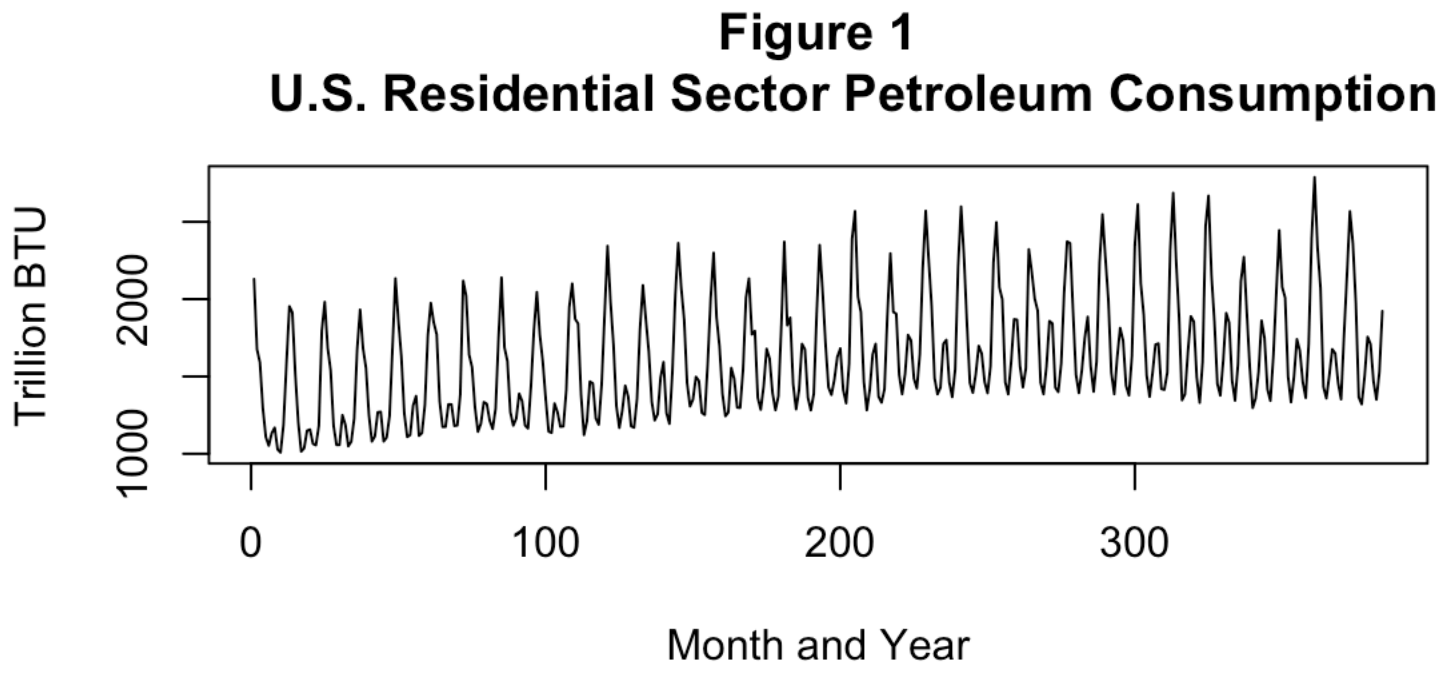
Appendix: Diagnostic Plots and R Code

Audrey Chu

Original Data

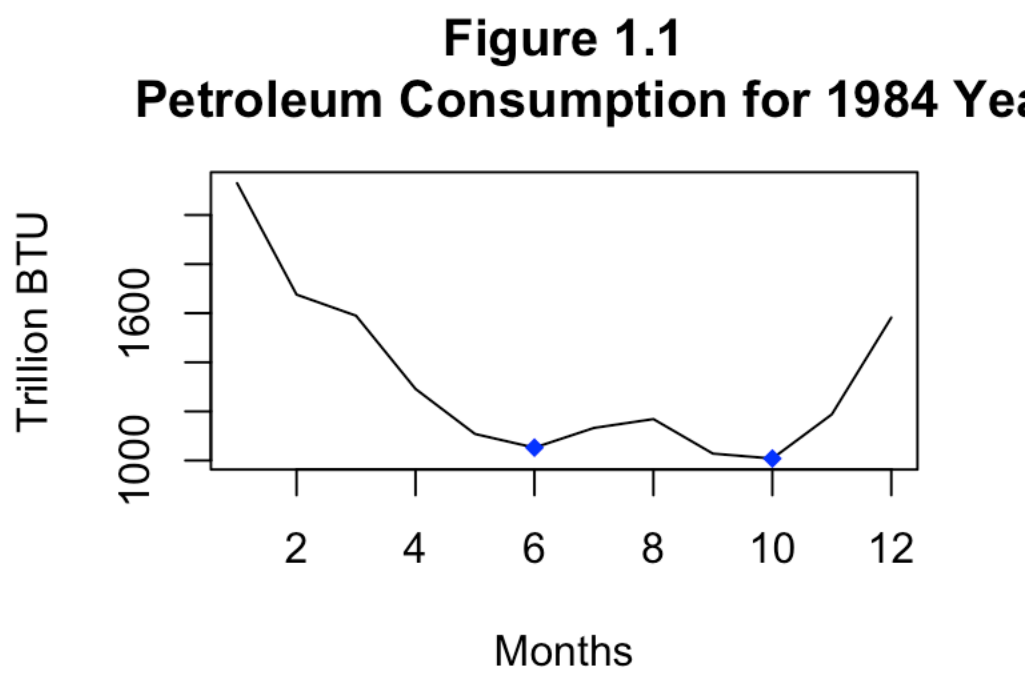
```
time = 1:384
y = read.table("~/Documents/3rd Year/STA137/EnergyConsumption.txt", row.names=NULL)[,
3]
plot(time, y, type='l', main = "Figure 1
     U.S. Residential Sector Petroleum Consumption", ylab="Trillion BTU", xlab="Month
and Year")
```




```
model = lm(y~time)
summary(model)
```

```
##
## Call:
## lm(formula = y ~ time)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -535.3 -278.2 -127.4  241.4  953.8
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1360.6754    35.4581  38.374  < 2e-16 ***
## time           1.3132     0.1596   8.227 3.05e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 346.7 on 382 degrees of freedom
## Multiple R-squared:  0.1505, Adjusted R-squared:  0.1483
## F-statistic: 67.68 on 1 and 382 DF,  p-value: 3.052e-15
```

```
plot(1:12, y[1:12], type='l', main="Figure 1.1
     Petroleum Consumption for 1984 Year", ylab="Trillion BTU", xlab="Months")
points(6, y[6], col='blue', pch=18)
points(10, y[10], col='blue', pch=18)
```



Box-Cox Transformation

Figure 2.0

```
par(mfrow=c(2,2))
plot.ts(y^(-1),ylab=expression(paste("1/",Y[t],")")), main=expression(paste("1/(",Y[t],")")))
plot.ts(y^.5,ylab=expression(paste("sqrt(",Y[t],")")), main=expression(paste("Plot of Sqrt(",Y[t],")")))
plot.ts(y^-.5,ylab=expression(paste("1/sqrt(",Y[t],")")), main=expression(paste("Plot of 1/sqrt(",Y[t],")")))
plot.ts(log(y),ylab=expression(paste("ln(",Y[t],")")), main=expression(paste("Plot of ln(",Y[t],")")))
```

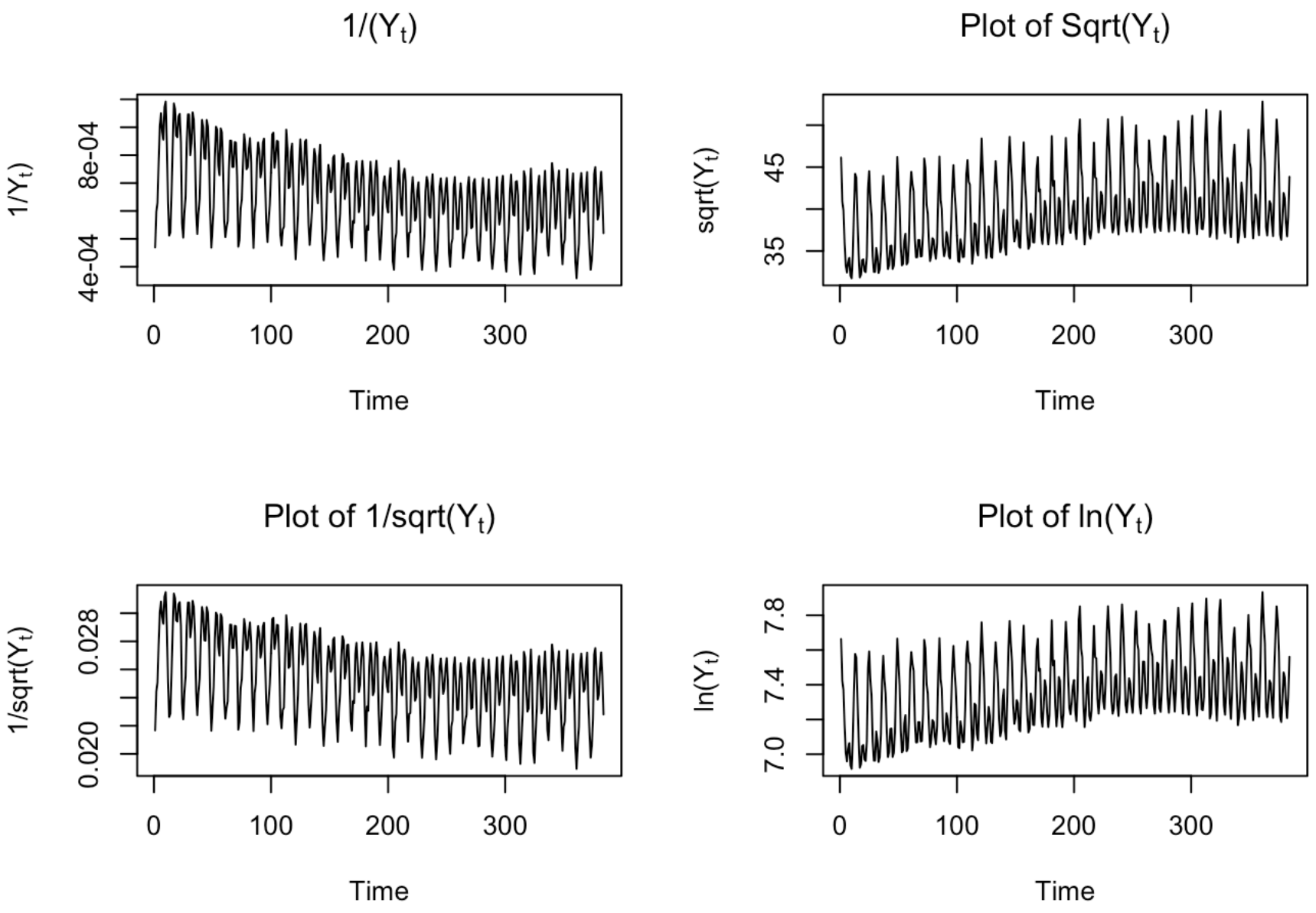
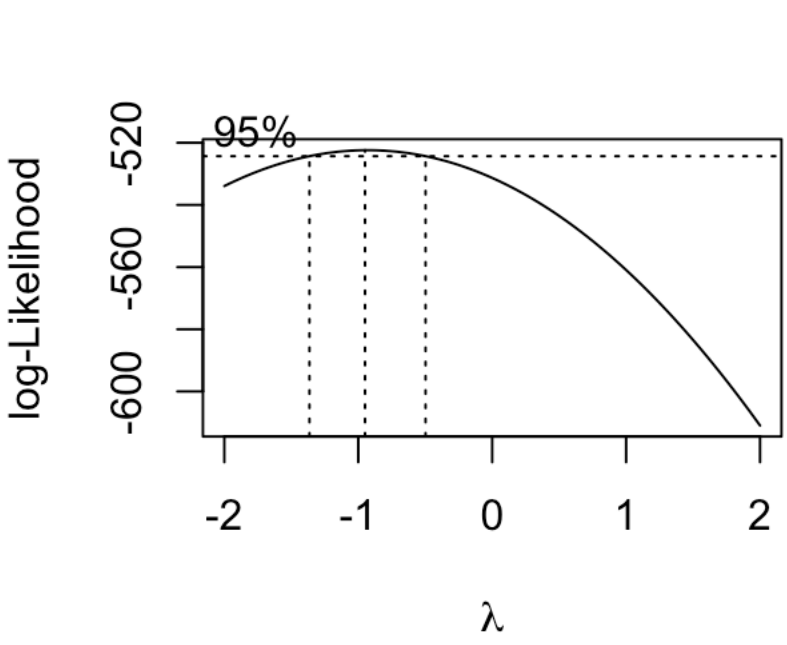


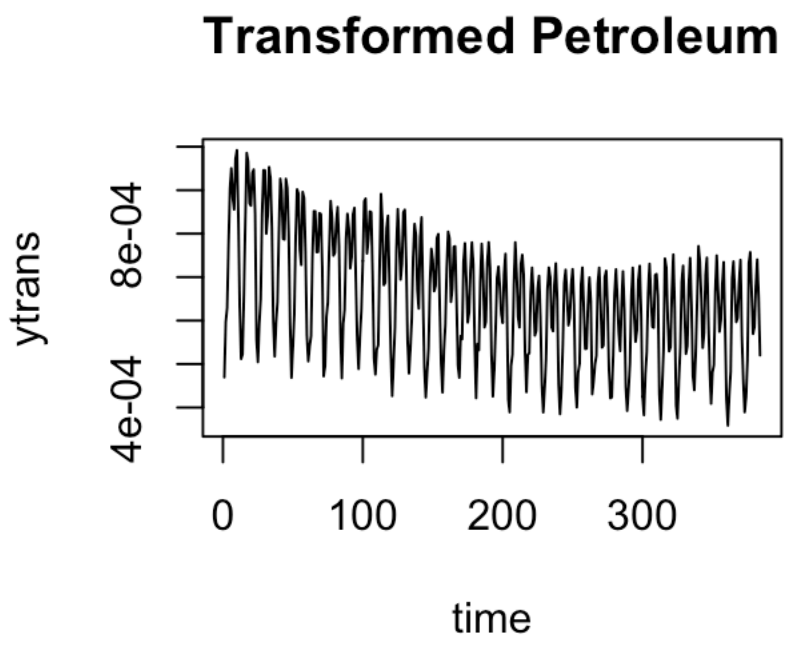
Figure 2.1

```
par(mfrow=c(1,2))
boxcox(model)
powerTransform(model)
```

```
## Estimated transformation parameters
##         Y1
## -0.9299752
```

```
ytrans = y^(-1)
modtrans = lm(ytrans~time)
plot(time, ytrans, type = 'l', main = "Transformed Petroleum")
```





Trend and Seasonal Estimation

Figure 3.0

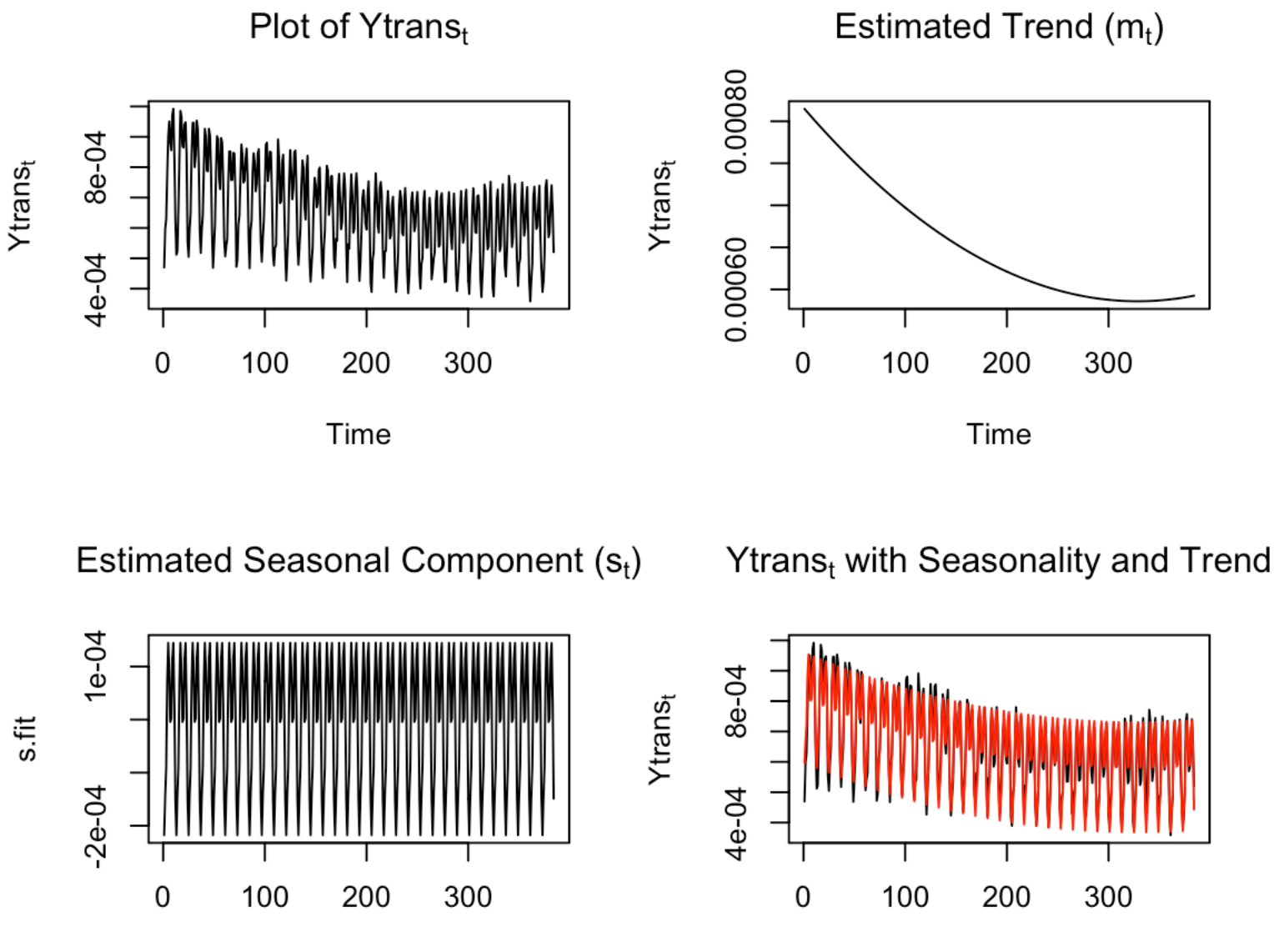
```
mod = trndseas(ytrans,degtrnd=2, seas=12)
mod$rsq
```

```
## [1] 0.9285714
```

```
mod$lamopt
```

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

```
m.fit = mod$trend
s.fit = rep(mod$season,length.out=384)
par(mfrow=c(2,2))
plot.ts(ytrans ,ylab=expression(paste("",Ytrans[t],"")), main=expression(paste("Plot
of ",Ytrans[t],"")))
plot.ts(m.fit, ylab=expression(paste("",Ytrans[t],"")), main=expression(paste("Estima
ted Trend (",m[t],")")))
plot.ts(s.fit,main=expression(paste("Estimated Seasonal Component (",s[t],")")))
plot.ts(ytrans,ylab=expression(paste("",Ytrans[t],"")), main=expression(paste("",Ytra
ns[t]," with Seasonality and Trend")))
points(mod$fit,type='l',col='red')
```



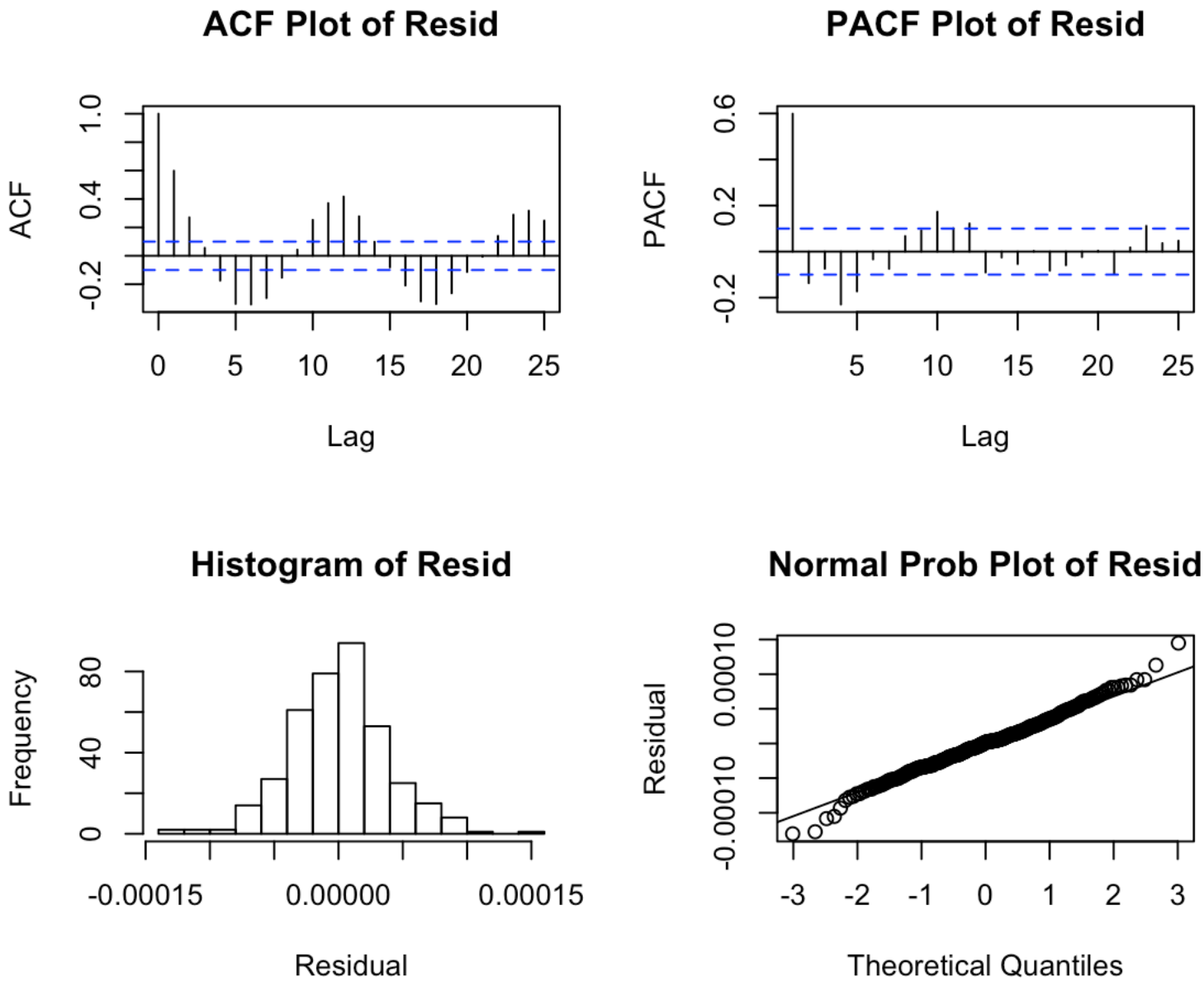

```
mod$coef
```

```
##                [,1]
##       8.161081e-04
## 1    -5.381070e-04
## 2     3.145122e-04
## x21  -2.177859e-04
## x22  -1.440324e-04
## x23  -8.967068e-05
## x24   7.204337e-05
## x25   1.446329e-04
## x26   9.314625e-05
## x27  -4.488970e-06
## x28   2.180358e-06
## x29   1.147087e-04
## x210  1.442814e-04
## x211  3.402836e-05
```

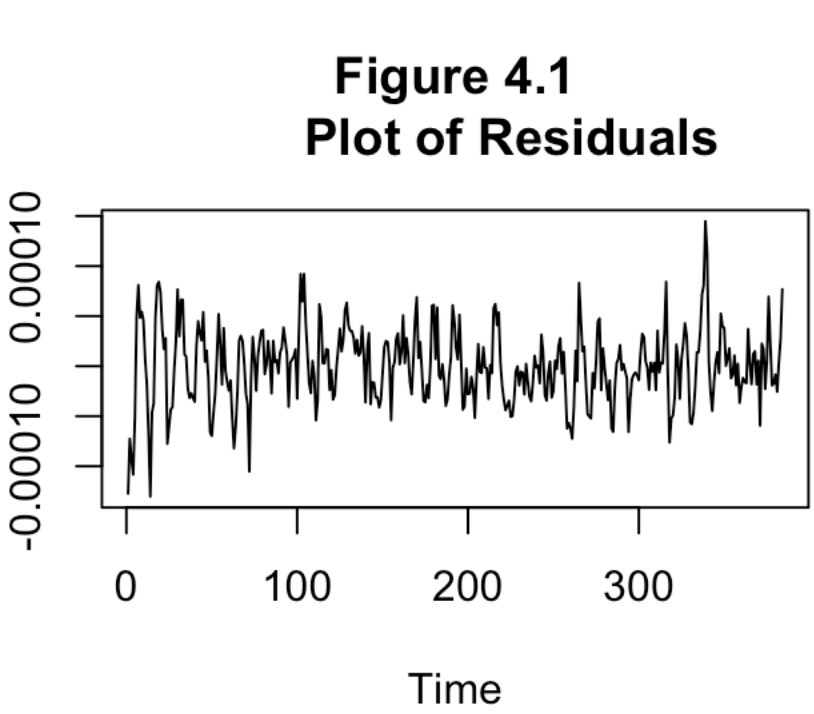
Rough Estimation

Figure 4.0

```
x = ytrans - m.fit - s.fit
par(mfrow=c(2,2))
acf(x, main= "ACF Plot of Resid", ylab = "ACF", xlab ="Lag")
pacf(x, main= "PACF Plot of Resid", ylab = "PACF", xlab = "Lag")
hist(x, main= "Histogram of Resid", ylab = "Frequency", xlab = "Residual")
qqnorm(x, main= "Normal Prob Plot of Resid",ylab="Residual"); qqline(x)
```




```
par(mfrow=c(1,1))
plot.ts(x, main = "Figure 4.1
        Plot of Residuals", ylab = "")
```

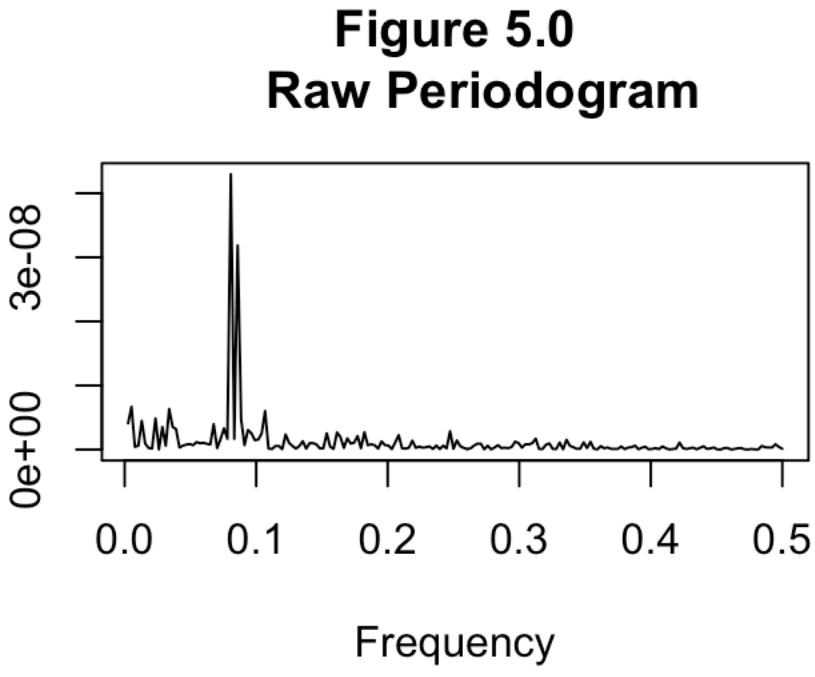



```
Box.test(x,lag=10,'Ljung-Box')
```

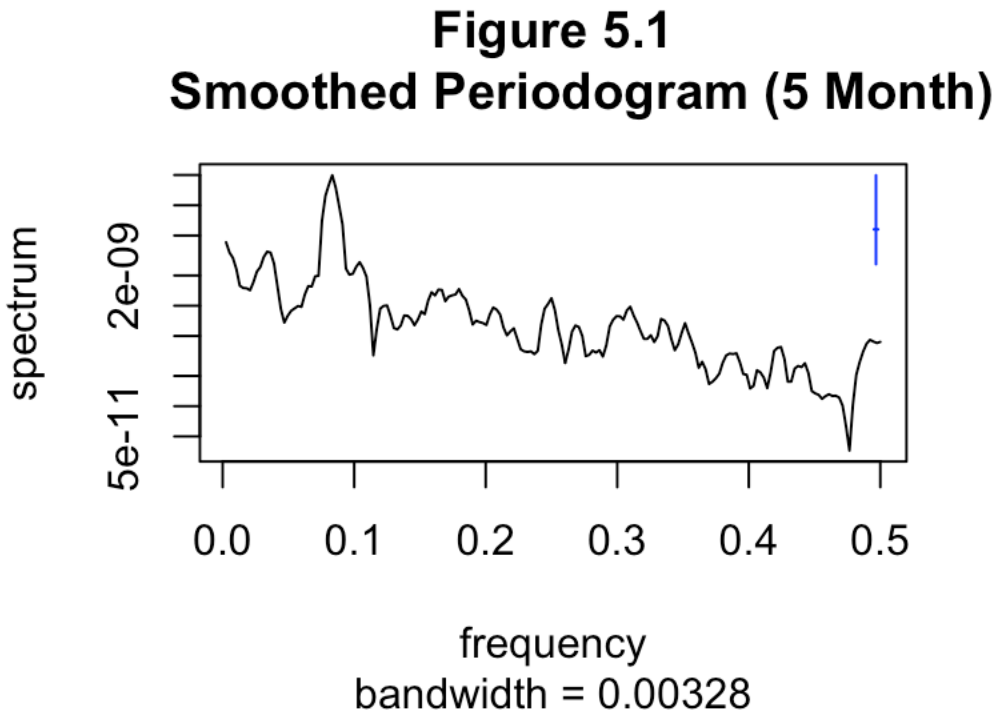
```
##
##  Box-Ljung test
##
## data:  x
## X-squared = 341.056, df = 10, p-value < 2.2e-16
```

Box-Ljung test shows a very low p-value, which can be intepreted as significant. This means that the rough series is not independent.

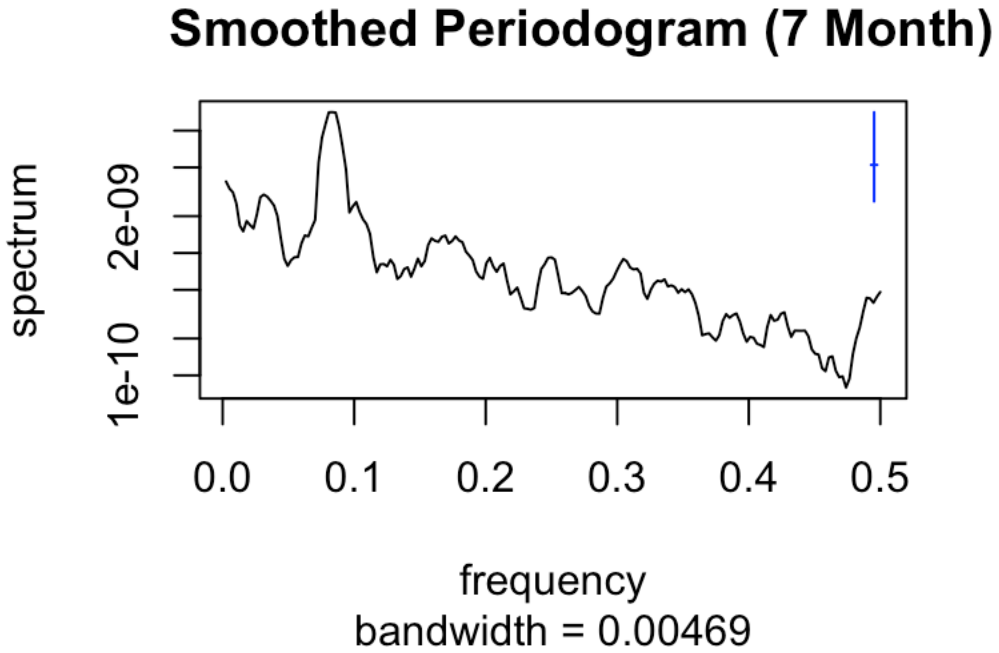
```
xpgrm = spec.pgram(x,log='no',plot=F)
plot(xpgrm$freq,xpgrm$spec,type='l',xlab='Frequency',ylab='', main = 'Figure 5.0
    Raw Periodogram')
```



```
xpgrm5 = spec.pgram(x,spans=5, main = "Figure 5.1
    Smoothed Periodogram (5 Month)")
```

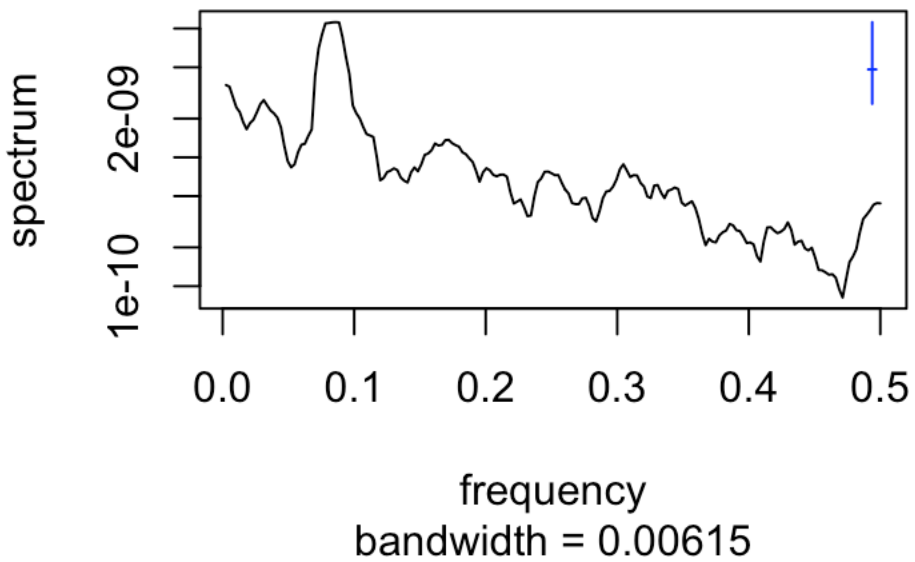



```
xpgrm7 = spec.pgram(x,spans=7, main="Figure 5.2
    Smoothed Periodogram (7 Month)")
```



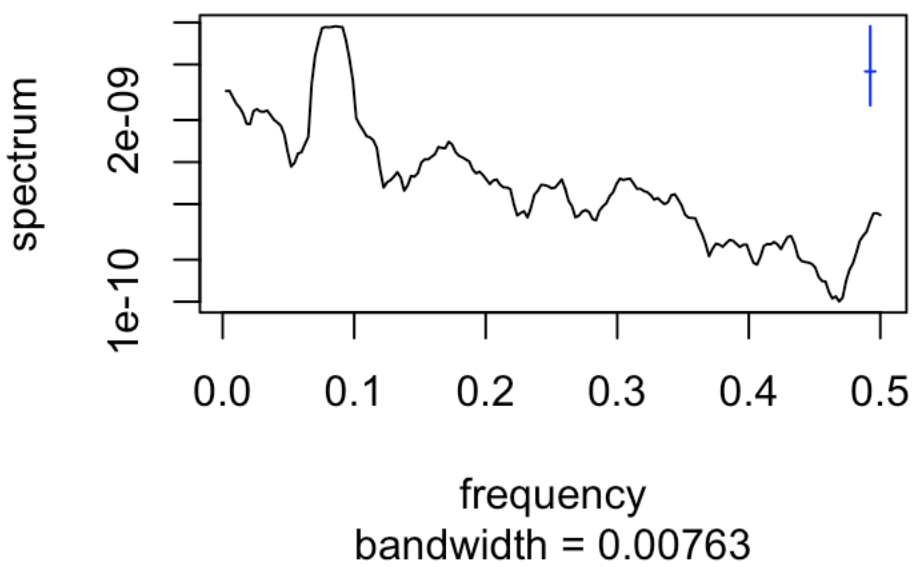

```
xpgrm9 = spec.pgram(x,spans=9, main="Figure 5.3
    Smoothed Periodogram (11 Month)")
```

Figure 5.3
Smoothed Periodogram (11 Month)




```
xpgrm11 = spec.pgram(x,spans=11, main="Figure 5.4
    Smoothed Periodogram (11 Month)")
```

Figure 5.4
Smoothed Periodogram (11 Month)



Preliminary ARMA(p, q) or ARIMA(p, d, q) Model

```
fitARMA0 = arima(x, order=c(0,0,0))
fitARMA14 = arima(x, order=c(1,0,4))
fitARMA24 = arima(x, order=c(2,0,4))
fitARMA34 = arima(x, order=c(3,0,4))
fitARMA44 = arima(x, order=c(4,0,4))
fitARMA54 = arima(x, order=c(5,0,4))
fitARMA45 = arima(x, order=c(4,0,5))
aicc(fitARMA0)
```

```
## [1] -6727.784
```

```
aicc(fitARMA14)
```

```
## [1] -6914.465
```

```
aicc(fitARMA24)
```

```
## [1] -6916.829
```

```
aicc(fitARMA34)
```

```
## [1] -6975.005
```

```
aicc(fitARMA44)
```

```
## [1] -6972.908
```

```
aicc(fitARMA54)
```

```
## [1] -6975.393
```

```
aicc(fitARMA45)
```

```
## [1] -6967.331
```

```
auto = auto.arima(x, max.p = 8, max.q =8, max.d = 2); auto
```

```
## Warning in auto.arima(x, max.p = 8, max.q = 8, max.d = 2): Unable to fit
## final model using maximum likelihood. AIC value approximated
```

```
## Series: x
## ARIMA(4,0,5) with zero mean
##
## Coefficients:
##          ar1      ar2     ar3      ar4      ma1     ma2      ma3     ma4
##       1.8990  -1.9961  1.3589  -0.6564  -1.3891  1.1768  -0.5111  0.0139
## s.e.  0.1223   0.2237  0.1910   0.0862   0.1211  0.1852   0.1442  0.0831
##          ma5
##       0.3097
## s.e.  0.0579
##
## sigma^2 estimated as 6.914e-10:  log likelihood=3504.84
## AIC=-6989.19   AICc=-6988.6   BIC=-6949.69
```

```
fitARIMA405 = arima(x, order=c(4, 0, 5))
aicc(fitARIMA405)
```

```
## [1] -6967.331
```

```
par(mfrow=c(1,1))
res = fitARIMA405$res
ts.plot(res, main = "Figure 6.0
  Plot of Residual of ARIMA(4,0,5)")
h=12
n = 372
fcast = predict(fitARIMA405,n.ahead=h)
fc = fcast$pred
upper = fc+qnorm(0.975)*fcast$se
lower = fc-qnorm(0.975)*fcast$se
polygon(x=c(n+1:h,n+h:1),y=c(upper,rev(lower)),col='lightblue',border=NA)
lines(x=n+(1:h),y=fc,col='blue')
```

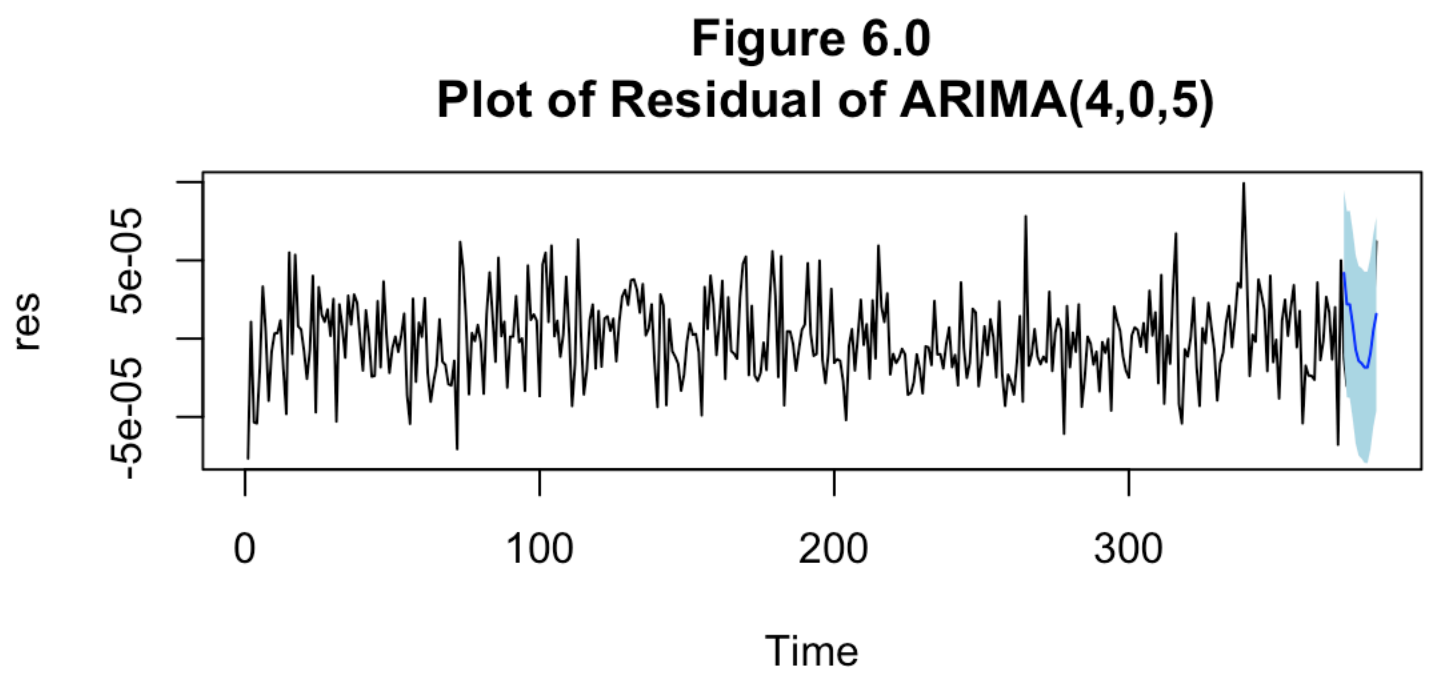


Figure 7.0

```
par(mfrow=c(1,2))
acf(res, main = "ACF of ARIMA(4,0,5) Residual")
pacf(res, main = "PACF of ARIMA(4,0,5) Residual")
```

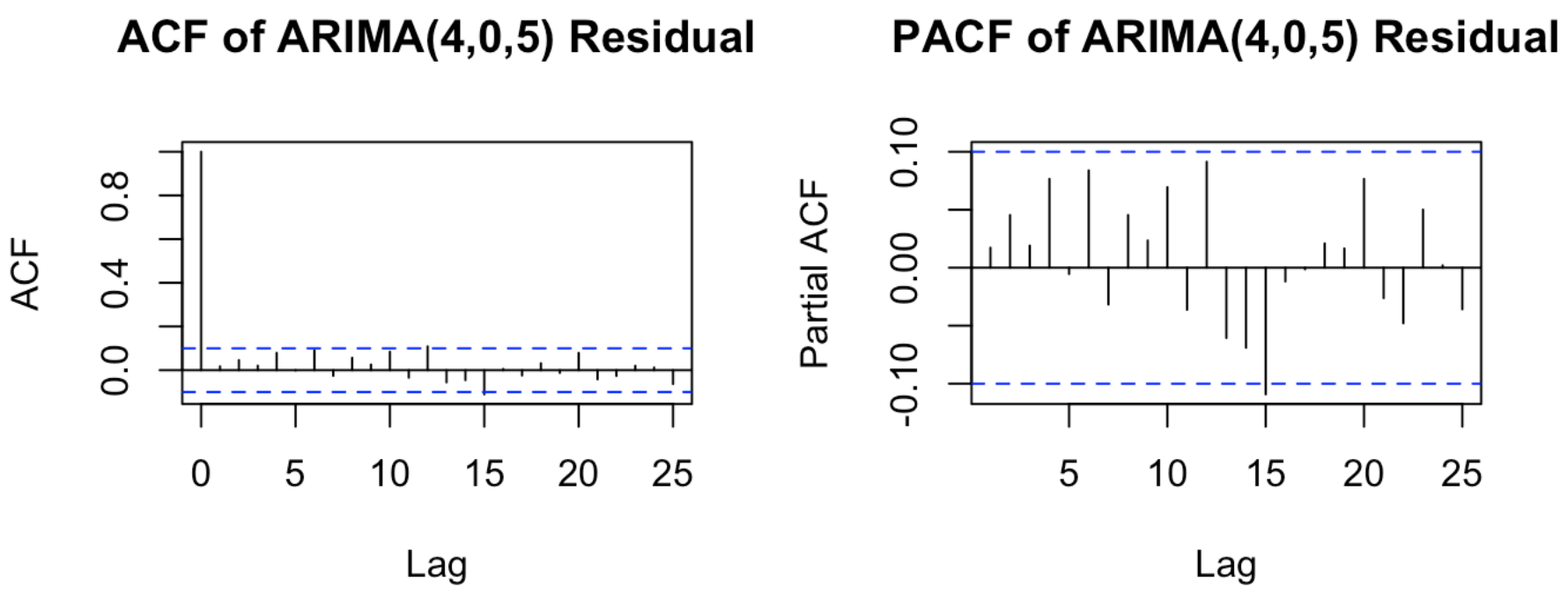
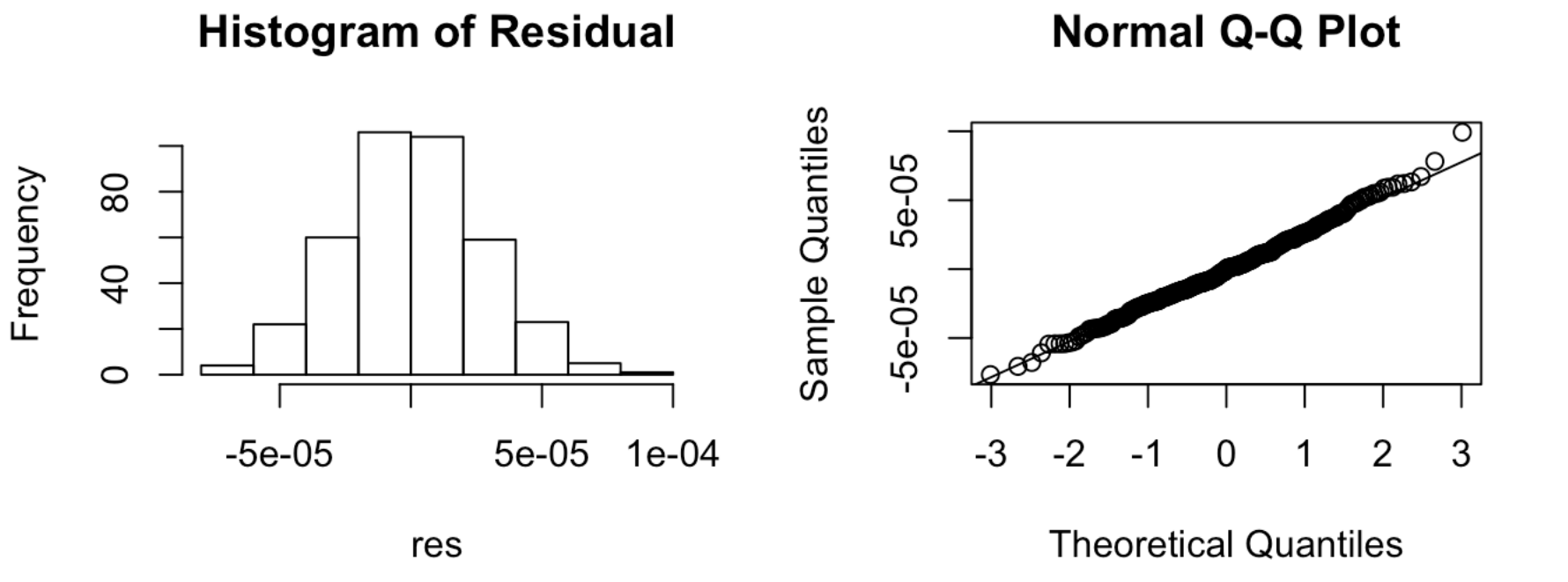
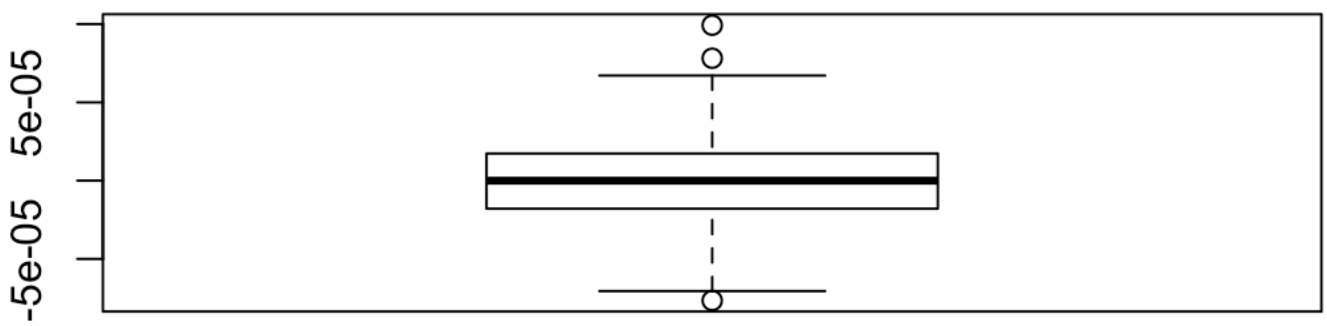


Figure 7.1

```
par(mfrow=c(1,2))
hist(res, main = "Histogram of Residual")
qqnorm(res); qqline(res)
```




```
par(mfrow=c(1,1))
boxplot(res, main = "Figure 7.2
        Boxplot of Residuals")
```



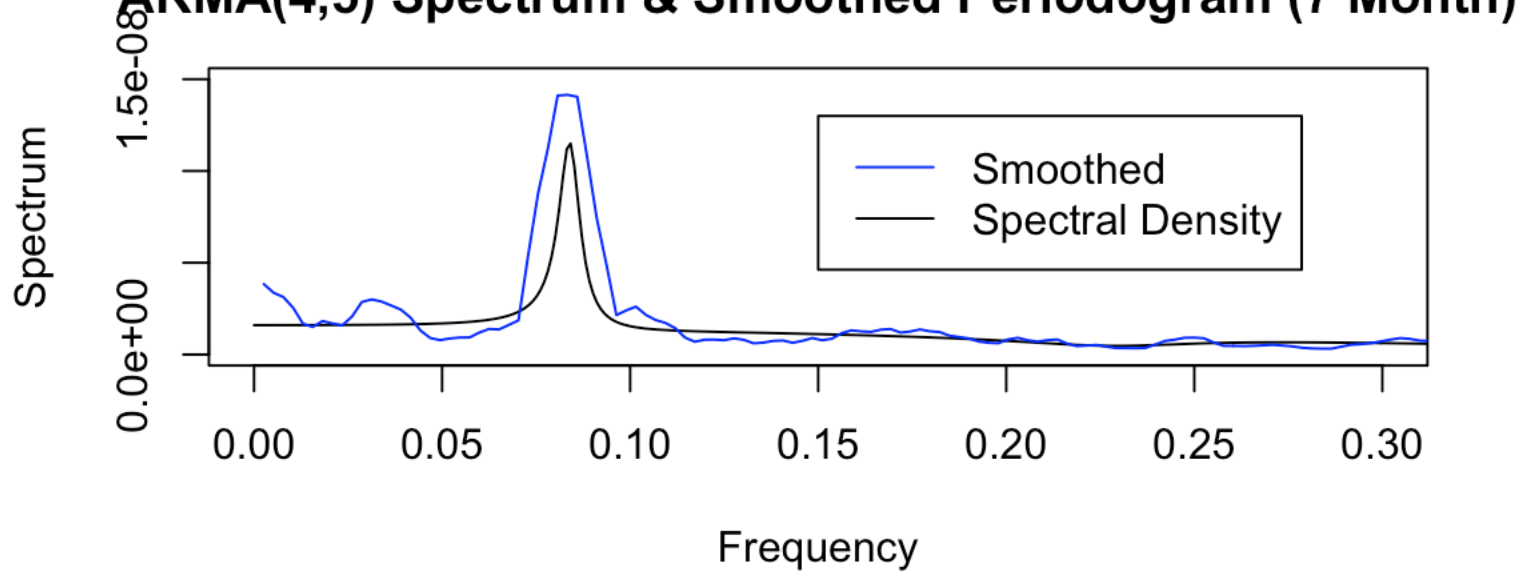

```
Box.test(res,lag=10,'Ljung-Box')
```

```
##
##  Box-Ljung test
##
## data:  res
## X-squared = 11.3325, df = 10, p-value = 0.3322
```

Spectral Density

```
coef.ar = auto$coef[1:4]
coef.ma = auto$coef[5:9]
sigma2 = auto$sigma2
```

```
mod1spec = arma.spec(ar=coef.ar, ma=coef.ma, var.noise=sigma2, log='no', main="Figure
8.0
ARMA(4,5) Spectrum & Smoothed Periodogram (7 Month)", xlim=c(0.0,0.3), ylab="Spectrum
", xlab="Frequency", ylim=c(0,15.0e-09))
points(xpgrm7$freq, xpgrm7$spec, type='l', col='blue')
legend(0.15, 13.0e-09, c("Smoothed", "Spectral Density"), lty=c(1,1), col=c('blue', '
black'))
```



Prediction

```
# Prediction of time 372 to 384
y372 = y[1:372]
n = 372
h=12
# Forecast the trend
deg = 2
coef = mod$coef[1:(deg+1)]
time = (n+(1:h))/n; time
```

```
##  [1] 1.002688 1.005376 1.008065 1.010753 1.013441 1.016129 1.018817
##  [8] 1.021505 1.024194 1.026882 1.029570 1.032258
```

```
predmat = matrix(rep(time,deg)^rep(1:deg,each=h),nrow=h,byrow=F)
predmat = cbind(rep(1,h),predmat); predmat
```

```
##       [,1]     [,2]     [,3]
##  [1,]    1 1.002688 1.005384
##  [2,]    1 1.005376 1.010782
##  [3,]    1 1.008065 1.016194
##  [4,]    1 1.010753 1.021621
##  [5,]    1 1.013441 1.027062
##  [6,]    1 1.016129 1.032518
##  [7,]    1 1.018817 1.037988
##  [8,]    1 1.021505 1.043473
##  [9,]    1 1.024194 1.048972
## [10,]    1 1.026882 1.054486
## [11,]    1 1.029570 1.060014
## [12,]    1 1.032258 1.065557
```

```
m.fc = predmat %*% coef
# Forecast the seasonality
s.fc = rep(mod$season,length.out=n+h)
s.fc = s.fc[-(1:n)]
# Forecast the rough part
fcast = predict(fitARIMA405,n.ahead=h)
x.fc = fcast$pred
# Combine forecasts
y.fc = ( m.fc + s.fc + x.fc)^(-1)
y.fc
```

```
## Time Series:
## Start = 385
## End = 396
## Frequency = 1
##           [,1]
##  [1,] 2399.258
##  [2,] 2123.120
##  [3,] 1903.211
##  [4,] 1483.402
##  [5,] 1367.182
##  [6,] 1485.714
##  [7,] 1742.031
##  [8,] 1728.739
##  [9,] 1446.591
## [10,] 1369.519
## [11,] 1575.130
## [12,] 2162.323
```

```
par(mfrow=c(1,1))
oldy=y[1:372]
plot.ts(oldy,xlim=c(0,n+h), main = "Figure 9.0
Petroleum Prediction from December 2014", ylab= "Trillion BTU", pch=1)
points(x=n+1:h, y=y.fc, col='blue',type='l',pch=8)
legend(0, 2800, c("Forecasted 2015"), lty=c(1,1), col=c('blue'))
```

Figure 9.0
Petroleum Prediction from December 2014