Project 5

David Contento

January 16, 2019

I) Introduction

The data used in the following analysis is time series data which tracks monthly milk production as measured by pounds per cow. The data was gathered online from a website called the datamarket. The data was originally published in a book for about time series analysis by Jonathan Cryer (1986). The data shows some upward trend with very strong and consistent seasonal components and starts in January 1962, and ends in December 1975.

II) Results

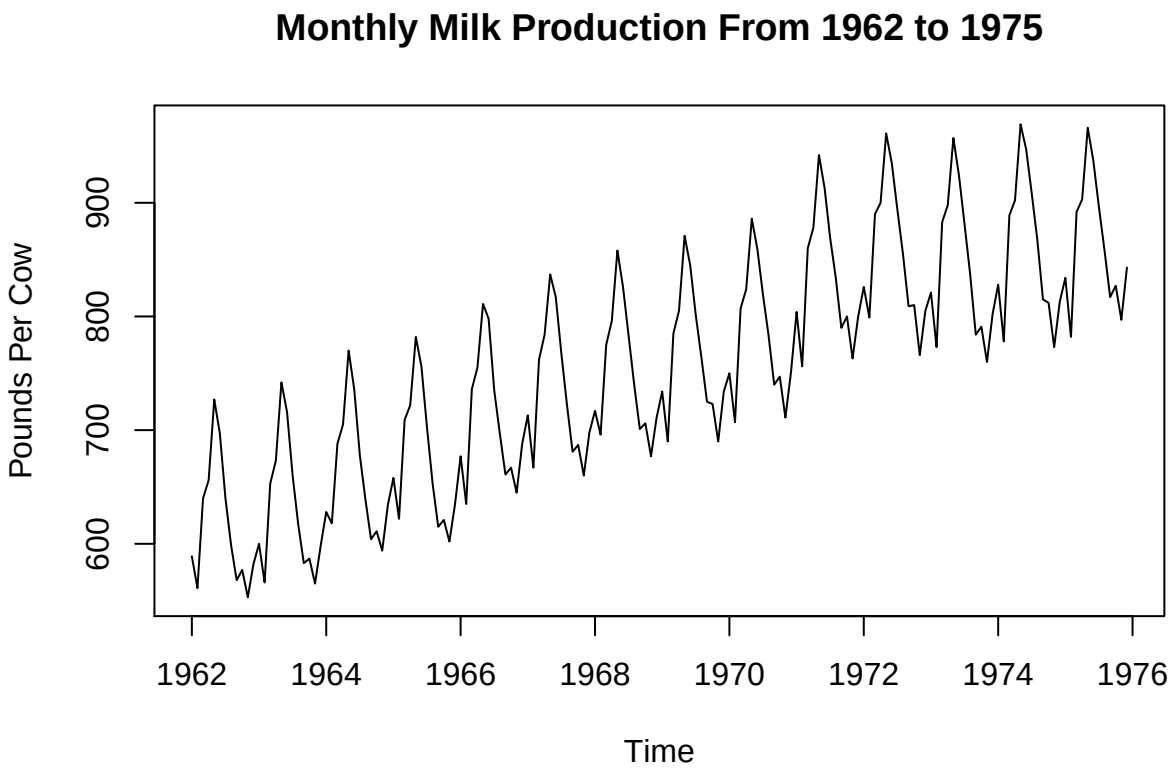
Modeling and Forecasting Trend

1A) Plot of times series data

```
#setting up data and converting to timeseries
setwd("C:/Users/David/Desktop/Grad school work/Winter 2019/403B/project 1")
data = read.csv("milk.csv",header = F)
names(data) = c('date','milkproduction')
data = na.exclude(data)
attach(data)
datats = ts(milkproduction,start=1962,freq=12)

#Creating time variable
time = seq(1962,1976,length=length(datats))

#Times series plot of data
plot.ts(datats,main='Monthly Milk Production From 1962 to 1975',ylab='Pounds Per Cow')
```



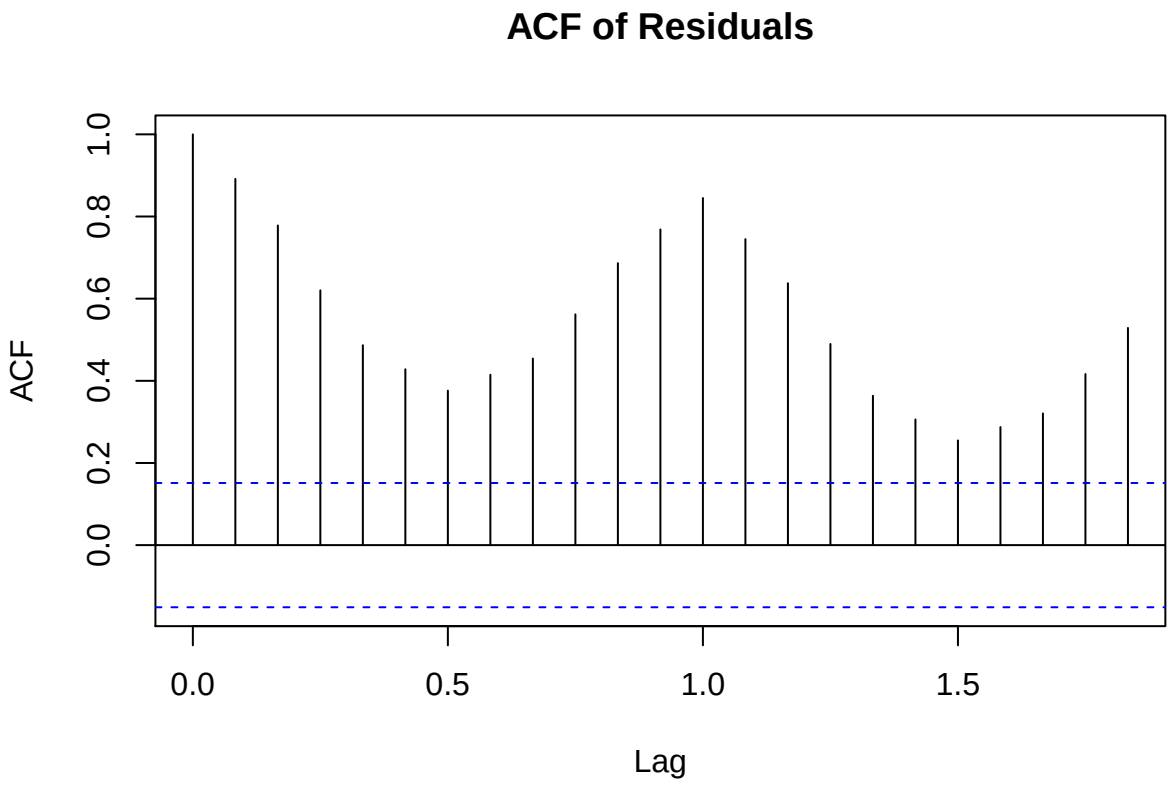
1B) Does the plot suggest covariance stationary

The data does not seem to be covariant stationary. This is evident by the fact that we clearly observe an upward trend.

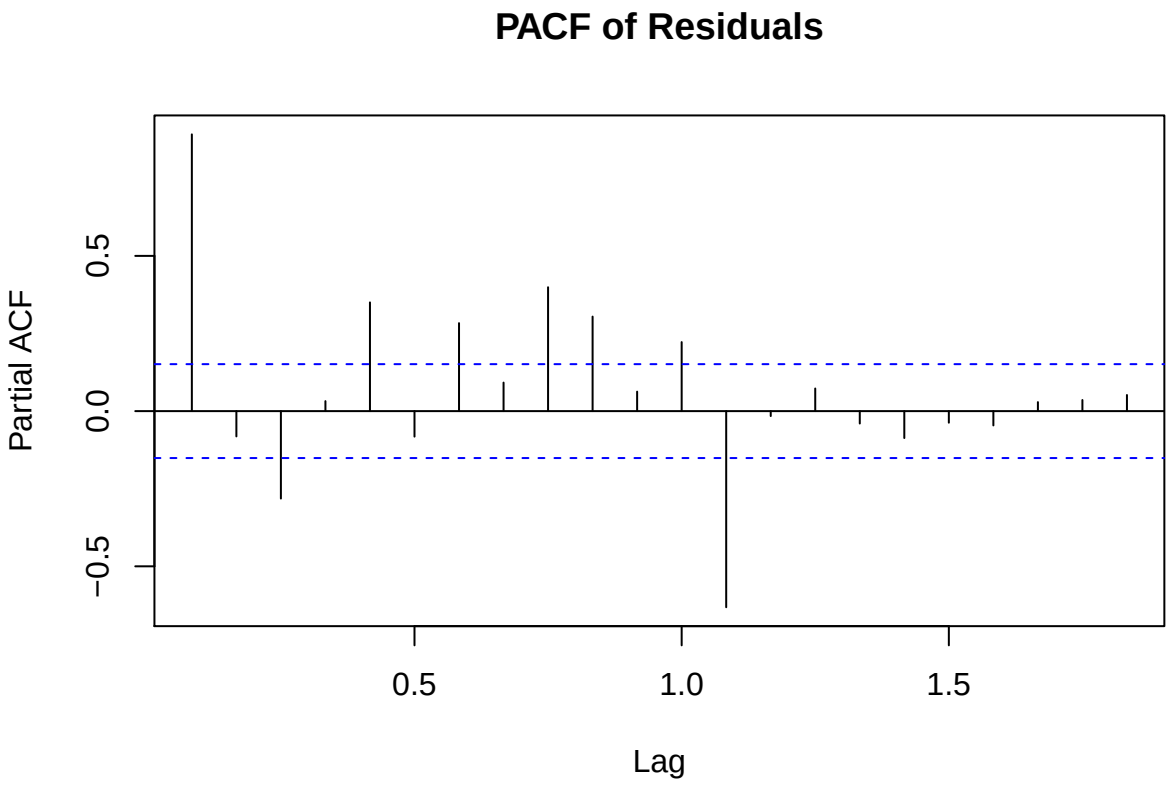
1C) ACF and PACF plots

From the ACF and PACF plots We observe large amounts of auto correlation, which is evident by the large significant spikes in each lag. This suggests that the data is nonstationary.

```
#acf and pcf plots
acf(datats,main='ACF of Residuals')
```



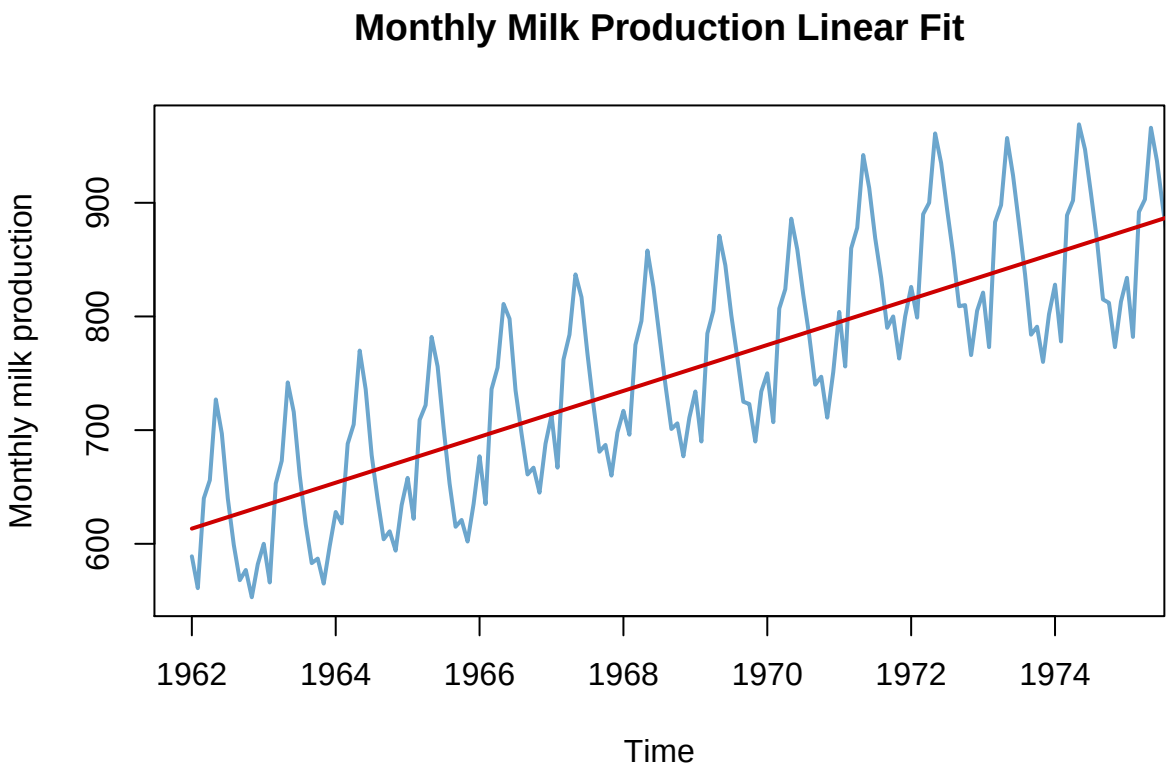

```
pacf(datats,main='PACF of Residuals')
```



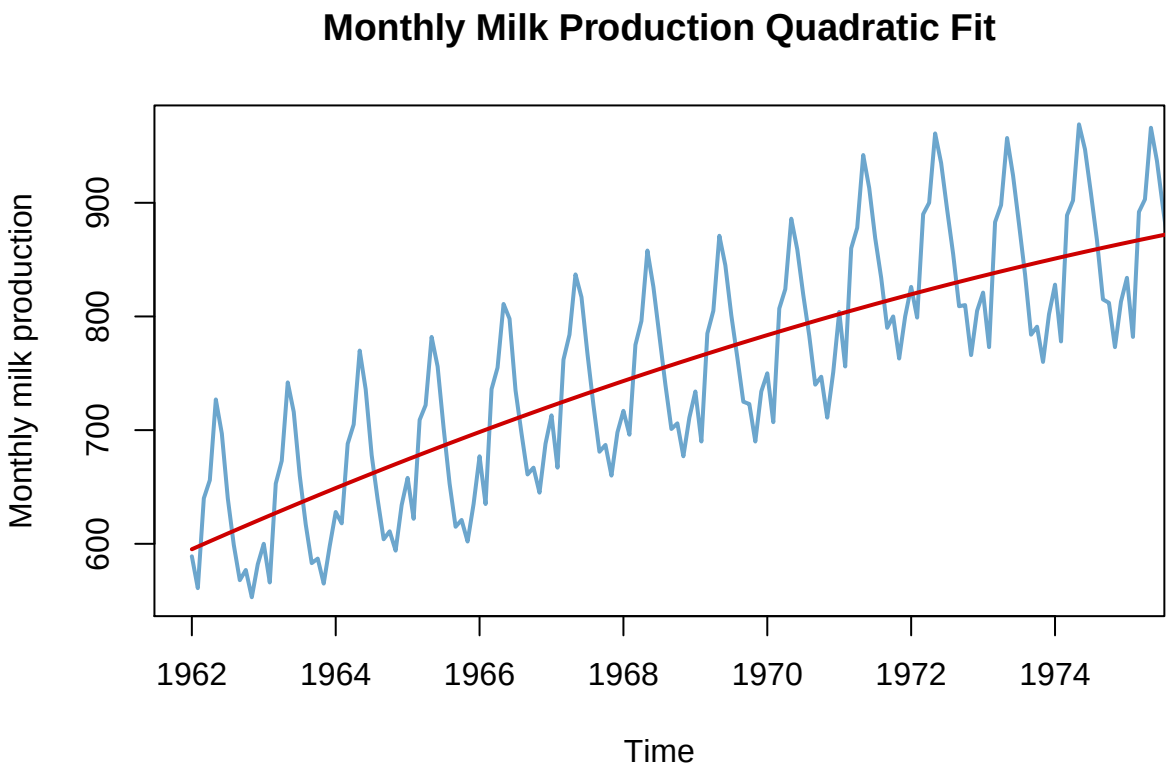
1D) Fitting linear and non-linear model

The code below fits a linear, quadratic, and quadratic+periodic trend to the data.

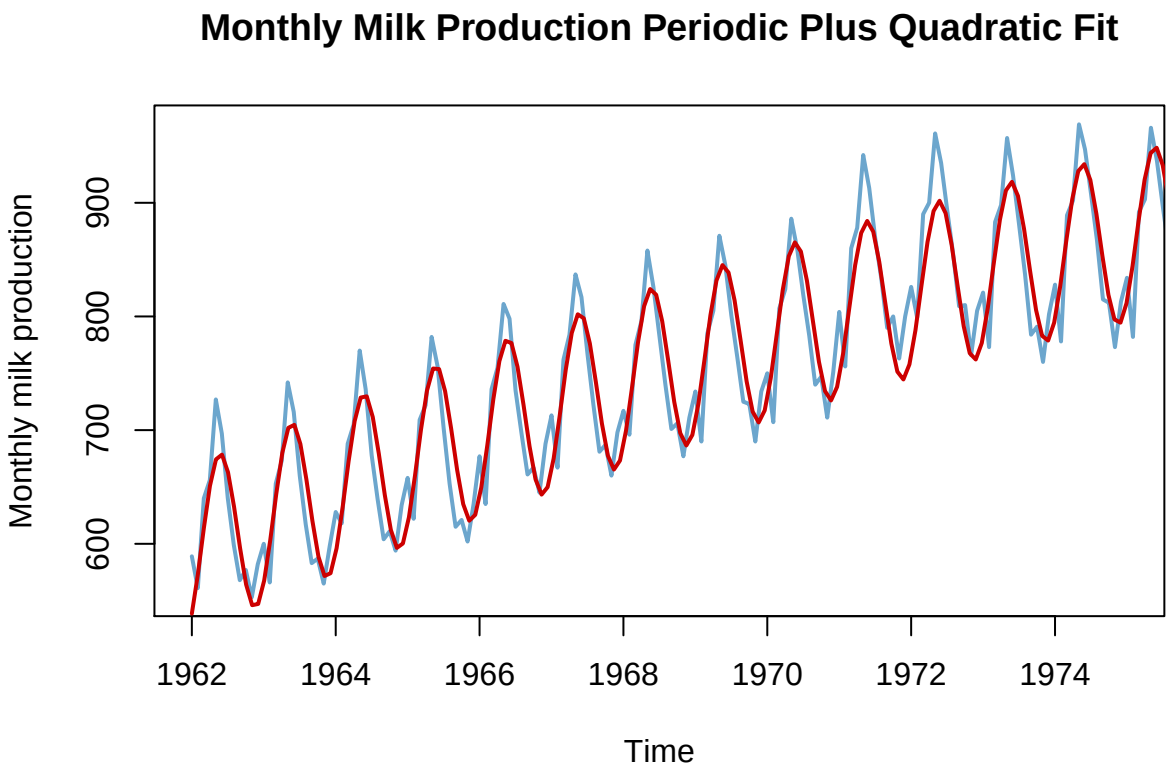
```
#, fig.width=6, fig.height=6
#Linear Fit
mod1=lm(datats~time)
#par(mfrow=c(2,1))
plot(datats, ylab="Monthly milk production",main='Monthly Milk Production Linear Fit', xlab="Time", lwd=
#plot(datats)
lines(time,mod1$fitted.values,col="red3",lwd=2)
```




```
#quadratic fit
mod2=lm(datats~time+I(time^2))
#par(mfrow=c(2,1))
plot(datats,ylab="Monthly milk production",main='Monthly Milk Production Quadratic Fit', xlab="Time", lw
lines(time,mod2$fitted.values,col="red3",lwd=2)
```



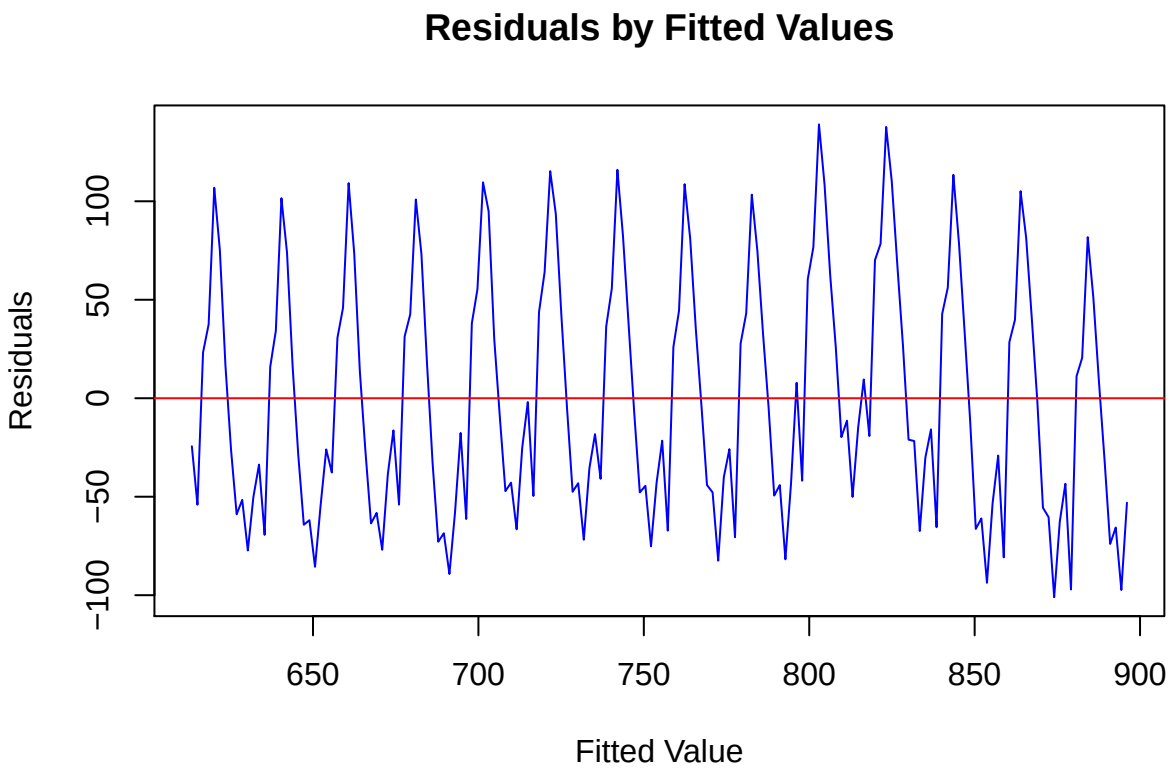

```
#periodic + quadratic fit
sin.t<-sin(2*pi*time)
cos.t<-cos(2*pi*time)
mod3=lm(datats~time+I(time^2) + sin.t + cos.t)
#par(mfrow=c(2,1))
plot(datats,ylab="Monthly milk production",main='Monthly Milk Production Periodic Plus Quadratic Fit', 
lines(time,mod3$fitted.values,col="red3",lwd=2)
```



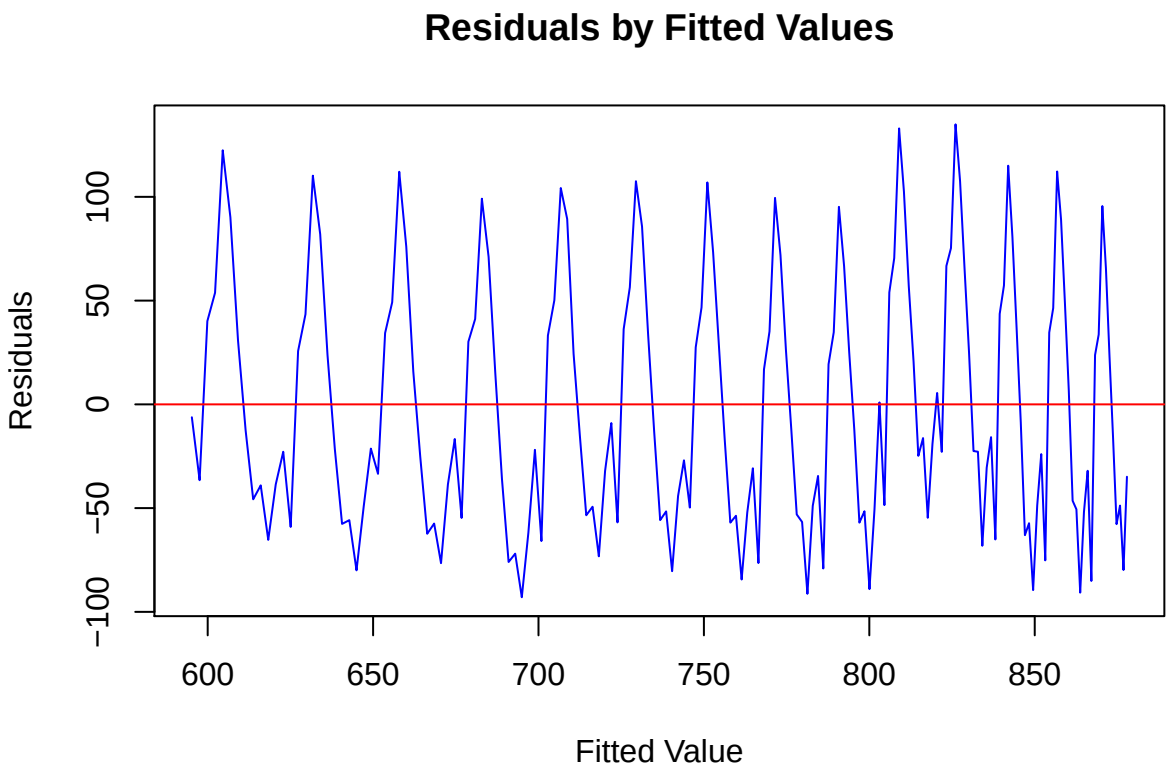
1E) Residuals vs fitted values plot

The residuals for the linear and quadratic+periodic model have a mean of zero (mean of residuals indicated by red line), which shows us that the forecasts will be unbiased. The residuals also have constant variance, which makes the calculation of prediction intervals easier.

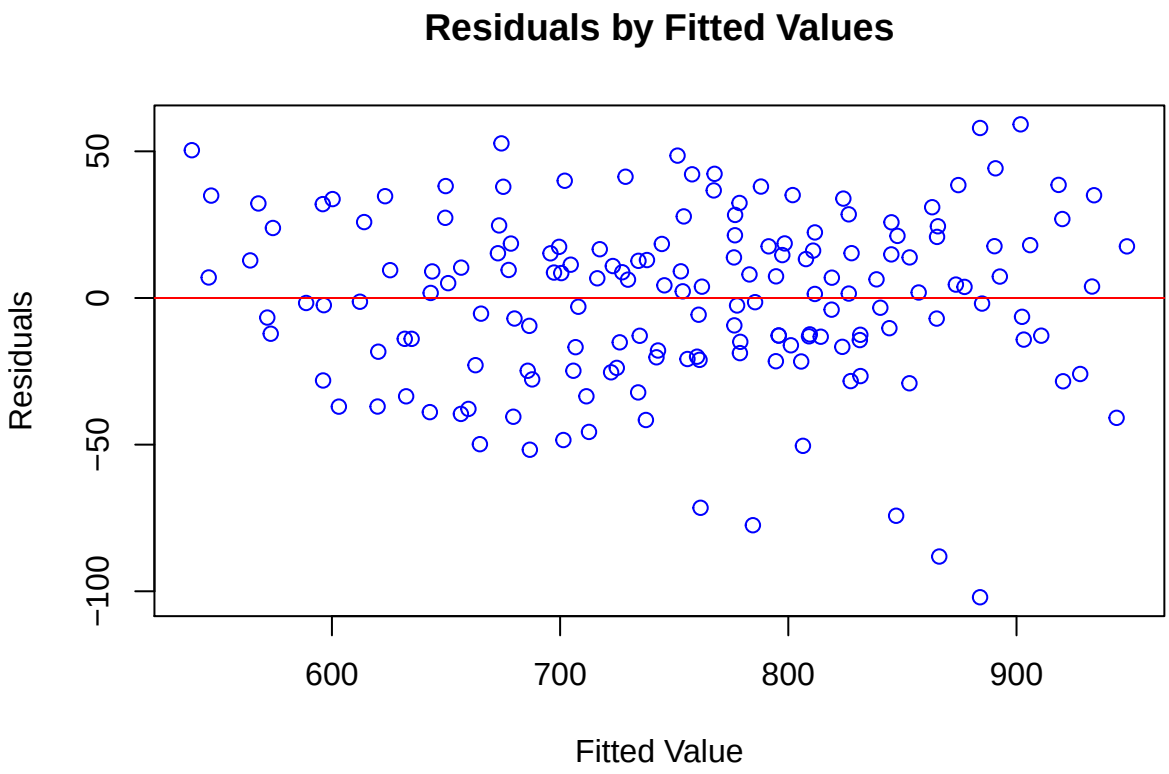
```
#plot of residuals for linear trend model
plot(mod1$fitted.values,mod1$residuals, main='Residuals by Fitted Values', ylab="Residuals",type='l',xla
abline(h=mean(mod1$residuals), col="red")
```




```
#plot of residuals for quadratic trend model
plot(mod2$fitted.values,mod2$residuals, main='Residuals by Fitted Values', ylab="Residuals",type='l',xla
abline(h=mean(mod2$residuals), col="red")
```



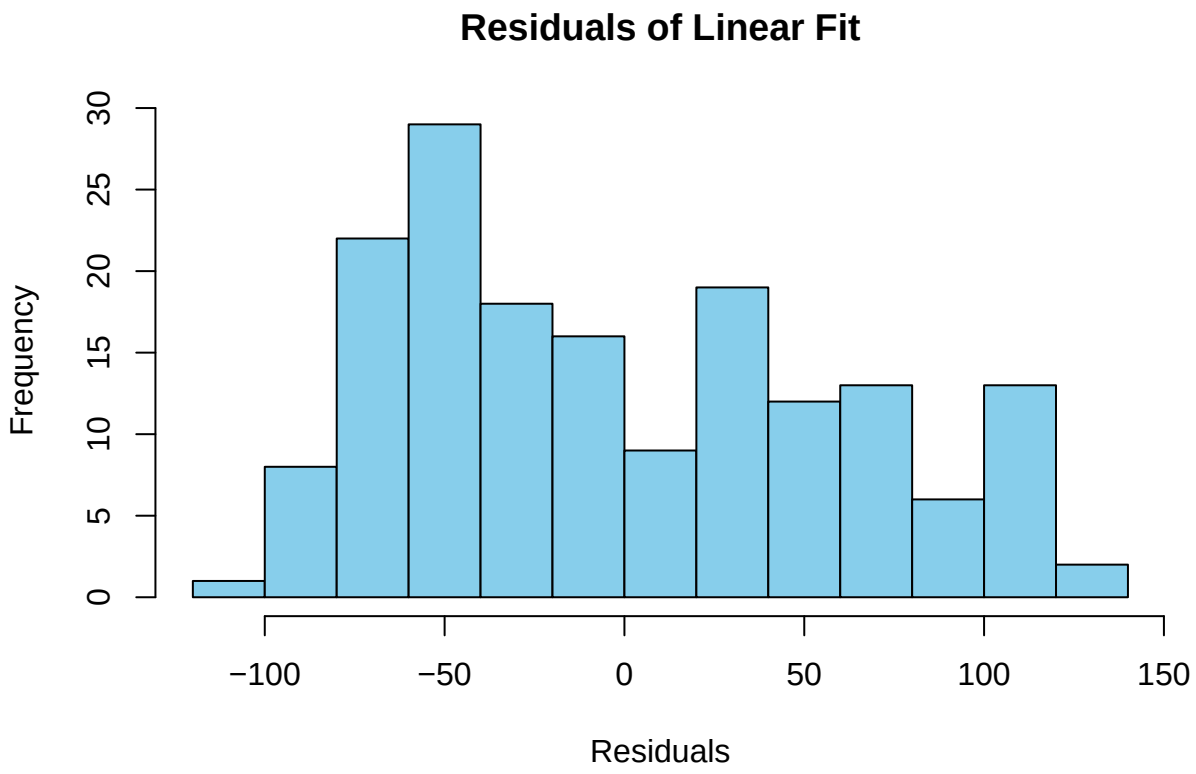

```
#plot of residuals for quadratic+periodic linear trend model
plot(mod3$fitted.values,mod3$residuals, main='Residuals by Fitted Values', ylab="Residuals",xlab="Fitte
abline(h=mean(mod3$residuals), col="red")
```



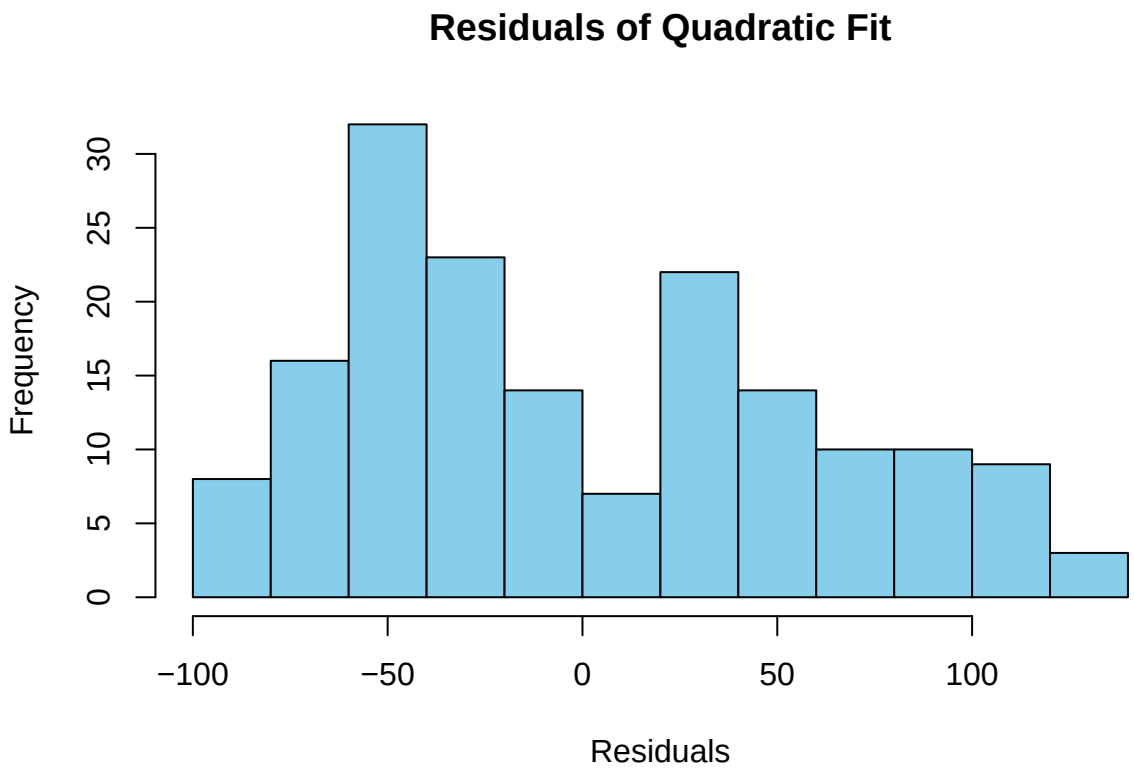
1F) Histogram of residuals

The histograms of the residuals for the linear and quadratic fit do not appear to be normally distributed. The residuals for the quadratic fit on the other hand do appear to be normally distributed, with perhaps a slight left skew.

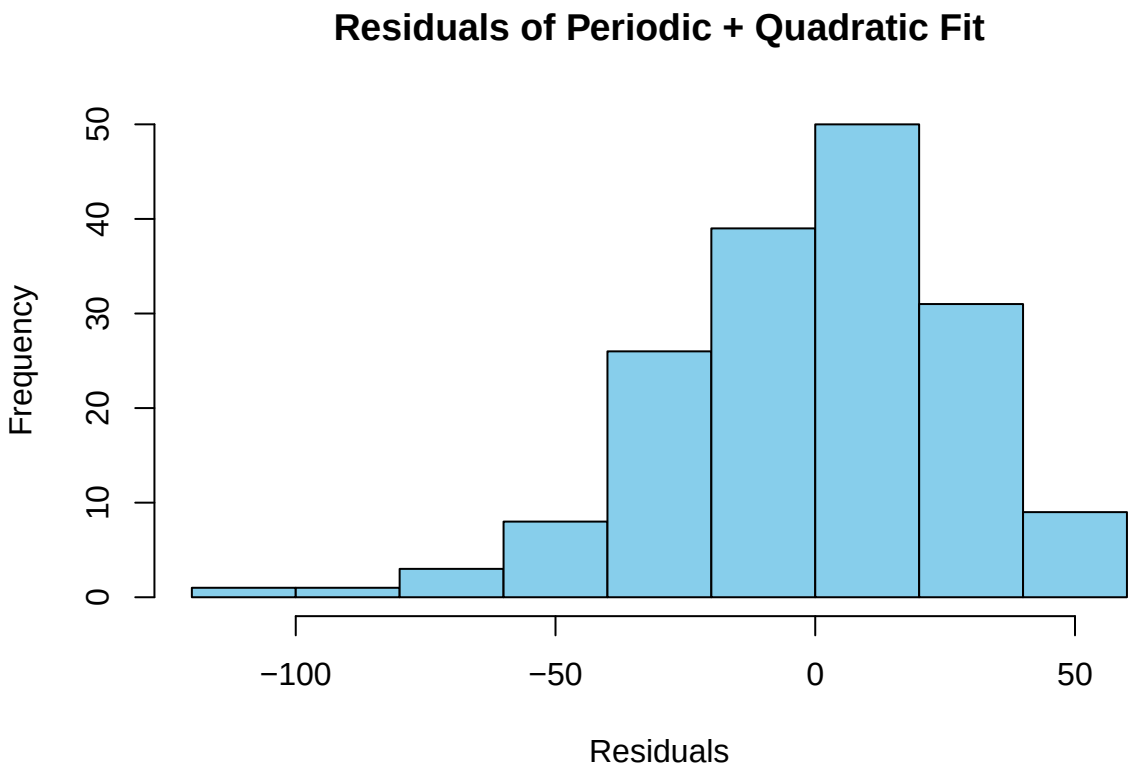
```
#Histogram of Linear model residuals
hist(mod1$residuals,col='skyblue',xlab='Residuals',main='Residuals of Linear Fit')
```




```
#Histogram of Quadratic model residuals
hist(mod2$residuals,col='skyblue',xlab='Residuals',main='Residuals of Quadratic Fit')
```




```
#Histogram of Quadratic+periodic model residuals
hist(mod3$residuals,col='skyblue',xlab='Residuals',main='Residuals of Periodic + Quadratic Fit')
```



1G) Jarque Bera test

Using the code below we ran a Jaque Bera test for normality on the residuals of each model.

```
library(tsoutliers)
#test for normality for linear model
JarqueBera.test(mod1$residuals)
```

```
##
##  Jarque Bera Test
##
## data:  mod1$residuals
## X-squared = 10.77, df = 2, p-value = 0.004585
##
##
##  Skewness
##
## data:  mod1$residuals
## statistic = 0.40457, p-value = 0.03229
##
##
##  Kurtosis
##
## data:  mod1$residuals
## statistic = 2.0599, p-value = 0.01287
```

The low p-value above means that we reject the null hypothesis that the residuals are normally distributed.

```
#test for normality for quadratic model
JarqueBera.test(mod2$residuals)
```

```
##
##  Jarque Bera Test
##
## data:  mod2$residuals
## X-squared = 11.24, df = 2, p-value = 0.003624
##
##
##  Skewness
##
## data:  mod2$residuals
## statistic = 0.41165, p-value = 0.02939
##
##
##  Kurtosis
##
## data:  mod2$residuals
## statistic = 2.0367, p-value = 0.01081
```

Again, the low p-value above means that we reject the null hypothesis that the residuals are normally distributed for model 2.

```
#test for normality for quadratic+periodic model
JarqueBera.test(mod3$residuals)
```

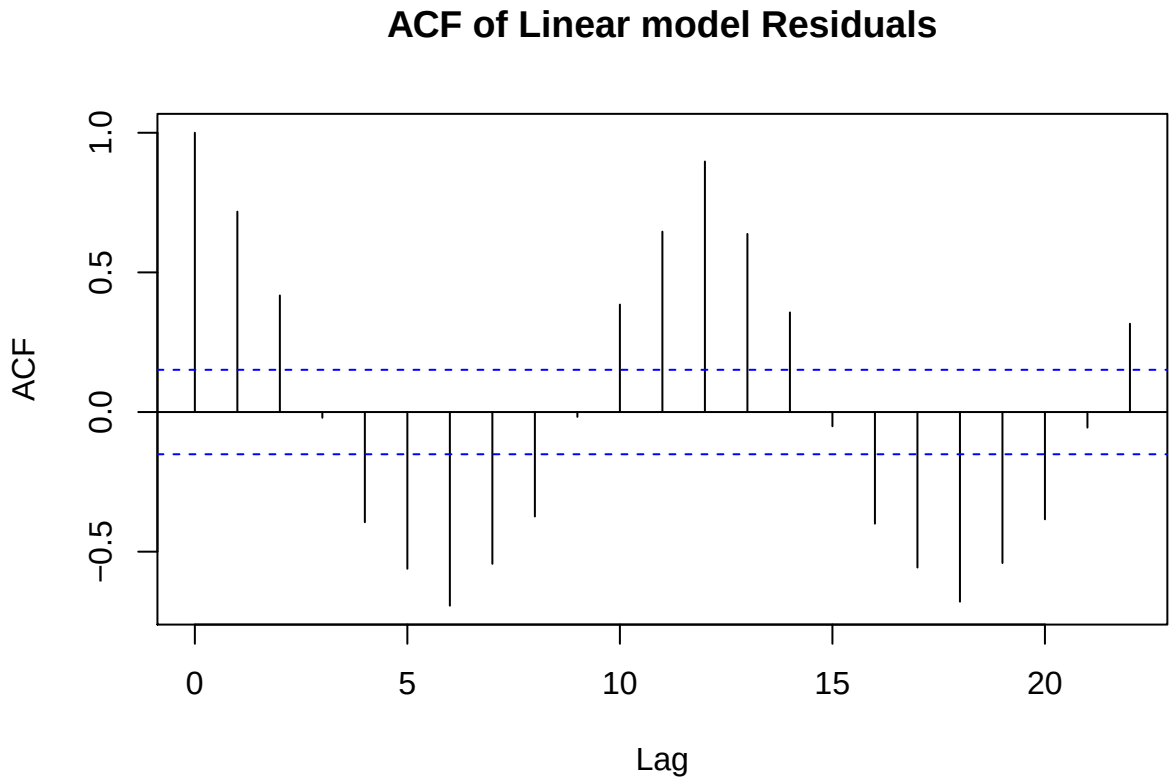
```
##
##  Jarque Bera Test
##
## data:  mod3$residuals
## X-squared = 13.078, df = 2, p-value = 0.001446
##
##
##  Skewness
##
## data:  mod3$residuals
## statistic = 0.59431, p-value = 0.001662
##
##
##  Kurtosis
##
## data:  mod3$residuals
## statistic = 3.6749, p-value = 0.07416
```

we reject the null hypothesis that the residuals for model 3 are normally distributed due to the low p-value of the Jarque Bera test. For all the models we were unable to confirm that the residuals were normally distributed.

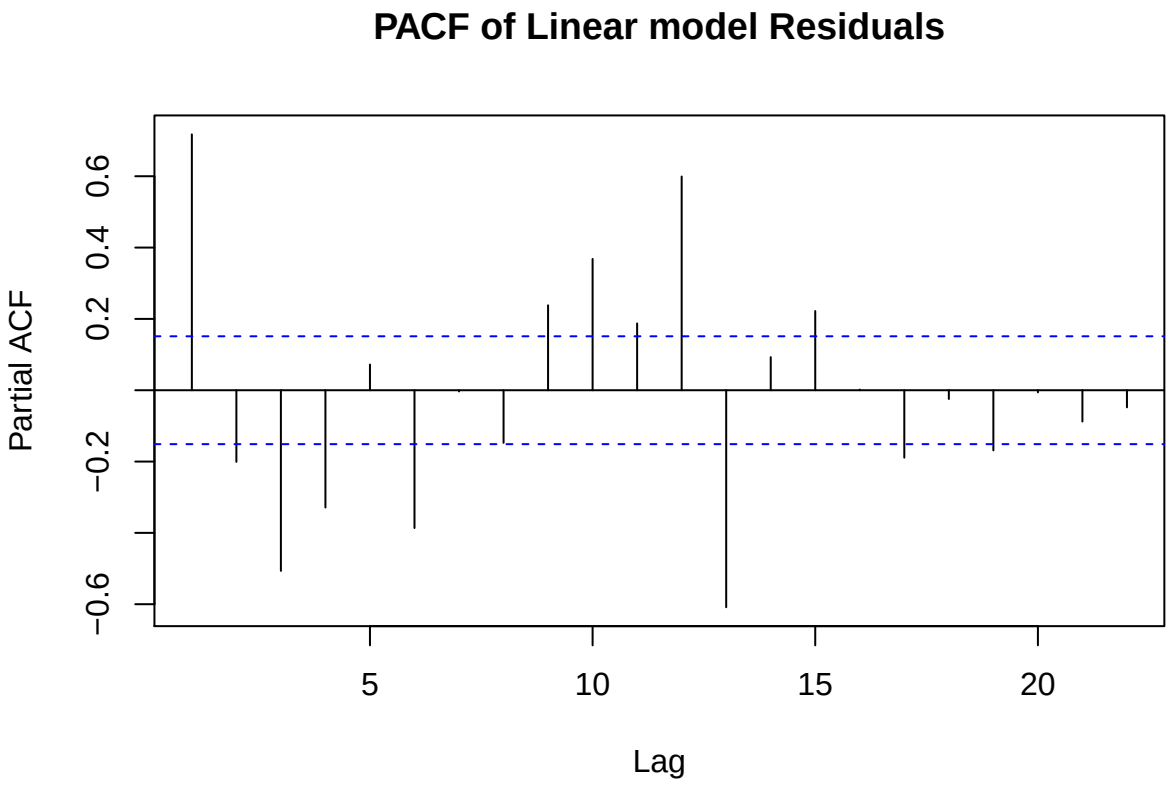
1H)ACF and PACF plots

From the plots below we can see that the Linear model still has large significant spikes in both its ACF and PACF plot. This suggests that the data still has large amounts of autocorrelation and that there is still information left in the residuals which should be used in computing forecasts. The ACF and PACF plots for the quadratic+periodic model are much better, with less significant spikes for each lag. This suggests that quadratic+periodic model has less autocorrelation than the linear model and leaves less information in the residuals that can be used for forecasting.

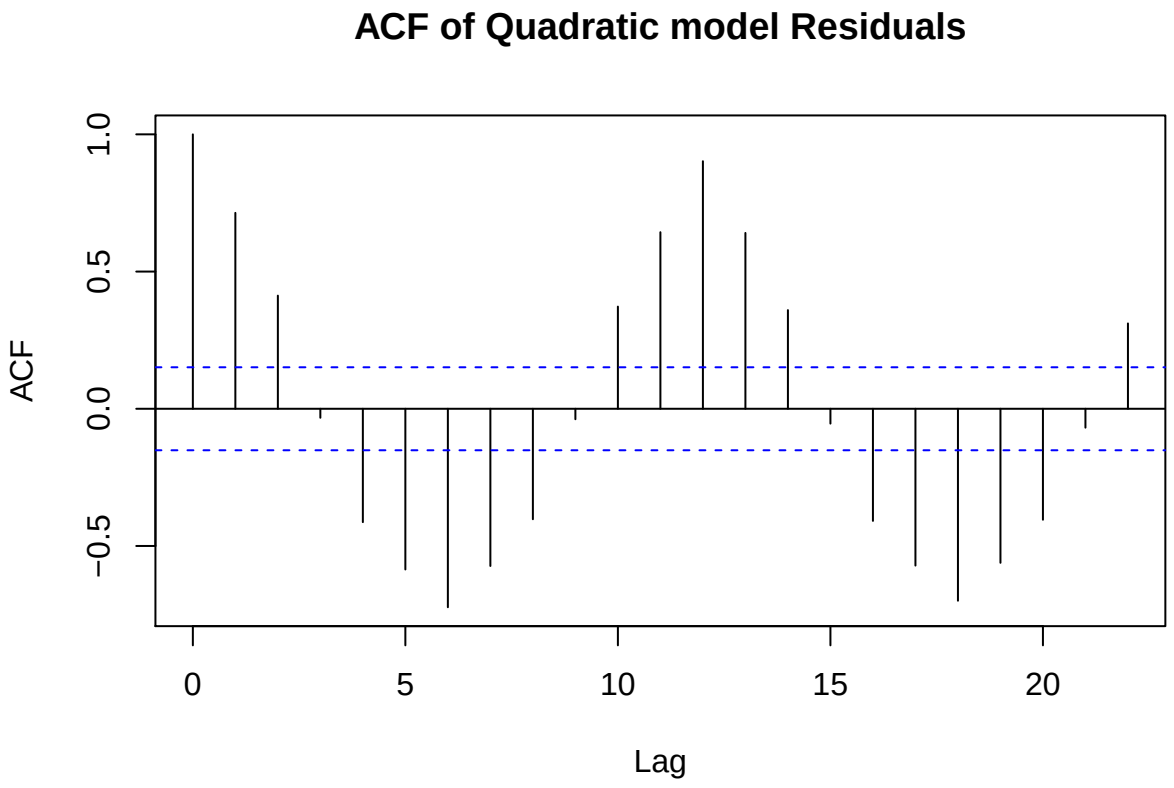
```
#ACF and PACF plots for linear model
acf(mod1$residuals, main="ACF of Linear model Residuals")
```



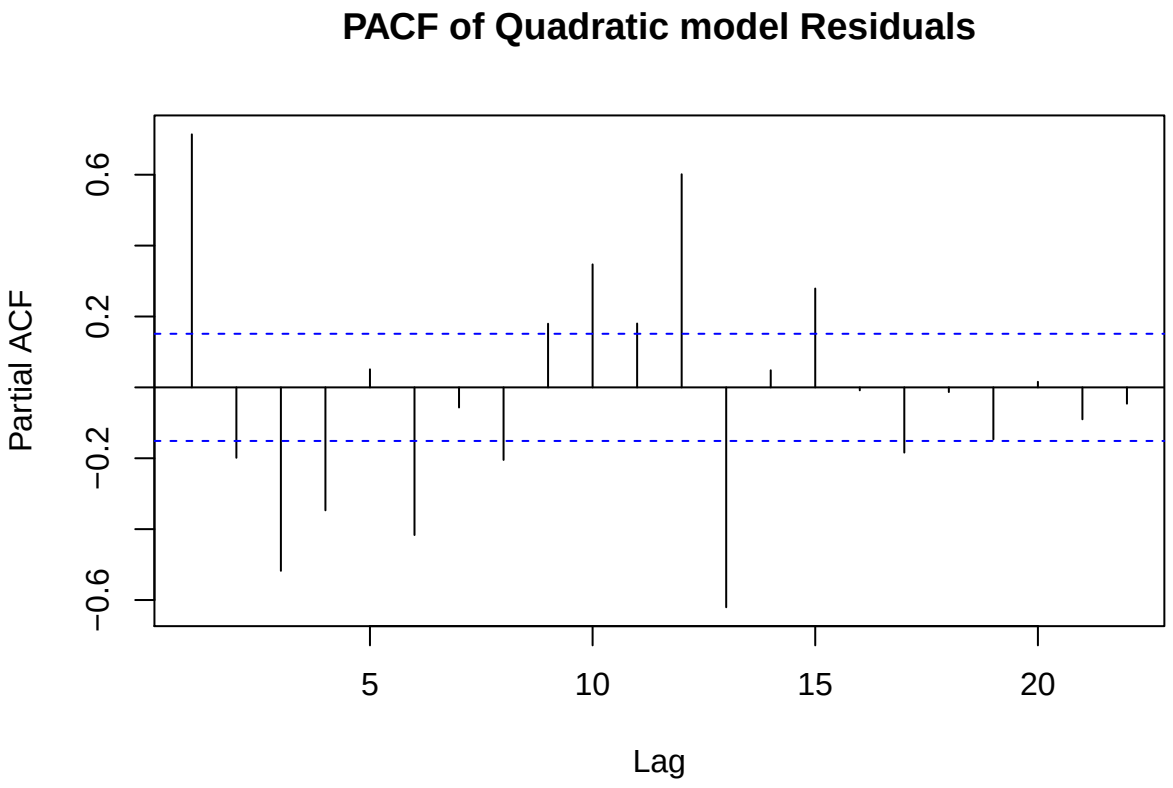

```
pacf(mod1$residuals, main="PACF of Linear model Residuals")
```



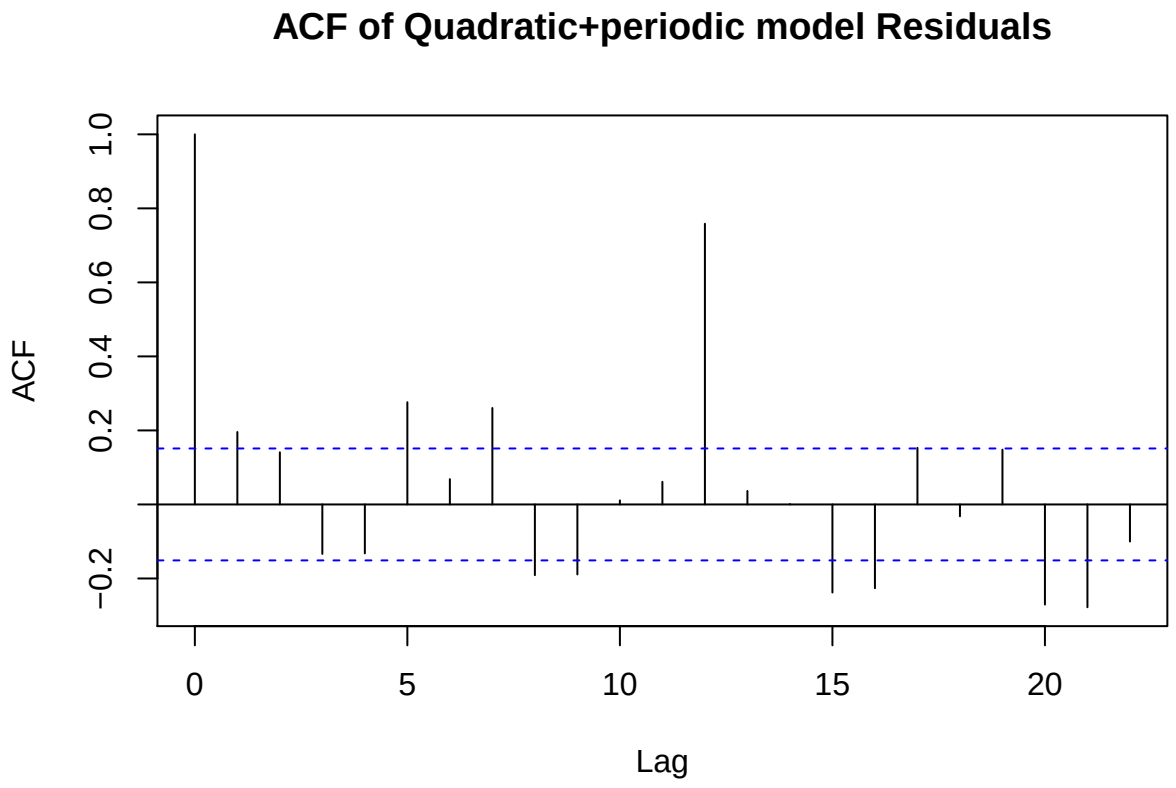

```
#ACF and PACF plots for quadratic model
acf(mod2$residuals, main="ACF of Quadratic model Residuals")
```



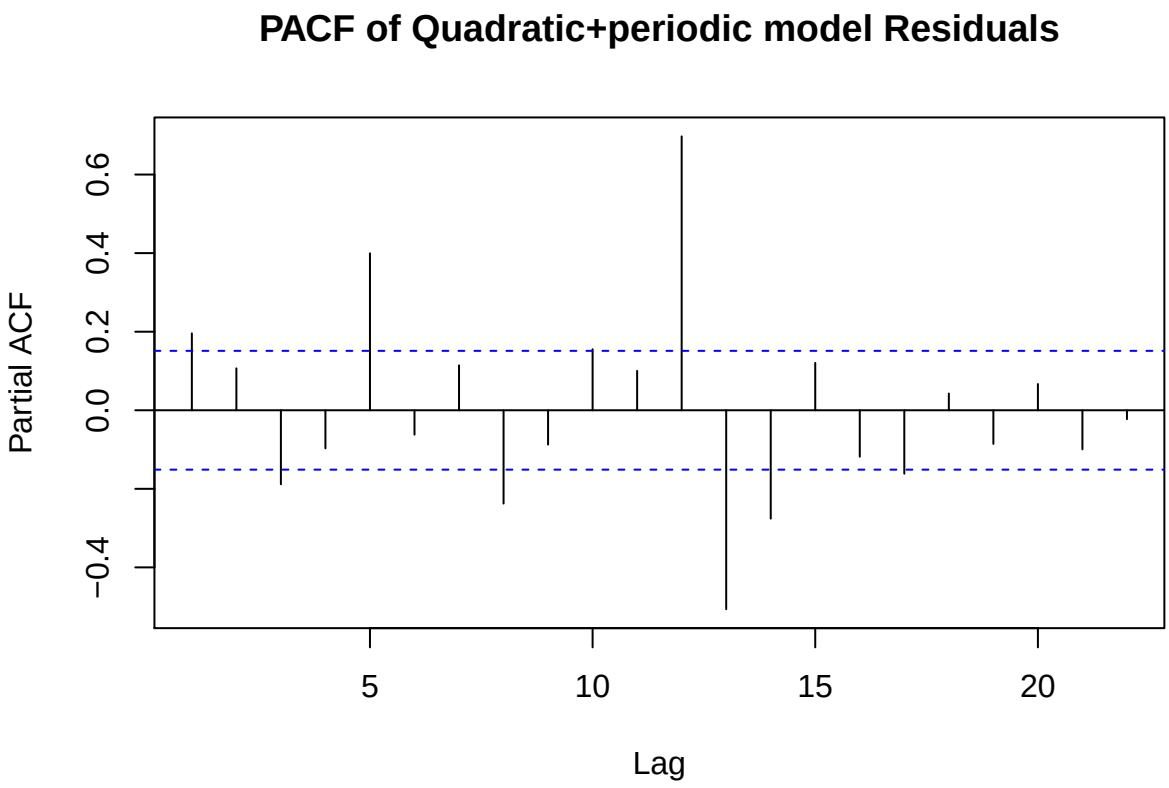

```
pacf(mod2$residuals, main="PACF of Quadratic model Residuals")
```




```r
#ACF and PACF plots for quadratic+periodic model
acf(mod3$residuals, main="ACF of Quadratic+periodic model Residuals")
```




```
pacf(mod3$residuals, main="PACF of Quadratic+periodic model Residuals")
```



1I) Diagnostic statistics

(R2, t???distribution, F???distribution,etc.) The linear model has an adjusted R-squared of 0.5525 which is relatively high for a univariate linear model.

The quadratic model has an adjusted R-squared of 0.5528 which is the same as the linear model. This would favor the linear model since we do not want to increase complexity if it does not explain more variance.

The periodic model has an adjusted R-squared of 0.7922. It is more complex than the previous two models but the great increase in adjusted R-squared may justify its use.

For all three models the F-Statistic rejects the null hypothesis suggesting there is a relationship between our predictors and the response variable.

```
#Summary statistics of each model
summary(mod1)
```

```
##
## Call:
## lm(formula = datats ~ time)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -101.04  -50.02  -15.30   42.88  139.05
##
## Coefficients:
```

```
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -39000.349   2269.588  -17.18   <2e-16 ***
## time            20.190      1.153   17.52   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 60.74 on 166 degrees of freedom
## Multiple R-squared:  0.6489, Adjusted R-squared:  0.6468
## F-statistic: 306.8 on 1 and 166 DF,  p-value: < 2.2e-16
```

```
summary(mod2)
```

```
##
## Call:
## lm(formula = datats ~ time + I(time^2))
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -92.95 -51.66 -15.73  44.30 134.94
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.208e+06  1.221e+06  -1.808   0.0724 .
## time         2.223e+03  1.240e+03   1.792   0.0749 .
## I(time^2)   -5.594e-01  3.150e-01  -1.776   0.0776 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 60.35 on 165 degrees of freedom
## Multiple R-squared:  0.6555, Adjusted R-squared:  0.6513
## F-statistic:   157 on 2 and 165 DF,  p-value: < 2.2e-16
```

```
summary(mod3)
```

```
##
## Call:
## lm(formula = datats ~ time + I(time^2) + sin.t + cos.t)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -102.031  -17.050    3.819   18.451   59.203
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.967e+06  5.877e+05  -3.346  0.00102 **
## time         1.978e+03  5.970e+02   3.313  0.00114 **
## I(time^2)   -4.970e-01  1.516e-01  -3.278  0.00128 **
## sin.t        4.915e+01  3.183e+00  15.444  < 2e-16 ***
## cos.t       -5.574e+01  3.160e+00 -17.638  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 29.04 on 163 degrees of freedom
```

```
## Multiple R-squared:  0.9212, Adjusted R-squared:  0.9193
## F-statistic: 476.4 on 4 and 163 DF,  p-value: < 2.2e-16
```

1J) AIC and BIC plots

Both AIC and BIC agree and suggest that the quadratic+periodic model is the best choice and fits the data the best.

```
#AIC and BIC for each model
AIC(mod1,mod2,mod3)
```

```
##      df      AIC
## mod1  3 1860.572
## mod2  4 1859.390
## mod3  6 1615.555
```

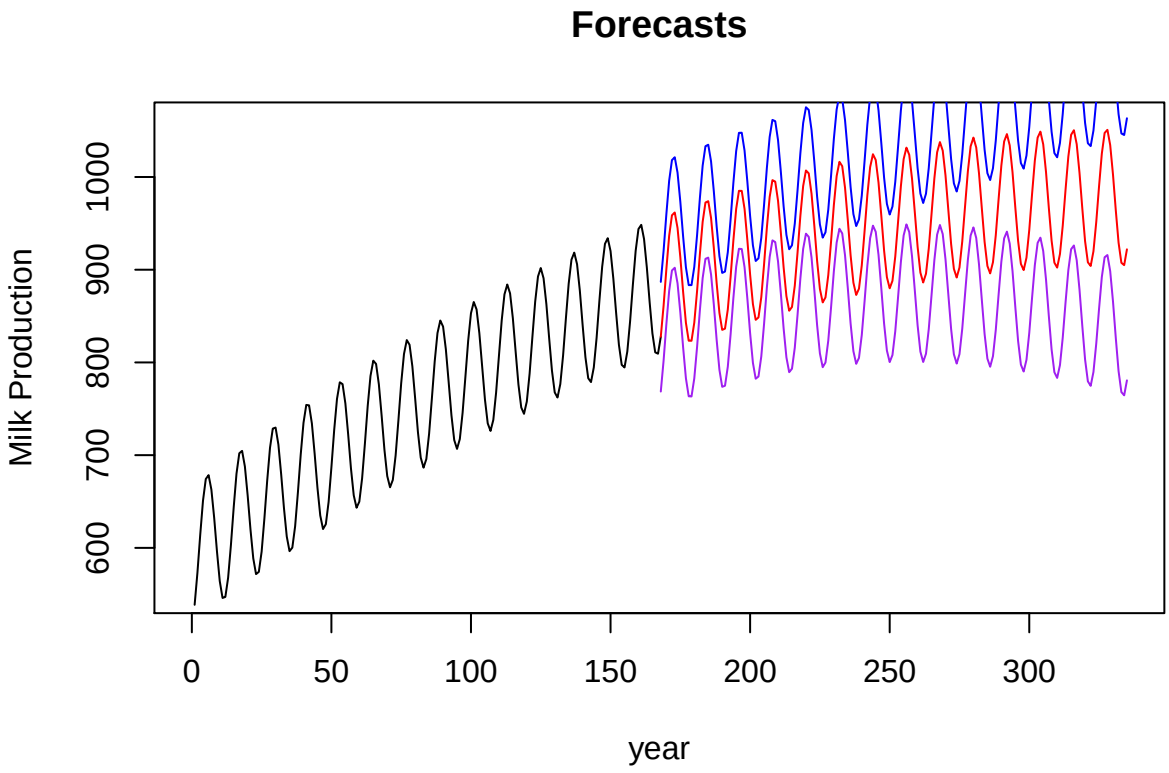
```
BIC(mod1,mod2,mod3)
```

```
##      df      BIC
## mod1  3 1869.944
## mod2  4 1871.886
## mod3  6 1634.298
```

1K) Forecasting trend

With the code below we forecasted the quadratic+periodic trend 20 steps (months) into the future. The red line indicate our point forecasts. The blue and purple lines represent the top and bottom of the 95% prediction interval respectively. As we can see our prediction interval grows the farther into future we forecast.

```
#forcasting quadratic+periodic trend
library(forecast)
t2=seq(1976,1990,length=length(datats))
x=forecast(mod3, newdata=t2, h = 20, level=.95)
plot(mod3$fitted.values,main="Forecasts",xlab="year", ylab="Milk Production",type="l", xlim=c(0,335), y
j=seq(168,335)
lines(j, x$mean, type="l",add=T, col="red")
lines(j, x$upper, type="l",add=T, col="blue")
lines(j, x$lower, type="l", add=T, col="purple")
```



Modeling and Forecasting Seasonality

2A) Creating seasonal dummies

The code below creates a model with a full set of seasonal dummies. One of the seasonal dummy variables (season 1) is removed in order to avoid the dummy variable trap. According to our f-static the seasonal dummies are jointly significant.

```
#Creating seasonal dummies
library(forecast)
seasonal=tslm(datats~season)
summary(seasonal)
```

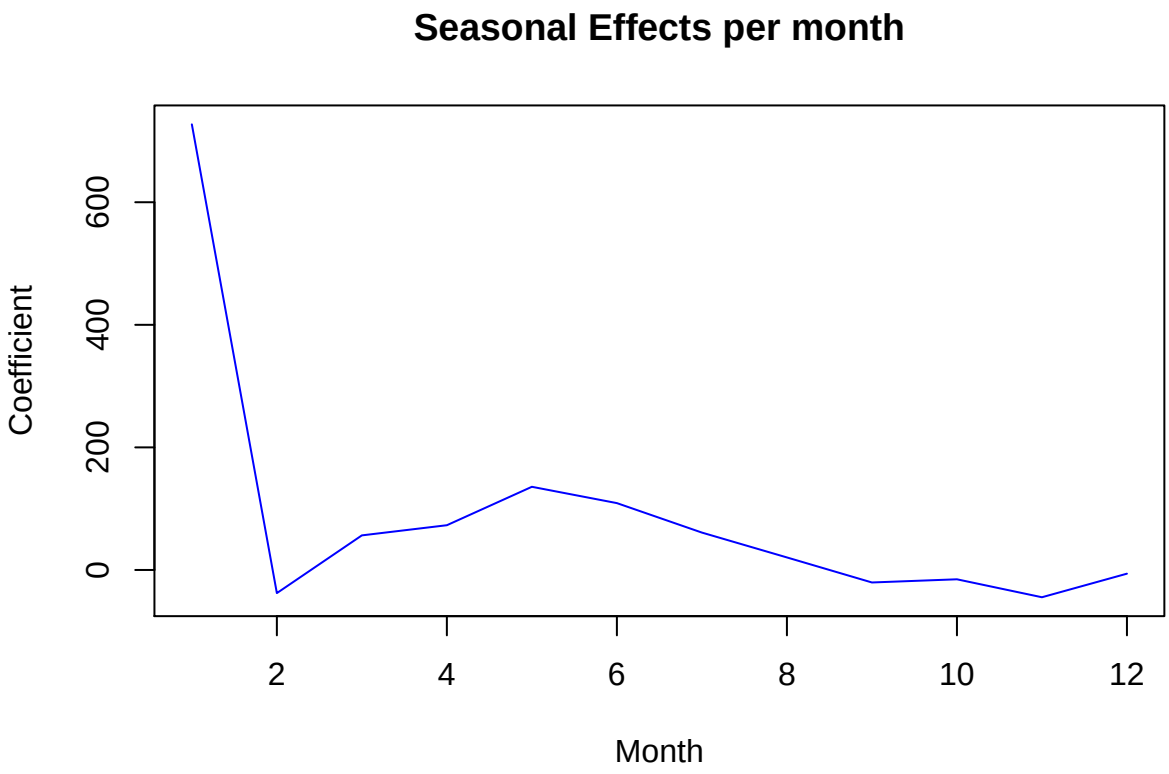
```
##
## Call:
## tslm(formula = datats ~ season)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -148.500  -80.250    1.107   86.839  122.000
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  727.071     23.576  30.840  < 2e-16 ***
```

```
## season2      -37.786     33.341  -1.133  0.25883
## season3       56.429     33.341   1.692  0.09256 .
## season4       73.000     33.341   2.189  0.03005 *
## season5      135.714     33.341   4.070 7.44e-05 ***
## season6      109.071     33.341   3.271  0.00132 **
## season7       61.000     33.341   1.830  0.06923 .
## season8       20.429     33.341   0.613  0.54096
## season9      -20.429     33.341  -0.613  0.54096
## season10     -15.214     33.341  -0.456  0.64880
## season11     -44.500     33.341  -1.335  0.18393
## season12      -6.071     33.341  -0.182  0.85574
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 88.21 on 156 degrees of freedom
## Multiple R-squared:  0.3041, Adjusted R-squared:  0.2551
## F-statistic: 6.198 on 11 and 156 DF,  p-value: 2.055e-08
```

2B) plotting seasonal factors

Based on the following plot the seasonal effects trend upwards until May (month 5) when they peak then trend downwards until they bottom out around November and December. we can ignore the large spike at the beginning since it is an anomaly that stems from removing the first seasonal dummy.

```
#plot factors (I dont think this is what he wants)
plot(seasonal$coefficients,col='blue',xlab='Month',ylab='Coefficient',main='Seasonal Effects per month'
```

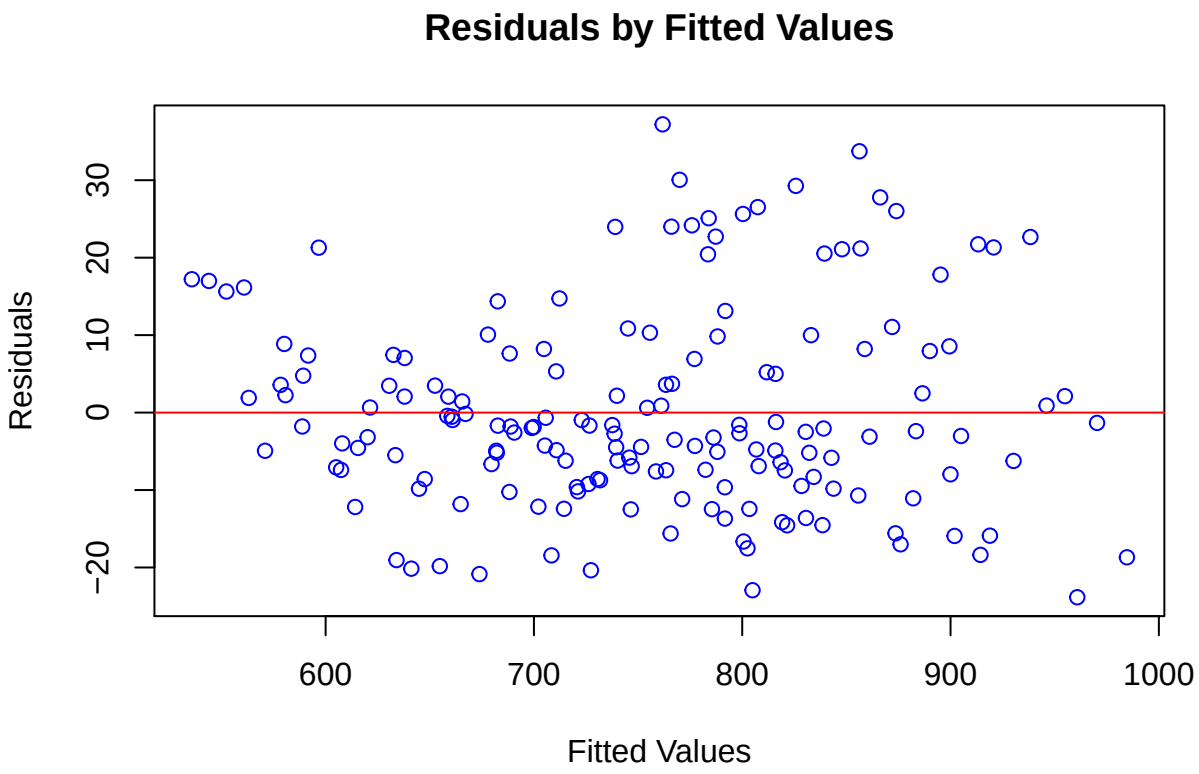


2C) Adding trend to seasonal model

The residuals for the full model have a mean of zero (indicated by the red line), which suggests that our forecasts will be unbiased. We can also see that the residuals also have constant variance. This makes calculating prediction intervals easier.

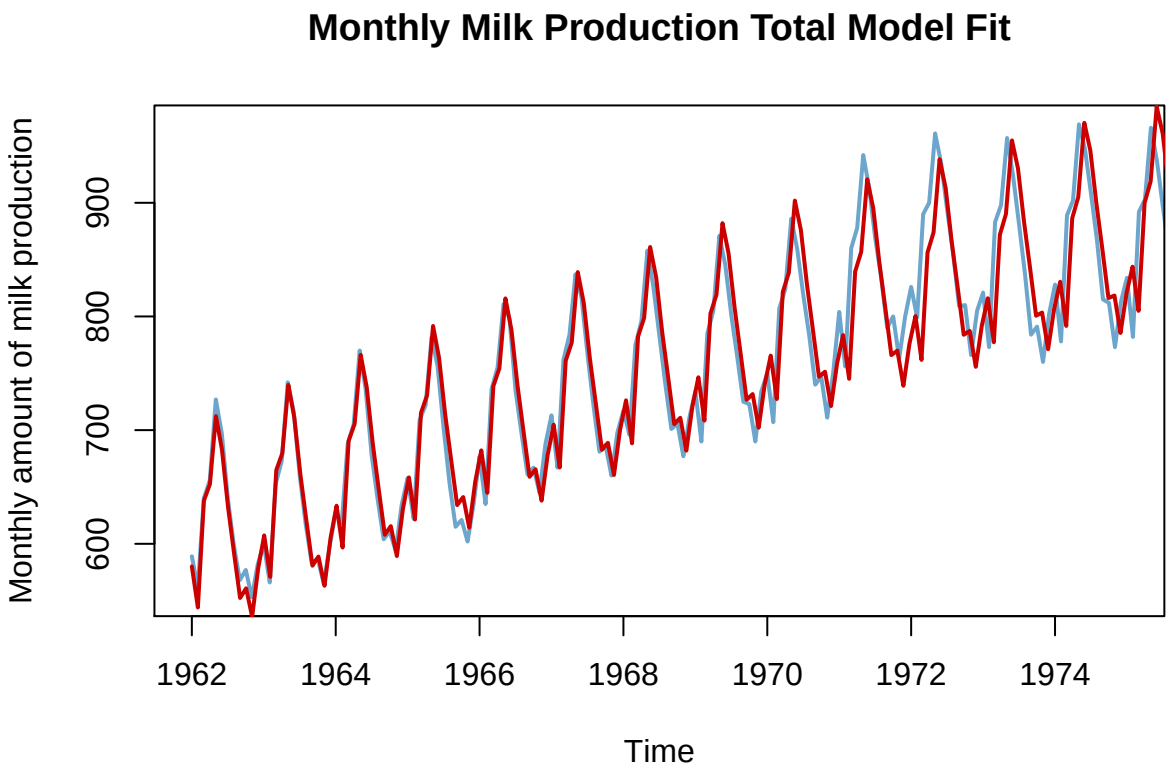
```
#Adding Seasonal effects to our quadratic+periodic trend model
seasontrend = tslm(datats~time + I(time^2) + sin.t + cos.t + season)

#Plotting full model against the data
plot(seasontrend$fitted.values,seasontrend$residuals,col='blue',xlab='Fitted Values',ylab='Residuals',ma
abline(h=mean(seasontrend$residuals), col="red")
```



The Residuals by Fitted Values graph below is fairly consistent and could almost be described as linear. However there does seem to be some weak sinusoidal tendencies.

```
#plotting full model resdiuals vs fitted values
plot(datats,ylab="Monthly amount of milk production", xlab="Time", lwd=2, col='skyblue3', xlim=c(1962,19
lines(time,seasontrend$fitted.values,col="red3",lwd=2)
```



2D) Interpret respective summary statistics

In the full model almost every variable is statistically significant at the 5% confidence level except for the cos.t and season 8 variable. Our f-statistic suggests that the variables used are jointly significant and help explain variation in the explanatory variable. Our R-squared is very high at .98, which suggests that our model can almost completely explain the entire variation in the explanatory variable.

```
#Summary statistic of full model
summary(seasontrend)
```

```
##
## Call:
## tslm(formula = datats ~ time + I(time^2) + sin.t + cos.t + season)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -23.840  -8.580  -2.443   7.388  37.211
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.228e+06  2.755e+05  -8.086 1.81e-13 ***
## time         2.243e+03  2.798e+02   8.015 2.71e-13 ***
## I(time^2)   -5.643e-01  7.106e-02  -7.941 4.13e-13 ***
## sin.t       -2.227e+01  9.735e+00  -2.287 0.023557 *
## cos.t        1.926e+01  9.871e+00   1.951 0.052854 .
```

```
## season2     -2.472e+01  7.166e+00  -3.449 0.000728 ***
## season3      8.211e+01  1.100e+01   7.466 5.98e-12 ***
## season4      1.070e+02  1.473e+01   7.262 1.84e-11 ***
## season5      1.709e+02  1.770e+01   9.656  < 2e-16 ***
## season6      1.376e+02  1.957e+01   7.031 6.50e-11 ***
## season7      7.631e+01  2.015e+01   3.786 0.000220 ***
## season8      1.913e+01  1.941e+01   0.985 0.326087
## season9     -3.773e+01  1.743e+01  -2.165 0.031984 *
## season10    -4.402e+01  1.440e+01  -3.057 0.002645 **
## season11    -7.767e+01  1.068e+01  -7.276 1.71e-11 ***
## season12    -3.574e+01  6.919e+00  -5.166 7.42e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.51 on 152 degrees of freedom
## Multiple R-squared:  0.9841, Adjusted R-squared:  0.9825
## F-statistic: 626.9 on 15 and 152 DF,  p-value: < 2.2e-16
```

2E) Jarque Bera test

The Jarque Bera Test rejects the null hypothesis that the residuals are normally distributed.

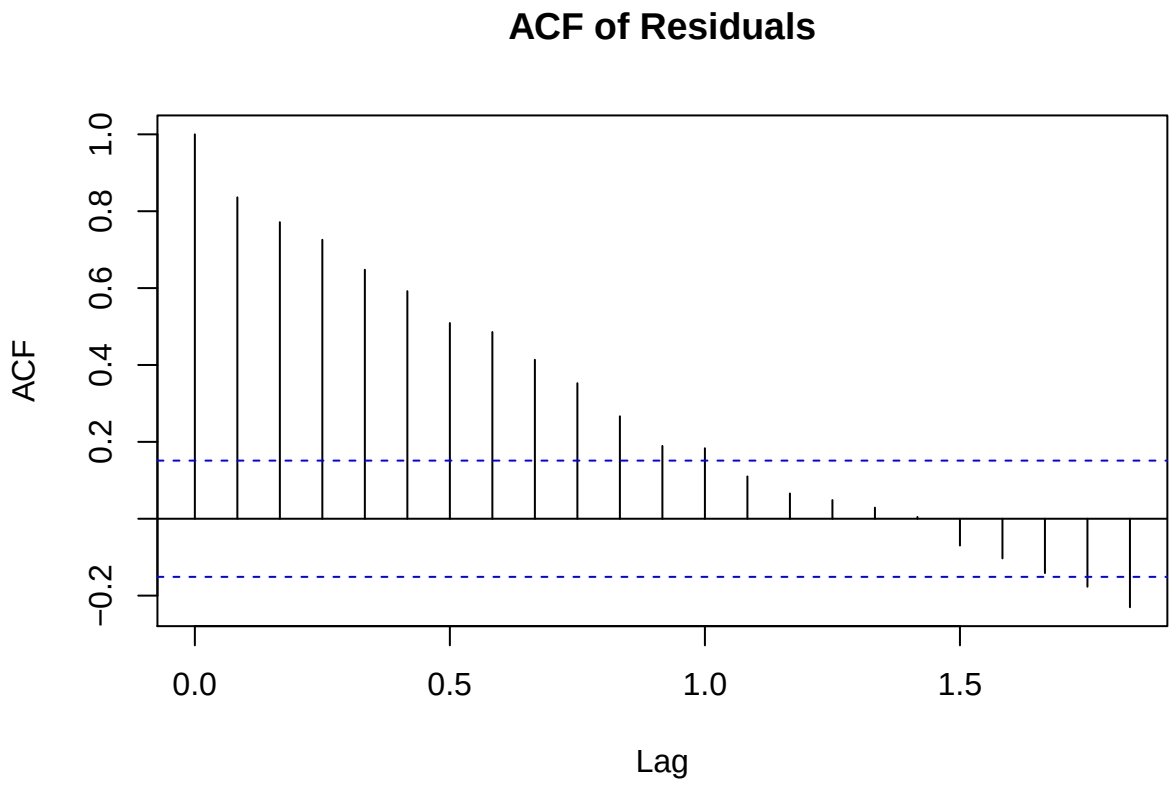
```
#test for normality for full model(trend+seasonality) model
JarqueBera.test(seasontrend$residuals)
```

```
##
##  Jarque Bera Test
##
## data:  seasontrend$residuals
## X-squared = 13.134, df = 2, p-value = 0.001406
##
##
##  Skewness
##
## data:  seasontrend$residuals
## statistic = 0.68433, p-value = 0.0002933
##
##
##  Kurtosis
##
## data:  seasontrend$residuals
## statistic = 2.9447, p-value = 0.8837
```

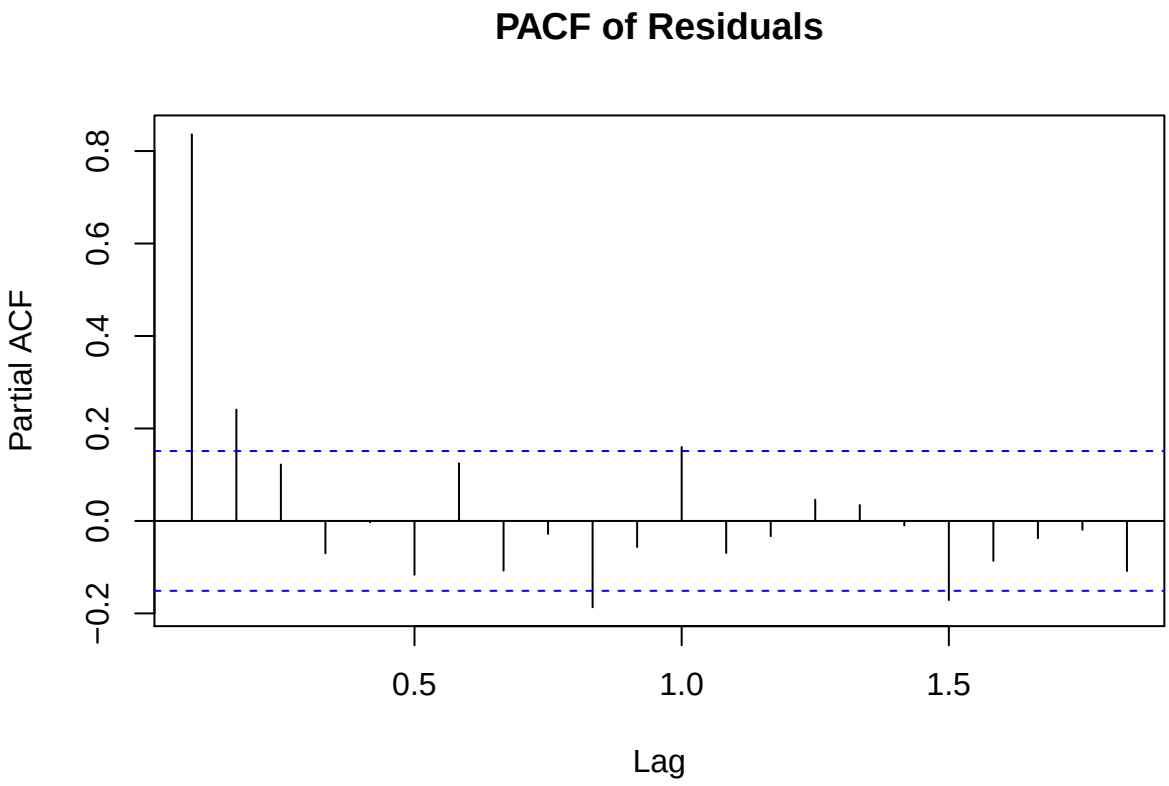
2F) ACF and PACF plots

The ACF and PACF plots are more tame compared to the plots in 1h. Every value is in the bounds suggesting they are statistically indistinguishable from 0. This suggests that our data has less auto correlation and that there is little information left in the residuals that can be used for computing forecasts.

```
#ACF and PACF of full model residuals
acf(seasontrend$residuals,main='ACF of Residuals')
```




```
pacf(seasontrend$residuals,main='PACF of Residuals')
```

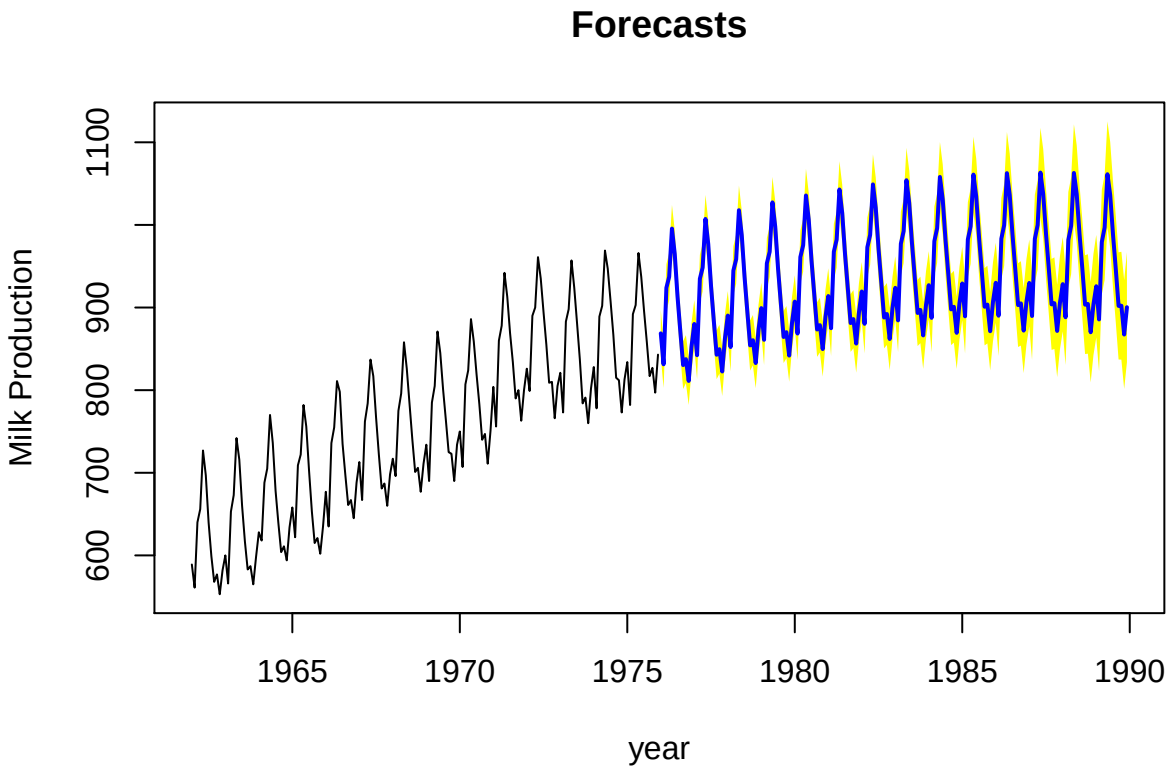


2G) Forecasting trend and seasonality model

With the code below we predicted the full model (trend+season) 20 steps (months) into the future. The blue line on the plot represents our point forecasts while the yellow shaded regions represent our 95% prediction interval.

```
#creating new time variable
t2=seq(1976,1990,length=length(datats))

#Forecasting Full model (trend+Seasonality)
pred2=forecast(seasontrend,newdata=t2, h = 20, level=.95)
plot(pred2,main="Forecasts",xlab="year", ylab="Milk Production",shadecols="oldstyle")
```



III) Conclusions and Future Works

Our final model included a dummy variable for each month, a periodic term, and a quadratic term over time. With these simple features, we are able to have a fitted model that matches the data almost exactly. We've considered some possible work we can add to this:

It would be interesting to get data after 1975 and use our model to make predictions on it. We can also use this to make future predictions in 2019. Additionally, our model doesn't fit the data perfectly. There may be some other predictors we can use when we fit our model.

IV) References

https://datamarket.com/data/set/22ox/monthly-milk-production-pounds-per-cow-jan-62-dec-75#!ds=22ox&display=line

Agriculture, Source: Cryer (1986), in file: data/milk, Description: Monthly milk production: pounds per cow. Jan 62 - Dec 75

V) R Source Code

R Code i