuals of this model. The residuals have a structure that can be modeled, indicating that our model can be improved.

```
pacf(residuals(m3),main="PACF of m3 Residuals")
```

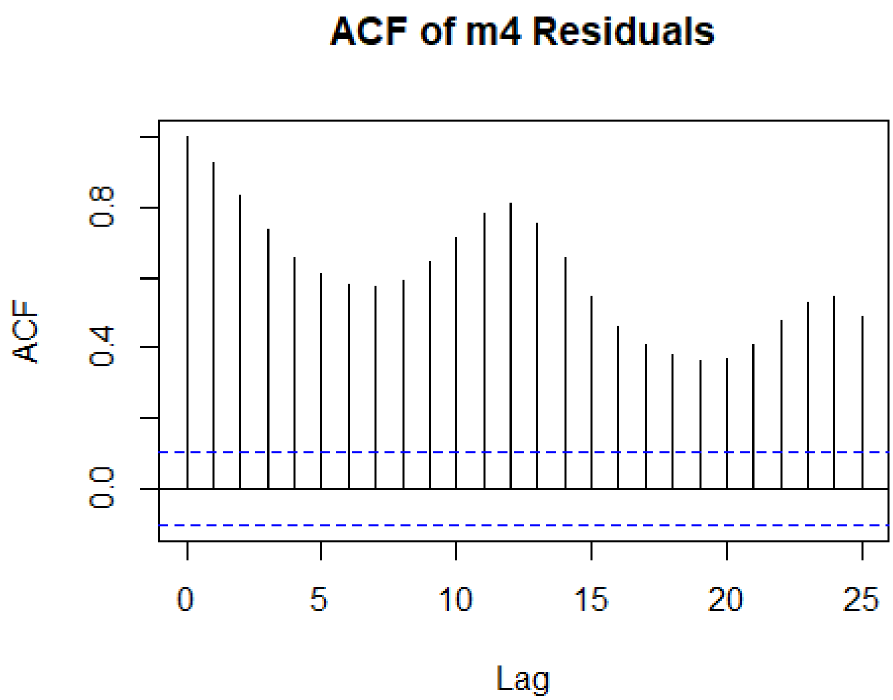



```
#pacf(diff(residuals(m3)),main="PACF of m3 Residuals")
```

The PACF has the highest significant lags at 15, 20, and 25.

Model 4: Log-quadratic model

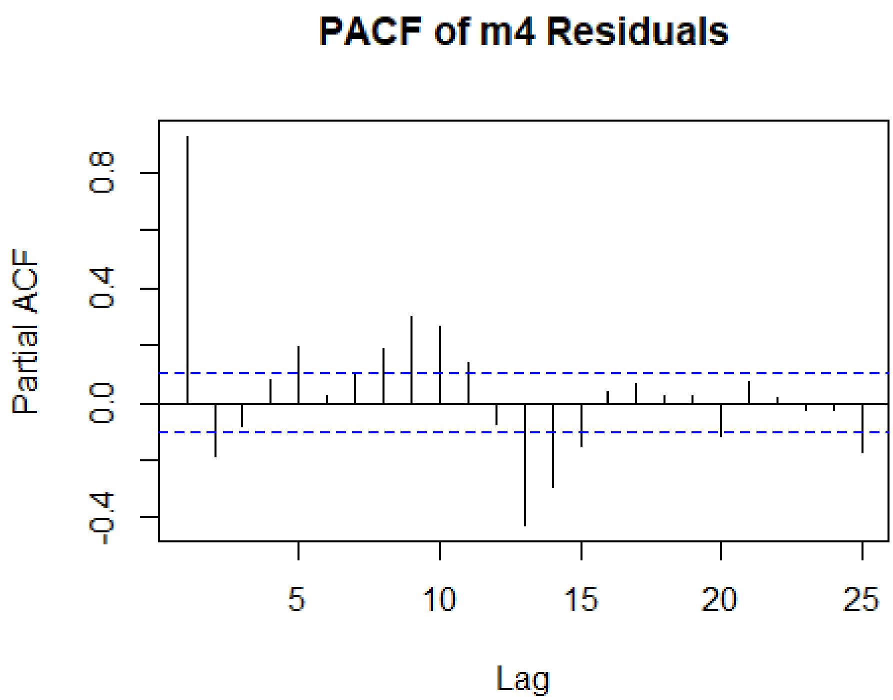
```
acf(residuals(m4),main="ACF of m4 Residuals")
```




```
#acf(diff(residuals(m5)),main="ACF of m5 Residuals")
```

The ACF indicates that an MA(25) model would be appropriate for the residuals of this model. The residuals have a structure that can be modeled, indicating that our model can be improved.

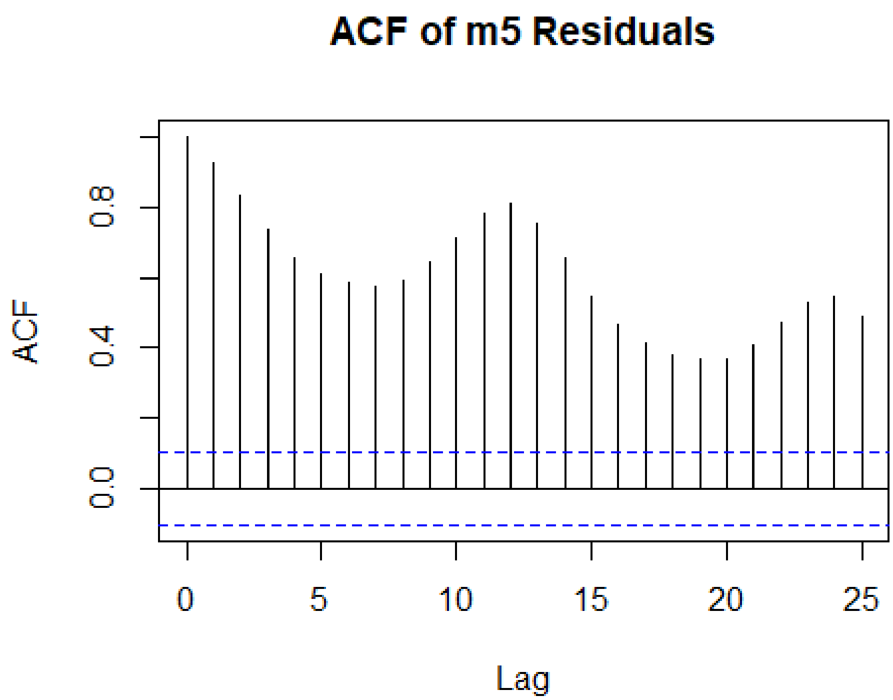
```
pacf(residuals(m4),main="PACF of m4 Residuals")
```




```
#pacf(diff(residuals(m5)),main="PACF of m5 Residuals")
```

The PACF has the highest significant lags at 15, 20, and 25.

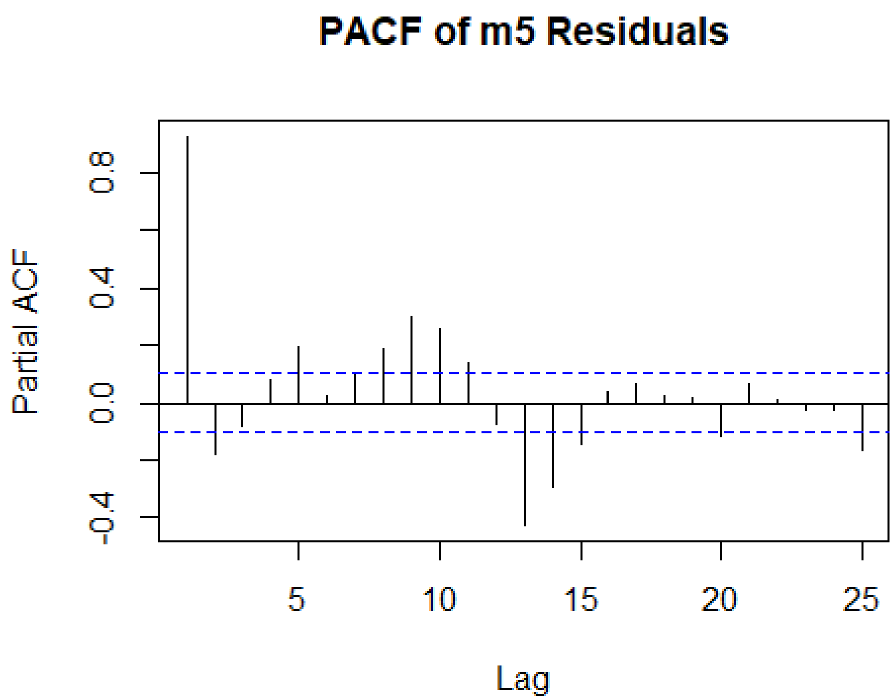
```
# Model 5: Log-quadratic-periodic
acf(residuals(m5),main="ACF of m5 Residuals")
```




```
#acf(diff(residuals(m6)),main="ACF of m6 Residuals")
```

The ACF indicates that an MA(25) model would be appropriate for the residuals of this model. The residuals have a structure that can be modeled, indicating that our model can be improved.

```
pacf(residuals(m5),main="PACF of m5 Residuals")
```




```
#pacf(diff(residuals(m6)),main="PACF of m6 Residuals")
```

Part (i)

Model 1: Linear

```
summary(m1)

##
## Call:
## lm(formula = Value ~ t)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -73.386 -26.406  -0.744  23.375  91.771
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2733.8031   416.9401  -6.557 1.85e-10 ***
## t               1.4205     0.2083   6.821 3.70e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 34.88 on 370 degrees of freedom
## Multiple R-squared:  0.1117, Adjusted R-squared:  0.1093
## F-statistic: 46.53 on 1 and 370 DF,  p-value: 3.701e-11
```

The adjusted R-squared value is 0.1093 indicating that the model is not particularly good at predicting New Privately Owned Housing Units Started.

Both intercept and time variable are significant, with very low p-values.

Model 2: Quadratic

```
summary(m2)

##
## Call:
## lm(formula = Value ~ t + I(t^2))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -75.248 -23.377   1.181  23.744  80.076
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -6.959e+05  1.014e+05  -6.864 2.85e-11 ***
## t            6.939e+02  1.013e+02   6.851 3.08e-11 ***
## I(t^2)      -1.729e-01  2.530e-02  -6.837 3.36e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 32.91 on 369 degrees of freedom
## Multiple R-squared:  0.2116, Adjusted R-squared:  0.2073
## F-statistic: 49.51 on 2 and 369 DF,  p-value: < 2.2e-16
```

The adjusted R-squared value is 0.2073 indicating that the model is not particularly good at predicting New Privately Owned Housing Units Started, though it can explain a lot more variation in the dependent variable than the linear model.

The intercept and both time variables are significant, with very low p-values.

Model 3: Log-Linear

```
summary(m3)

##
## Call:
## lm(formula = log(Value) ~ t)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.06986 -0.22316  0.05579  0.25292  0.69753
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept) -27.872466   4.249906  -6.558 1.83e-10 ***
## t             0.016238   0.002123   7.649 1.77e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3555 on 370 degrees of freedom
## Multiple R-squared:  0.1365, Adjusted R-squared:  0.1342
## F-statistic: 58.51 on 1 and 370 DF,  p-value: 1.767e-13
```

The adjusted R-squared value is 0.1342 indicating that the model is not particularly good at predicting New Privately Owned Housing Units Started, and is worse than the quadratic model.

The intercept and time variable are both significant, with very low p-values.

Model 4: Log-quadratic model

```
summary(m4)

##
## Call:
## lm(formula = lvalue ~ t + t2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.11766 -0.19744  0.05231  0.24300  0.60546
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.485e+03  1.060e+03  -5.176 3.72e-07 ***
## t            5.468e+00  1.058e+00   5.165 3.93e-07 ***
## t2          -1.362e-03  2.644e-04  -5.150 4.25e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3439 on 369 degrees of freedom
## Multiple R-squared:  0.1944, Adjusted R-squared:  0.1901
## F-statistic: 44.54 on 2 and 369 DF,  p-value: < 2.2e-16
```

The adjusted R-squared value is 0.1901 indicating that the model is not particularly good at predicting New Privately Owned Housing Units Started.

The intercept and both time variables are significant, with very low p-values.

Model 5: Log-quadratic-periodic

```
summary(m5)
```

```
##
## Call:
## lm(formula = lvalue ~ t + t2 + sin.t + cos.t)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.11799 -0.18840  0.05228  0.24197  0.60999
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.488e+03  1.062e+03  -5.168 3.90e-07 ***
## t            5.472e+00  1.061e+00   5.157 4.12e-07 ***
## t2          -1.362e-03  2.650e-04  -5.141 4.45e-07 ***
## sin.t       -7.302e-03  2.532e-02  -0.288    0.773
## cos.t        9.963e-03  2.524e-02   0.395    0.693
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3447 on 367 degrees of freedom
## Multiple R-squared:  0.195,  Adjusted R-squared:  0.1862
## F-statistic: 22.22 on 4 and 367 DF,  p-value: < 2.2e-16
```

The adjusted R-squared value is 0.1862, indicating that the model is not particularly good at predicting New Privately Owned Housing Units Started.

The intercept and time variables are significant, with very low p-values, but the periodic variables are not significant.

Part (j)

AIC

```
AIC(m1)

## [1] 3702.306

AIC(m2)

## [1] 3659.936

AIC(m3)

## [1] 290.2882

AIC(m4)

## [1] 266.4658

AIC(m5)

## [1] 270.2237
```

Based on the AIC, the Log-quadratic model seems to be the best, followed closely by the Log-quadratic-periodic model.

BIC

```
BIC(m1)

## [1] 3714.063

BIC(m2)

## [1] 3675.612

BIC(m3)

## [1] 302.0448

BIC(m4)

## [1] 282.1414

BIC(m5)

## [1] 293.737
```

Based on the BIC, the Log-quadratic model seems to be the best, followed closely by the Log-quadratic-periodic model.

Therefore, we We choose the Log-quadratic model. Both AIC and BIC agree that it is the best model.

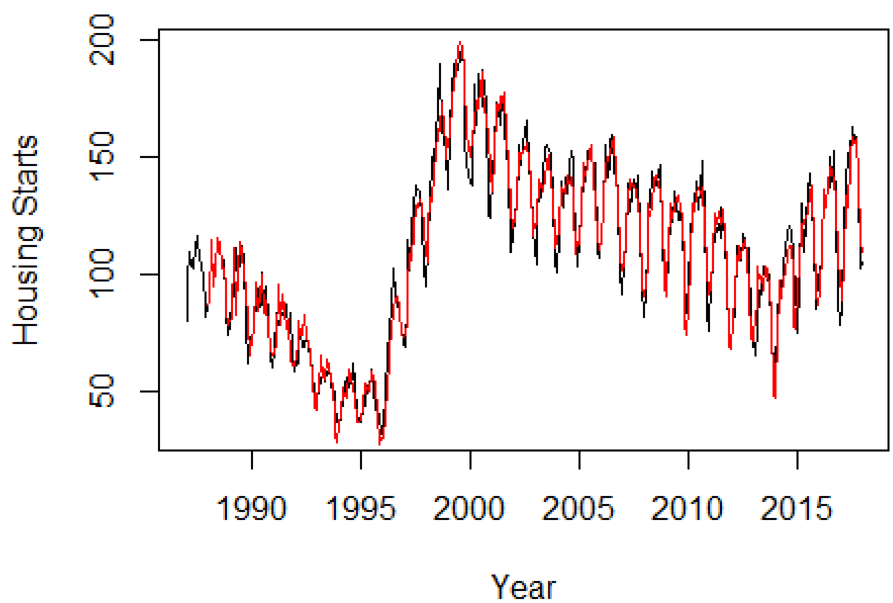
Part (k)

Our h-steps are in months.

```
HoltWinters(Value)

## Holt-Winters exponential smoothing with trend and additive seasonal
component.
##
## Call:
## HoltWinters(x = Value)
##
## Smoothing parameters:
##  alpha: 0.4077281
##  beta : 0.05293857
##  gamma: 0.3265617
##
## Coefficients:
##            [,1]
## a   137.5735786
## b     0.7399720
## s1  -29.0343990
## s2   -8.5696990
```

```
## s3   12.8627849
## s4    8.7111674
## s5   12.0233129
## s6   16.3119075
## s7   23.2953316
## s8   17.5276703
## s9   17.3514074
## s10  -0.2040141
## s11 -31.3478530
## s12 -30.7286658
```

```
plot(Value,xlab="Year", ylab="Housing Starts")
lines(HoltWinters(Value)$fitted[,1],col="red")
```

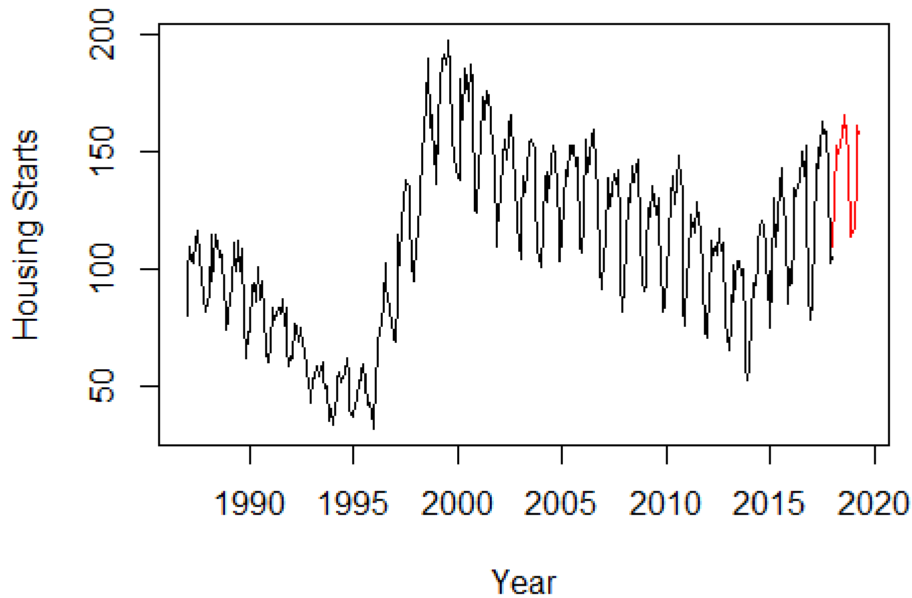


Try Holt-Winters Prediction

```
value.hw<-HoltWinters(Value)
predict(value.hw,n.ahead=16)
```

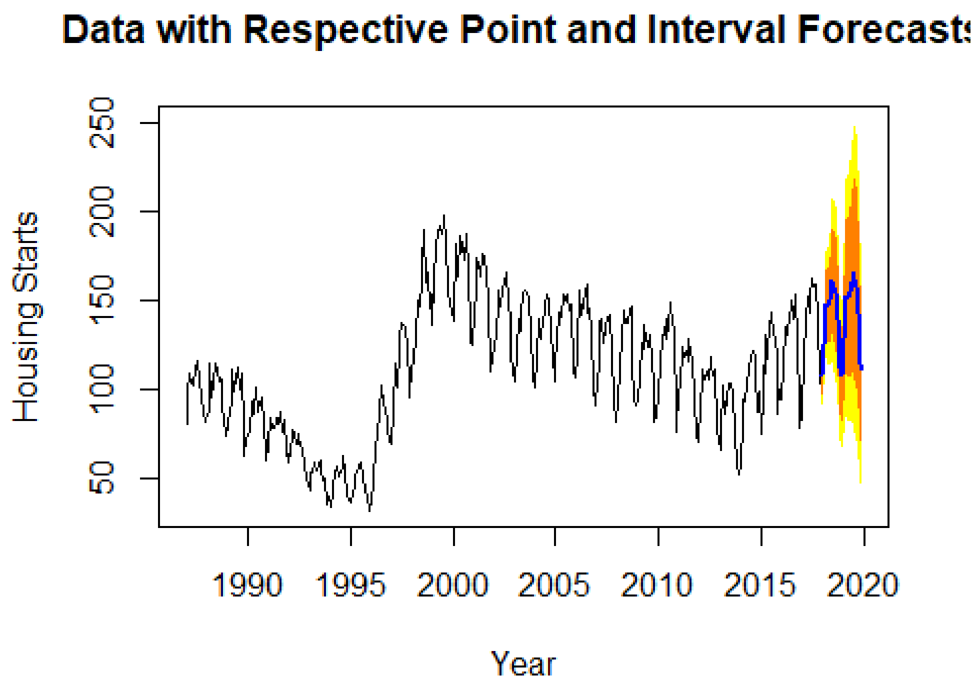
```
##           Jan      Feb      Mar      Apr      May      Jun      Jul
## 2018 109.2792 130.4838 152.6563 149.2446 153.2968 158.3253 166.0487
## 2019 118.1588 139.3635 161.5359 158.1243
##           Aug      Sep      Oct      Nov      Dec
## 2018 161.0210 161.5847 144.7693 114.3654 115.7246
## 2019
```

```r
plot(Value,xlim=c(1987,2019.4),xlab="Year", ylab="Housing Starts")
lines(predict(value.hw,n.ahead=16),col=2)
```

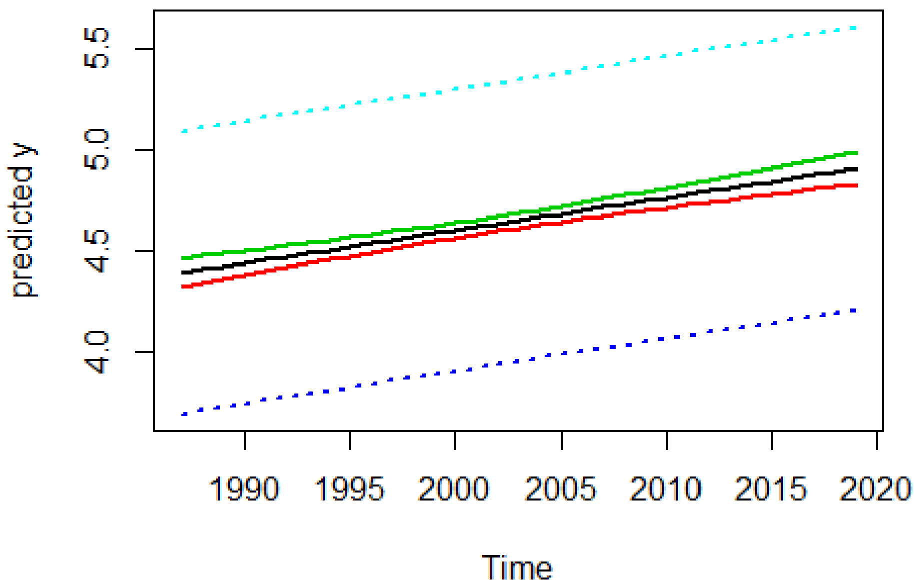


Forecast using preferred model with uncertainty prediction interval

```r
#plot(Value,main="Data",xlab="Year", ylab="Housing Starts")
plot(forecast(Value),main="Data with Respective Point and Interval Fore
casts",xlab="Year", ylab="Housing Starts",shadecols="oldstyle")
```




```
tn=data.frame(t=seq(1987,2019.4))
pred=predict(lm(lvalue~t+t^2), tn, se.fit = TRUE)
pred.plim = predict(lm(lvalue~t+t^2), tn, level=0.95, interval="predict
ion")
pred.clim = predict(lm(lvalue~t+t^2), tn, level=0.95, interval="confide
nce")
matplot(tn$t,cbind(pred.clim, pred.plim[,-1]),
        lty=c(1,1,1,3,3), type="l", lwd=2, ylab="predicted y",xlab="Tim
e")
```



2. Modeling and Forecasting Seasonality

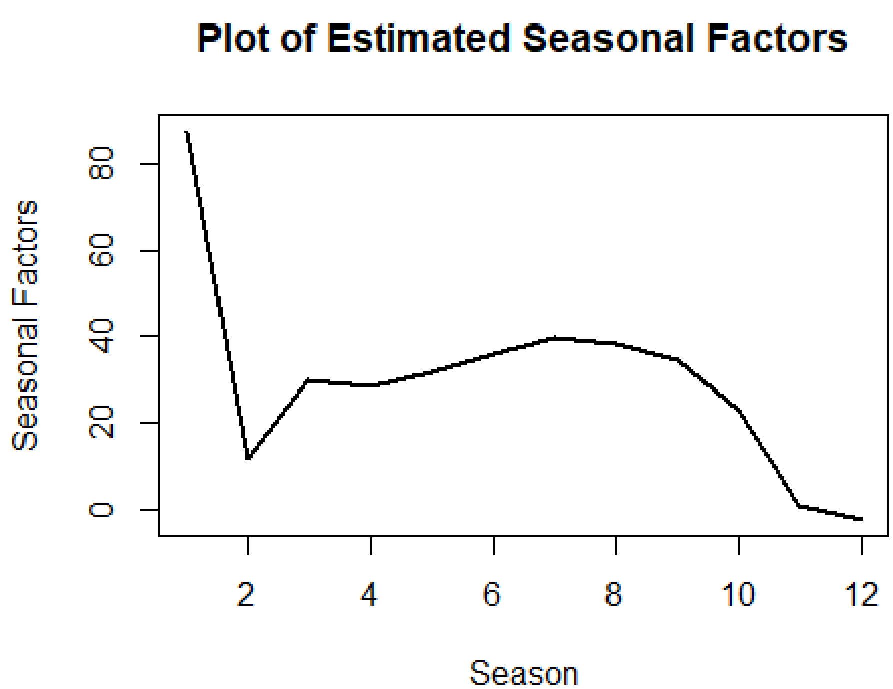
Part (a)

```
# Creat dummy variables (monthly)
library(dummies)
library(zoo)

df<-data.frame(date=seq(as.Date("1987/1/1"), as.Date("2017/12/1"), "mon
ths"))
df$Month<-format(as.yearmon(df$date), "M%m")
df<-dummy.data.frame(df, sep="_")
colnames(df)<-gsub("Month_", "", colnames(df))
# Now we have a new dataframe "df" contains monthly dummies

dummyreg<-lm(Value~M02+M03+M04+M05+M06+M07+M08+M09+M10+M11+M12,data=df)
summary(dummyreg)

##
## Call:
## lm(formula = Value ~ M02 + M03 + M04 + M05 + M06 + M07 + M08 +
##     M09 + M10 + M11 + M12, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -79.461 -22.165   3.847  22.959  72.690
##
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  87.3710     6.1450  14.218  < 2e-16 ***
## M02          11.8839     8.6903   1.367 0.172326
## M03          30.0871     8.6903   3.462 0.000600 ***
## M04          28.4000     8.6903   3.268 0.001187 **
## M05          31.9387     8.6903   3.675 0.000274 ***
## M06          35.8258     8.6903   4.123 4.66e-05 ***
## M07          39.7000     8.6903   4.568 6.76e-06 ***
## M08          38.5194     8.6903   4.432 1.24e-05 ***
## M09          34.5903     8.6903   3.980 8.33e-05 ***
## M10          23.0097     8.6903   2.648 0.008459 **
## M11           0.8839     8.6903   0.102 0.919045
## M12          -2.1839     8.6903  -0.251 0.801726
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 34.21 on 360 degrees of freedom
## Multiple R-squared:  0.1684, Adjusted R-squared:  0.143
## F-statistic: 6.628 on 11 and 360 DF,  p-value: 4.143e-10
```

Part (b)

```
plot(dummyreg$coef,type='l',ylab='Seasonal Factors', xlab="Season",lwd=
2, main="Plot of Estimated Seasonal Factors")
```

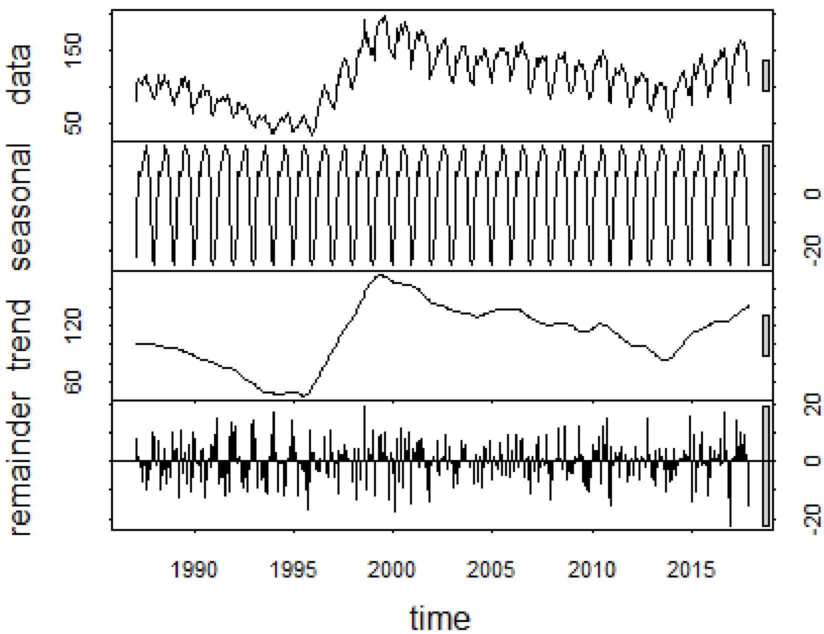



```
#hist(dummyreg$res,main="Histogram of Residuals",col="skyblue3")
```

It appears that between April and September - the summer months - more housing units are started, which makes sense.

Part (c)

```
plot(stl(Value,s.window="periodic"))
```




```
forecast(Value)

##          Point Forecast     Lo 80    Hi 80     Lo 95    Hi 95
## Jan 2018       108.0967  97.07366 119.1197  91.23843 124.9549
## Feb 2018       124.5808 109.99022 139.1715 102.26642 146.8953
## Mar 2018       147.4710 128.07392 166.8681 117.80571 177.1364
## Apr 2018       146.9855 125.60759 168.3633 114.29082 179.6801
## May 2018       150.4539 126.53349 174.3742 113.87081 187.0369
## Jun 2018       156.0971 129.20770 182.9866 114.97329 197.2210
## Jul 2018       161.0553 131.20798 190.9025 115.40779 206.7027
## Aug 2018       158.3136 126.93284 189.6945 110.32084 206.3065
## Sep 2018       153.9406 121.46201 186.4192 104.26887 203.6124
## Oct 2018       139.3623 108.19613 170.5284  91.69777 187.0268
## Nov 2018       112.3532  85.81593 138.8905  71.76794 152.9385
## Dec 2018       108.2477  81.32873 135.1666  67.07872 149.4166
## Jan 2019       111.5842  82.44957 140.7188  67.02665 156.1417
## Feb 2019       128.4825  93.34773 163.6172  74.74851 182.2165
## Mar 2019       151.9547 108.53085 195.3786  85.54364 218.3658
## Apr 2019       151.3243 106.22479 196.4239  82.35052 220.2982
## May 2019       154.7662 106.74916 202.7833  81.33046 228.2020
## Jun 2019       160.4418 108.70868 212.1749  81.32284 239.5607
```

```
## Jul 2019       165.4085 110.06389 220.7531  80.76622 250.0508
## Aug 2019       162.4696 106.13911 218.8000  76.31957 248.6196
## Sep 2019       157.8657 101.22272 214.5087  71.23773 244.4937
## Oct 2019       142.8138  89.84873 195.7790  61.81069 223.8170
## Nov 2019       115.0563  71.00065 159.1120  47.67898 182.4337
## Dec 2019       110.7777  67.02966 154.5258  43.87082 177.6846
```

```
summary(forecast(Value))
```

```
##
## Forecast method: ETS(M,Ad,M)
##
## Model Information:
## ETS(M,Ad,M)
##
## Call:
##  ets(y = object, lambda = lambda, allow.multiplicative.trend = allow.
multiplicative.trend)
##
##   Smoothing parameters:
##     alpha = 0.5408
##     beta  = 0.024
##     gamma = 1e-04
##     phi   = 0.9737
##
##   Initial states:
##     l = 106.8174
##     b = -0.2208
##     s=0.7683 0.7993 0.9938 1.1005 1.1346 1.1572
##            1.1246 1.0869 1.0649 1.0715 0.9079 0.7902
##
##   sigma:  0.0796
##
##      AIC     AICc      BIC
## 3802.595 3804.532 3873.135
##
## Error measures:
##                     ME     RMSE      MAE        MPE     MAPE      MA
SE
## Training set 0.1218285 7.960887 6.275775 -0.1858535 6.160249 0.46126
56
##                    ACF1
## Training set -0.0706053
##
## Forecasts:
##          Point Forecast     Lo 80    Hi 80     Lo 95    Hi 95
## Jan 2018       108.0967  97.07366 119.1197  91.23843 124.9549
## Feb 2018       124.5808 109.99022 139.1715 102.26642 146.8953
## Mar 2018       147.4710 128.07392 166.8681 117.80571 177.1364
## Apr 2018       146.9855 125.60759 168.3633 114.29082 179.6801
```

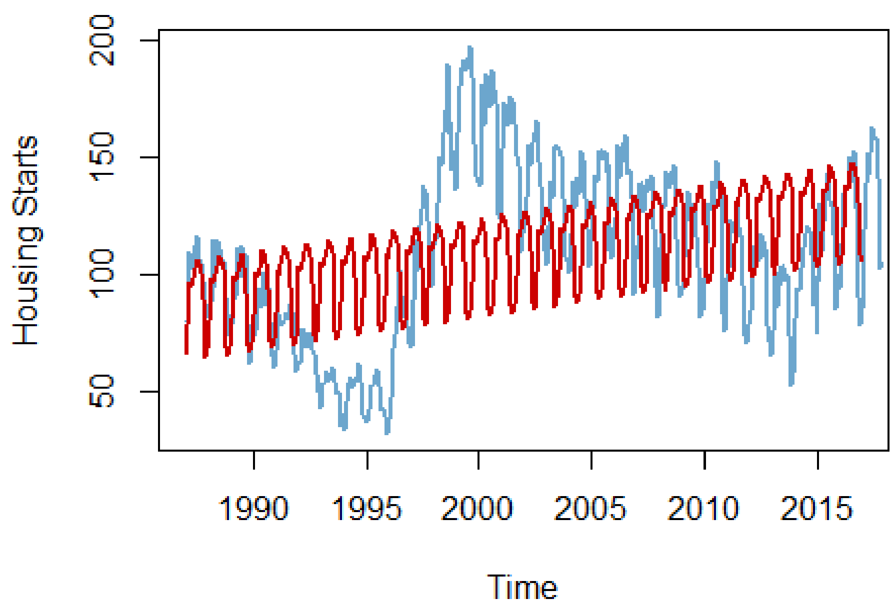
```
## May 2018       150.4539 126.53349 174.3742 113.87081 187.0369
## Jun 2018       156.0971 129.20770 182.9866 114.97329 197.2210
## Jul 2018       161.0553 131.20798 190.9025 115.40779 206.7027
## Aug 2018       158.3136 126.93284 189.6945 110.32084 206.3065
## Sep 2018       153.9406 121.46201 186.4192 104.26887 203.6124
## Oct 2018       139.3623 108.19613 170.5284  91.69777 187.0268
## Nov 2018       112.3532  85.81593 138.8905  71.76794 152.9385
## Dec 2018       108.2477  81.32873 135.1666  67.07872 149.4166
## Jan 2019       111.5842  82.44957 140.7188  67.02665 156.1417
## Feb 2019       128.4825  93.34773 163.6172  74.74851 182.2165
## Mar 2019       151.9547 108.53085 195.3786  85.54364 218.3658
## Apr 2019       151.3243 106.22479 196.4239  82.35052 220.2982
## May 2019       154.7662 106.74916 202.7833  81.33046 228.2020
## Jun 2019       160.4418 108.70868 212.1749  81.32284 239.5607
## Jul 2019       165.4085 110.06389 220.7531  80.76622 250.0508
## Aug 2019       162.4696 106.13911 218.8000  76.31957 248.6196
## Sep 2019       157.8657 101.22272 214.5087  71.23773 244.4937
## Oct 2019       142.8138  89.84873 195.7790  61.81069 223.8170
## Nov 2019       115.0563  71.00065 159.1120  47.67898 182.4337
## Dec 2019       110.7777  67.02966 154.5258  43.87082 177.6846
```

Full Model

```
seasonal<-tslm(Value~season)
summary(seasonal)

## 
## Call:
## tslm(formula = Value ~ season)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -79.461 -22.165   3.847  22.959  72.690 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  87.3710     6.1450  14.218  < 2e-16 ***
## season2      11.8839     8.6903   1.367 0.172326    
## season3      30.0871     8.6903   3.462 0.000600 ***
## season4      28.4000     8.6903   3.268 0.001187 ** 
## season5      31.9387     8.6903   3.675 0.000274 ***
## season6      35.8258     8.6903   4.123 4.66e-05 ***
## season7      39.7000     8.6903   4.568 6.76e-06 ***
## season8      38.5194     8.6903   4.432 1.24e-05 ***
## season9      34.5903     8.6903   3.980 8.33e-05 ***
## season10     23.0097     8.6903   2.648 0.008459 ** 
## season11      0.8839     8.6903   0.102 0.919045    
## season12     -2.1839     8.6903  -0.251 0.801726    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
```

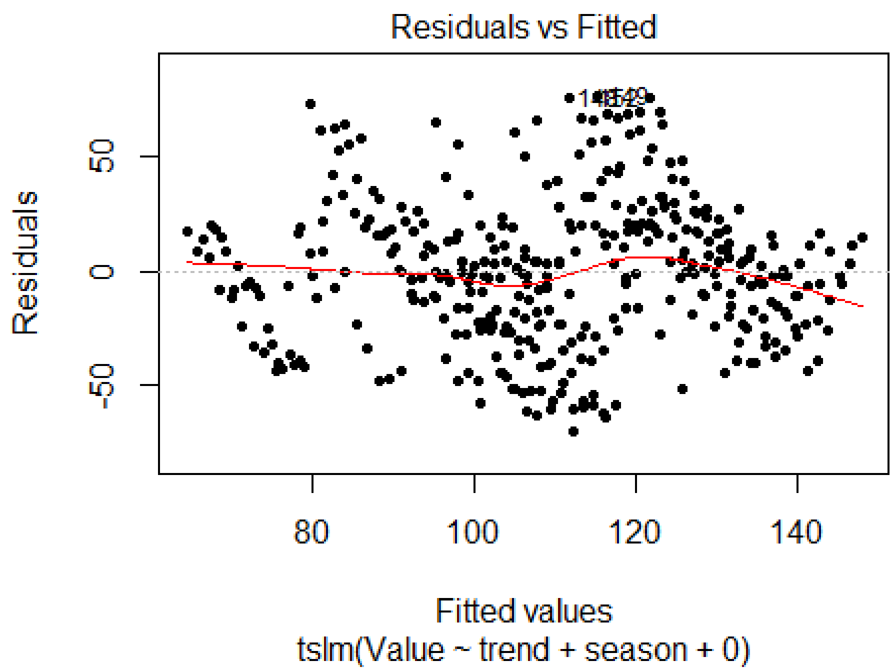
```
## Residual standard error: 34.21 on 360 degrees of freedom
## Multiple R-squared:  0.1684, Adjusted R-squared:  0.143
## F-statistic: 6.628 on 11 and 360 DF,  p-value: 4.143e-10
```

```
fullmodel<-tslm(Value~trend+season+0)

plot(Value,ylab="Housing Starts", xlab="Time", lwd=2, col='skyblue3', x
lim=c(1987,2017))
lines(t,fullmodel$fit,col="red3",lwd=2)
```



Plot of Residuals vs. Fitted Values

```
plot(fullmodel,pch=20,which = c(1))
```



We observe that there is not a particularly discernible trend in the residuals, especially not compared to our previous models.

Part (d)

```
summary(fullmodel)

##
## Call:
## tslm(formula = Value ~ trend + season + 0)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -69.767 -23.219  -0.572  19.342  76.845
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## trend      0.11541    0.01539   7.500 5.06e-13 ***
## season1   66.48121    6.36355  10.447  < 2e-16 ***
## season2   78.24967    6.37030  12.284  < 2e-16 ***
## season3   96.33748    6.37708  15.107  < 2e-16 ***
## season4   94.53497    6.38389  14.808  < 2e-16 ***
## season5   97.95827    6.39073  15.328  < 2e-16 ***
## season6  101.72995    6.39760  15.901  < 2e-16 ***
## season7  105.48873    6.40450  16.471  < 2e-16 ***
## season8  104.19267    6.41143  16.251  < 2e-16 ***
## season9  100.14823    6.41839  15.603  < 2e-16 ***
```

```
## season10  88.45217    6.42538  13.766  < 2e-16 ***
## season11  66.21095    6.43240  10.293  < 2e-16 ***
## season12  63.02780    6.43944   9.788  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 31.86 on 359 degrees of freedom
## Multiple R-squared:  0.9274, Adjusted R-squared:  0.9247
## F-statistic: 352.6 on 13 and 359 DF,  p-value: < 2.2e-16
```

We observe that the Adjusted R-Squared value is 0.9247, so our model is much better than the models we had before. The season dummies have much larger standard errors than the trend, which makes sense as their estimated coefficients are much larger.

Part (e)

```
jarque.bera.test(residuals(fullmodel))
```

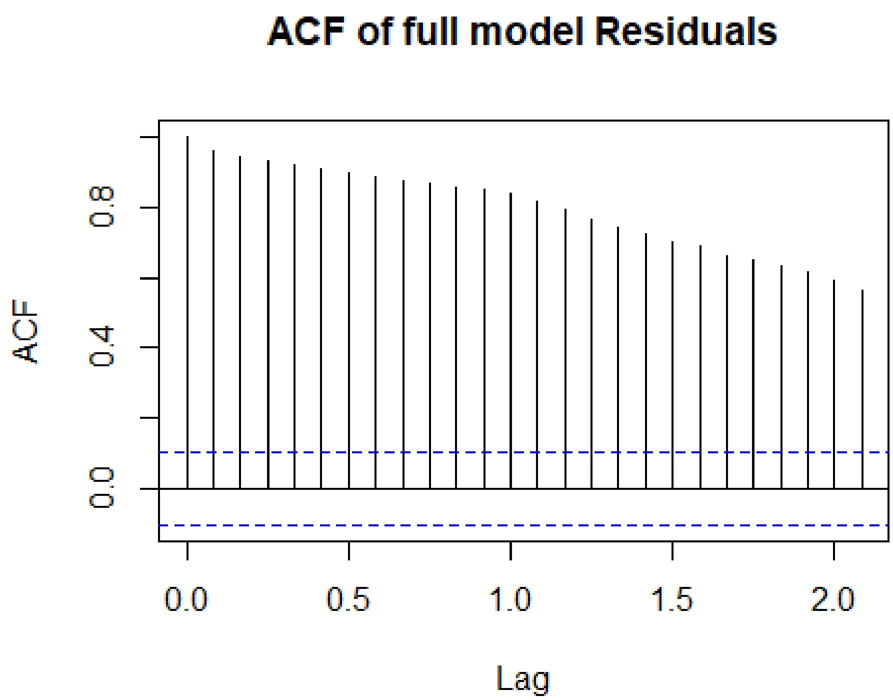
```
##
##  Jarque Bera Test
##
## data:  residuals(fullmodel)
## X-squared = 3.6087, df = 2, p-value = 0.1646
```

The p-value is 0.1646, so we are not able to reject the null hypothesis that the residuals follow a normal distribution.

Part (f)

ACF of residuals

```
acf(residuals(fullmodel),main="ACF of full model Residuals")
```

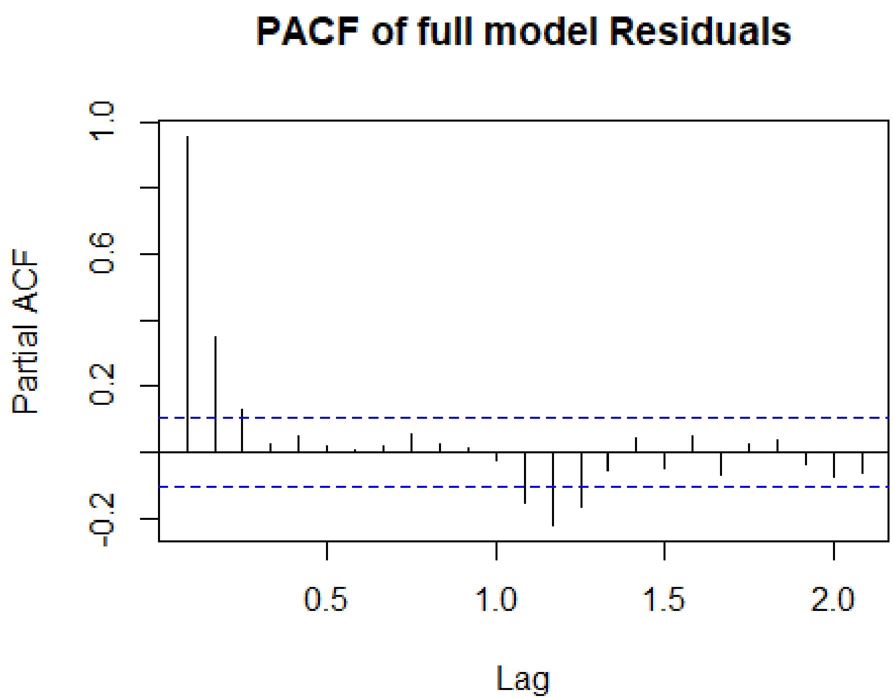



```
#acf(diff(residuals(fullmodel)),main="ACF of full model Residuals")
```

Since the ACF values end at about 2, an MA(2) model could be used to predict them. However, this is much better than the ACF plots of previous model residuals, which were dependent on 20 lags and even 25 lags.

PACF of residuals

```
pacf(residuals(fullmodel),main="PACF of full model Residuals")
```

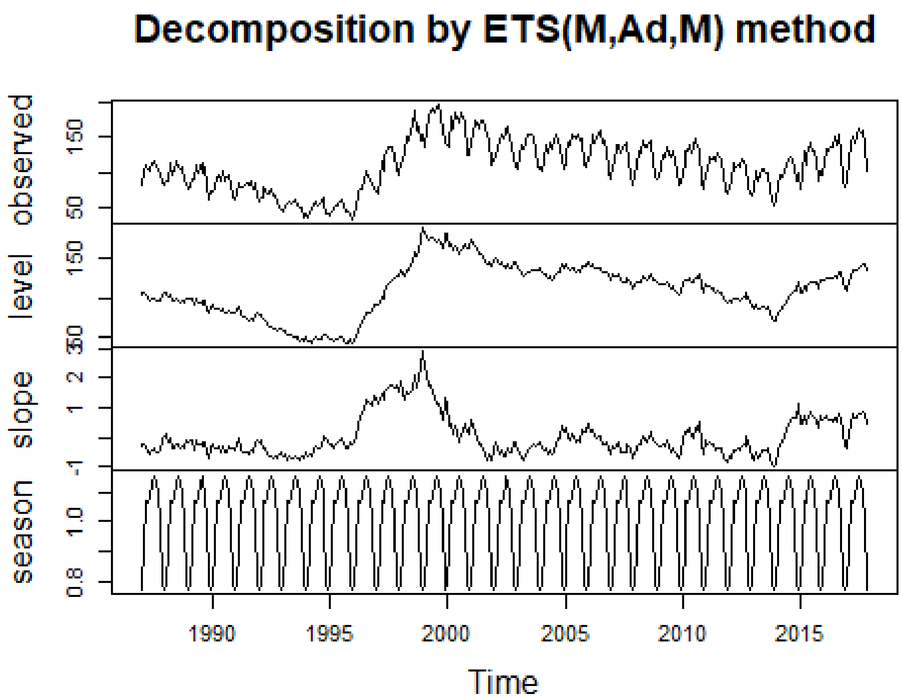



```
#pacf(diff(residuals(fullmodel)),main="PACF of full model Residuals")
```

The significant PACF values end after about 2 months.

Part (g)

```
fit=ets(Value)
plot(fit)
```




```
accuracy(fit)

##                      ME     RMSE      MAE        MPE     MAPE       MA
SE
## Training set 0.1218285 7.960887 6.275775 -0.1858535 6.160249 0.46126
56
##                     ACF1
## Training set -0.0706053

plot(forecast(fit,level=c(50,80,95),h=16))
```

Forecasts from ETS(M,Ad,M)