Time Series Analysis: Milk Production

Chaewon Shin

PSTAT 174

June 10, 2020

Abstract

A time series is a set of observations taken over time, and time series analysis aims to develop models of that best describes the patterns and behavior of a dataset. In this report, we will be building a time series model on monthly milk production in pounds from January 1962 to December 1975 by using the Box-Jenkins approach. The goal of this project is to forecast the milk production in the year of 1975 based on the data from the first 12 years. The techniques used in this analysis include differencing, ACF and PACF analysis, residual analysis, and forecasting. Using these techniques, we were able to develop a SARIMA model that forecasts the last year of milk production.

Introduction

The purpose of this project report is to analyze and construct a time series model and forecast future outcomes based on the data set, milk. This data set measures monthly milk production in pounds from January 1962 to December 1975 with 156 observations. (Makridakis, Wheelwright and Hyndman (1998) Forecasting: methods and applications, John Wiley & Sons: New York. Chapter 2.) I found this dataset to be interesting due to the recent popularity of various types of plant-based milk, and I was interested to see how milk production trends behaved before the vast introduction of other milk types. In order to build a time series model to forecast future outcomes, I will be using the Box-Jenkins approach to model building, which includes:

1. Analyzing the time series
2. Transformations and Differencing
3. ACF and PACF Analysis
4. Model fitting
5. Diagnostic Checking
6. Forecasting

As a result, I was able to forecast a SARIMA model that mirrors the pattern and behavior of the original dataset, milk. The software used to conduct this project is RStudio, Version 1.1.442.

Data Preparation

The data set is made of 156 observations over the course of 13 years. Since no new data is expected, the time series analysis will be conducted using the first 144 observations, denoted as U_t. The remaining 12 data points will be used to test the time series model in forecasting.

Analyzing the Time Series

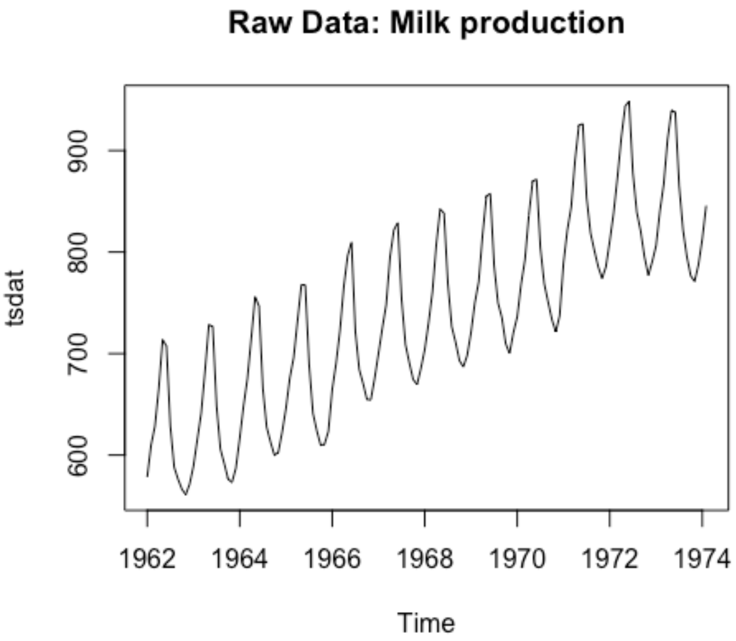


Figure 1. The plot of monthly milk production from Jan. 1962 to Dec. 1975

The time series plot for raw monthly milk production data is shown in figure 1. Immediately, we can observe that there is an upward trend and periodic behavior. Note that at 1971, there is a small spike in variance, whereas the data points from 1962 to 1971 exhibits stable variance.

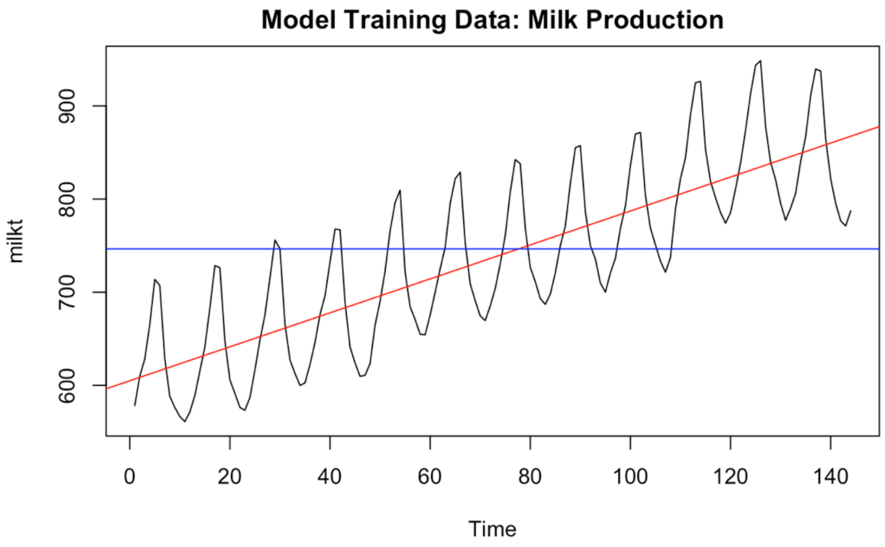
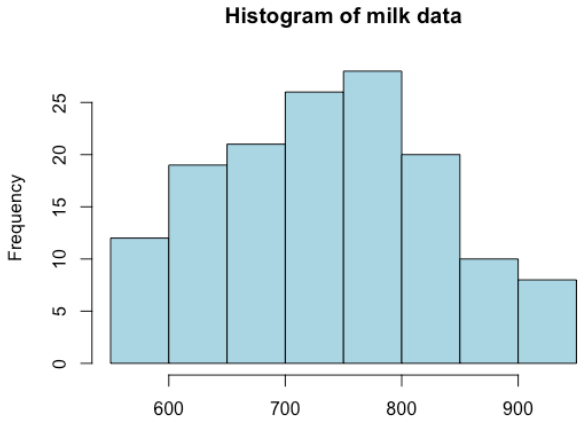


Figure 2. The plot of model training data Ut with mean and trend

The model training data for milk production is shown in figure 2, along with the mean (in blue) and the trend (in red). The trend and periodic fluctuations indicate that the data is nonstationary. In figure 3 (below), the histogram of the milk data remains relatively normal, suggesting that the variance is stable. However, we can confirm the nonstationary of the data as the ACFs remain large and periodic in figure 4.



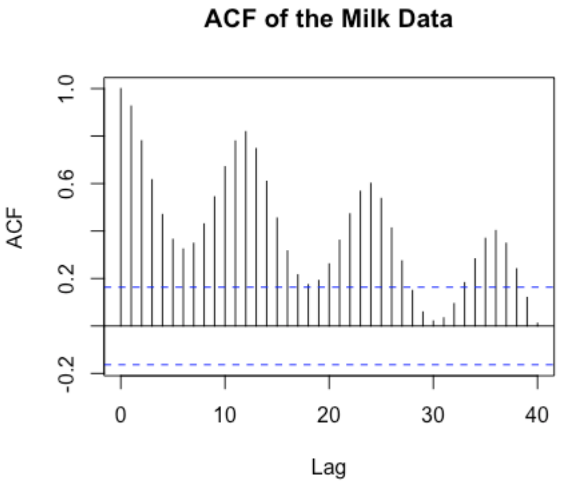


Figure 3. The histogram of milk data, Ut

Figure 4. The ACF plot of milk data, Ut

Transformation and Differencing

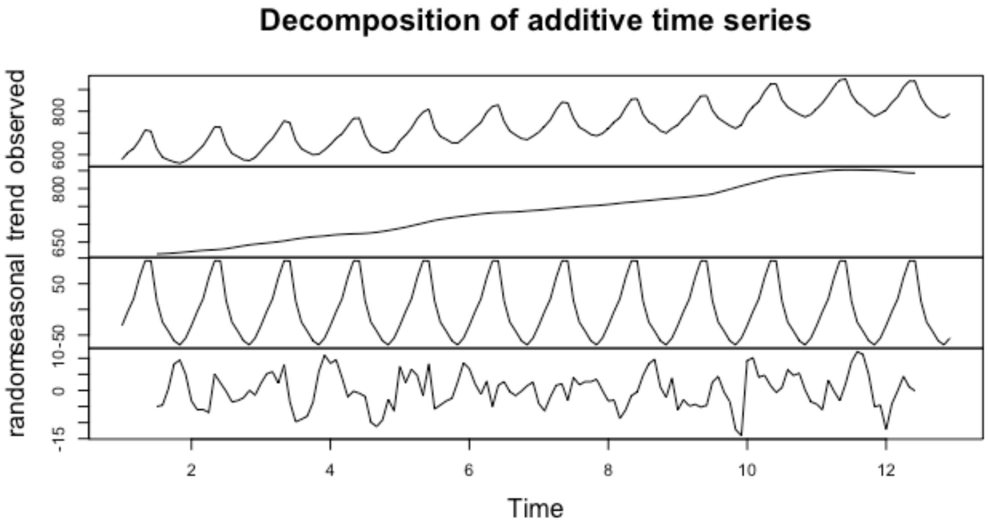


Figure 5. The decomposition of time series Ut

The decomposition of Ut shows the seasonal, trend, and irregular components of the time series. In figure 5, we can see in the random component that the variance remains similar throughout the plot, indicating that we do not need to apply transformation technique to stabilize variance. However, in the trend and seasonal components, the data remains nonstationary. In order to remove trend and seasonality, we will apply differencing techniques.

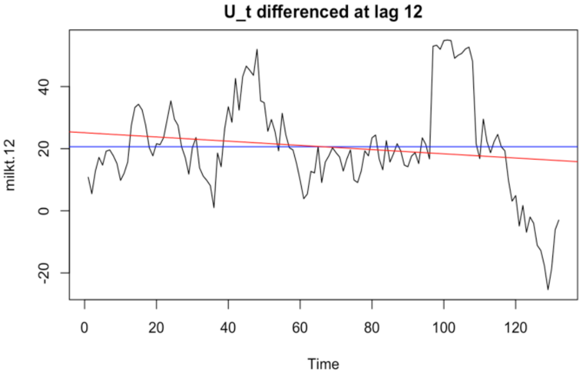


Figure 6. The plot of Ut differenced at lag 12

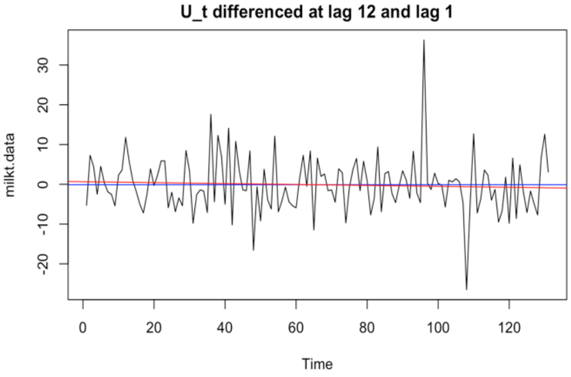


Figure 7. The plot of Ut differenced at lag 12 and lag 1

Note that the variance of Ut before differencing is 9106.941. To remove seasonality, we applied differencing at lag 12. In figure 6, we can see that the seasonality is no longer apparent. Despite the variance significantly decreasing to 254.5098, there is still a slight downward trend. To remove this trend, we applied differencing again, but at lag 1. In figure 7, we can see that both the trend and seasonality is no longer apparent. The variance decreased again to 51.612, which indicates that there was no over differencing applied.

Once the seasonality and trend have been removed, we can see that the data appears to be stationary. To confirm, we will check the ACF's of the differenced data.

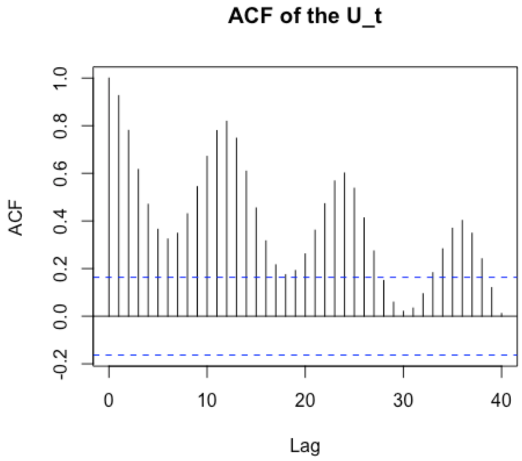


Figure 8. The ACF plot of Ut, training data

Before differencing, we can see that the ACF is periodic, indicating seasonality. The slow decay also indicates non-stationarity.

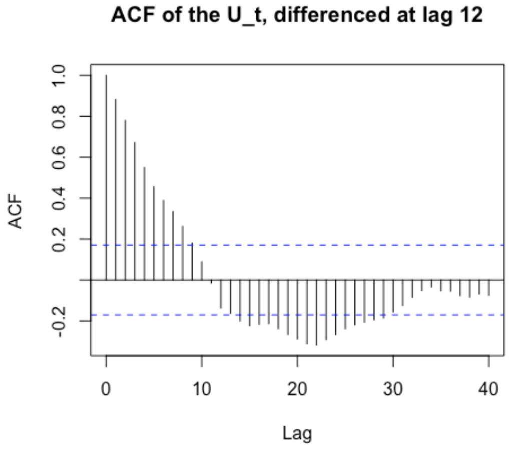


Figure 9. The ACF plot of Ut, differenced at lag 12

After differencing at lag 12, the seasonality is no longer apparent. The ACF remains large with slow decay, indicating that non-stationarity is still present.

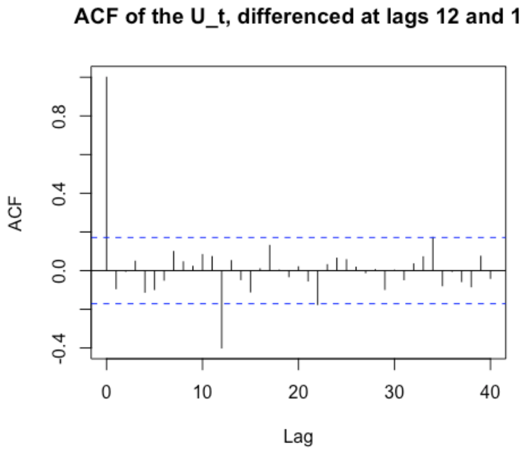
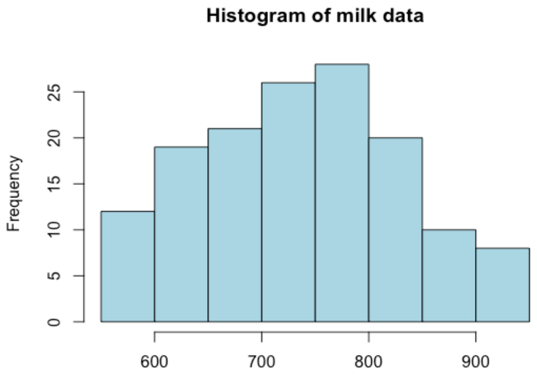


Figure 10. The ACF plot of Ut, differenced at lags 12 and 1

After differencing at lag 12 and lag 1, the ACF is no longer large and the decay corresponds to a stationary process.



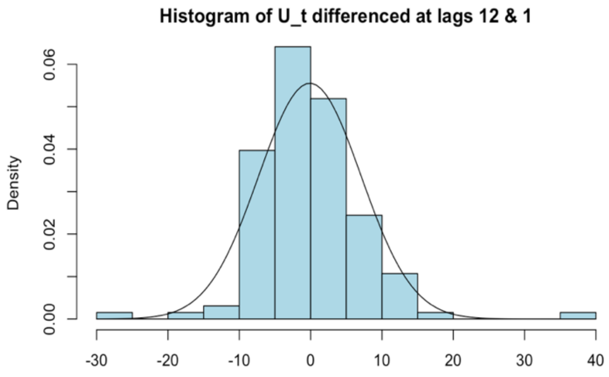


Figure 3. The histogram of milk data, Ut

Figure 11. Histogram of Ut, differenced at lags 12 and 1

Comparing the histograms, we can see that differencing the data at lags 12 and 1 in figure 11 creates a relatively symmetrical and normal plot, as indicated by the normal curve. Compared to figure 3, Ut differenced at lag 12 and 1 offers a more normal distribution, which confirms that the data is stationary.

ACF and PACF Analysis

Given that the data was differenced at lag 12, we will look at SARIMA models. In order to preliminary identify the SARIMA model of the data, we will construct and analyze the ACF and PACF of the differenced data.

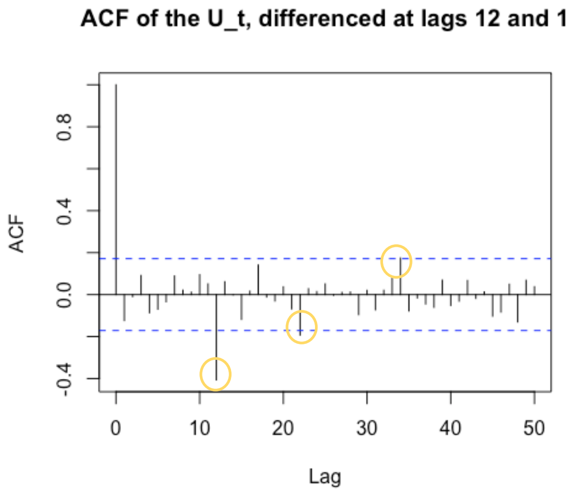


Figure 12. ACF plot of Ut, differenced at lags 12 and 1

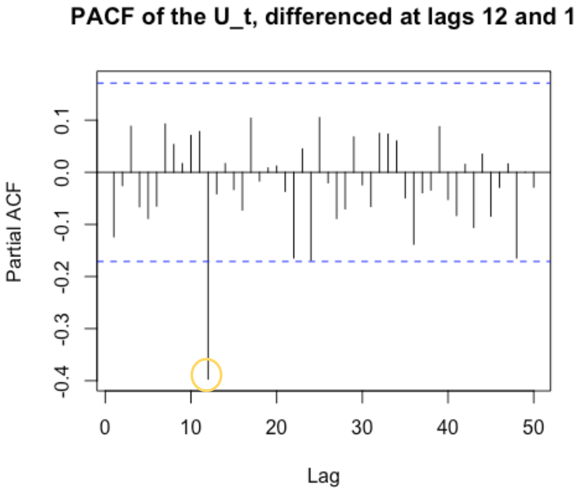


Figure 13. PACF plot of Ut, differenced at lags 12 and 1

Observing the ACF, we can see that the first ACF that lies outside of the confidence interval is at around lag 12, then again at around lag 23, and again at lag 34. We can approximate this by lag 1s, 2s, and 3s where s = 12.

Observing the PACF, we can see that the first PACF that lies outside of the confidence interval is at around lag 12. Afterward, there are no instances of the PACF outside of the confidence interval.

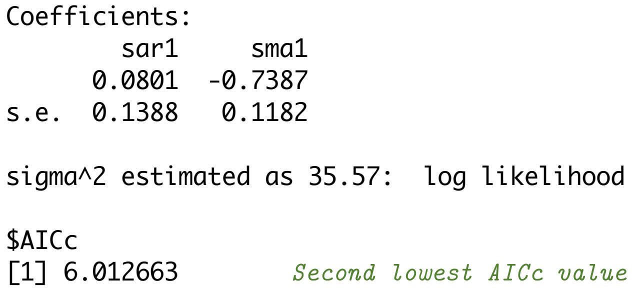
Based on these observations, a list of suitable values for a SARIMA model are:

- D = 1 (differencing at lag 12), d = 1 (differencing at lag 1), s = 12
- Q = 1, 3 (For MA part, strong ACF peak at lag 1s, decay afterward)
- P = 0, 1, 4 (For AR part, strong PACF peak at lag 1s, smaller peak afterward)
- q = 0 (the ACF appears to decay, tail off after lag 1s)
- p = 0 (the PACF cuts off after lag 1s)

Fitting the Model

Based on the possible values, we fit possible combinations of the SARIMA model and choose the models with the lowest Akaike's second-order corrected information criterion (AICc) score. After testing various combinations of possible SARIMA models, the two models with the lowest AICc are:

Model A: SARIMA (0,1,0)x(1,1,1)s=12 , $(1 - 0.0801B^{12})Y_t = (1 - 0.7387B^{12})Z_t$



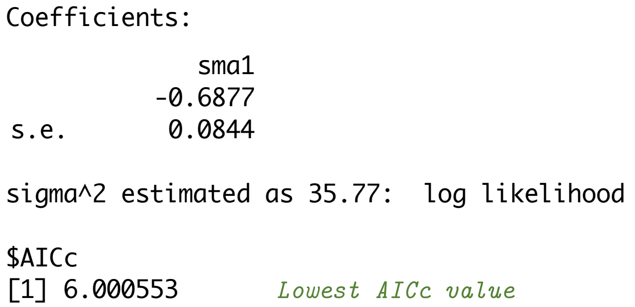
```
Coefficients:
        sar1     sma1
      0.0801  -0.7387
s.e.  0.1388   0.1182

sigma^2 estimated as 35.57:  log likelihood = -423.86,  aic = 853.71

$AICc
[1] 6.012663        Second lowest AICc value
```

Figure 14: Output of coefficients, sigma, and AICc values for model A

Model B: SARIMA (0,1,0)x(0,1,1)s=12, $Y_t = (1 - 0.6877B^{12})Z_t$



```
Coefficients:

             sma1
          -0.6877
s.e.       0.0844

sigma^2 estimated as 35.77:  log likelihood = -424.03,  aic = 852.05

$AICc
[1] 6.000553       Lowest AICc value
```

Figure 15: Output of coefficients, sigma, and AICc values for model B

Before proceeding with diagnostic checking, we must check that the models are both stationary and invertible. For model A, $|\theta_1| < 1$, holds true for MA part, so it is invertible. For AR part of model A, $|\Phi_1| < 1$ holds true, so it is stationary. For model B, since it is a MA process, it is stationary. To check invertibility, we see that $|\theta_1| < 1, |\Theta_1| < 1$ holds true. Thus, model B is invertible and stationary.

Diagnostic Checking

To determine which model to use for forecasting, we will perform diagnostic checking using residual analysis.

For Model A:

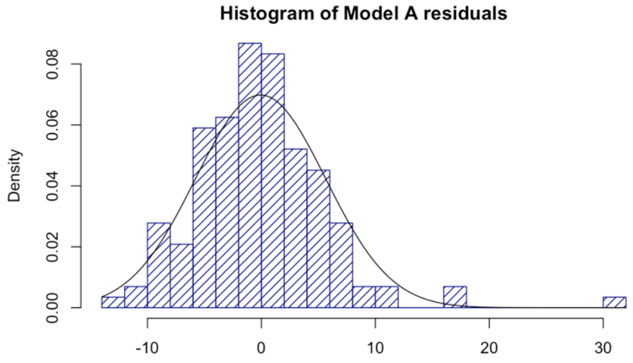


Figure 16. Histogram of residuals for model A

The histogram for model A appears to be normal with mean zero, despite the outlier.

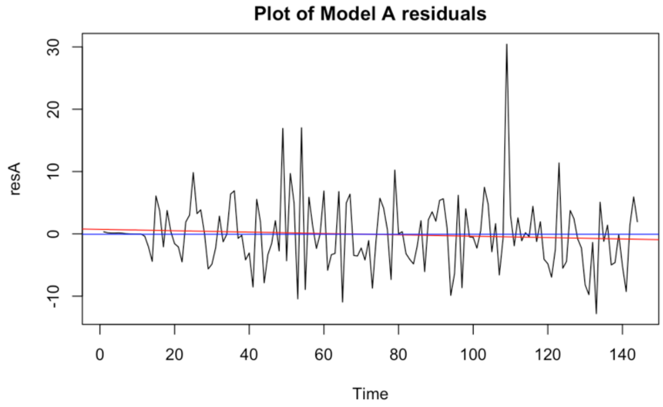


Figure 17. Plot of residuals for model A

The residual plot appears to have no seasonality, no change in variance, and slight, insignificant trend. Sample mean is also zero.

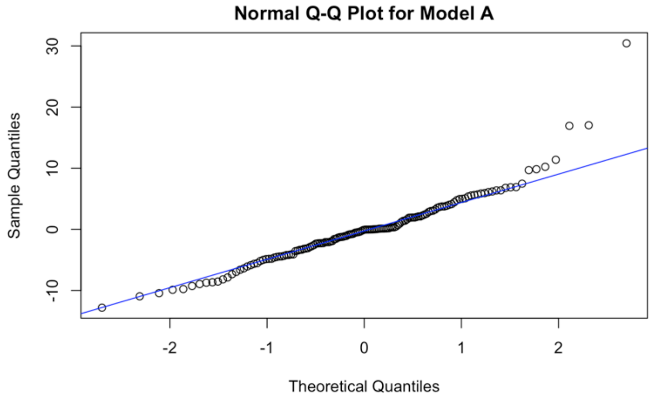


Figure 18. Normal Q-Q plot for model A

Around 95% of the data points lie within two standard deviations from the center, resembling a normal distribution.

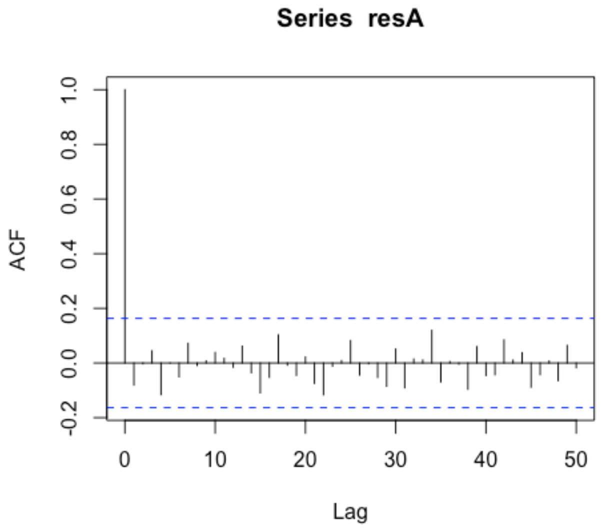


Figure 19. ACF plot of residuals for model A

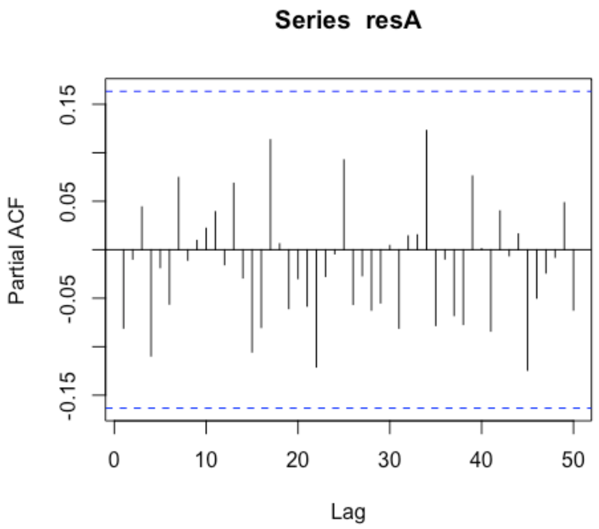
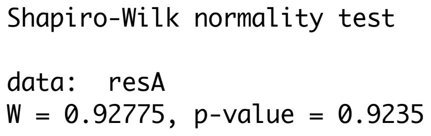


Figure 20. PACF plot of residuals for model A

Observing the ACF plot of residuals for model A, most of the ACF are within the confidence intervals. Similarly for the PACF plot of residuals, most of the PACF are also within the confidence intervals.

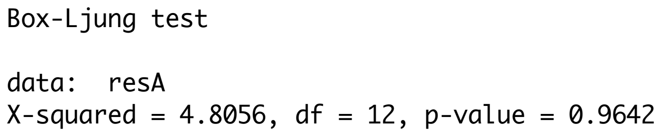
To check for the independence and normality of residuals, we will perform the Box-Ljung test and Shapiro-Wilk test, respectively.



```
Shapiro-Wilk normality test

data:  resA
W = 0.92775, p-value = 0.9235
```

Figure 21a. Output for Shapiro-Wilk normality test for model A



```
Box-Ljung test

data:  resA
X-squared = 4.8056, df = 12, p-value = 0.9642
```

Figure 21b. Output for Box-Ljung independence test for model A

Since the p-value for both tests are greater than 0.05, the residuals are independence and normal.

For Model B:

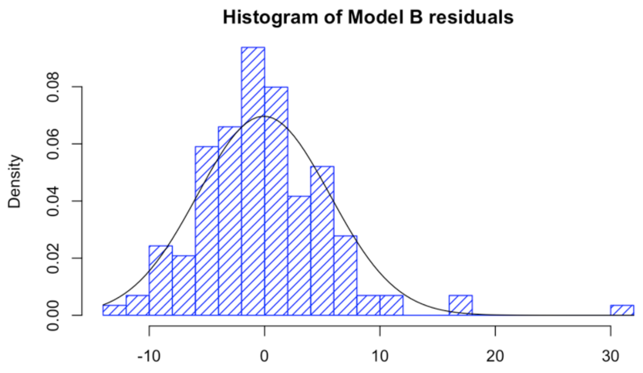


Figure 22. Histogram of residuals for model B

The histogram for model B appears to be normal with mean zero, despite the outlier.

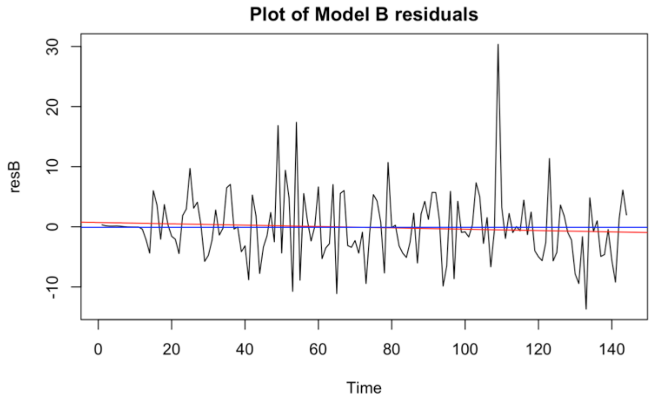


Figure 23. Plot of residuals for model B

The residual plot appears to have no seasonality, no change in variance, and slight trend. Sample mean is also zero.

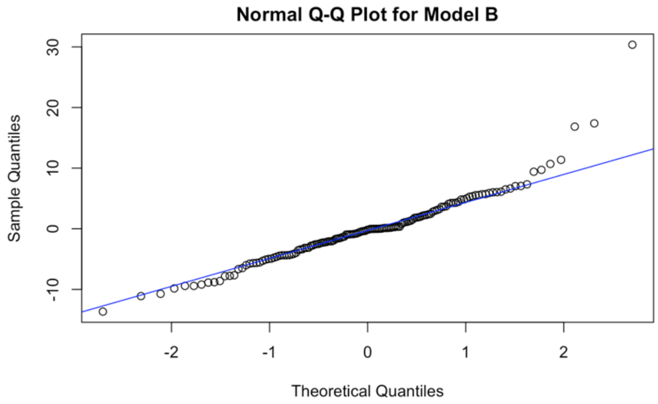


Figure 24. Normal Q-Q plot for model B

Around 95% of the data points lie within two standard deviations from the center, resembling a normal distribution.

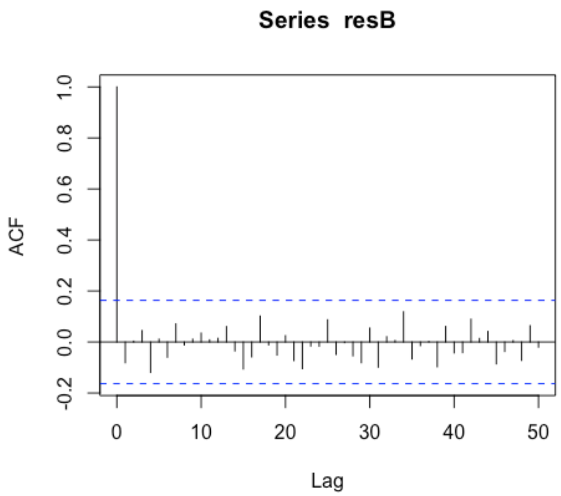


Figure 25. ACF plot of residuals for model B

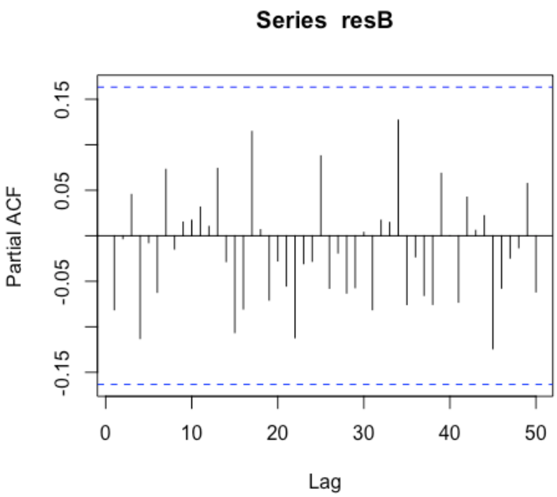


Figure 26. PACF plot of residuals for model B

Observing the ACF plot of residuals for model B, most of the ACF are within the confidence intervals. Similarly for the PACF plot of residuals, most of the PACF are also within the confidence intervals.

To check for the independence and normality of residuals, we will perform the Box-Ljung test and Shapiro-Wilk test, respectively.

```
Shapiro-Wilk normality test

data:  resB
W = 0. 0.92832, p-value = 0.9145
```

Figure 27a. Output for Shapiro-Wilk normality test for model A

```
Box-Ljung test

data:  resA
X-squared = 5.025, df = 12, p-value = 0.01165
```

Figure 27b. Output for Box-Ljung independence test for model A

Here, the p-value for the Box-Ljung test is less than 0.05, so it does not pass the independence test.

From the diagnostic checking, we conclude that model A is ideal model to proceed with as the residuals display normality, independence, and stationarity. Despite the higher value in AICc, based on the other tests that were conducted, model A stood more favorable. Model B behaved similarly, but did not pass the Box-Ljung test, so we proceed with model A. The final model is given by

$$(1 - 0.0801B^{12})Y_t = (1 - 0.7387B^{12})Z_t$$

where Zt ~ WN(0, 35.57).

Forecasting

Using model A, we can forecast the next 12 data points of Ut using the `forecast()` function in R, and plotting the results as shown in figure 28.

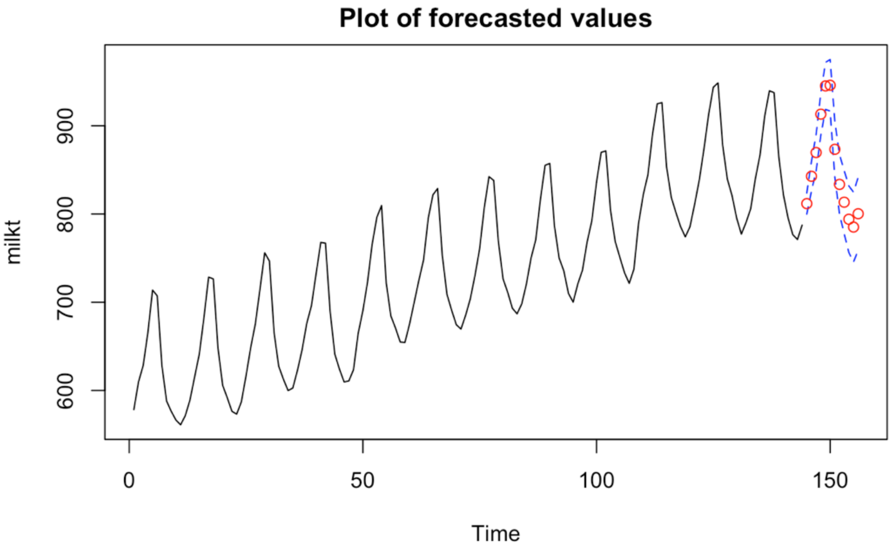


Figure 28. The plot of forecasted values for Ut by model A

The forecasted data points are shown in red, while the confidence intervals are shown as the dashed blue lines. Since we used training data, we can compare these results to the original data set in figure 29.

```
pred.tr
[1] 811.7693 842.7451 869.6512 913.1131 945.1368 945.8775 873.3154 833.5454
[9] 813.3699 794.0550 785.1878 800.3059
milktest
[1] 813.0 845.7 872.9 915.2 951.4 960.8 891.5 851.3 826.9 797.3 784.3 798.2
```

Figure 29. Predicted values (pred.tr) and original data (milktest)

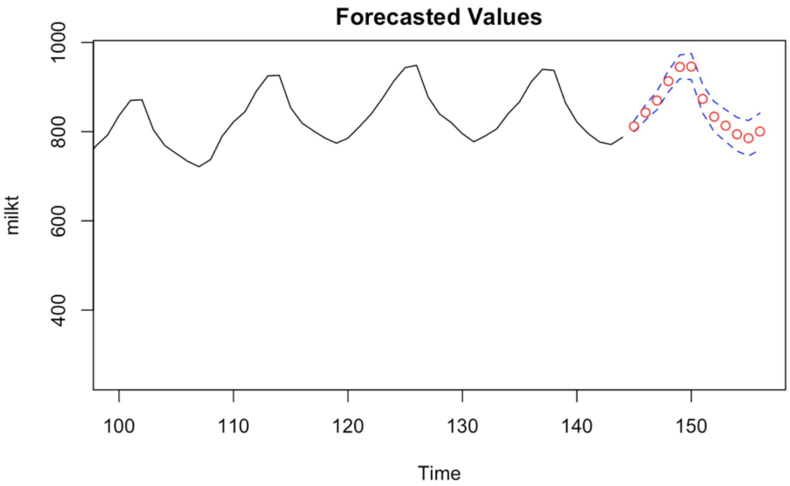


Figure 30. The plot of forecasted values for Ut by model A, close up

Figure 30 displays a closer view at the forecasted values, where the red circles represent the forecasted data points, the dashed blue lines represent the confidence intervals, and the black line represents the training set data Ut.

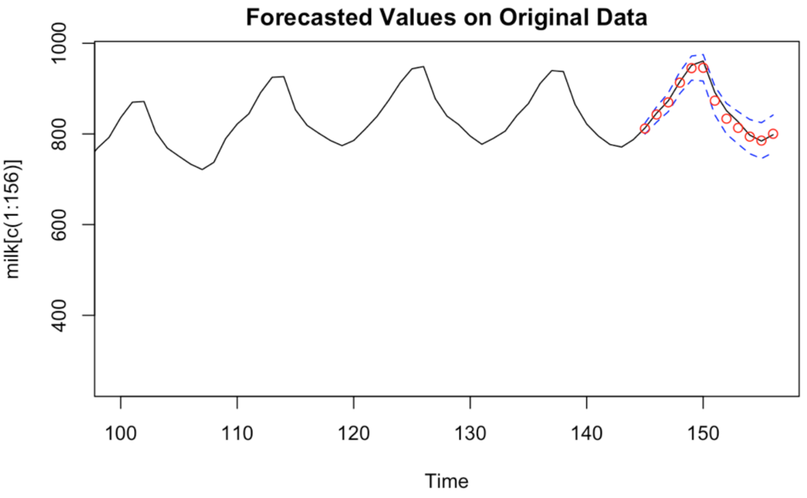


Figure 31. The plot of forecasted values over original data by model A, close up

Figure 31 displays a closer look the forecasted values, but over the original data points. Here, the red circles represent the forecasted data points, the dashed blue lines represent the confidence intervals, and the black line represents the original dataset milk.

Conclusion

In conclusion, by using the Box-Jenkins method, I was able to forecast the last 12 data points of the dataset milk using forecasting techniques such as differencing, diagnostic checking, and analysis of ACF and PACF. By comparing two models, I was able to select the ideal, final SARIMA model which is given by

$$(1 - 0.0801B^{12})Y_t = (1 - 0.7387B^{12})Z_t$$

where Zt ~ WN(0, 35.57). Comparing the test data and the forecasted data, I was able to build a model that mirrored the patterns and behavior of the original dataset, milk.

References

Makridakis, Wheelwright and Hyndman (1998) Forecasting: methods and applications, John Wiley & Sons: New York. Chapter 2

Feldman, R 2020, *Lecture notes for week 3*, lecture notes, PSTAT 174/274, University of California, Santa Barbara, 13 April 2020

Feldman, R 2020, *Lecture notes for week 4*, lecture notes, PSTAT 174/274, University of California, Santa Barbara, 20 April 2020

Feldman, R 2020, *Lecture notes for week 5*, lecture notes, PSTAT 174/274, University of California, Santa Barbara, 27 April 2020

Appendix

Data Preparation

```
#Loading packages
library(fma)
library(MASS)
library(astsa)
library(qpcR)

#Preparing the data
milkt <- milk[c(1:144)] #Training dataset U to build a model
milktest <- milk[c(145:156)] #Test dataset to forecast
```

Analyzing the Time Series

```
# Raw Data Time Series Plot
tsdat <- ts(milk, start = c(1962,1), end = c(1974,2), frequency = 12)
ts.plot(tsdat, main = "Raw Data") #figure 1

#Plotting the training data
plot.ts(milkt, main = "Model Training Data: Milk Production") #figure 2
#Adding the trend line to plot
fitt <- lm(milkt ~ as.numeric(1:length(milkt)))
abline(fitt, col = 'red')
#Adding mean to the plot
abline(h=mean(milk), col = 'blue')

#Plotting the histogram, figure 3
hist(milkt, col="light blue", xlab="", main="Histogram of milk data")

#Plotting the ACF
acf(milkt,lag.max=40, main="ACF of the Milk Data") #figure 4
```

Transformations and Differencing

```
#Decomposition of data
library(ggplot2) #Installing necessary packages
library(ggfortify)
y <- ts(as.ts(milkt), frequency = 12)
decomp <- decompose(y)
plot(decomp) #figure 5

#Differencing
var(milkt) #Initial variance of milkt
#Differencing at lag 12
milkt.12 <- diff(milkt, lag=12)
plot.ts(milkt.12, main="ln(U_t) differenced at lag 12") #figure 6
```

```
var(milkt.12)
abline(h=mean(milkt.12), col="blue")
fit <- lm(milkt.12 ~ as.numeric(1:length(milkt.12)))
abline(fit, col="red")
#Differencing again at lag 1
milkt.data <- diff(milkt.12, lag= 1)
plot.ts(milkt.data, main = "U_t differenced at lag 12 and lag 1")#figure 7
var(milkt.data)
abline(h=mean(milkt.data), col="blue")
fit.1 <- lm(milkt.data ~ as.numeric(1:length(milkt.data)))
abline(fit.1, col="red")
#ACF of data
acf(milkt, lag.max=40, main="ACF of the U_t") #figure 8
acf(milkt.12, lag.max=40, main="ACF of the U_t, differenced at lag
12")#figure 9
acf(milkt.data, lag.max=40, main="ACF of the U_t, differenced at lags 12
and 1")#figure 10
# Plotting histogram with normal curve, figure 11
hist(milkt.data, density=20,breaks=20, col="blue", xlab="", prob=TRUE)
m <- mean(milkt.data)
std <- sqrt(var(milkt.data))
curve( dnorm(x,m,std), add=TRUE )
```

ACF and PACF Analysis

```
#ACF and PACF Analysis
acf(milkt.data, lag.max=50, main="ACF of the U_t, differenced at lags 12
and 1") #figure 12
pacf(milkt.data, lag.max=50, main="PACF of the U_t, differenced at lags 12
and 1")#figure 13
```

Fitting the model

```
#Testing possible models
fit1 <- sarima(xdata = milkt, p = 0, d = 1, q = 0, P = 1 , D = 1, Q = 1, S
= 12, details = F) #P=1, Q=1
fit1f <- sarima(xdata = milkt, p = 0, d = 1, q = 0, P = 1 , D = 1, Q = 1, S
= 12, fixed = c(0,NA), details = F) #P=1, Q=1, fixed coefficients
fit2 <- sarima(xdata = milkt, p = 0, d = 1, q = 0, P = 4 , D = 1, Q = 1, S
= 12, details = F) #P=4, Q=1
fit2f <- sarima(xdata = milkt, p = 0, d = 1, q = 0, P = 4 , D = 1, Q = 1, S
= 12, fixed = c(NA, NA, NA, NA, 0), details = F) #P=4, Q=1, fixed
coefficients
fit3 <- sarima(xdata = milkt, p = 0, d = 1, q = 0, P = 1 , D = 1, Q = 3, S
= 12, details = F) #P=1, Q=3
fit4 <- sarima(xdata = milkt, p = 0, d = 1, q = 0, P = 4 , D = 1, Q = 3, S
= 12, details = F) #P=3, Q=3
fit5 <- <- sarima(xdata = milkt, p = 0, d = 1, q = 0, P = 0 , D = 1, Q = 1,
S = 12, details = F) #P=0, Q=1
```

```r
#FINAL MODELS:
modelA <- sarima(xdata = milkt, p = 0, d = 1, q = 0, P = 1 , D = 1, Q = 1,
S = 12, details = F) #fit2f
modelA #figure 14
modelB <- sarima(xdata = milkt, p = 0, d = 1, q = 0, P = 0 , D = 1, Q = 1,
S = 12, details = F) #fit5
modelB #figure 15
```

Diagnostic Checking

```r
#MODEL A
fitA <- arima(milkt, order=c(0,1,0), seasonal = list(order = c(1,1,1),
period = 12), method="ML")
resA <- residuals(fitA)
hist(resA,density=20,breaks=20, col="blue", xlab="", prob=TRUE) #figure 16
mA <- mean(resA)
stdA <- sqrt(var(resA))
curve(dnorm(x,mA,stdA), add=TRUE ) #Adding curve to histogram
plot.ts(resA) #Plotting residuals, figure 17

#MODEL B
fitB <- arima(milkt, order=c(0,1,0), seasonal = list(order = c(0,1,1),
period = 12), method="ML")
resB <- residuals(fitB)
hist(resB,density=20,breaks=20, col="blue", xlab="", prob=TRUE) #figure 22
mB <- mean(resB)
stdB <- sqrt(var(resB))
curve(dnorm(x,mB,stdB), add=TRUE ) #Adding curve to histogram
plot.ts(resB) #Plotting residuals, figure 23

#Normal Q-Q Plots
#Model A
plot.ts(resA) #figure 18
fittA <- lm(resA ~ as.numeric(1:length(resA)))
abline(fittA, col="red")
abline(h=mean(resA), col="blue")
qqnorm(resA,main= "Normal Q-Q Plot for Model A")
qqline(resA,col="blue")
#Model B
plot.ts(resB) #figure 24
fittB <- lm(resB ~ as.numeric(1:length(resB)))
abline(fittB, col="red")
abline(h=mean(resB), col="blue")
qqnorm(resB,main= "Normal Q-Q Plot for Model B")
qqline(resB,col="blue")

#ACF/PACF of residuals
acf(resA, lag.max = 50) #figure 19
pacf(resA,lag.max = 50) #figure 20
```

```
acf(resB,lag.max = 50) #figure 25
pacf(resB,lag.max = 50) #figure 26

#Tests for residual correlations and independence
shapiro.test(resA) #figure 21a
Box.test(resA, lag = 12, type = c("Ljung-Box"), fitdf = 0) #figure 21b

shapiro.test(resB) #figure 27a
Box.test(resB, lag = 12, type = c("Ljung-Box"), fitdf = 0) #figure 27b
```

Forecasting

```
#Loading packages
library(forecast)
fitA <- arima(milkt, order=c(0,1,0), seasonal = list(order = c(1,1,1),
period = 12), method="ML")
forecast(fitA)
#Plotting the forecast on training data, figure 28
pred.tr <- predict(fitA, n.ahead = 12)
U.tr= pred.tr$pred + 2*pred.tr$se
L.tr= pred.tr$pred - 2*pred.tr$se
ts.plot(milkt, xlim=c(1,length(milkt)+12), ylim = c(min(milkt),max(U.tr)))
lines(U.tr, col="blue", lty="dashed")
lines(L.tr, col="blue", lty="dashed")
points((length(milkt)+1):(length(milkt)+12), pred.tr$pred, col="red")

#Comparing the forecasted data to the original data
pred.tr #From plotting figure 28
milktest #From data preparation

#Plotting the forecast on training data, close up, figure 30
ts.plot(milkt, xlim = c(100,length(milkt)+12), ylim = c(250,max(U.tr)))
lines(U.tr, col="blue", lty="dashed")
lines(L.tr, col="blue", lty="dashed")
points((length(milkt)+1):(length(milkt)+12), pred.tr$pred, col="red")

#Plotting the forecast on original data, close up , figure 31
ts.plot(milk[c(1:156)], xlim = c(100,length(milkt)+12), ylim =
c(250,max(U.tr)))
lines(U.tr, col="blue", lty="dashed")
lines(L.tr, col="blue", lty="dashed")
points((length(milkt)+1):(length(milkt)+12), pred.tr$pred, col="red")
```