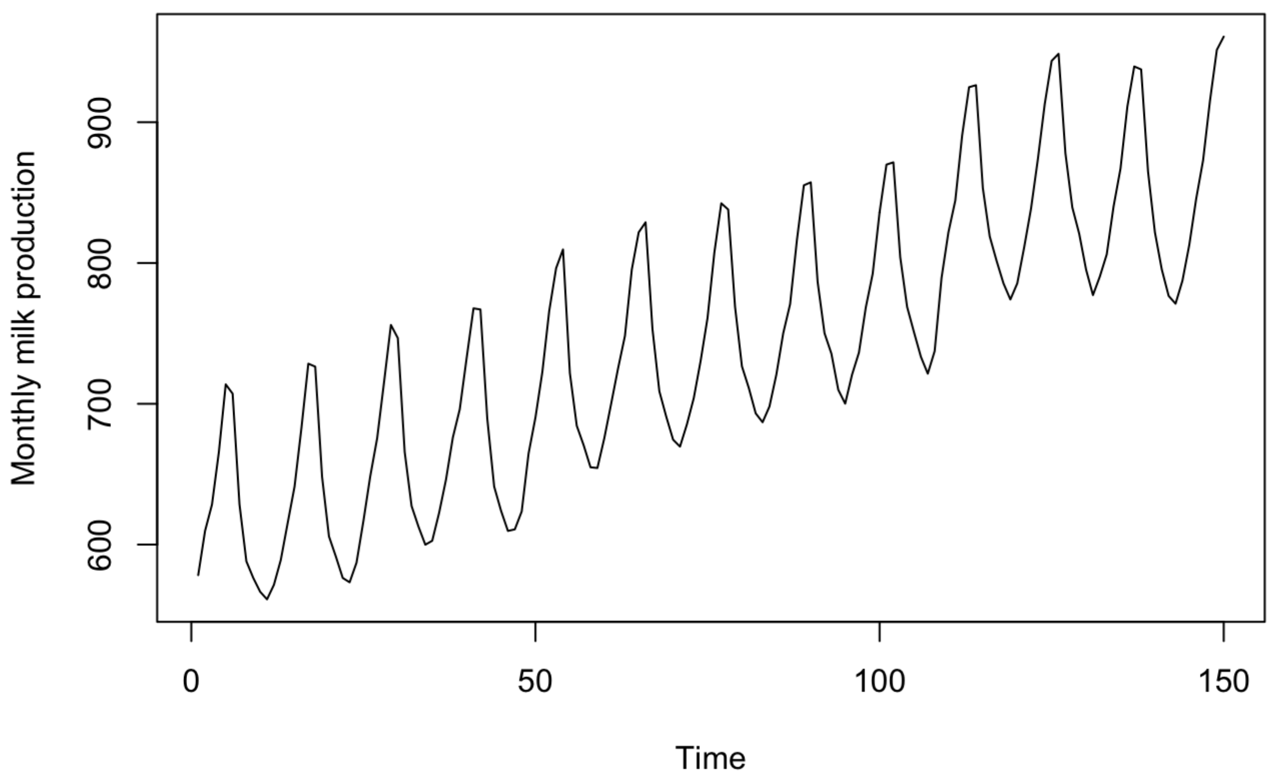
pstat274_lab07_aoxu

AO XU

2022-11-09

```
library(tsdl)
milk <- subset(tsdl, 12, "Agriculture")[[3]]
train <- milk[1:150]
test <- milk[151:156]
ts.plot(train, ylab = "Monthly milk production")
```



To

make it more stationary, we need to remove trend and seasonality:

```
dmilk <- diff(train, 12)
ddmilk <- diff(dmilk, 1)
dmilk
```

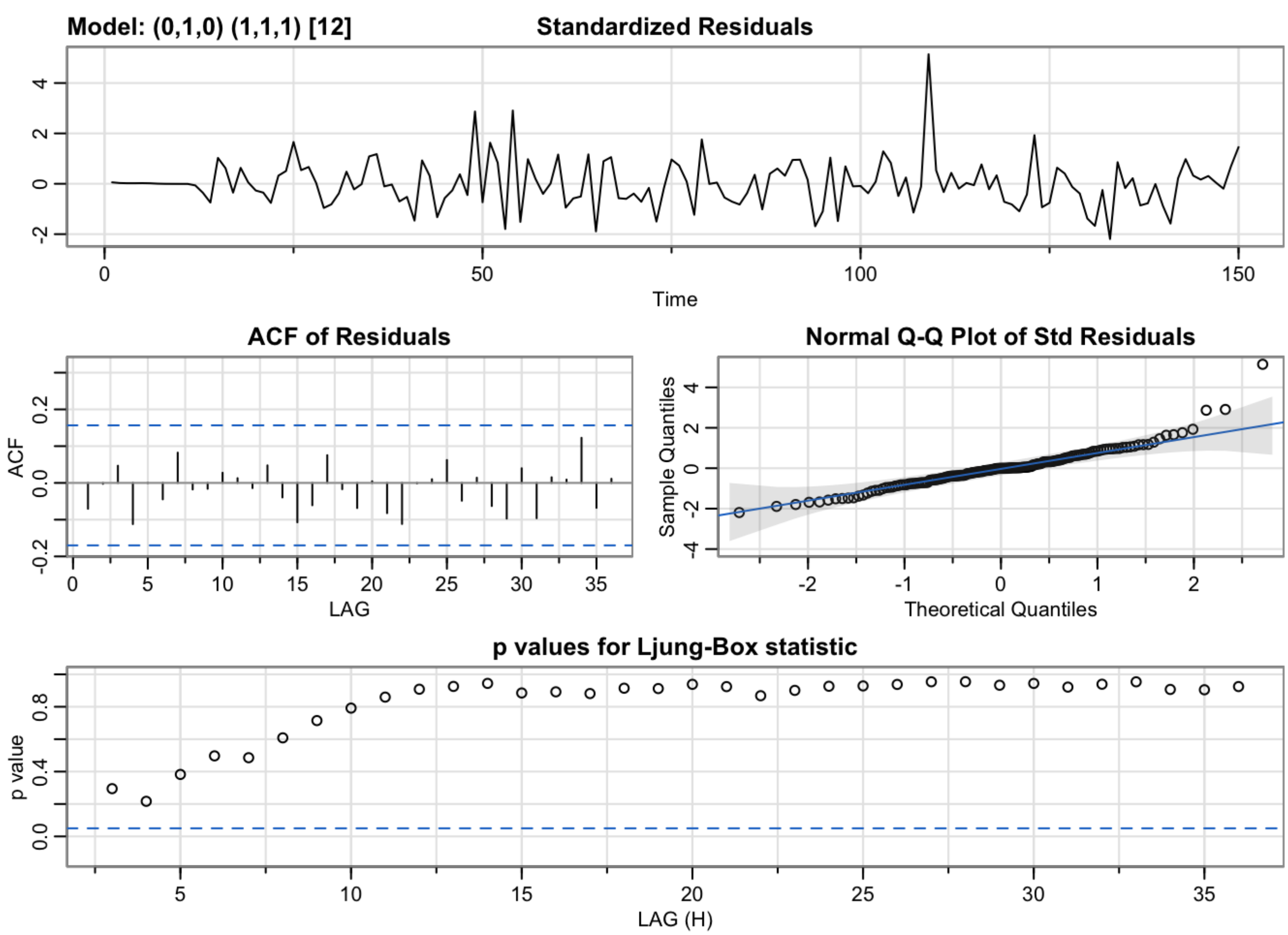
```
##   [1]  10.8   5.5  12.8  17.2  14.7  19.2  19.6  17.7  15.2   9.8  12.1  15.7
##  [13]  27.5  33.3  34.3  32.5  27.5  20.3  17.7  21.6  21.3  23.6  29.5  35.4
##  [25]  29.5  27.5  20.6  17.2  11.8  20.3  23.6  13.8  11.2   9.8   8.1   1.0
##  [37]  18.6  14.2  26.5  33.5  28.5  42.6  32.4  43.2  46.6  45.2  43.6  52.0
##  [49]  35.4  34.8  25.6  29.4  25.5  19.3  31.4  24.5  20.3  19.6  15.2   9.8
##  [61]   3.9   5.4  12.7  12.2  20.6   9.1  15.7  17.7  20.3  18.7  17.3  12.8
##  [73]  16.7  19.6   9.9   9.1  12.8  19.3  17.7  23.5  24.4  16.7  13.2  22.6
##  [85]  15.7  18.4  21.6  19.3  14.7  14.2  17.6  18.7  15.2  23.5  21.3  16.7
##  [97]  53.0  53.3  52.0  54.8  55.0  54.8  49.1  50.1  50.7  52.1  52.7  48.1
## [109]  21.6  16.8  29.5  22.3  18.7  22.3  24.6  20.6  19.3   9.8   3.1   4.9
## [121]  -4.9   1.7  -6.9  -2.0  -4.0 -11.1 -12.8 -17.7 -25.4 -18.7  -6.1  -3.0
## [133]   6.9   5.4   5.9   4.1  11.8  23.3
```

```
ddmilk
```

```
##   [1]  -5.3   7.3   4.4  -2.5   4.5   0.4  -1.9  -2.5  -5.4   2.3   3.6  11.8
##  [13]   5.8   1.0  -1.8  -5.0  -7.2  -2.6   3.9  -0.3   2.3   5.9   5.9  -5.9
##  [25]  -2.0  -6.9  -3.4  -5.4   8.5   3.3  -9.8  -2.6  -1.4  -1.7  -7.1  17.6
##  [37]  -4.4  12.3   7.0  -5.0  14.1 -10.2  10.8   3.4  -1.4  -1.6   8.4 -16.6
##  [49]  -0.6  -9.2   3.8  -3.9  -6.2  12.1  -6.9  -4.2  -0.7  -4.4  -5.4  -5.9
##  [61]   1.5   7.3  -0.5   8.4 -11.5   6.6   2.0   2.6  -1.6  -1.4  -4.5   3.9
##  [73]   2.9  -9.7  -0.8   3.7   6.5  -1.6   5.8   0.9  -7.7  -3.5   9.4  -6.9
##  [85]   2.7   3.2  -2.3  -4.6  -0.5   3.4   1.1  -3.5   8.3  -2.2  -4.6  36.3
##  [97]   0.3  -1.3   2.8   0.2  -0.2  -5.7   1.0   0.6   1.4   0.6  -4.6 -26.5
## [109]  -4.8  12.7  -7.2  -3.6   3.6   2.3  -4.0  -1.3  -9.5  -6.7   1.8  -9.8
## [121]   6.6  -8.6   4.9  -2.0  -7.1  -1.7  -4.9  -7.7   6.7  12.6   3.1   9.9
## [133]  -1.5   0.5  -1.8   7.7  11.5
```

```
library(astsa)
fit.i <- sarima(xdata=train, p=0, d=1, q=0, P=1, D=1, Q=1, S=12)
```

```
## initial  value 1.989465
## iter   2 value 1.850408
## iter   3 value 1.824156
## iter   4 value 1.800049
## iter   5 value 1.791131
## iter   6 value 1.789958
## iter   7 value 1.789636
## iter   8 value 1.789235
## iter   9 value 1.789186
## iter  10 value 1.789182
## iter  10 value 1.789182
## iter  10 value 1.789182
## final  value 1.789182
## converged
## initial  value 1.803940
## iter   2 value 1.803675
## iter   3 value 1.803165
## iter   4 value 1.803164
## iter   5 value 1.803163
## iter   6 value 1.803163
## iter   6 value 1.803163
## final  value 1.803163
## converged
```



```
# Residual plots:
res <- residuals(fit.i)
res
```

```
## NULL
```

a.

There is no residual since res is null.

From the ACF of residuals plot, we could get that residuals lie within the 95% CI, so it could be assumed as zero.

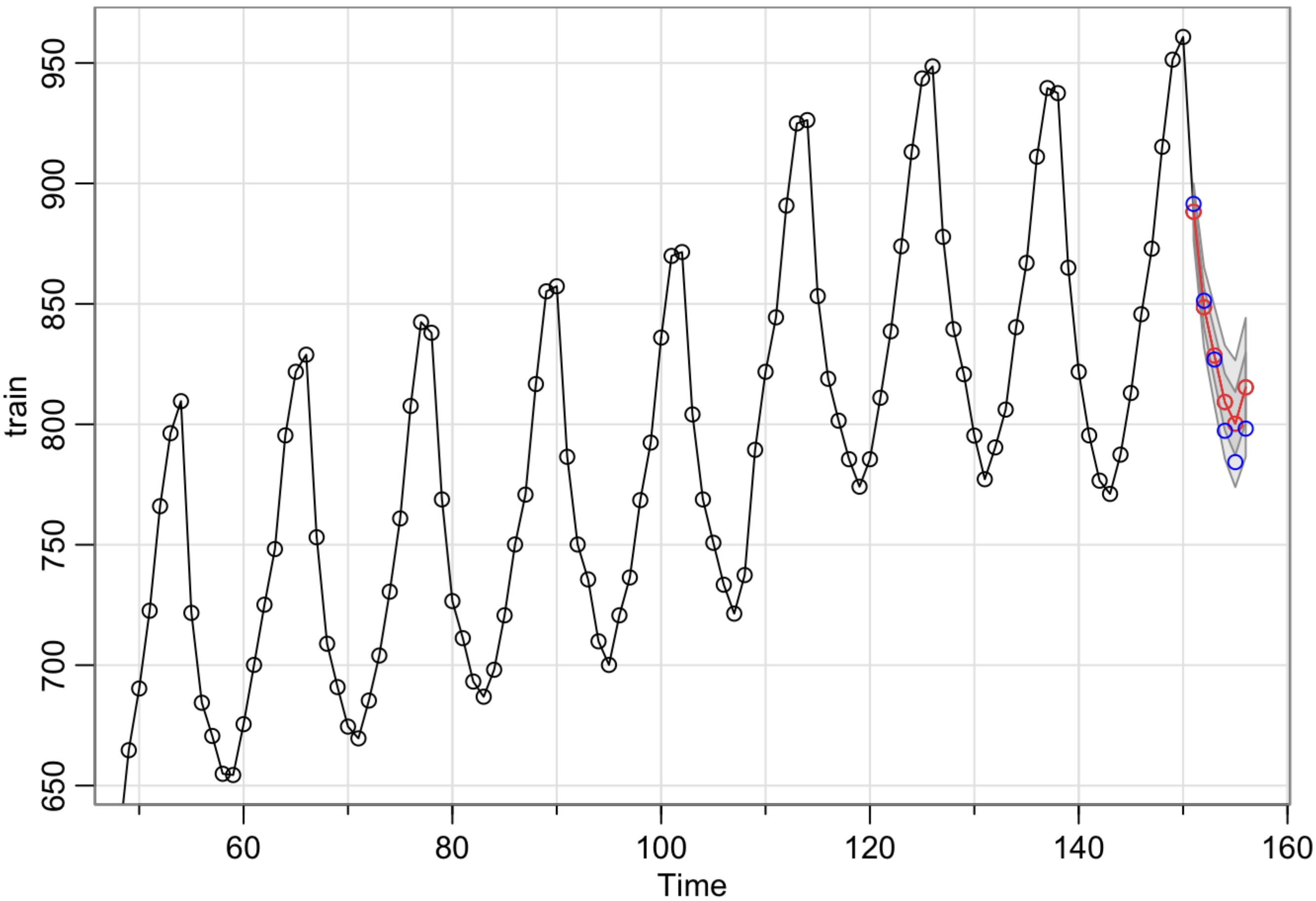
From QQ plot, we could get that all points lie on a line, meaning residuals are normally distributed.

Nearly all p-values are greater than 5%, so we fail to reject the hypothesis.

Therefore, the residuals are white noise.

b.

```
# Predict 6 future observations and plot
par(mfrow=c(1, 1))
mypred <- sarima.for(train, n.ahead = 6, p=0, d=1, q=0, P=1, D=1, Q=1, S=12)
#ts.plot(c(milk), xlim=c(1, length(milk) + 6), ylim=c(min(milk)*0.8, max(milk)*1.2))
#points((length(milk) + 1):(length(milk) + 6), col="red",(lambda*mypred$pred + 1)^(1/lam
bda))
#lines((length(wine) + 1):(length(wine) + 6),(lambda*mypred$pred + 1.96*mypred$se + 1)^
(1/lambda), lty=2)
points(151:156, test, col="blue")
```



950
900
850
800
750
700
650
train
60
80
100
120
140
160
Time