STAT 626 Project - Version 1

Ken Marciel

6/12/2021

## US Meat Production

Packages

library(readxl) # for read\_excel function  
library(astsa) # for tsplot function  
library(dplyr) # data cleaning

Data

# read data from Excel file  
setwd("C:/Users/keoka/OneDrive - Texas A&M University/Courses/STAT\_626/Project/Analysis/Ken/")  
meat <- read\_excel("MeatStatsFull.xlsx", sheet = 2)  
head(meat)

## # A tibble: 6 × 17  
## `Red meat and po…` ...2 ...3 ...4 ...5 ...6 ...7 ...8 ...9 ...10 ...11  
## <chr> <chr> <chr> <chr> <chr> <lgl> <chr> <chr> <chr> <chr> <chr>  
## 1 Type 1/ Comm… <NA> <NA> <NA> NA <NA> Fede… <NA> <NA> <NA>   
## 2 <NA> Beef… Veal… Pork… Lamb… NA Tota… Beef… Veal… Pork… Lamb…  
## 3 Jan-Jun 2021 1385… 26.3 1395… 71 NA 2790… 1362… 25.4 1389… 63.3   
## 4 Jan-Jun 2020 1298… 32.9 1373… 71.1 NA 26833 1278… 31.8 13666 64.4   
## 5 Jun-2021 2399… 4.1 2254… 11.3 NA 4668… 2360 4 2243… 9.9   
## 6 May-2021 2210… 3.7 2070 11 NA 4295… 2173… 3.5 2059… 9.6   
## # … with 6 more variables: ...12 <chr>, ...13 <chr>, ...14 <chr>, ...15 <chr>,  
## # ...16 <chr>, ...17 <chr>

# total monthly red meat and poultry production from January 1983 to April 2021  
meat <- meat[464:5, c(1,17)] # changes data from reverse to chronological order  
colnames(meat) <- c("Month","MeatProd")  
head(meat)

## # A tibble: 6 × 2  
## Month MeatProd  
## <chr> <chr>   
## 1 Mar-1983 4489.982  
## 2 Apr-1983 4207.824  
## 3 May-1983 4376.39   
## 4 Jun-1983 4587.926  
## 5 Jul-1983 4139.788  
## 6 Aug-1983 4749.126

dim(meat) # 460 months = 38 1/3 years

## [1] 460 2

class(meat) # data frame

## [1] "tbl\_df" "tbl" "data.frame"

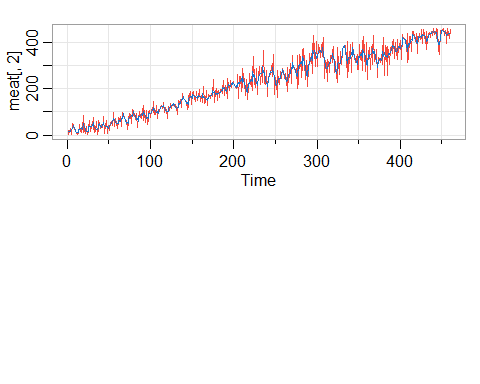
meat = ts(meat) # change data frame to time series  
class(meat) # time series

## [1] "mts" "ts" "matrix"

### Exploratory Data Analysis

Upward trend shows that the raw data is not stationary.

# 3-term moving average  
meat\_ma = stats::filter(meat[,2], sides = 2, filter = rep(1/3,3))  
par(mfrow = c(2,1))  
tsplot(meat[,2], col=2)  
lines(meat\_ma, col=4)

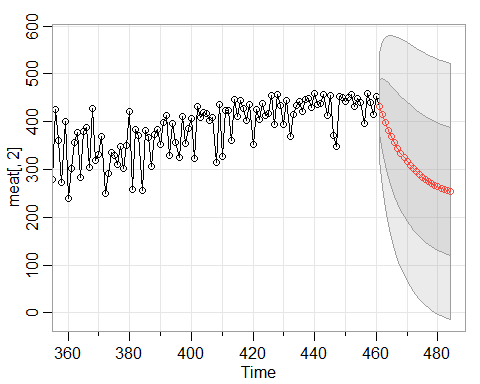


### Forecasting

#### No Transformation

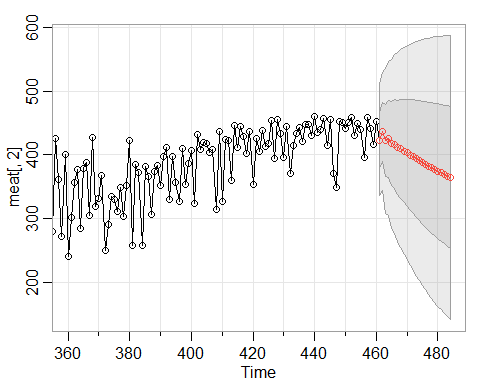
AR models seem to work best, but the data is nonstationary.

sarima.for(meat[,2], n.ahead=24, p=1, d=0, q=0) # AR(1)



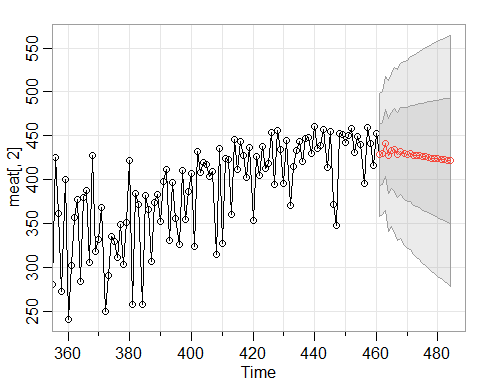
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 432.6623 414.1965 397.4302 382.2070 368.3849 355.8349 344.4400 334.0939  
## [9] 324.6999 316.1706 308.4262 301.3947 295.0102 289.2134 283.9501 279.1713  
## [17] 274.8322 270.8925 267.3154 264.0675 261.1186 258.4410 256.0099 253.8026  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 56.54347 76.37341 89.47509 98.98040 106.17855 111.76461 116.16791  
## [8] 119.67621 122.49291 124.76717 126.61135 128.11173 129.33555 130.33582  
## [15] 131.15470 131.82596 132.37679 132.82917 133.20096 133.50668 133.75819  
## [22] 133.96518 134.13558 134.27590

sarima.for(meat[,2], n.ahead=24, p=2, d=0, q=0) # AR(2)



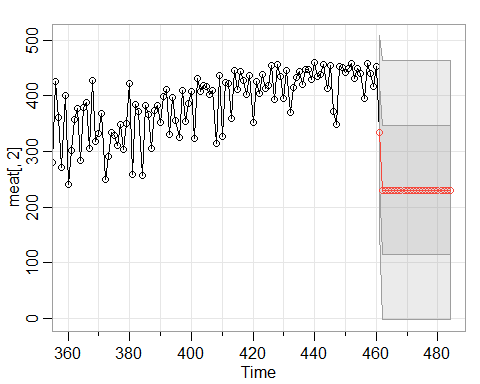
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 423.0503 436.2353 421.8138 425.2412 417.3064 416.7782 411.5941 409.5033  
## [9] 405.5244 402.8567 399.4424 396.6021 393.4855 390.6372 387.7057 384.9152  
## [17] 382.1200 379.4122 376.7310 374.1139 371.5354 369.0104 366.5279 364.0936  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 43.91549 46.37073 56.76903 60.35512 66.36866 70.01062 74.32984  
## [8] 77.66731 81.11168 84.07787 86.97363 89.59358 92.09776 94.41704  
## [15] 96.61821 98.68102 100.63598 102.47996 104.22871 105.88472 107.45735  
## [22] 108.95056 110.37069 111.72177

sarima.for(meat[,2], n.ahead=24, p=3, d=0, q=0) # AR(3)



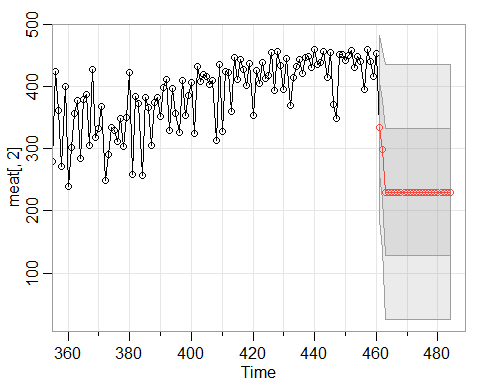
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 428.1295 430.1254 441.3920 426.9847 433.5846 433.8375 428.0790 432.3441  
## [9] 429.8467 428.3455 429.8687 427.6719 427.5162 427.4820 426.1045 426.0483  
## [17] 425.4346 424.6109 424.3429 423.6306 423.0497 422.6036 421.9440 421.4290  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 35.18547 35.21011 38.37298 43.21065 43.50631 46.89559 48.77081 49.80294  
## [9] 52.26774 53.57483 54.99939 56.77489 58.00241 59.45652 60.87210 62.09172  
## [17] 63.43485 64.67380 65.86149 67.08062 68.22185 69.35227 70.47051 71.54281

sarima.for(meat[,2], n.ahead=24, p=0, d=0, q=1) # MA(1)



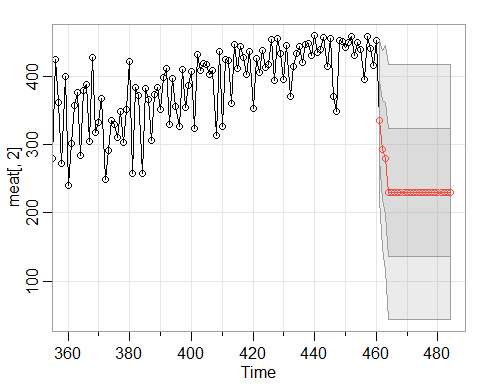
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 333.3657 230.4737 230.4737 230.4737 230.4737 230.4737 230.4737 230.4737  
## [9] 230.4737 230.4737 230.4737 230.4737 230.4737 230.4737 230.4737 230.4737  
## [17] 230.4737 230.4737 230.4737 230.4737 230.4737 230.4737 230.4737 230.4737  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 87.7267 116.6536 116.6536 116.6536 116.6536 116.6536 116.6536 116.6536  
## [9] 116.6536 116.6536 116.6536 116.6536 116.6536 116.6536 116.6536 116.6536  
## [17] 116.6536 116.6536 116.6536 116.6536 116.6536 116.6536 116.6536 116.6536

sarima.for(meat[,2], n.ahead=24, p=0, d=0, q=2) # MA(2)



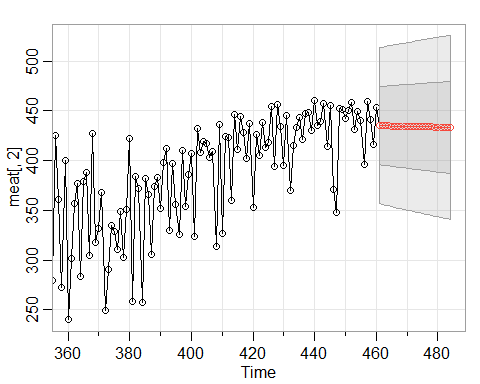
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 333.7481 298.3729 230.3813 230.3813 230.3813 230.3813 230.3813 230.3813  
## [9] 230.3813 230.3813 230.3813 230.3813 230.3813 230.3813 230.3813 230.3813  
## [17] 230.3813 230.3813 230.3813 230.3813 230.3813 230.3813 230.3813 230.3813  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 74.27900 79.82708 102.67653 102.67653 102.67653 102.67653 102.67653  
## [8] 102.67653 102.67653 102.67653 102.67653 102.67653 102.67653 102.67653  
## [15] 102.67653 102.67653 102.67653 102.67653 102.67653 102.67653 102.67653  
## [22] 102.67653 102.67653 102.67653

sarima.for(meat[,2], n.ahead=24, p=0, d=0, q=3) # MA(3)



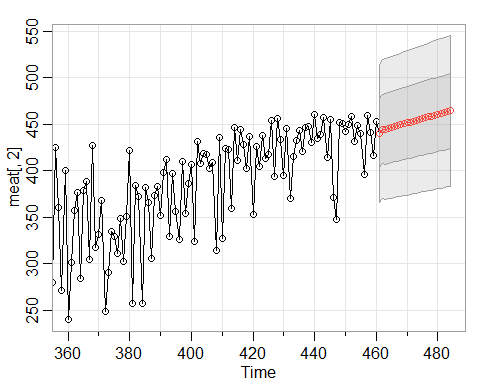
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 335.7462 292.1823 279.7890 230.2667 230.2667 230.2667 230.2667 230.2667  
## [9] 230.2667 230.2667 230.2667 230.2667 230.2667 230.2667 230.2667 230.2667  
## [17] 230.2667 230.2667 230.2667 230.2667 230.2667 230.2667 230.2667 230.2667  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 58.10111 72.50145 82.35172 93.42266 93.42266 93.42266 93.42266 93.42266  
## [9] 93.42266 93.42266 93.42266 93.42266 93.42266 93.42266 93.42266 93.42266  
## [17] 93.42266 93.42266 93.42266 93.42266 93.42266 93.42266 93.42266 93.42266

sarima.for(meat[,2], n.ahead=24, p=1, d=0, q=1) # ARMA(1,1)



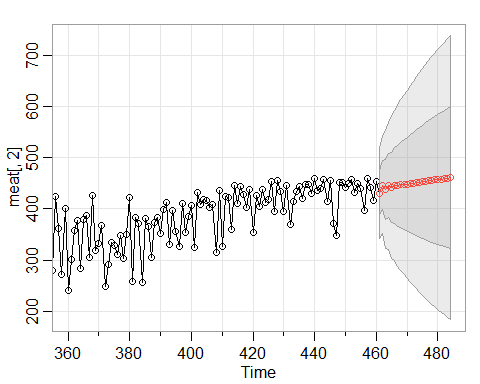
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 435.0480 434.9732 434.8984 434.8236 434.7488 434.6741 434.5994 434.5248  
## [9] 434.4501 434.3755 434.3010 434.2264 434.1519 434.0774 434.0029 433.9285  
## [17] 433.8540 433.7797 433.7053 433.6310 433.5567 433.4824 433.4081 433.3339  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 39.19707 39.53212 39.86410 40.19311 40.51920 40.84246 41.16294 41.48071  
## [9] 41.79584 42.10838 42.41838 42.72592 43.03103 43.33377 43.63419 43.93234  
## [17] 44.22826 44.52200 44.81361 45.10311 45.39056 45.67598 45.95943 46.24093

sarima.for(meat[,2], n.ahead=24, p=1, d=1, q=1) # ARIMA(1,1,1)



## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 440.1571 444.7538 444.7058 445.8949 446.7544 447.7018 448.6257 449.5559  
## [9] 450.4844 451.4134 452.3422 453.2711 454.1999 455.1288 456.0577 456.9865  
## [17] 457.9154 458.8442 459.7731 460.7020 461.6308 462.5597 463.4886 464.4174  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 36.95797 37.41349 37.82623 37.91130 38.05871 38.18731 38.32019 38.45136  
## [9] 38.58241 38.71293 38.84303 38.97270 39.10193 39.23074 39.35913 39.48710  
## [17] 39.61466 39.74181 39.86855 39.99490 40.12084 40.24639 40.37155 40.49632

sarima.for(meat[,2], n.ahead=24, p=1, d=2, q=1) # ARIMA(1,2,1)

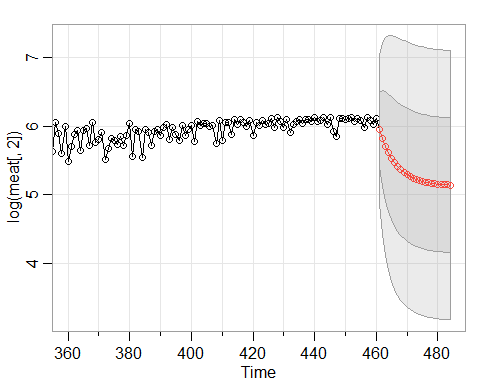


## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 430.6239 446.5987 437.8028 445.0062 441.8756 445.4198 444.6528 446.6703  
## [9] 446.8893 448.2700 448.9003 450.0153 450.8173 451.8214 452.6950 453.6529  
## [17] 454.5563 455.4949 456.4108 457.3414 458.2625 459.1897 460.1129 461.0387  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 44.15583 46.87450 58.01682 62.17728 69.07774 73.52923 78.84322  
## [8] 83.15990 87.69563 91.76749 95.84168 99.66606 103.42599 107.02989  
## [15] 110.55474 113.96916 117.30715 120.55972 123.74329 126.85693 129.90955  
## [22] 132.90289 135.84244 138.73061

#### Log Transformation

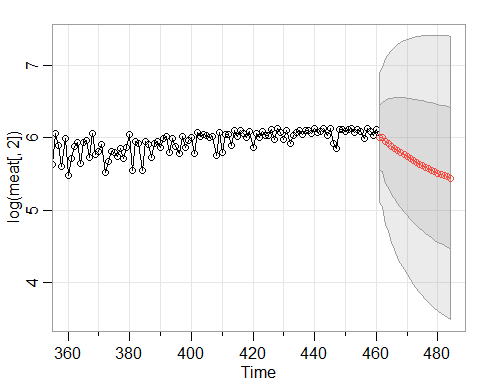
The log transformation removes the trend, resulting in stationarity. Again, the AR models seem to perform best.

sarima.for(log(meat[,2]), n.ahead=24, p=1, d=0, q=0) # AR(1)



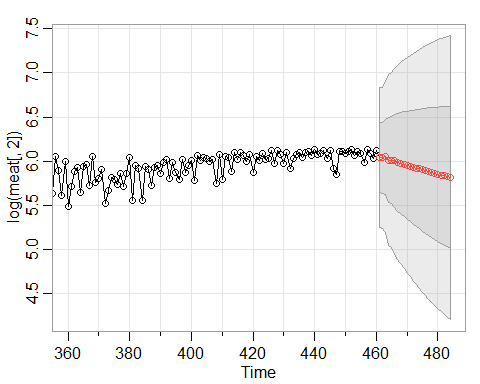
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 5.953996 5.818625 5.705433 5.610786 5.531646 5.465472 5.410139 5.363873  
## [9] 5.325186 5.292838 5.265790 5.243173 5.224262 5.208449 5.195226 5.184171  
## [17] 5.174926 5.167196 5.160733 5.155328 5.150809 5.147030 5.143871 5.141229  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 0.5387408 0.7022600 0.7968999 0.8568824 0.8964393 0.9230898 0.9412749  
## [8] 0.9537834 0.9624324 0.9684335 0.9726074 0.9755150 0.9775428 0.9789580  
## [15] 0.9799463 0.9806367 0.9811191 0.9814562 0.9816919 0.9818566 0.9819717  
## [22] 0.9820522 0.9821085 0.9821479

sarima.for(log(meat[,2]), n.ahead=24, p=2, d=0, q=0) # AR(2)



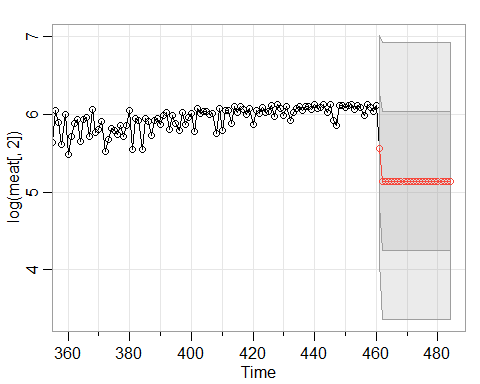
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 6.001491 6.006704 5.944249 5.923842 5.881057 5.853578 5.819222 5.790913  
## [9] 5.760992 5.733873 5.706893 5.681543 5.656881 5.633393 5.610731 5.589039  
## [17] 5.568174 5.548164 5.528940 5.510491 5.492774 5.475766 5.459436 5.443758  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 0.4480187 0.4782850 0.5725821 0.6105371 0.6621947 0.6962614 0.7316357  
## [8] 0.7598399 0.7865345 0.8094865 0.8305602 0.8492582 0.8662817 0.8816017  
## [15] 0.8955358 0.9081658 0.9196678 0.9301365 0.9396871 0.9484031 0.9563682  
## [22] 0.9636511 0.9703163 0.9764196

sarima.for(log(meat[,2]), n.ahead=24, p=3, d=0, q=0) # AR(3)



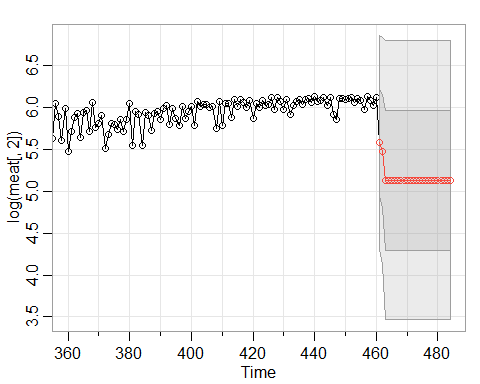
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 6.040742 6.037823 6.048322 6.013015 6.011672 6.002706 5.984531 5.978316  
## [9] 5.966301 5.953943 5.944915 5.933407 5.922750 5.912791 5.902077 5.891949  
## [17] 5.881923 5.871781 5.861942 5.852140 5.842405 5.832836 5.823329 5.813926  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 0.3962533 0.3988635 0.4296109 0.4838706 0.4960435 0.5258150 0.5518201  
## [8] 0.5689619 0.5919213 0.6110194 0.6282843 0.6465785 0.6628338 0.6785352  
## [15] 0.6939413 0.7083074 0.7222604 0.7357014 0.7485226 0.7609391 0.7728877  
## [22] 0.7843898 0.7955140 0.8062455

sarima.for(log(meat[,2]), n.ahead=24, p=0, d=0, q=1) # MA(1)



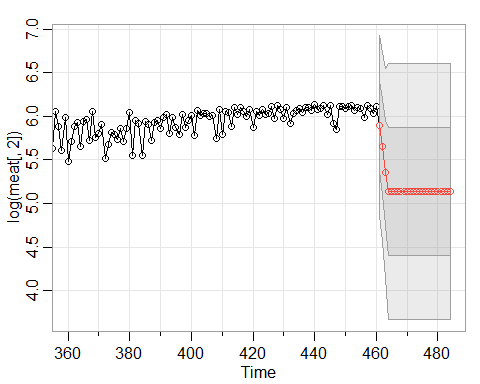
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 5.554049 5.139127 5.139127 5.139127 5.139127 5.139127 5.139127 5.139127  
## [9] 5.139127 5.139127 5.139127 5.139127 5.139127 5.139127 5.139127 5.139127  
## [17] 5.139127 5.139127 5.139127 5.139127 5.139127 5.139127 5.139127 5.139127  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 0.7295608 0.8899660 0.8899660 0.8899660 0.8899660 0.8899660 0.8899660  
## [8] 0.8899660 0.8899660 0.8899660 0.8899660 0.8899660 0.8899660 0.8899660  
## [15] 0.8899660 0.8899660 0.8899660 0.8899660 0.8899660 0.8899660 0.8899660  
## [22] 0.8899660 0.8899660 0.8899660

sarima.for(log(meat[,2]), n.ahead=24, p=0, d=0, q=2) # MA(2)



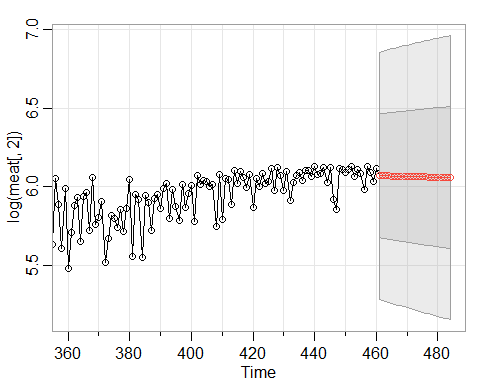
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 5.581843 5.481208 5.132749 5.132749 5.132749 5.132749 5.132749 5.132749  
## [9] 5.132749 5.132749 5.132749 5.132749 5.132749 5.132749 5.132749 5.132749  
## [17] 5.132749 5.132749 5.132749 5.132749 5.132749 5.132749 5.132749 5.132749  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 0.6406634 0.6790581 0.8340306 0.8340306 0.8340306 0.8340306 0.8340306  
## [8] 0.8340306 0.8340306 0.8340306 0.8340306 0.8340306 0.8340306 0.8340306  
## [15] 0.8340306 0.8340306 0.8340306 0.8340306 0.8340306 0.8340306 0.8340306  
## [22] 0.8340306 0.8340306 0.8340306

sarima.for(log(meat[,2]), n.ahead=24, p=0, d=0, q=3) # MA(3)



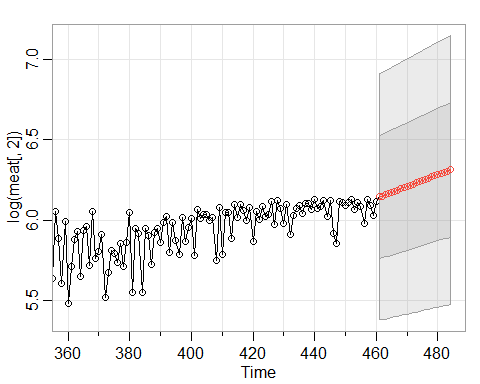
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 5.896220 5.648799 5.350824 5.135185 5.135185 5.135185 5.135185 5.135185  
## [9] 5.135185 5.135185 5.135185 5.135185 5.135185 5.135185 5.135185 5.135185  
## [17] 5.135185 5.135185 5.135185 5.135185 5.135185 5.135185 5.135185 5.135185  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 0.5162632 0.5446181 0.5988439 0.7348222 0.7348222 0.7348222 0.7348222  
## [8] 0.7348222 0.7348222 0.7348222 0.7348222 0.7348222 0.7348222 0.7348222  
## [15] 0.7348222 0.7348222 0.7348222 0.7348222 0.7348222 0.7348222 0.7348222  
## [22] 0.7348222 0.7348222 0.7348222

sarima.for(log(meat[,2]), n.ahead=24, p=1, d=0, q=1) # ARMA(1,1)



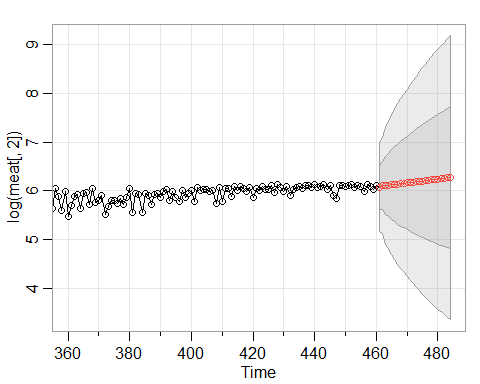
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 6.069757 6.069242 6.068727 6.068213 6.067698 6.067184 6.066670 6.066157  
## [9] 6.065643 6.065130 6.064617 6.064104 6.063592 6.063079 6.062567 6.062055  
## [17] 6.061544 6.061032 6.060521 6.060010 6.059499 6.058988 6.058478 6.057968  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 0.3933465 0.3961192 0.3988703 0.4016002 0.4043094 0.4069983 0.4096673  
## [8] 0.4123168 0.4149472 0.4175588 0.4201520 0.4227271 0.4252845 0.4278244  
## [15] 0.4303472 0.4328532 0.4353427 0.4378159 0.4402732 0.4427148 0.4451409  
## [22] 0.4475519 0.4499479 0.4523292

sarima.for(log(meat[,2]), n.ahead=24, p=1, d=1, q=1) # ARIMA(1,1,1)



## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 6.145421 6.149126 6.157166 6.164478 6.171913 6.179327 6.186744 6.194161  
## [9] 6.201578 6.208994 6.216411 6.223828 6.231245 6.238662 6.246079 6.253496  
## [17] 6.260913 6.268329 6.275746 6.283163 6.290580 6.297997 6.305414 6.312831  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 0.3821013 0.3827999 0.3854244 0.3868762 0.3884925 0.3900731 0.3916521  
## [8] 0.3932240 0.3947898 0.3963493 0.3979027 0.3994501 0.4009916 0.4025271  
## [15] 0.4040568 0.4055807 0.4070989 0.4086115 0.4101185 0.4116200 0.4131160  
## [22] 0.4146066 0.4160919 0.4175719

sarima.for(log(meat[,2]), n.ahead=24, p=1, d=2, q=1) # ARIMA(1,2,1)

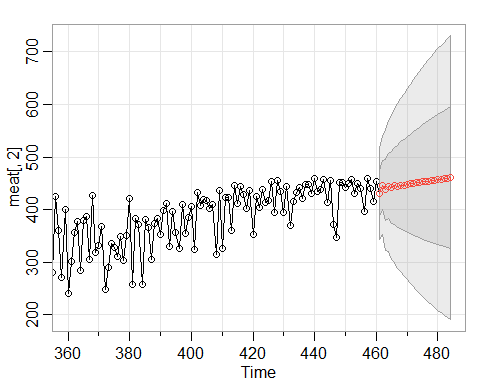


## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 6.077524 6.112868 6.104243 6.121845 6.123803 6.135092 6.140815 6.149859  
## [9] 6.156922 6.165166 6.172706 6.180666 6.188375 6.196234 6.204003 6.211826  
## [17] 6.219617 6.227427 6.235226 6.243032 6.250833 6.258637 6.266439 6.274242  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 0.4526568 0.4884860 0.5980840 0.6481388 0.7174279 0.7679970 0.8227227  
## [8] 0.8702010 0.9176828 0.9617076 1.0047202 1.0457059 1.0855516 1.1239995  
## [15] 1.1614079 1.1977470 1.2331851 1.2677549 1.3015506 1.3346165 1.3670139  
## [22] 1.3987843 1.4299726 1.4606142

#### First Order Differencing

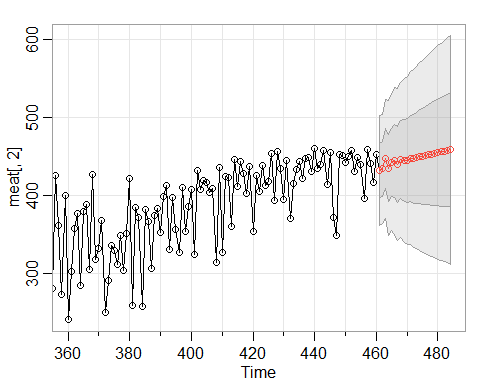
First order differencing doesn’t remove the trend as well as the log transformation.

sarima.for(meat[,2], n.ahead=24, p=1, d=1, q=0) # AR(1)



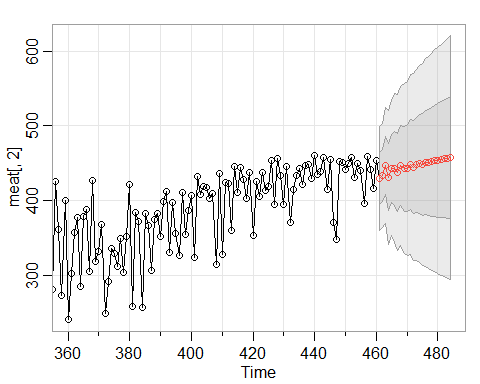
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 430.5963 446.6068 437.7762 445.0094 441.8548 445.4176 444.6365 446.6644  
## [9] 446.8758 448.2619 448.8884 450.0061 450.8061 451.8116 452.6842 453.6428  
## [17] 454.5457 455.4846 456.4003 457.3310 458.2520 459.1792 460.1024 461.0282  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 44.05955 46.72902 57.78481 61.85886 68.65508 72.99707 78.19087  
## [8] 82.38098 86.78220 90.71321 94.63987 98.31072 101.91118 105.34999  
## [15] 108.70428 111.94290 115.09999 118.16674 121.15969 124.07806 126.93087  
## [22] 129.72000 132.45101 135.12646

sarima.for(meat[,2], n.ahead=24, p=2, d=1, q=0) # AR(2)



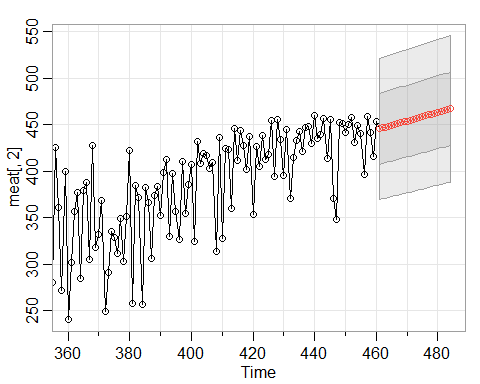
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 432.1068 433.9469 447.0422 434.7901 442.0508 444.3266 440.0327 445.5502  
## [9] 444.8461 444.6909 447.7094 447.1067 448.3498 449.8574 449.9803 451.3799  
## [17] 452.2887 452.9383 454.1520 454.9367 455.8269 456.8657 457.6870 458.6444  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 35.11842 35.14238 38.34325 43.24381 43.55027 47.02525 48.95766 50.03418  
## [9] 52.59781 53.96897 55.47198 57.34767 58.65572 60.21030 61.72976 63.04808  
## [17] 64.50455 65.85515 67.15707 68.49859 69.76132 71.01810 72.26688 73.47071

sarima.for(meat[,2], n.ahead=24, p=3, d=1, q=0) # AR(3)



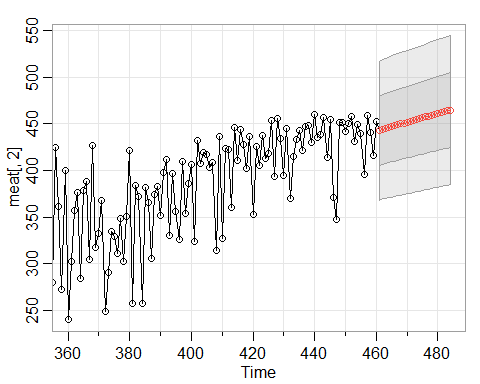
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 428.8385 433.5806 447.2983 430.8380 442.9826 442.9268 437.2548 446.4445  
## [9] 442.3738 443.3599 447.6398 444.6469 447.7680 448.8467 448.0736 450.8388  
## [17] 450.8009 451.5591 453.3279 453.3849 454.7191 455.7590 456.2596 457.5840  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 34.79003 34.82559 38.87355 44.81431 45.18340 49.98104 52.11563 53.39992  
## [9] 56.94564 58.31206 60.28144 62.66169 64.02158 66.10312 67.86342 69.36059  
## [17] 71.24318 72.75253 74.31736 75.96334 77.38999 78.92435 80.39928 81.79253

sarima.for(meat[,2], n.ahead=24, p=0, d=1, q=1) # MA(1)



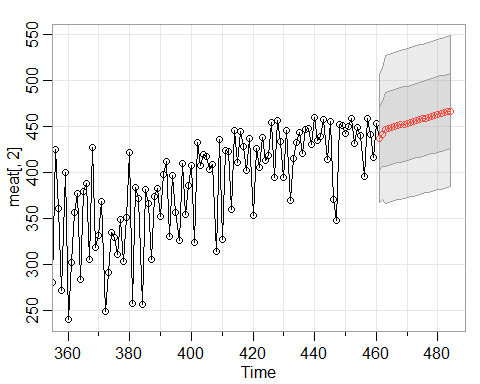
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 445.4219 446.3620 447.3021 448.2423 449.1824 450.1225 451.0626 452.0028  
## [9] 452.9429 453.8830 454.8231 455.7633 456.7034 457.6435 458.5836 459.5238  
## [17] 460.4639 461.4040 462.3441 463.2842 464.2244 465.1645 466.1046 467.0447  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 38.08434 38.14585 38.20728 38.26860 38.32982 38.39095 38.45198 38.51291  
## [9] 38.57375 38.63449 38.69513 38.75568 38.81614 38.87650 38.93677 38.99694  
## [17] 39.05702 39.11701 39.17691 39.23672 39.29643 39.35606 39.41559 39.47504

sarima.for(meat[,2], n.ahead=24, p=0, d=1, q=2) # MA(2)



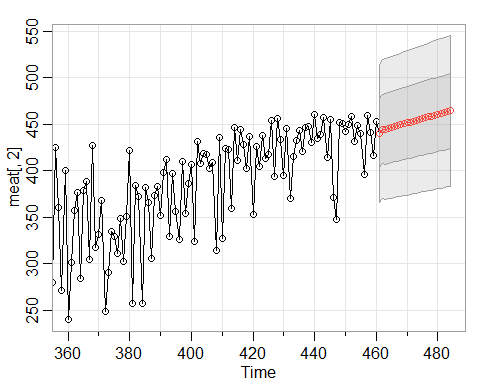
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 442.4076 444.4201 445.3504 446.2806 447.2109 448.1412 449.0715 450.0018  
## [9] 450.9321 451.8623 452.7926 453.7229 454.6532 455.5835 456.5138 457.4440  
## [17] 458.3743 459.3046 460.2349 461.1652 462.0955 463.0257 463.9560 464.8863  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 37.25044 37.42997 37.55067 37.67099 37.79093 37.91048 38.02966 38.14847  
## [9] 38.26691 38.38499 38.50270 38.62005 38.73705 38.85369 38.96999 39.08594  
## [17] 39.20155 39.31681 39.43174 39.54634 39.66060 39.77454 39.88815 40.00144

sarima.for(meat[,2], n.ahead=24, p=0, d=1, q=3) # MA(3)



## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 437.0163 441.5194 446.8398 447.7807 448.7216 449.6626 450.6035 451.5444  
## [9] 452.4853 453.4262 454.3672 455.3081 456.2490 457.1899 458.1308 459.0718  
## [17] 460.0127 460.9536 461.8945 462.8355 463.7764 464.7173 465.6582 466.5991  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 34.57067 35.62381 40.02183 40.07291 40.12392 40.17487 40.22575 40.27656  
## [9] 40.32732 40.37801 40.42863 40.47919 40.52969 40.58013 40.63050 40.68081  
## [17] 40.73106 40.78125 40.83138 40.88144 40.93144 40.98138 41.03126 41.08109

sarima.for(meat[,2], n.ahead=24, p=1, d=1, q=1) # ARIMA(1,1,1)

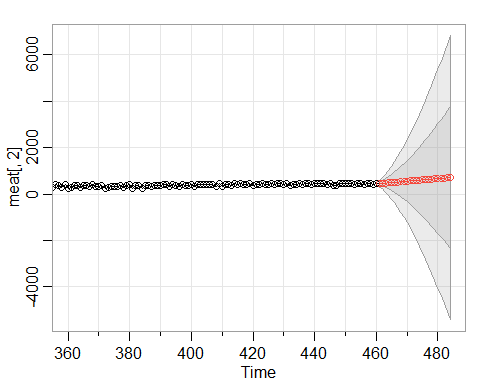


## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 440.1571 444.7538 444.7058 445.8949 446.7544 447.7018 448.6257 449.5559  
## [9] 450.4844 451.4134 452.3422 453.2711 454.1999 455.1288 456.0577 456.9865  
## [17] 457.9154 458.8442 459.7731 460.7020 461.6308 462.5597 463.4886 464.4174  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 36.95797 37.41349 37.82623 37.91130 38.05871 38.18731 38.32019 38.45136  
## [9] 38.58241 38.71293 38.84303 38.97270 39.10193 39.23074 39.35913 39.48710  
## [17] 39.61466 39.74181 39.86855 39.99490 40.12084 40.24639 40.37155 40.49632

#### Second Order Differencing

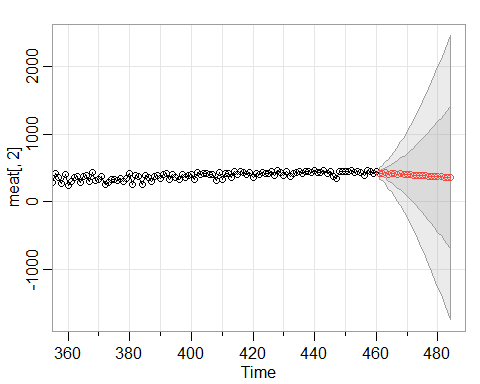
Second order differencing creates too much stationarity for AR models, and not enough for MA models.

sarima.for(meat[,2], n.ahead=24, p=1, d=2, q=0) # AR(1)



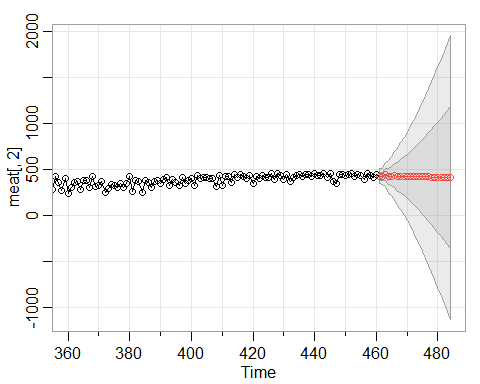
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 445.5991 469.9956 471.6206 489.5533 495.8073 510.4250 519.0531 531.9705  
## [9] 541.8162 553.8617 564.3318 575.9301 586.7205 598.0894 609.0440 620.2954  
## [17] 631.3342 642.5252 653.6073 664.7673 675.8715 687.0158 698.1313 709.2674  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 73.2016 119.1251 193.3559 266.7315 355.1074 446.8999 548.5514  
## [8] 654.9303 768.7649 887.6276 1012.6668 1142.6665 1278.0814 1418.2632  
## [15] 1563.3591 1712.9966 1867.1885 2025.7016 2188.4924 2355.4015 2526.3643  
## [22] 2701.2626 2880.0266 3062.5626

sarima.for(meat[,2], n.ahead=24, p=2, d=2, q=0) # AR(2)



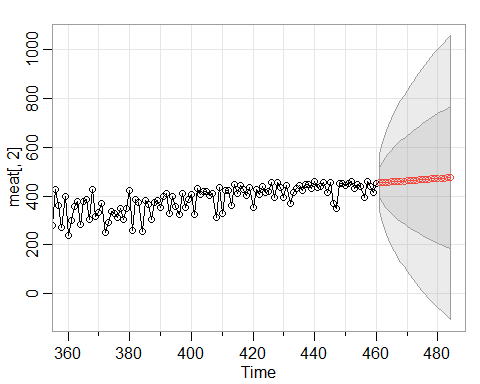
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 415.6001 424.7025 433.1896 405.3577 424.8722 412.2906 403.3298 415.3132  
## [9] 397.3866 401.3318 400.9309 388.6813 395.1863 386.9522 382.7654 385.1202  
## [17] 375.8055 376.3249 373.4706 367.1203 367.9727 362.3229 359.3132 358.0869  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 43.83574 53.75826 77.89869 110.07902 132.55418 169.52112  
## [7] 202.95002 236.99939 279.18600 317.07243 360.01612 405.26635  
## [13] 449.17810 498.10726 546.64922 596.62102 649.66577 702.23103  
## [19] 757.39814 813.86831 870.83373 930.20261 990.09244 1051.33345

sarima.for(meat[,2], n.ahead=24, p=3, d=2, q=0) # AR(3)



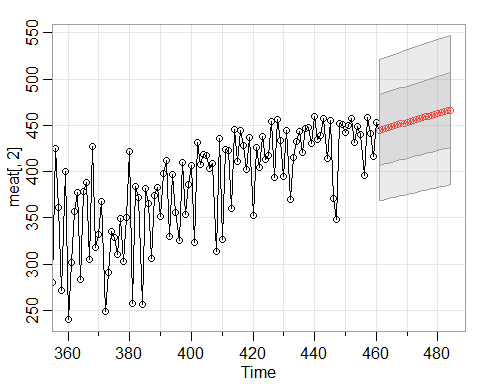
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 428.2490 426.2527 444.0714 423.3151 431.0375 435.0797 422.6093 431.0946  
## [9] 428.3064 422.8492 428.5062 423.7776 422.5172 424.9270 420.7605 421.2264  
## [17] 421.3166 418.4717 419.1512 418.0486 416.3710 416.6199 415.1499 414.1910  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 41.59908 45.75461 64.40265 88.36780 103.63049 130.32710 155.67720  
## [8] 179.48734 210.14711 238.67358 268.71093 302.19467 334.49729 369.08700  
## [15] 405.24868 441.29302 479.41767 518.31798 557.81058 598.91348 640.58242  
## [22] 683.17798 726.97126 771.36211

sarima.for(meat[,2], n.ahead=24, p=0, d=2, q=1) # MA(1)



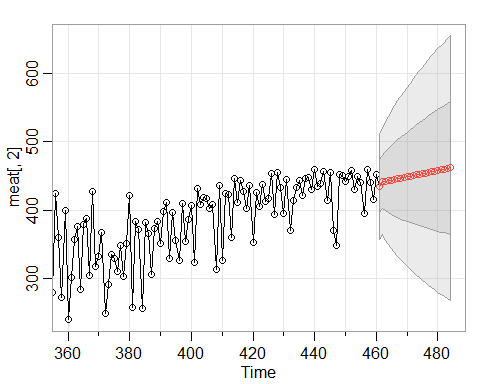
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 453.9412 454.8824 455.8235 456.7647 457.7059 458.6471 459.5882 460.5294  
## [9] 461.4706 462.4118 463.3529 464.2941 465.2353 466.1765 467.1176 468.0588  
## [17] 469.0000 469.9412 470.8824 471.8235 472.7647 473.7059 474.6471 475.5882  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 57.95714 82.05282 100.60270 116.29165 130.15835 142.73489 154.33699  
## [8] 165.17012 175.37683 185.06081 194.30023 203.15575 211.67556 219.89868  
## [15] 227.85724 235.57806 243.08378 250.39375 257.52464 264.49094 271.30532  
## [22] 277.97895 284.52175 290.94254

sarima.for(meat[,2], n.ahead=24, p=0, d=2, q=2) # MA(2)



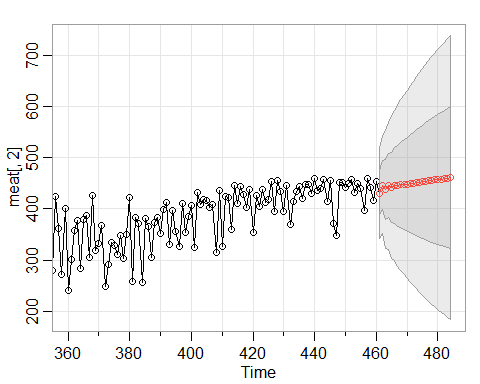
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 445.0402 445.9773 446.9144 447.8515 448.7885 449.7256 450.6627 451.5998  
## [9] 452.5368 453.4739 454.4110 455.3480 456.2851 457.2222 458.1593 459.0963  
## [17] 460.0334 460.9705 461.9075 462.8446 463.7817 464.7188 465.6558 466.5929  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 38.19057 38.28153 38.37268 38.46403 38.55556 38.64729 38.73920 38.83130  
## [9] 38.92358 39.01605 39.10870 39.20153 39.29454 39.38773 39.48110 39.57464  
## [17] 39.66836 39.76226 39.85633 39.95057 40.04498 40.13957 40.23432 40.32924

sarima.for(meat[,2], n.ahead=24, p=0, d=2, q=3) # MA(3)



## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 435.1925 441.4999 442.4327 443.3654 444.2981 445.2309 446.1636 447.0963  
## [9] 448.0291 448.9618 449.8945 450.8273 451.7600 452.6927 453.6254 454.5582  
## [17] 455.4909 456.4236 457.3564 458.2891 459.2218 460.1546 461.0873 462.0200  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 38.62655 38.73849 42.93184 46.76636 50.32412 53.66021 56.81370 59.81346  
## [9] 62.68157 65.43536 68.08868 70.65286 73.13728 75.54980 77.89711 80.18493  
## [17] 82.41822 84.60130 86.73795 88.83153 90.88501 92.90105 94.88205 96.83014

sarima.for(meat[,2], n.ahead=24, p=1, d=2, q=1) # ARIMA(1,1,1)

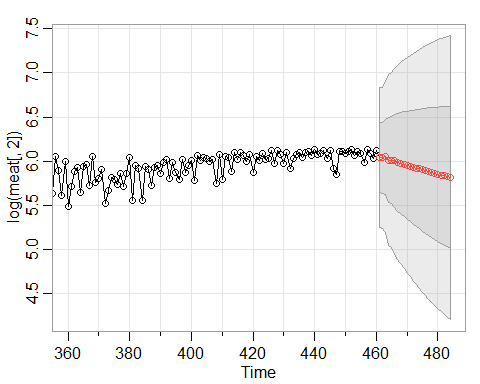


## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 430.6239 446.5987 437.8028 445.0062 441.8756 445.4198 444.6528 446.6703  
## [9] 446.8893 448.2700 448.9003 450.0153 450.8173 451.8214 452.6950 453.6529  
## [17] 454.5563 455.4949 456.4108 457.3414 458.2625 459.1897 460.1129 461.0387  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 44.15583 46.87450 58.01682 62.17728 69.07774 73.52923 78.84322  
## [8] 83.15990 87.69563 91.76749 95.84168 99.66606 103.42599 107.02989  
## [15] 110.55474 113.96916 117.30715 120.55972 123.74329 126.85693 129.90955  
## [22] 132.90289 135.84244 138.73061

#### Differencing for AR(3)

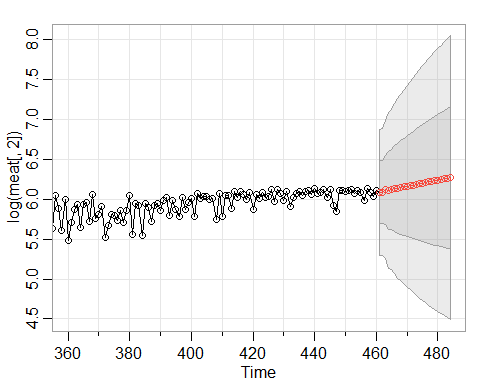
Differencing doesn’t appear to improve the stationarity or forecast. AR(3) on log transformed data seems to work the best of the models investigated so far.

sarima.for(log(meat[,2]), n.ahead=24, p=3, d=0, q=0) # AR(1)



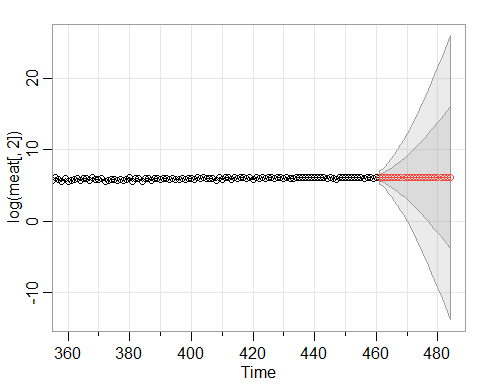
## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 6.040742 6.037823 6.048322 6.013015 6.011672 6.002706 5.984531 5.978316  
## [9] 5.966301 5.953943 5.944915 5.933407 5.922750 5.912791 5.902077 5.891949  
## [17] 5.881923 5.871781 5.861942 5.852140 5.842405 5.832836 5.823329 5.813926  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 0.3962533 0.3988635 0.4296109 0.4838706 0.4960435 0.5258150 0.5518201  
## [8] 0.5689619 0.5919213 0.6110194 0.6282843 0.6465785 0.6628338 0.6785352  
## [15] 0.6939413 0.7083074 0.7222604 0.7357014 0.7485226 0.7609391 0.7728877  
## [22] 0.7843898 0.7955140 0.8062455

sarima.for(log(meat[,2]), n.ahead=24, p=2, d=1, q=0) # AR(2)



## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 6.087362 6.089881 6.119758 6.110641 6.122678 6.134849 6.136733 6.147603  
## [9] 6.155513 6.161707 6.170834 6.178205 6.185711 6.193942 6.201471 6.209269  
## [17] 6.217168 6.224849 6.232673 6.240475 6.248228 6.256035 6.263818 6.271596  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 0.3964774 0.3993333 0.4316786 0.4888112 0.5023673 0.5350940 0.5641289  
## [8] 0.5839195 0.6104447 0.6329895 0.6537941 0.6760528 0.6962456 0.7160695  
## [15] 0.7358001 0.7545453 0.7730374 0.7911391 0.8087072 0.8259933 0.8429052  
## [22] 0.8594591 0.8757311 0.8916912

sarima.for(log(meat[,2]), n.ahead=24, p=3, d=2, q=0) # AR(3)



## $pred  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 6.070420 6.063161 6.099620 6.070562 6.077266 6.090312 6.075016 6.083063  
## [9] 6.086251 6.079613 6.085449 6.085386 6.083251 6.086774 6.086140 6.085991  
## [17] 6.087958 6.087587 6.088151 6.089268 6.089272 6.090004 6.090727 6.091006  
##   
## $se  
## Time Series:  
## Start = 461   
## End = 484   
## Frequency = 1   
## [1] 0.4866403 0.5717353 0.7900766 1.0941554 1.3253591 1.6398403 1.9773964  
## [8] 2.3015711 2.6752790 3.0583251 3.4493292 3.8702452 4.2997892 4.7433105  
## [15] 5.2067221 5.6801366 6.1681343 6.6711743 7.1851720 7.7127900 8.2531255  
## [22] 8.8046887 9.3687068 9.9442497

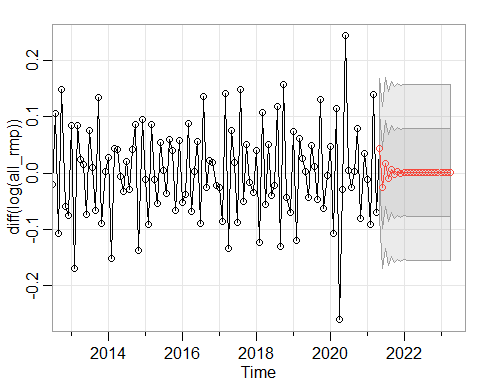
#### Forecasting using data sets provided by Jocelyn Hunyadi

Jocelyn’s code for data loading and cleaning

# Load Data  
FI\_long <- readRDS("FI\_long.rds") %>% group\_by(Time) %>%   
 summarize(Production\_capita = sum(Production\_capita, na.rm = TRUE))  
FI\_long\_rmp <- readRDS("FI\_long\_rmp.rds")  
FI\_long\_big <- readRDS("FI\_long\_big.rds")  
FI\_long\_small <- readRDS("FI\_long\_small.rds")  
  
# Time Series Objects:  
FI\_long\_rm <- FI\_long\_rmp %>%   
 dplyr::select(-Time) %>% arrange(Time2, decreasing = TRUE) %>%   
 filter(Meat\_Type == "TotalRedMeat")   
all\_rmp <-ts(FI\_long\_rm$Production, start=c(1983,1), end=c(2021,4), deltat=1/12)  
  
FI\_long\_p <- FI\_long\_rmp %>% filter(Meat\_Type != "TotalRedMeat") %>%   
 arrange(Time2, decreasing = TRUE)  
all\_p <- ts(FI\_long\_p$Production, start=c(1983,1), end=c(2021,4), deltat=1/12)

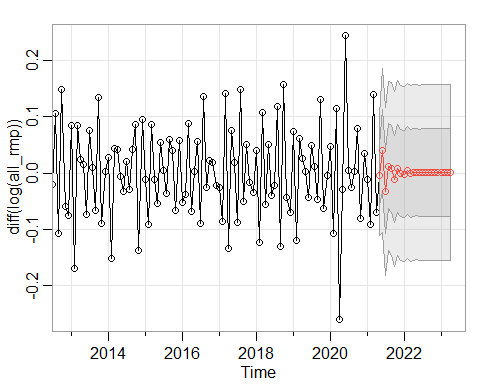
24-Month Forecast: All Red Meat and Poultry

sarima.for(diff(log(all\_rmp)), n.ahead=24, p=1, d=0, q=0) # AR(1)



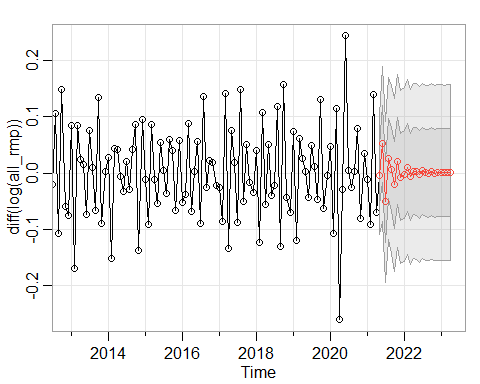
## $pred  
## Jan Feb Mar Apr May  
## 2021 0.0445167266  
## 2022 0.0019720463 0.0005767208 0.0014320013 0.0009077475 0.0012290948  
## 2023 0.0011094093 0.0011054840 0.0011078900 0.0011064152   
## Jun Jul Aug Sep Oct  
## 2021 -0.0255015206 0.0174169534 -0.0088904091 0.0072349854 -0.0026492570  
## 2022 0.0010321214 0.0011528585 0.0010788513 0.0011242148 0.0010964087  
## 2023   
## Nov Dec  
## 2021 0.0034094008 -0.0003043218  
## 2022 0.0011134528 0.0011030054  
## 2023   
##   
## $se  
## Jan Feb Mar Apr May Jun  
## 2021 0.06164717 0.07230668  
## 2022 0.07801742 0.07802105 0.07802242 0.07802293 0.07802313 0.07802320  
## 2023 0.07802324 0.07802324 0.07802324 0.07802324   
## Jul Aug Sep Oct Nov Dec  
## 2021 0.07592590 0.07724191 0.07773060 0.07791342 0.07798200 0.07800775  
## 2022 0.07802323 0.07802324 0.07802324 0.07802324 0.07802324 0.07802324  
## 2023

sarima.for(diff(log(all\_rmp)), n.ahead=24, p=2, d=0, q=0) # AR(2)



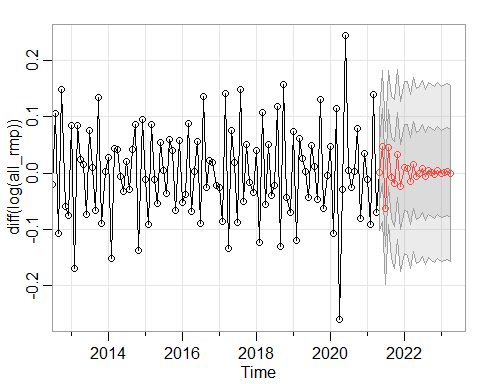
## $pred  
## Jan Feb Mar Apr May  
## 2021 -0.0033952524  
## 2022 -0.0017064994 0.0040555146 -0.0003059913 0.0008142561 0.0019596802  
## 2023 0.0011222038 0.0010325536 0.0010498504 0.0010786309   
## Jun Jul Aug Sep Oct  
## 2021 0.0403268439 -0.0326148905 0.0123019035 0.0075300829 -0.0104486403  
## 2022 0.0003556984 0.0012519107 0.0012306052 0.0008043378 0.0012045053  
## 2023   
## Nov Dec  
## 2021 0.0083556419 0.0001120474  
## 2022 0.0010507950 0.0009922460  
## 2023   
##   
## $se  
## Jan Feb Mar Apr May Jun  
## 2021 0.05355144 0.07254660  
## 2022 0.07787826 0.07790664 0.07790693 0.07791682 0.07792361 0.07792427  
## 2023 0.07792583 0.07792583 0.07792583 0.07792583   
## Jul Aug Sep Oct Nov Dec  
## 2021 0.07476895 0.07517524 0.07688674 0.07765015 0.07767346 0.07775979  
## 2022 0.07792457 0.07792540 0.07792571 0.07792571 0.07792576 0.07792582  
## 2023

sarima.for(diff(log(all\_rmp)), n.ahead=24, p=3, d=0, q=0) # AR(3)



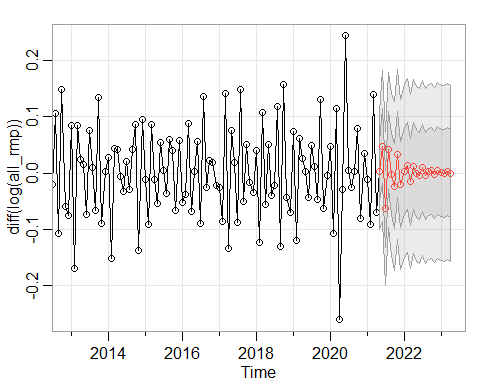
## $pred  
## Jan Feb Mar Apr May  
## 2021 -3.422033e-03  
## 2022 -2.618696e-03 1.026906e-02 -6.623685e-03 3.665345e-03 3.079398e-03  
## 2023 1.542160e-03 4.413936e-04 1.458360e-03 1.043053e-03   
## Jun Jul Aug Sep Oct  
## 2021 5.188732e-02 -5.104473e-02 2.595948e-02 6.744712e-03 -2.073493e-02  
## 2022 -2.746498e-03 3.962194e-03 3.206584e-04 6.948640e-05 2.630476e-03  
## 2023   
## Nov Dec  
## 2021 2.123880e-02 -7.181501e-03  
## 2022 1.017267e-06 1.253958e-03  
## 2023   
##   
## $se  
## Jan Feb Mar Apr May Jun  
## 2021 0.05287449 0.06890385  
## 2022 0.07752796 0.07779466 0.07781108 0.07784518 0.07792433 0.07796049  
## 2023 0.07798998 0.07799058 0.07799058 0.07799087   
## Jul Aug Sep Oct Nov Dec  
## 2021 0.07127896 0.07181401 0.07475723 0.07670204 0.07690746 0.07704879  
## 2022 0.07796141 0.07796889 0.07798169 0.07798643 0.07798644 0.07798797  
## 2023

sarima.for(diff(log(all\_rmp)), n.ahead=24, p=4, d=0, q=0) # AR(4)



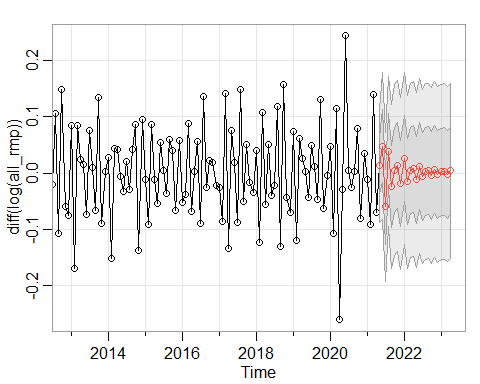
## $pred  
## Jan Feb Mar Apr May  
## 2021 0.0016336539  
## 2022 0.0095764396 0.0085736467 -0.0144690002 0.0150207866 -0.0051988766  
## 2023 -0.0014736610 0.0013640643 0.0024859732 -0.0009606225   
## Jun Jul Aug Sep Oct  
## 2021 0.0478282533 -0.0623662547 0.0453584467 -0.0079415412 -0.0190999424  
## 2022 -0.0012114411 0.0083866454 -0.0064719576 0.0051627590 0.0013458846  
## 2023   
## Nov Dec  
## 2021 0.0327968891 -0.0241511302  
## 2022 -0.0022754658 0.0049979931  
## 2023   
##   
## $se  
## Jan Feb Mar Apr May Jun  
## 2021 0.05204540 0.06689625  
## 2022 0.07634058 0.07715834 0.07744837 0.07744884 0.07757835 0.07779407  
## 2023 0.07801856 0.07803154 0.07804197 0.07804403   
## Jul Aug Sep Oct Nov Dec  
## 2021 0.06802514 0.06850655 0.07176210 0.07490319 0.07564190 0.07567055  
## 2022 0.07789690 0.07790124 0.07792341 0.07797783 0.07801166 0.07801545  
## 2023

sarima.for(diff(log(all\_rmp)), n.ahead=24, p=5, d=0, q=0) # AR(5)



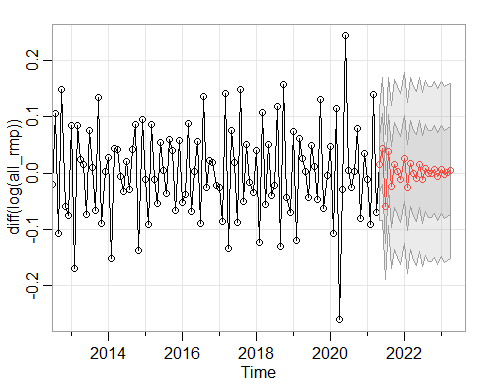
## $pred  
## Jan Feb Mar Apr May  
## 2021 0.0034259591  
## 2022 0.0036748273 0.0129894804 -0.0152946225 0.0122530562 -0.0007592687  
## 2023 0.0003491386 -0.0003126774 0.0031892613 -0.0005641353   
## Jun Jul Aug Sep Oct  
## 2021 0.0481356438 -0.0633071219 0.0414274636 -0.0020387473 -0.0231201337  
## 2022 -0.0047718537 0.0093623304 -0.0048315497 0.0021947956 0.0038678437  
## 2023   
## Nov Dec  
## 2021 0.0331948703 -0.0203025811  
## 2022 -0.0031753821 0.0041288994  
## 2023   
##   
## $se  
## Jan Feb Mar Apr May Jun  
## 2021 0.05193810 0.06712605  
## 2022 0.07651181 0.07726583 0.07743241 0.07744708 0.07763646 0.07783142  
## 2023 0.07800688 0.07802041 0.07802443 0.07802447   
## Jul Aug Sep Oct Nov Dec  
## 2021 0.06843208 0.06869217 0.07201984 0.07513831 0.07567189 0.07574972  
## 2022 0.07788006 0.07788252 0.07792955 0.07798088 0.07799489 0.07799526  
## 2023

sarima.for(diff(log(all\_rmp)), n.ahead=24, p=6, d=0, q=0) # AR(6)



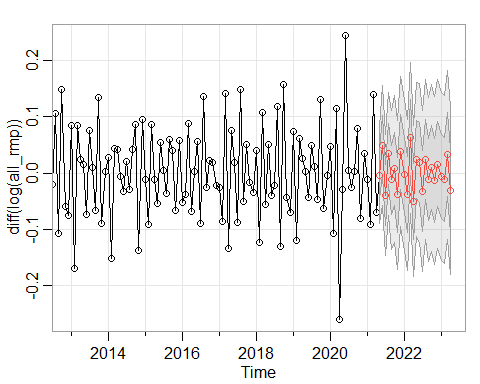
## $pred  
## Jan Feb Mar Apr May  
## 2021 0.012709705  
## 2022 0.026154362 -0.015335259 0.004754233 0.007425063 -0.011937521  
## 2023 0.002101034 0.002340529 -0.001694503 0.003905617   
## Jun Jul Aug Sep Oct  
## 2021 0.047955163 -0.060085985 0.038926691 -0.024083192 0.005155970  
## 2022 0.012569279 -0.005962300 0.002897919 0.004197088 -0.004292060  
## 2023   
## Nov Dec  
## 2021 0.013057541 -0.018684206  
## 2022 0.006603148 -0.002894404  
## 2023   
##   
## $se  
## Jan Feb Mar Apr May Jun  
## 2021 0.04995407 0.06512118  
## 2022 0.07639100 0.07668117 0.07711008 0.07750239 0.07753779 0.07755276  
## 2023 0.07798835 0.07800621 0.07800890 0.07800981   
## Jul Aug Sep Oct Nov Dec  
## 2021 0.06607441 0.06676630 0.07303487 0.07569913 0.07614398 0.07619876  
## 2022 0.07768796 0.07786247 0.07791984 0.07792682 0.07793247 0.07796087  
## 2023

sarima.for(diff(log(all\_rmp)), n.ahead=24, p=7, d=0, q=0) # AR(7)



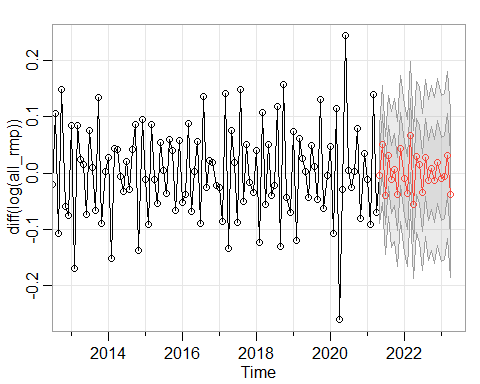
## $pred  
## Jan Feb Mar Apr May  
## 2021 0.0144792298  
## 2022 0.0261214815 -0.0259534336 0.0165558428 -0.0005379099 -0.0093926766  
## 2023 0.0071044750 -0.0020295310 0.0007009912 0.0040778742   
## Jun Jul Aug Sep Oct  
## 2021 0.0430133376 -0.0596536426 0.0382276640 -0.0244977389 0.0149241207  
## 2022 0.0160943647 -0.0112829252 0.0087108769 -0.0012625074 -0.0013464494  
## 2023   
## Nov Dec  
## 2021 0.0023418155 -0.0110827420  
## 2022 0.0065271183 -0.0059276664  
## 2023   
##   
## $se  
## Jan Feb Mar Apr May Jun  
## 2021 0.04963994 0.06374859  
## 2022 0.07594250 0.07610442 0.07637735 0.07695441 0.07724700 0.07726045  
## 2023 0.07789575 0.07793231 0.07796205 0.07797116   
## Jul Aug Sep Oct Nov Dec  
## 2021 0.06456782 0.06540205 0.07217628 0.07549277 0.07583328 0.07586574  
## 2022 0.07733470 0.07760344 0.07779954 0.07785203 0.07785370 0.07786533  
## 2023

sarima.for(diff(log(all\_rmp)), n.ahead=24, p=8, d=0, q=0) # AR(8)



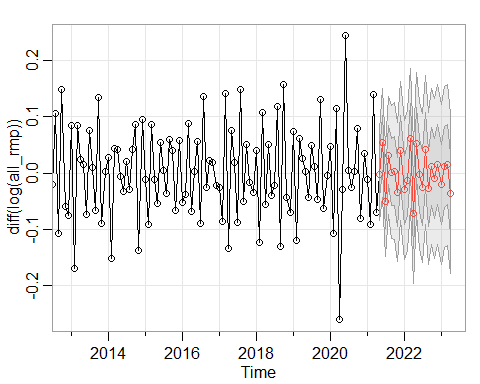
## $pred  
## Jan Feb Mar Apr May  
## 2021 -0.003377495  
## 2022 -0.002956033 -0.038124447 0.062869148 -0.049639614 0.023986968  
## 2023 -0.005453579 -0.010807799 0.032482361 -0.031152803   
## Jun Jul Aug Sep Oct  
## 2021 0.048598241 -0.038908726 0.034145947 -0.012012800 0.007464844  
## 2022 0.018747341 -0.031720485 0.023588757 -0.012020141 0.009712295  
## 2023   
## Nov Dec  
## 2021 -0.038429834 0.038275368  
## 2022 -0.013850272 0.016222982  
## 2023   
##   
## $se  
## Jan Feb Mar Apr May Jun  
## 2021 0.04277541 0.05345157  
## 2022 0.06693803 0.06712837 0.06715099 0.06739466 0.06867404 0.07055490  
## 2023 0.07472638 0.07474686 0.07476404 0.07479595   
## Jul Aug Sep Oct Nov Dec  
## 2021 0.05345982 0.05480072 0.06156685 0.06510698 0.06632253 0.06651693  
## 2022 0.07106048 0.07108378 0.07279704 0.07421526 0.07462016 0.07465879  
## 2023

sarima.for(diff(log(all\_rmp)), n.ahead=24, p=9, d=0, q=0) # AR(9)



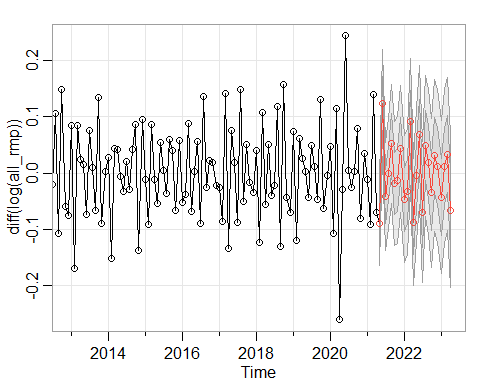
## $pred  
## Jan Feb Mar Apr May  
## 2021 -0.004144047  
## 2022 -0.009603886 -0.035344606 0.066613027 -0.055455117 0.029147602  
## 2023 -0.010108908 -0.006509592 0.032043084 -0.037243772   
## Jun Jul Aug Sep Oct  
## 2021 0.051411946 -0.039376235 0.030873907 -0.011278960 0.005913052  
## 2022 0.014766959 -0.034034277 0.027523509 -0.012864458 0.008849245  
## 2023   
## Nov Dec  
## 2021 -0.037187846 0.043515009  
## 2022 -0.013738006 0.019774768  
## 2023   
##   
## $se  
## Jan Feb Mar Apr May Jun  
## 2021 0.04261248 0.05213641  
## 2022 0.06544148 0.06560616 0.06560760 0.06583654 0.06714046 0.06916044  
## 2023 0.07381766 0.07385091 0.07386657 0.07389505   
## Jul Aug Sep Oct Nov Dec  
## 2021 0.05214079 0.05372009 0.06040715 0.06361034 0.06472095 0.06490318  
## 2022 0.06976601 0.06977190 0.07139339 0.07318289 0.07370862 0.07375598  
## 2023

sarima.for(diff(log(all\_rmp)), n.ahead=24, p=10, d=0, q=0) # AR(10)



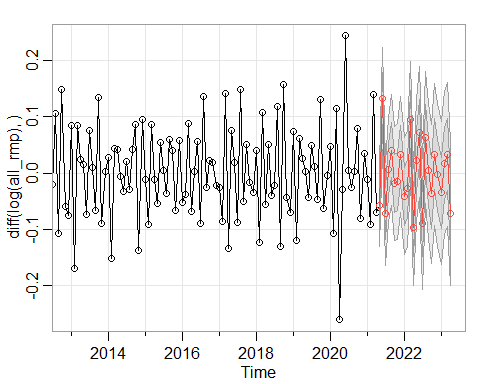
## $pred  
## Jan Feb Mar Apr May  
## 2021 -0.003214204  
## 2022 -0.029908061 -0.012528980 0.060955046 -0.072413034 0.052826685  
## 2023 -0.019558444 0.010928188 0.014665144 -0.035815543   
## Jun Jul Aug Sep Oct  
## 2021 0.053596959 -0.049664911 0.031812847 0.001999276 0.003712382  
## 2022 -0.003231638 -0.026342817 0.041582681 -0.027723260 0.012245527  
## 2023   
## Nov Dec  
## 2021 -0.033593002 0.040120059  
## 2022 -0.009437482 0.014950441  
## 2023   
##   
## $se  
## Jan Feb Mar Apr May Jun  
## 2021 0.04034426 0.04873141  
## 2022 0.06205798 0.06207557 0.06211329 0.06211532 0.06318881 0.06538860  
## 2023 0.07100045 0.07107439 0.07121417 0.07122292   
## Jul Aug Sep Oct Nov Dec  
## 2021 0.04938090 0.05227327 0.05902199 0.06107407 0.06137649 0.06173277  
## 2022 0.06623649 0.06625256 0.06758011 0.06932771 0.07075954 0.07092844  
## 2023

sarima.for(diff(log(all\_rmp)), n.ahead=24, p=11, d=0, q=0) # AR(11)



## $pred  
## Jan Feb Mar Apr May  
## 2021 -0.0895919925  
## 2022 -0.0464386914 -0.0324664941 0.0912252577 -0.0874125685 -0.0035869123  
## 2023 -0.0425792829 0.0119025126 0.0337105143 -0.0669552464   
## Jun Jul Aug Sep Oct  
## 2021 0.1242942873 -0.0414745408 -0.0006750937 0.0522845450 -0.0182037947  
## 2022 0.0680365777 -0.0696298133 0.0488527896 0.0196979192 -0.0350515553  
## 2023   
## Nov Dec  
## 2021 -0.0125354365 0.0431039374  
## 2022 0.0313617190 0.0124342074  
## 2023   
##   
## $se  
## Jan Feb Mar Apr May Jun  
## 2021 0.03718665 0.04776889  
## 2022 0.05630966 0.05631574 0.05634288 0.05646373 0.05911515 0.06198914  
## 2023 0.06791901 0.06809887 0.06813828 0.06820734   
## Jul Aug Sep Oct Nov Dec  
## 2021 0.04783583 0.04863614 0.05286985 0.05521460 0.05612290 0.05612342  
## 2022 0.06262954 0.06264689 0.06423810 0.06614260 0.06768132 0.06773557  
## 2023

sarima.for(diff(log(all\_rmp), ), n.ahead=24, p=12, d=0, q=0) # AR(12)

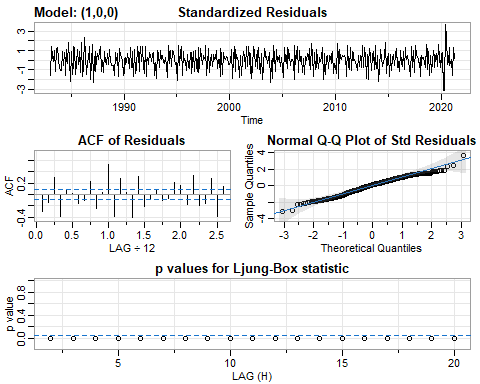


## $pred  
## Jan Feb Mar Apr May  
## 2021 -0.057279953  
## 2022 -0.040809497 -0.027404471 0.094758735 -0.096233761 0.022150614  
## 2023 -0.034882425 0.017760964 0.032953828 -0.071755084   
## Jun Jul Aug Sep Oct  
## 2021 0.132342247 -0.071897230 0.006753334 0.040872153 -0.018446909  
## 2022 0.073052388 -0.090260435 0.063142355 0.004154285 -0.036436358  
## 2023   
## Nov Dec  
## 2021 -0.015294621 0.033735385  
## 2022 0.033555488 -0.002759931  
## 2023   
##   
## $se  
## Jan Feb Mar Apr May Jun  
## 2021 0.03667642 0.04557007  
## 2022 0.05162028 0.05162142 0.05162879 0.05185977 0.05431121 0.05778279  
## 2023 0.06348096 0.06368164 0.06377308 0.06388253   
## Jul Aug Sep Oct Nov Dec  
## 2021 0.04564367 0.04613179 0.04926954 0.05078371 0.05153252 0.05153275  
## 2022 0.05843953 0.05844253 0.05995576 0.06175625 0.06325433 0.06332524  
## 2023

Checking the Model Conditions: All Red Meat and Poultry

sarima(diff(log(all\_rmp)), p=1, d=0, q=0)# AR(1)

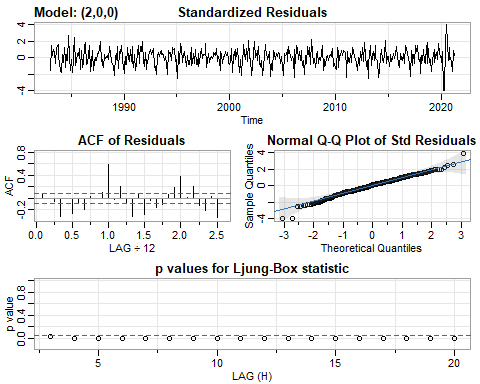
## initial value -2.553084   
## iter 2 value -2.787849  
## iter 3 value -2.787860  
## iter 4 value -2.787861  
## iter 5 value -2.787861  
## iter 5 value -2.787861  
## iter 5 value -2.787861  
## final value -2.787861   
## converged  
## initial value -2.785810   
## iter 2 value -2.785814  
## iter 3 value -2.785815  
## iter 3 value -2.785815  
## iter 3 value -2.785815  
## final value -2.785815   
## converged



## $fit  
##   
## Call:  
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),   
## xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,   
## optim.control = list(trace = trc, REPORT = 1, reltol = tol))  
##   
## Coefficients:  
## ar1 xmean  
## -0.613 0.0011  
## s.e. 0.037 0.0018  
##   
## sigma^2 estimated as 0.0038: log likelihood = 627.4, aic = -1248.79  
##   
## $degrees\_of\_freedom  
## [1] 457  
##   
## $ttable  
## Estimate SE t.value p.value  
## ar1 -0.6130 0.0370 -16.5785 0.0000  
## xmean 0.0011 0.0018 0.6198 0.5357  
##   
## $AIC  
## [1] -2.72068  
##   
## $AICc  
## [1] -2.720623  
##   
## $BIC  
## [1] -2.693693

sarima(diff(log(all\_rmp)), p=2, d=0, q=0) # AR(2)

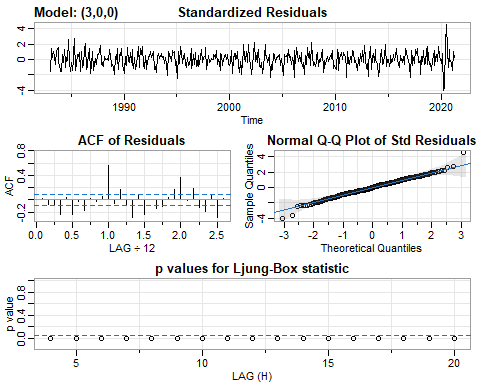
## initial value -2.556648   
## iter 2 value -2.755918  
## iter 3 value -2.915015  
## iter 4 value -2.915675  
## iter 5 value -2.929618  
## iter 6 value -2.929683  
## iter 7 value -2.929722  
## iter 8 value -2.929724  
## iter 9 value -2.929725  
## iter 10 value -2.929725  
## iter 10 value -2.929725  
## iter 10 value -2.929725  
## final value -2.929725   
## converged  
## initial value -2.925972   
## iter 2 value -2.925978  
## iter 3 value -2.925986  
## iter 4 value -2.925986  
## iter 5 value -2.925986  
## iter 5 value -2.925986  
## iter 5 value -2.925986  
## final value -2.925986   
## converged



## $fit  
##   
## Call:  
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),   
## xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,   
## optim.control = list(trace = trc, REPORT = 1, reltol = tol))  
##   
## Coefficients:  
## ar1 ar2 xmean  
## -0.9139 -0.4974 0.0011  
## s.e. 0.0405 0.0408 0.0010  
##   
## sigma^2 estimated as 0.002868: log likelihood = 691.73, aic = -1375.47  
##   
## $degrees\_of\_freedom  
## [1] 456  
##   
## $ttable  
## Estimate SE t.value p.value  
## ar1 -0.9139 0.0405 -22.5581 0.0000  
## ar2 -0.4974 0.0408 -12.2024 0.0000  
## xmean 0.0011 0.0010 1.0180 0.3092  
##   
## $AIC  
## [1] -2.996665  
##   
## $AICc  
## [1] -2.99655  
##   
## $BIC  
## [1] -2.960682

sarima(diff(log(all\_rmp)), p=3, d=0, q=0) # AR(3)

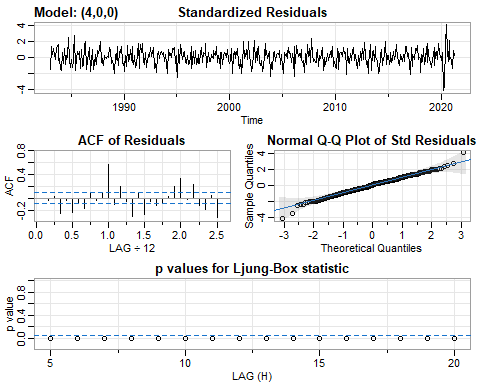
## initial value -2.556431   
## iter 2 value -2.884880  
## iter 3 value -2.904106  
## iter 4 value -2.929416  
## iter 5 value -2.939858  
## iter 6 value -2.941141  
## iter 7 value -2.941450  
## iter 8 value -2.941451  
## iter 8 value -2.941451  
## iter 8 value -2.941451  
## final value -2.941451   
## converged  
## initial value -2.938608   
## iter 2 value -2.938614  
## iter 3 value -2.938621  
## iter 4 value -2.938621  
## iter 5 value -2.938621  
## iter 5 value -2.938621  
## iter 5 value -2.938621  
## final value -2.938621   
## converged



## $fit  
##   
## Call:  
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),   
## xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,   
## optim.control = list(trace = trc, REPORT = 1, reltol = tol))  
##   
## Coefficients:  
## ar1 ar2 ar3 xmean  
## -0.8356 -0.3531 0.1588 0.0011  
## s.e. 0.0461 0.0582 0.0463 0.0012  
##   
## sigma^2 estimated as 0.002796: log likelihood = 697.53, aic = -1385.07  
##   
## $degrees\_of\_freedom  
## [1] 455  
##   
## $ttable  
## Estimate SE t.value p.value  
## ar1 -0.8356 0.0461 -18.1438 0.0000  
## ar2 -0.3531 0.0582 -6.0651 0.0000  
## ar3 0.1588 0.0463 3.4286 0.0007  
## xmean 0.0011 0.0012 0.8787 0.3800  
##   
## $AIC  
## [1] -3.017578  
##   
## $AICc  
## [1] -3.017386  
##   
## $BIC  
## [1] -2.972599

sarima(diff(log(all\_rmp)), p=4, d=0, q=0) # AR(4)

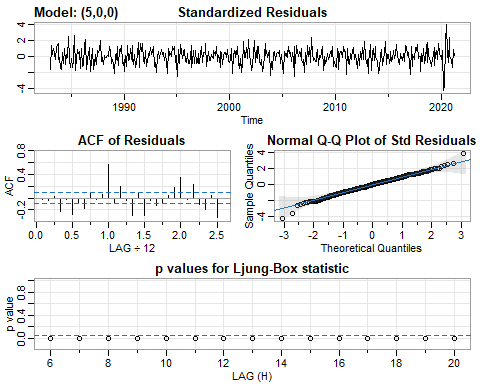
## initial value -2.555585   
## iter 2 value -2.758273  
## iter 3 value -2.904481  
## iter 4 value -2.937520  
## iter 5 value -2.947497  
## iter 6 value -2.954731  
## iter 7 value -2.956907  
## iter 8 value -2.957322  
## iter 9 value -2.957327  
## iter 10 value -2.957336  
## iter 11 value -2.957338  
## iter 12 value -2.957338  
## iter 12 value -2.957338  
## final value -2.957338   
## converged  
## initial value -2.954266   
## iter 2 value -2.954279  
## iter 3 value -2.954284  
## iter 4 value -2.954284  
## iter 5 value -2.954284  
## iter 5 value -2.954284  
## iter 5 value -2.954284  
## final value -2.954284   
## converged



## $fit  
##   
## Call:  
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),   
## xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,   
## optim.control = list(trace = trc, REPORT = 1, reltol = tol))  
##   
## Coefficients:  
## ar1 ar2 ar3 ar4 xmean  
## -0.8075 -0.4150 0.0122 -0.1766 0.0011  
## s.e. 0.0459 0.0596 0.0596 0.0462 0.0010  
##   
## sigma^2 estimated as 0.002709: log likelihood = 704.72, aic = -1397.45  
##   
## $degrees\_of\_freedom  
## [1] 454  
##   
## $ttable  
## Estimate SE t.value p.value  
## ar1 -0.8075 0.0459 -17.5835 0.0000  
## ar2 -0.4150 0.0596 -6.9646 0.0000  
## ar3 0.0122 0.0596 0.2039 0.8385  
## ar4 -0.1766 0.0462 -3.8226 0.0002  
## xmean 0.0011 0.0010 1.0314 0.3029  
##   
## $AIC  
## [1] -3.044548  
##   
## $AICc  
## [1] -3.044259  
##   
## $BIC  
## [1] -2.990573

sarima(diff(log(all\_rmp)), p=5, d=0, q=0) # AR(5)

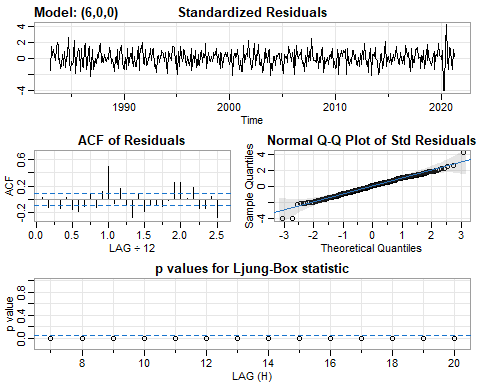
## initial value -2.554798   
## iter 2 value -2.577322  
## iter 3 value -2.900936  
## iter 4 value -2.921756  
## iter 5 value -2.932988  
## iter 6 value -2.949687  
## iter 7 value -2.955794  
## iter 8 value -2.957711  
## iter 9 value -2.958307  
## iter 10 value -2.958445  
## iter 11 value -2.958452  
## iter 12 value -2.958456  
## iter 12 value -2.958456  
## iter 12 value -2.958456  
## final value -2.958456   
## converged  
## initial value -2.956306   
## iter 2 value -2.956320  
## iter 3 value -2.956326  
## iter 4 value -2.956326  
## iter 5 value -2.956327  
## iter 5 value -2.956327  
## iter 5 value -2.956327  
## final value -2.956327   
## converged



## $fit  
##   
## Call:  
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),   
## xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,   
## optim.control = list(trace = trc, REPORT = 1, reltol = tol))  
##   
## Coefficients:  
## ar1 ar2 ar3 ar4 ar5 xmean  
## -0.8188 -0.4142 -0.0144 -0.2282 -0.0642 0.001  
## s.e. 0.0466 0.0595 0.0626 0.0595 0.0468 0.001  
##   
## sigma^2 estimated as 0.002698: log likelihood = 705.66, aic = -1397.32  
##   
## $degrees\_of\_freedom  
## [1] 453  
##   
## $ttable  
## Estimate SE t.value p.value  
## ar1 -0.8188 0.0466 -17.5849 0.0000  
## ar2 -0.4142 0.0595 -6.9642 0.0000  
## ar3 -0.0144 0.0626 -0.2298 0.8184  
## ar4 -0.2282 0.0595 -3.8341 0.0001  
## ar5 -0.0642 0.0468 -1.3706 0.1712  
## xmean 0.0010 0.0010 1.0935 0.2747  
##   
## $AIC  
## [1] -3.044275  
##   
## $AICc  
## [1] -3.04387  
##   
## $BIC  
## [1] -2.981305

sarima(diff(log(all\_rmp)), p=6, d=0, q=0) # AR(6)

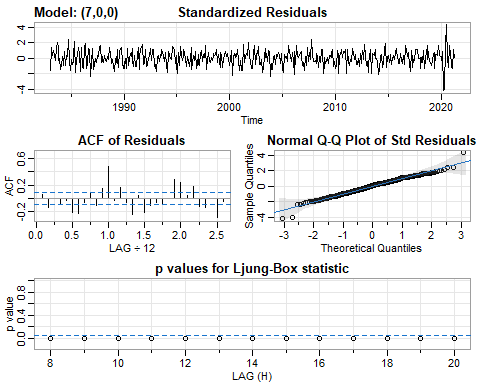
## initial value -2.555352   
## iter 2 value -2.707763  
## iter 3 value -2.874728  
## iter 4 value -2.890036  
## iter 5 value -2.949570  
## iter 6 value -2.974339  
## iter 7 value -2.989233  
## iter 8 value -2.994296  
## iter 9 value -2.994595  
## iter 10 value -2.995595  
## iter 11 value -2.996427  
## iter 12 value -2.996439  
## iter 13 value -2.996441  
## iter 14 value -2.996441  
## iter 14 value -2.996441  
## iter 14 value -2.996441  
## final value -2.996441   
## converged  
## initial value -2.994739   
## iter 2 value -2.994752  
## iter 3 value -2.994756  
## iter 4 value -2.994761  
## iter 5 value -2.994768  
## iter 6 value -2.994769  
## iter 7 value -2.994769  
## iter 7 value -2.994769  
## iter 7 value -2.994769  
## final value -2.994769   
## converged



## $fit  
##   
## Call:  
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),   
## xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,   
## optim.control = list(trace = trc, REPORT = 1, reltol = tol))  
##   
## Coefficients:  
## ar1 ar2 ar3 ar4 ar5 ar6 xmean  
## -0.8363 -0.4756 -0.0186 -0.3412 -0.2870 -0.2732 1e-03  
## s.e. 0.0449 0.0581 0.0602 0.0602 0.0581 0.0451 7e-04  
##   
## sigma^2 estimated as 0.002495: log likelihood = 723.31, aic = -1430.61  
##   
## $degrees\_of\_freedom  
## [1] 452  
##   
## $ttable  
## Estimate SE t.value p.value  
## ar1 -0.8363 0.0449 -18.6433 0.0000  
## ar2 -0.4756 0.0581 -8.1889 0.0000  
## ar3 -0.0186 0.0602 -0.3085 0.7578  
## ar4 -0.3412 0.0602 -5.6701 0.0000  
## ar5 -0.2870 0.0581 -4.9397 0.0000  
## ar6 -0.2732 0.0451 -6.0628 0.0000  
## xmean 0.0010 0.0007 1.4387 0.1509  
##   
## $AIC  
## [1] -3.116802  
##   
## $AICc  
## [1] -3.116261  
##   
## $BIC  
## [1] -3.044836

sarima(diff(log(all\_rmp)), p=7, d=0, q=0) # AR(7)

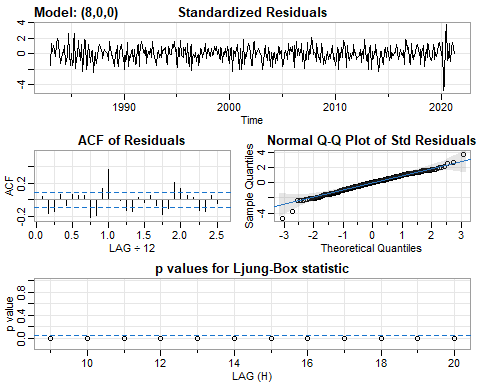
## initial value -2.557589   
## iter 2 value -2.710865  
## iter 3 value -2.900994  
## iter 4 value -2.928243  
## iter 5 value -2.931114  
## iter 6 value -2.987438  
## iter 7 value -2.995125  
## iter 8 value -3.001252  
## iter 9 value -3.003258  
## iter 10 value -3.003472  
## iter 11 value -3.003528  
## iter 12 value -3.003547  
## iter 13 value -3.003569  
## iter 14 value -3.003570  
## iter 15 value -3.003570  
## iter 15 value -3.003570  
## iter 15 value -3.003570  
## final value -3.003570   
## converged  
## initial value -3.000871   
## iter 2 value -3.000917  
## iter 3 value -3.000971  
## iter 4 value -3.000973  
## iter 5 value -3.000979  
## iter 6 value -3.000981  
## iter 7 value -3.000981  
## iter 7 value -3.000981  
## iter 7 value -3.000981  
## final value -3.000981   
## converged



## $fit  
##   
## Call:  
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),   
## xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,   
## optim.control = list(trace = trc, REPORT = 1, reltol = tol))  
##   
## Coefficients:  
## ar1 ar2 ar3 ar4 ar5 ar6 ar7 xmean  
## -0.8057 -0.4427 0.0195 -0.3388 -0.2339 -0.1802 0.1117 0.0011  
## s.e. 0.0464 0.0593 0.0619 0.0598 0.0619 0.0593 0.0466 0.0008  
##   
## sigma^2 estimated as 0.002464: log likelihood = 726.16, aic = -1434.32  
##   
## $degrees\_of\_freedom  
## [1] 451  
##   
## $ttable  
## Estimate SE t.value p.value  
## ar1 -0.8057 0.0464 -17.3794 0.0000  
## ar2 -0.4427 0.0593 -7.4625 0.0000  
## ar3 0.0195 0.0619 0.3150 0.7529  
## ar4 -0.3388 0.0598 -5.6680 0.0000  
## ar5 -0.2339 0.0619 -3.7807 0.0002  
## ar6 -0.1802 0.0593 -3.0414 0.0025  
## ar7 0.1117 0.0466 2.3965 0.0170  
## xmean 0.0011 0.0008 1.2978 0.1950  
##   
## $AIC  
## [1] -3.12487  
##   
## $AICc  
## [1] -3.124173  
##   
## $BIC  
## [1] -3.043908

sarima(diff(log(all\_rmp)), p=8, d=0, q=0) # AR(8)

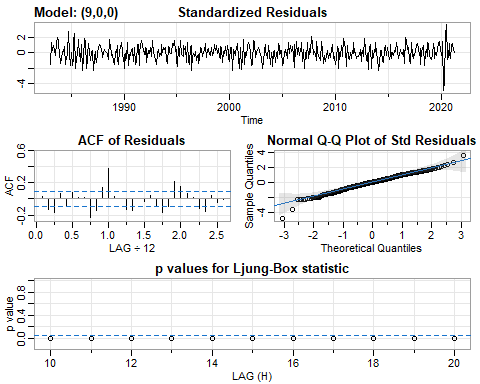
## initial value -2.556485   
## iter 2 value -2.635598  
## iter 3 value -3.026915  
## iter 4 value -3.048763  
## iter 5 value -3.072725  
## iter 6 value -3.119837  
## iter 7 value -3.128842  
## iter 8 value -3.152746  
## iter 9 value -3.153003  
## iter 10 value -3.155731  
## iter 11 value -3.155852  
## iter 12 value -3.155868  
## iter 13 value -3.155880  
## iter 14 value -3.155881  
## iter 14 value -3.155881  
## iter 14 value -3.155881  
## final value -3.155881   
## converged  
## initial value -3.147151   
## iter 2 value -3.147172  
## iter 3 value -3.147195  
## iter 4 value -3.147210  
## iter 5 value -3.147225  
## iter 6 value -3.147230  
## iter 7 value -3.147234  
## iter 8 value -3.147236  
## iter 9 value -3.147236  
## iter 10 value -3.147236  
## iter 10 value -3.147236  
## iter 10 value -3.147236  
## final value -3.147236   
## converged



## $fit  
##   
## Call:  
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),   
## xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,   
## optim.control = list(trace = trc, REPORT = 1, reltol = tol))  
##   
## Coefficients:  
## ar1 ar2 ar3 ar4 ar5 ar6 ar7 ar8  
## -0.7493 -0.5395 -0.1061 -0.5126 -0.2263 -0.4068 -0.2982 -0.5077  
## s.e. 0.0401 0.0516 0.0542 0.0535 0.0533 0.0541 0.0517 0.0404  
## xmean  
## 1e-03  
## s.e. 5e-04  
##   
## sigma^2 estimated as 0.00183: log likelihood = 793.29, aic = -1566.58  
##   
## $degrees\_of\_freedom  
## [1] 450  
##   
## $ttable  
## Estimate SE t.value p.value  
## ar1 -0.7493 0.0401 -18.6776 0.0000  
## ar2 -0.5395 0.0516 -10.4501 0.0000  
## ar3 -0.1061 0.0542 -1.9568 0.0510  
## ar4 -0.5126 0.0535 -9.5856 0.0000  
## ar5 -0.2263 0.0533 -4.2482 0.0000  
## ar6 -0.4068 0.0541 -7.5257 0.0000  
## ar7 -0.2982 0.0517 -5.7654 0.0000  
## ar8 -0.5077 0.0404 -12.5733 0.0000  
## xmean 0.0010 0.0005 2.1285 0.0338  
##   
## $AIC  
## [1] -3.413023  
##   
## $AICc  
## [1] -3.412149  
##   
## $BIC  
## [1] -3.323065

sarima(diff(log(all\_rmp)), p=9, d=0, q=0) # AR(9)

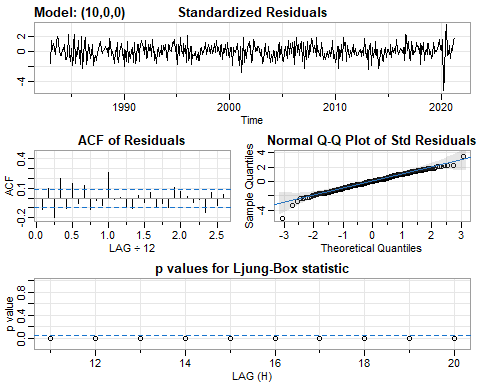
## initial value -2.555453   
## iter 2 value -2.723946  
## iter 3 value -2.908137  
## iter 4 value -2.935821  
## iter 5 value -3.064612  
## iter 6 value -3.089706  
## iter 7 value -3.116873  
## iter 8 value -3.134916  
## iter 9 value -3.148688  
## iter 10 value -3.160605  
## iter 11 value -3.161086  
## iter 12 value -3.161264  
## iter 13 value -3.161524  
## iter 14 value -3.161547  
## iter 15 value -3.161548  
## iter 16 value -3.161548  
## iter 16 value -3.161548  
## iter 16 value -3.161548  
## final value -3.161548   
## converged  
## initial value -3.150780   
## iter 2 value -3.150828  
## iter 3 value -3.150899  
## iter 4 value -3.150930  
## iter 5 value -3.150963  
## iter 6 value -3.150977  
## iter 7 value -3.150982  
## iter 8 value -3.150983  
## iter 9 value -3.150984  
## iter 10 value -3.150984  
## iter 11 value -3.150984  
## iter 11 value -3.150984  
## iter 11 value -3.150984  
## final value -3.150984   
## converged



## $fit  
##   
## Call:  
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),   
## xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,   
## optim.control = list(trace = trc, REPORT = 1, reltol = tol))  
##   
## Coefficients:  
## ar1 ar2 ar3 ar4 ar5 ar6 ar7 ar8  
## -0.7049 -0.5128 -0.0693 -0.4914 -0.1812 -0.3974 -0.2505 -0.4416  
## s.e. 0.0465 0.0534 0.0576 0.0545 0.0584 0.0541 0.0576 0.0537  
## ar9 xmean  
## 0.0874 1e-03  
## s.e. 0.0470 5e-04  
##   
## sigma^2 estimated as 0.001816: log likelihood = 795.01, aic = -1568.02  
##   
## $degrees\_of\_freedom  
## [1] 449  
##   
## $ttable  
## Estimate SE t.value p.value  
## ar1 -0.7049 0.0465 -15.1519 0.0000  
## ar2 -0.5128 0.0534 -9.6046 0.0000  
## ar3 -0.0693 0.0576 -1.2030 0.2296  
## ar4 -0.4914 0.0545 -9.0188 0.0000  
## ar5 -0.1812 0.0584 -3.1045 0.0020  
## ar6 -0.3974 0.0541 -7.3430 0.0000  
## ar7 -0.2505 0.0576 -4.3485 0.0000  
## ar8 -0.4416 0.0537 -8.2167 0.0000  
## ar9 0.0874 0.0470 1.8588 0.0637  
## xmean 0.0010 0.0005 1.9676 0.0497  
##   
## $AIC  
## [1] -3.416161  
##   
## $AICc  
## [1] -3.415091  
##   
## $BIC  
## [1] -3.317208

sarima(diff(log(all\_rmp)), p=10, d=0, q=0) # AR(10)

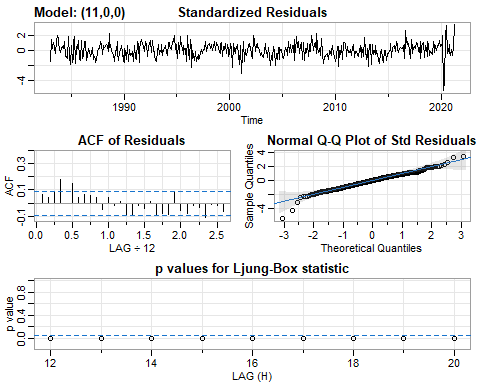
## initial value -2.554396   
## iter 2 value -2.731118  
## iter 3 value -2.923997  
## iter 4 value -3.013507  
## iter 5 value -3.093885  
## iter 6 value -3.114130  
## iter 7 value -3.131892  
## iter 8 value -3.150052  
## iter 9 value -3.177600  
## iter 10 value -3.200871  
## iter 11 value -3.204509  
## iter 12 value -3.215594  
## iter 13 value -3.216370  
## iter 14 value -3.216414  
## iter 15 value -3.216425  
## iter 16 value -3.216426  
## iter 17 value -3.216426  
## iter 17 value -3.216426  
## iter 17 value -3.216426  
## final value -3.216426   
## converged  
## initial value -3.204221   
## iter 2 value -3.204314  
## iter 3 value -3.204386  
## iter 4 value -3.204410  
## iter 5 value -3.204420  
## iter 6 value -3.204463  
## iter 7 value -3.204471  
## iter 8 value -3.204479  
## iter 9 value -3.204481  
## iter 10 value -3.204482  
## iter 10 value -3.204482  
## iter 10 value -3.204482  
## final value -3.204482   
## converged



## $fit  
##   
## Call:  
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),   
## xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,   
## optim.control = list(trace = trc, REPORT = 1, reltol = tol))  
##   
## Coefficients:  
## ar1 ar2 ar3 ar4 ar5 ar6 ar7 ar8  
## -0.6775 -0.6569 -0.1541 -0.6257 -0.2463 -0.5586 -0.2767 -0.6114  
## s.e. 0.0442 0.0543 0.0557 0.0548 0.0561 0.0559 0.0546 0.0559  
## ar9 ar10 xmean  
## -0.1433 -0.3220 1e-03  
## s.e. 0.0548 0.0446 4e-04  
##   
## sigma^2 estimated as 0.001628: log likelihood = 819.56, aic = -1615.13  
##   
## $degrees\_of\_freedom  
## [1] 448  
##   
## $ttable  
## Estimate SE t.value p.value  
## ar1 -0.6775 0.0442 -15.3319 0.0000  
## ar2 -0.6569 0.0543 -12.0886 0.0000  
## ar3 -0.1541 0.0557 -2.7637 0.0059  
## ar4 -0.6257 0.0548 -11.4118 0.0000  
## ar5 -0.2463 0.0561 -4.3894 0.0000  
## ar6 -0.5586 0.0559 -9.9995 0.0000  
## ar7 -0.2767 0.0546 -5.0648 0.0000  
## ar8 -0.6114 0.0559 -10.9280 0.0000  
## ar9 -0.1433 0.0548 -2.6151 0.0092  
## ar10 -0.3220 0.0446 -7.2194 0.0000  
## xmean 0.0010 0.0004 2.6595 0.0081  
##   
## $AIC  
## [1] -3.518799  
##   
## $AICc  
## [1] -3.517512  
##   
## $BIC  
## [1] -3.41085

sarima(diff(log(all\_rmp)), p=11, d=0, q=0) # AR(11)

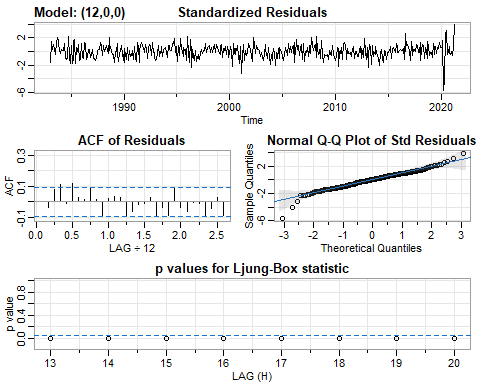
## initial value -2.553408   
## iter 2 value -2.725246  
## iter 3 value -2.876035  
## iter 4 value -2.980879  
## iter 5 value -3.108342  
## iter 6 value -3.131789  
## iter 7 value -3.168506  
## iter 8 value -3.179240  
## iter 9 value -3.260190  
## iter 10 value -3.289425  
## iter 11 value -3.289929  
## iter 12 value -3.295660  
## iter 13 value -3.296360  
## iter 14 value -3.296822  
## iter 15 value -3.296825  
## iter 16 value -3.296825  
## iter 16 value -3.296825  
## iter 16 value -3.296825  
## final value -3.296825   
## converged  
## initial value -3.283429   
## iter 2 value -3.283480  
## iter 3 value -3.283531  
## iter 4 value -3.283590  
## iter 5 value -3.283709  
## iter 6 value -3.283743  
## iter 7 value -3.283772  
## iter 8 value -3.283806  
## iter 9 value -3.283814  
## iter 10 value -3.283824  
## iter 11 value -3.283827  
## iter 12 value -3.283827  
## iter 13 value -3.283828  
## iter 13 value -3.283828  
## iter 13 value -3.283828  
## final value -3.283828   
## converged



## $fit  
##   
## Call:  
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),   
## xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,   
## optim.control = list(trace = trc, REPORT = 1, reltol = tol))  
##   
## Coefficients:  
## ar1 ar2 ar3 ar4 ar5 ar6 ar7 ar8  
## -0.8063 -0.7182 -0.3976 -0.7364 -0.4725 -0.6627 -0.5312 -0.6785  
## s.e. 0.0431 0.0504 0.0581 0.0519 0.0576 0.0528 0.0577 0.0519  
## ar9 ar10 ar11 xmean  
## -0.4100 -0.5967 -0.3994 9e-04  
## s.e. 0.0585 0.0511 0.0447 2e-04  
##   
## sigma^2 estimated as 0.001383: log likelihood = 855.98, aic = -1685.97  
##   
## $degrees\_of\_freedom  
## [1] 447  
##   
## $ttable  
## Estimate SE t.value p.value  
## ar1 -0.8063 0.0431 -18.6917 0e+00  
## ar2 -0.7182 0.0504 -14.2452 0e+00  
## ar3 -0.3976 0.0581 -6.8440 0e+00  
## ar4 -0.7364 0.0519 -14.1821 0e+00  
## ar5 -0.4725 0.0576 -8.1995 0e+00  
## ar6 -0.6627 0.0528 -12.5596 0e+00  
## ar7 -0.5312 0.0577 -9.2054 0e+00  
## ar8 -0.6785 0.0519 -13.0767 0e+00  
## ar9 -0.4100 0.0585 -7.0098 0e+00  
## ar10 -0.5967 0.0511 -11.6834 0e+00  
## ar11 -0.3994 0.0447 -8.9353 0e+00  
## xmean 0.0009 0.0002 3.8108 2e-04  
##   
## $AIC  
## [1] -3.673134  
##   
## $AICc  
## [1] -3.67161  
##   
## $BIC  
## [1] -3.556189

sarima(diff(log(all\_rmp)), p=12, d=0, q=0) # AR(12)

## initial value -2.552728   
## iter 2 value -2.845003  
## iter 3 value -3.109176  
## iter 4 value -3.155985  
## iter 5 value -3.191147  
## iter 6 value -3.220362  
## iter 7 value -3.231858  
## iter 8 value -3.242799  
## iter 9 value -3.255816  
## iter 10 value -3.271244  
## iter 11 value -3.278037  
## iter 12 value -3.298541  
## iter 13 value -3.301628  
## iter 14 value -3.309140  
## iter 15 value -3.309682  
## iter 16 value -3.309709  
## iter 17 value -3.309710  
## iter 17 value -3.309710  
## iter 17 value -3.309710  
## final value -3.309710   
## converged  
## initial value -3.296998   
## iter 2 value -3.297064  
## iter 3 value -3.297092  
## iter 4 value -3.297143  
## iter 5 value -3.297283  
## iter 6 value -3.297289  
## iter 7 value -3.297307  
## iter 8 value -3.297325  
## iter 9 value -3.297340  
## iter 10 value -3.297346  
## iter 11 value -3.297351  
## iter 12 value -3.297356  
## iter 13 value -3.297357  
## iter 14 value -3.297357  
## iter 14 value -3.297357  
## iter 14 value -3.297357  
## final value -3.297357   
## converged

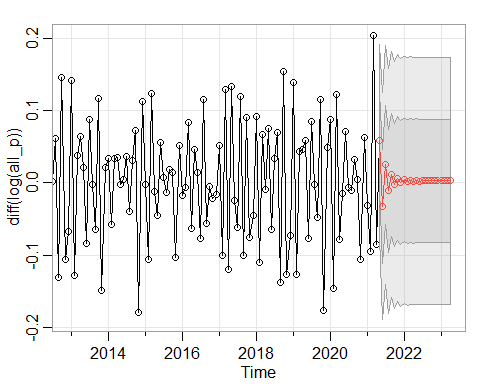


## $fit  
##   
## Call:  
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),   
## xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,   
## optim.control = list(trace = trc, REPORT = 1, reltol = tol))  
##   
## Coefficients:  
## ar1 ar2 ar3 ar4 ar5 ar6 ar7 ar8  
## -0.7374 -0.6144 -0.3227 -0.6185 -0.3791 -0.5454 -0.4423 -0.5468  
## s.e. 0.0467 0.0577 0.0611 0.0611 0.0626 0.0616 0.0622 0.0633  
## ar9 ar10 ar11 ar12 xmean  
## -0.3348 -0.4641 -0.2484 0.1751 9e-04  
## s.e. 0.0615 0.0628 0.0613 0.0493 3e-04  
##   
## sigma^2 estimated as 0.001345: log likelihood = 862.19, aic = -1696.39  
##   
## $degrees\_of\_freedom  
## [1] 446  
##   
## $ttable  
## Estimate SE t.value p.value  
## ar1 -0.7374 0.0467 -15.7764 0.0000  
## ar2 -0.6144 0.0577 -10.6527 0.0000  
## ar3 -0.3227 0.0611 -5.2832 0.0000  
## ar4 -0.6185 0.0611 -10.1308 0.0000  
## ar5 -0.3791 0.0626 -6.0518 0.0000  
## ar6 -0.5454 0.0616 -8.8527 0.0000  
## ar7 -0.4423 0.0622 -7.1073 0.0000  
## ar8 -0.5468 0.0633 -8.6393 0.0000  
## ar9 -0.3348 0.0615 -5.4402 0.0000  
## ar10 -0.4641 0.0628 -7.3872 0.0000  
## ar11 -0.2484 0.0613 -4.0488 0.0001  
## ar12 0.1751 0.0493 3.5515 0.0004  
## xmean 0.0009 0.0003 3.2810 0.0011  
##   
## $AIC  
## [1] -3.695835  
##   
## $AICc  
## [1] -3.694053  
##   
## $BIC  
## [1] -3.569895

Autocorrelation displayed by ACF of Residuals is lowest for AR(12).

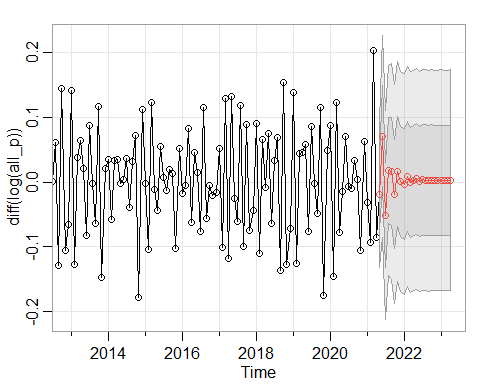
24-Month Forecast: All Poultry

sarima.for(diff(log(all\_p)), n.ahead=24, p=1, d=0, q=0) # AR(1)



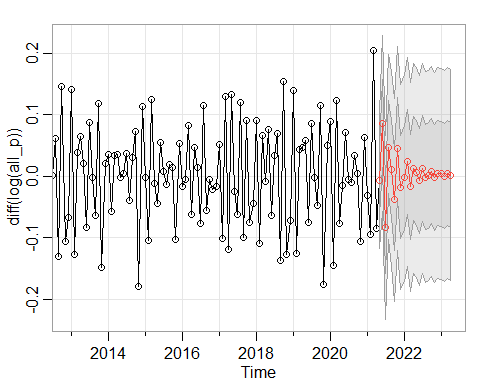
## $pred  
## Jan Feb Mar Apr May  
## 2021 0.0585146289  
## 2022 0.0042260413 0.0019775394 0.0033938524 0.0025017284 0.0030636700  
## 2023 0.0028518825 0.0028431108 0.0028486360 0.0028451557   
## Jun Jul Aug Sep Oct  
## 2021 -0.0322183991 0.0249335950 -0.0110659798 0.0116098587 -0.0026734669  
## 2022 0.0027097076 0.0029326656 0.0027922262 0.0028806878 0.0028249666  
## 2023   
## Nov Dec  
## 2021 0.0063234833 0.0006563778  
## 2022 0.0028600649 0.0028379568  
## 2023   
##   
## $se  
## Jan Feb Mar Apr May Jun  
## 2021 0.06633516 0.07839805  
## 2022 0.08539789 0.08540417 0.08540666 0.08540765 0.08540804 0.08540820  
## 2023 0.08540830 0.08540830 0.08540830 0.08540830   
## Jul Aug Sep Oct Nov Dec  
## 2021 0.08269803 0.08434339 0.08498738 0.08524154 0.08534217 0.08538207  
## 2022 0.08540826 0.08540828 0.08540829 0.08540830 0.08540830 0.08540830  
## 2023

sarima.for(diff(log(all\_p)), n.ahead=24, p=2, d=0, q=0) # AR(2)



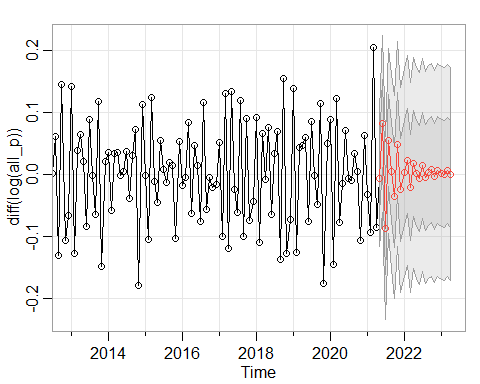
## $pred  
## Jan Feb Mar Apr May  
## 2021 -0.0194133021  
## 2022 -0.0030306447 0.0090130206 -0.0001900122 0.0022461610 0.0048005932  
## 2023 0.0029454318 0.0026538469 0.0027411526 0.0028123639   
## Jun Jul Aug Sep Oct  
## 2021 0.0710034781 -0.0510359426 0.0181257873 0.0165885005 -0.0187215490  
## 2022 0.0010505204 0.0032948551 0.0031331796 0.0020947563 0.0031784587  
## 2023   
## Nov Dec  
## 2021 0.0160216971 0.0014220946  
## 2022 0.0026895840 0.0025828700  
## 2023   
##   
## $se  
## Jan Feb Mar Apr May Jun  
## 2021 0.05633751 0.07812691  
## 2022 0.08530842 0.08537793 0.08537812 0.08539427 0.08541101 0.08541422  
## 2023 0.08541799 0.08541799 0.08541800 0.08541802   
## Jul Aug Sep Oct Nov Dec  
## 2021 0.08117629 0.08153266 0.08372619 0.08492896 0.08500342 0.08510505  
## 2022 0.08541442 0.08541635 0.08541755 0.08541765 0.08541773 0.08541792  
## 2023

sarima.for(diff(log(all\_p)), n.ahead=24, p=3, d=0, q=0) # AR(3)



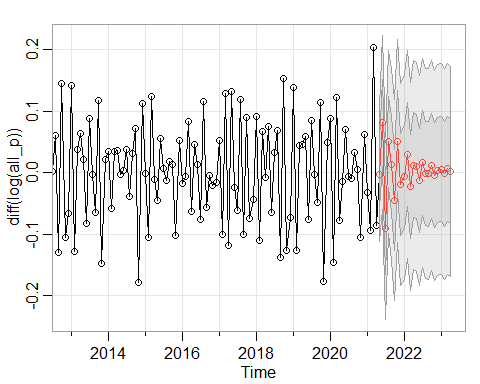
## $pred  
## Jan Feb Mar Apr May  
## 2021 -0.0078084452  
## 2022 -0.0024051788 0.0232612439 -0.0174911309 0.0122996707 0.0058583738  
## 2023 0.0037701761 0.0002931854 0.0050807402 0.0019029355   
## Jun Jul Aug Sep Oct  
## 2021 0.0855285230 -0.0837204202 0.0473295431 0.0113693012 -0.0381578018  
## 2022 -0.0073810006 0.0126053535 -0.0015506936 0.0010765388 0.0078750184  
## 2023   
## Nov Dec  
## 2021 0.0448139667 -0.0177674828  
## 2022 -0.0019135260 0.0048003181  
## 2023   
##   
## $se  
## Jan Feb Mar Apr May Jun  
## 2021 0.05484052 0.07159989  
## 2022 0.08412578 0.08484959 0.08496211 0.08500182 0.08521961 0.08538513  
## 2023 0.08553117 0.08553989 0.08554079 0.08554169   
## Jul Aug Sep Oct Nov Dec  
## 2021 0.07479518 0.07530891 0.07925725 0.08252804 0.08308675 0.08322314  
## 2022 0.08540790 0.08541934 0.08547242 0.08551043 0.08551499 0.08551824  
## 2023

sarima.for(diff(log(all\_p)), n.ahead=24, p=4, d=0, q=0) # AR(4)



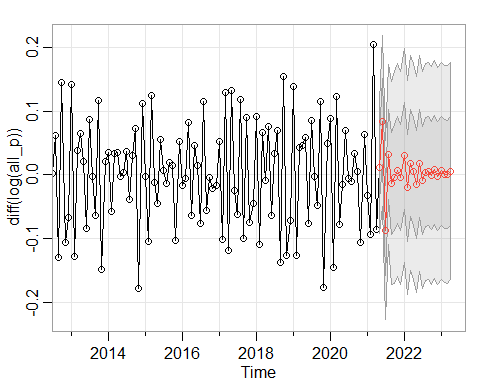
## $pred  
## Jan Feb Mar Apr May  
## 2021 -0.0060740795  
## 2022 0.0032434937 0.0221132779 -0.0209417057 0.0176658784 0.0018104493  
## 2023 0.0021351524 0.0005308815 0.0059985175 0.0005269905   
## Jun Jul Aug Sep Oct  
## 2021 0.0815660575 -0.0866054885 0.0540532540 0.0047216201 -0.0363796448  
## 2022 -0.0067023885 0.0149938734 -0.0051763880 0.0036797783 0.0074336073  
## 2023   
## Nov Dec  
## 2021 0.0488559544 -0.0248856896  
## 2022 -0.0034723136 0.0070436757  
## 2023   
##   
## $se  
## Jan Feb Mar Apr May Jun  
## 2021 0.05476439 0.07108179  
## 2022 0.08368096 0.08459941 0.08481615 0.08483097 0.08506248 0.08530733  
## 2023 0.08553066 0.08554821 0.08555385 0.08555387   
## Jul Aug Sep Oct Nov Dec  
## 2021 0.07373391 0.07425557 0.07835600 0.08191404 0.08264780 0.08273668  
## 2022 0.08537195 0.08537416 0.08543147 0.08549708 0.08551624 0.08551650  
## 2023

sarima.for(diff(log(all\_p)), n.ahead=24, p=5, d=0, q=0) # AR(5)



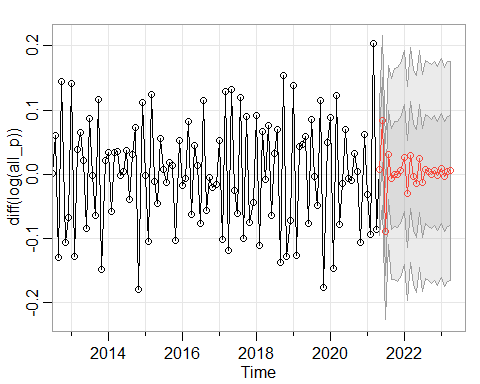
## $pred  
## Jan Feb Mar Apr May  
## 2021 -0.002760065  
## 2022 -0.006656821 0.030137196 -0.022605163 0.012558853 0.009827633  
## 2023 0.005886360 -0.002007599 0.006111314 0.002334095   
## Jun Jul Aug Sep Oct  
## 2021 0.081440996 -0.091351494 0.050011964 0.013187667 -0.044659582  
## 2022 -0.012732326 0.015861621 -0.001286328 -0.002054341 0.011427780  
## 2023   
## Nov Dec  
## 2021 0.051462221 -0.019351699  
## 2022 -0.003896938 0.004272817  
## 2023   
##   
## $se  
## Jan Feb Mar Apr May Jun  
## 2021 0.05458912 0.07099859  
## 2022 0.08360273 0.08448324 0.08458353 0.08468057 0.08504063 0.08526842  
## 2023 0.08554409 0.08555829 0.08555835 0.08556453   
## Jul Aug Sep Oct Nov Dec  
## 2021 0.07395690 0.07417339 0.07826338 0.08167439 0.08223169 0.08242721  
## 2022 0.08528358 0.08532545 0.08543527 0.08549302 0.08549464 0.08551123  
## 2023

sarima.for(diff(log(all\_p)), n.ahead=24, p=6, d=0, q=0) # AR(6)



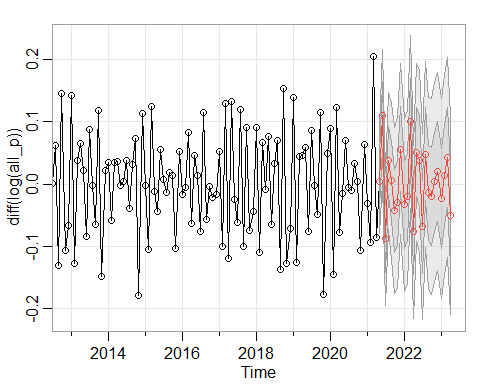
## $pred  
## Jan Feb Mar Apr May  
## 2021 0.0108717832  
## 2022 0.0302593103 -0.0202750207 0.0183920951 0.0055006648 -0.0155267111  
## 2023 0.0063428329 0.0012964025 0.0006130902 0.0060622542   
## Jun Jul Aug Sep Oct  
## 2021 0.0843498218 -0.0870917533 0.0319328092 -0.0132388756 -0.0038820219  
## 2022 0.0182395138 -0.0083300868 0.0044387641 0.0060986442 -0.0008569296  
## 2023   
## Nov Dec  
## 2021 0.0074004062 -0.0042860358  
## 2022 0.0081752792 -0.0017068179  
## 2023   
##   
## $se  
## Jan Feb Mar Apr May Jun  
## 2021 0.05106625 0.06731524  
## 2022 0.08393167 0.08399891 0.08429797 0.08460907 0.08478772 0.08482183  
## 2023 0.08538268 0.08540123 0.08540835 0.08541124   
## Jul Aug Sep Oct Nov Dec  
## 2021 0.07015801 0.07079276 0.07959489 0.08287554 0.08362732 0.08362847  
## 2022 0.08493279 0.08514139 0.08529839 0.08533415 0.08533934 0.08535588  
## 2023

sarima.for(diff(log(all\_p)), n.ahead=24, p=7, d=0, q=0) # AR(7)



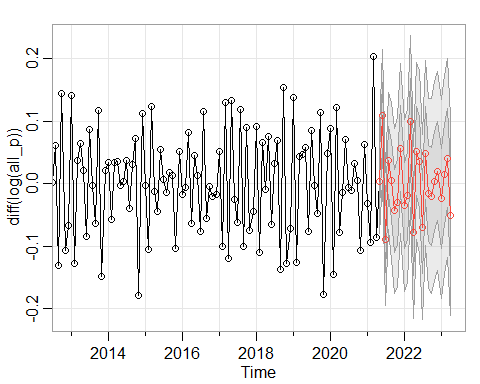
## $pred  
## Jan Feb Mar Apr May  
## 2021 0.0072986555  
## 2022 0.0261429453 -0.0303390528 0.0300829220 -0.0039272593 -0.0146024875  
## 2023 0.0094469481 -0.0034590705 0.0044804410 0.0057643034   
## Jun Jul Aug Sep Oct  
## 2021 0.0843842473 -0.0890232761 0.0303293300 -0.0069693788 0.0006689785  
## 2022 0.0246381172 -0.0131057823 0.0082879256 0.0044371498 0.0006047622  
## 2023   
## Nov Dec  
## 2021 -0.0003087408 0.0060709524  
## 2022 0.0059035794 -0.0023627348  
## 2023   
##   
## $se  
## Jan Feb Mar Apr May Jun  
## 2021 0.05087033 0.06604198  
## 2022 0.08326303 0.08329392 0.08344594 0.08385412 0.08427421 0.08446192  
## 2023 0.08528188 0.08530577 0.08532522 0.08534407   
## Jul Aug Sep Oct Nov Dec  
## 2021 0.06864025 0.06935316 0.07852925 0.08232069 0.08301402 0.08301500  
## 2022 0.08449235 0.08479651 0.08511867 0.08524072 0.08524121 0.08525732  
## 2023

sarima.for(diff(log(all\_p)), n.ahead=24, p=8, d=0, q=0) # AR(8)



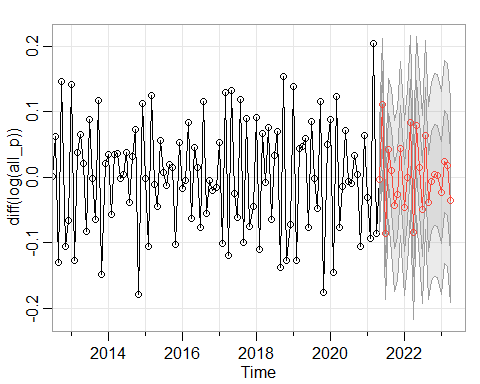
## $pred  
## Jan Feb Mar Apr May  
## 2021 0.004090777  
## 2022 -0.033559884 -0.020402804 0.099539463 -0.076360597 0.050949795  
## 2023 -0.022997366 0.013753246 0.042342306 -0.050080668   
## Jun Jul Aug Sep Oct  
## 2021 0.109574880 -0.087945430 0.037257438 0.005233874 -0.043056418  
## 2022 0.037169951 -0.068411971 0.047722903 -0.014324035 -0.019845070  
## 2023   
## Nov Dec  
## 2021 -0.030031604 0.056122073  
## 2022 0.003521584 0.020114489  
## 2023   
##   
## $se  
## Jan Feb Mar Apr May Jun  
## 2021 0.04200099 0.05326750  
## 2022 0.06923073 0.06935844 0.06935844 0.06937317 0.07113284 0.07341067  
## 2023 0.07980957 0.07982838 0.07986142 0.07986920   
## Jul Aug Sep Oct Nov Dec  
## 2021 0.05354404 0.05512081 0.06296342 0.06649143 0.06841782 0.06868971  
## 2022 0.07457787 0.07461070 0.07665951 0.07857692 0.07965425 0.07969126  
## 2023

sarima.for(diff(log(all\_p)), n.ahead=24, p=9, d=0, q=0) # AR(9)



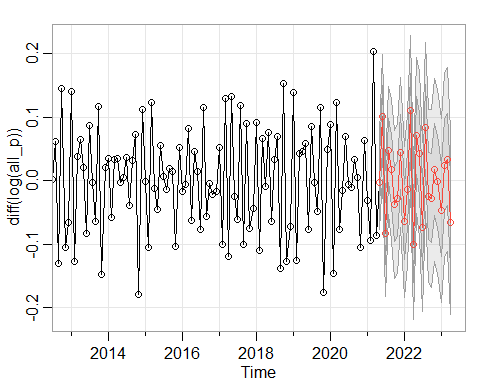
## $pred  
## Jan Feb Mar Apr May  
## 2021 0.004365606  
## 2022 -0.034612057 -0.019206319 0.099687228 -0.077983767 0.052428844  
## 2023 -0.023512285 0.015051492 0.041289826 -0.051210440   
## Jun Jul Aug Sep Oct  
## 2021 0.109435685 -0.088450074 0.037277089 0.004981152 -0.043134087  
## 2022 0.035485454 -0.068965191 0.049277622 -0.015415192 -0.019469691  
## 2023   
## Nov Dec  
## 2021 -0.029184141 0.056751538  
## 2022 0.004483031 0.020208357  
## 2023   
##   
## $se  
## Jan Feb Mar Apr May Jun  
## 2021 0.04199428 0.05299815  
## 2022 0.06887979 0.06900468 0.06900487 0.06901810 0.07078751 0.07309686  
## 2023 0.07957664 0.07959650 0.07962918 0.07963634   
## Jul Aug Sep Oct Nov Dec  
## 2021 0.05324000 0.05488810 0.06273370 0.06616025 0.06804974 0.06832264  
## 2022 0.07427161 0.07430919 0.07633318 0.07830779 0.07941604 0.07945711  
## 2023

sarima.for(diff(log(all\_p)), n.ahead=24, p=10, d=0, q=0) # AR(10)



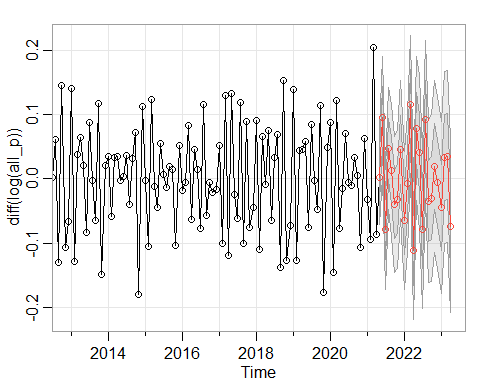
## $pred  
## Jan Feb Mar Apr May  
## 2021 -0.0033514473  
## 2022 -0.0467705531 -0.0005995298 0.0826980517 -0.0839573203 0.0785014782  
## 2023 -0.0236598045 0.0235989526 0.0184682280 -0.0357829336   
## Jun Jul Aug Sep Oct  
## 2021 0.1109152109 -0.0858634891 0.0425737243 0.0104396587 -0.0434570811  
## 2022 0.0149380910 -0.0493396473 0.0642893899 -0.0393727932 -0.0063425988  
## 2023   
## Nov Dec  
## 2021 -0.0260875110 0.0436592611  
## 2022 0.0035537876 0.0018244444  
## 2023   
##   
## $se  
## Jan Feb Mar Apr May Jun  
## 2021 0.04022200 0.05061803  
## 2022 0.06669952 0.06675795 0.06676883 0.06678706 0.06818161 0.07050281  
## 2023 0.07747115 0.07752622 0.07764621 0.07765425   
## Jul Aug Sep Oct Nov Dec  
## 2021 0.05065101 0.05385535 0.06258086 0.06514868 0.06573474 0.06631307  
## 2022 0.07191295 0.07197957 0.07359578 0.07536504 0.07726297 0.07740637  
## 2023

sarima.for(diff(log(all\_p)), n.ahead=24, p=11, d=0, q=0) # AR(11)



## $pred  
## Jan Feb Mar Apr May  
## 2021 -0.002373511  
## 2022 -0.064640861 -0.014357606 0.110420880 -0.100104874 0.071305895  
## 2023 -0.046876441 0.023792573 0.033350947 -0.065942637   
## Jun Jul Aug Sep Oct  
## 2021 0.100739157 -0.083007873 0.047209517 0.017009041 -0.037319479  
## 2022 0.042233530 -0.073041526 0.084587356 -0.024017169 -0.027852751  
## 2023   
## Nov Dec  
## 2021 -0.028534849 0.044786769  
## 2022 0.017233581 -0.001362971  
## 2023   
##   
## $se  
## Jan Feb Mar Apr May Jun  
## 2021 0.03726194 0.04938943  
## 2022 0.05926919 0.05926929 0.05927803 0.05944570 0.06174053 0.06496328  
## 2023 0.07238035 0.07253053 0.07260545 0.07270982   
## Jul Aug Sep Oct Nov Dec  
## 2021 0.04943454 0.05027963 0.05558328 0.05801509 0.05914432 0.05914438  
## 2022 0.06638471 0.06645595 0.06818521 0.07015082 0.07218729 0.07228468  
## 2023

sarima.for(diff(log(all\_p)), n.ahead=24, p=12, d=0, q=0) # AR(12)

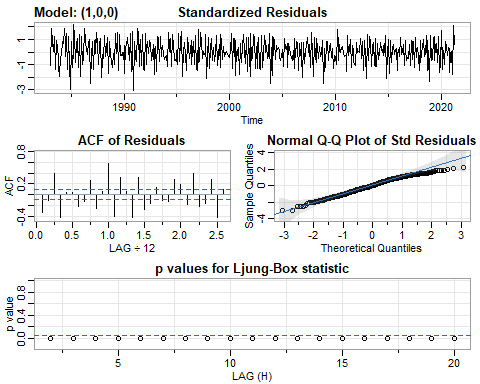


## $pred  
## Jan Feb Mar Apr May  
## 2021 0.003054730  
## 2022 -0.064386499 -0.006247674 0.115995011 -0.111101022 0.079072958  
## 2023 -0.043624549 0.033391930 0.034616240 -0.073660460   
## Jun Jul Aug Sep Oct  
## 2021 0.096037396 -0.078472542 0.046892765 0.012557702 -0.039464957  
## 2022 0.041243346 -0.078673766 0.092052846 -0.034972862 -0.029552056  
## 2023   
## Nov Dec  
## 2021 -0.031654604 0.045557116  
## 2022 0.019444550 -0.005442959  
## 2023   
##   
## $se  
## Jan Feb Mar Apr May Jun  
## 2021 0.03660896 0.04688029  
## 2022 0.05355666 0.05356083 0.05356120 0.05385451 0.05595170 0.05963166  
## 2023 0.06676030 0.06687985 0.06701896 0.06715002   
## Jul Aug Sep Oct Nov Dec  
## 2021 0.04690973 0.04750461 0.05129102 0.05267326 0.05353520 0.05353652  
## 2022 0.06115542 0.06119357 0.06292297 0.06472366 0.06662407 0.06670183  
## 2023

Checking the Model Conditions: All Poultry

sarima(diff(log(all\_p)), p=1, d=0, q=0)# AR(1)

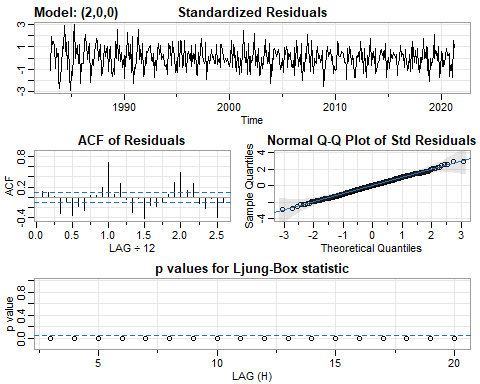
## initial value -2.460721   
## iter 2 value -2.713258  
## iter 3 value -2.713266  
## iter 3 value -2.713266  
## iter 3 value -2.713266  
## final value -2.713266   
## converged  
## initial value -2.712484   
## iter 2 value -2.712485  
## iter 2 value -2.712485  
## iter 2 value -2.712485  
## final value -2.712485   
## converged



## $fit  
##   
## Call:  
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),   
## xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,   
## optim.control = list(trace = trc, REPORT = 1, reltol = tol))  
##   
## Coefficients:  
## ar1 xmean  
## -0.6299 0.0028  
## s.e. 0.0363 0.0019  
##   
## sigma^2 estimated as 0.0044: log likelihood = 593.74, aic = -1181.48  
##   
## $degrees\_of\_freedom  
## [1] 457  
##   
## $ttable  
## Estimate SE t.value p.value  
## ar1 -0.6299 0.0363 -17.3619 0.0000  
## xmean 0.0028 0.0019 1.4967 0.1352  
##   
## $AIC  
## [1] -2.57402  
##   
## $AICc  
## [1] -2.573963  
##   
## $BIC  
## [1] -2.547033

sarima(diff(log(all\_p)), p=2, d=0, q=0) # AR(2)

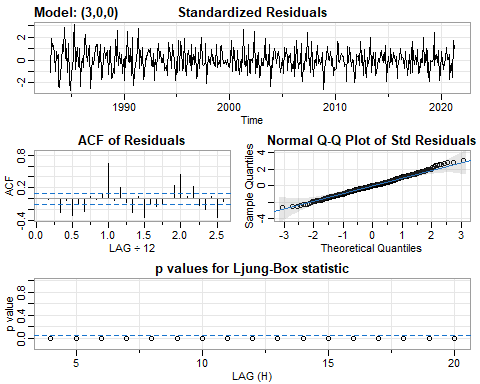
## initial value -2.464856   
## iter 2 value -2.672827  
## iter 3 value -2.857715  
## iter 4 value -2.871723  
## iter 5 value -2.879038  
## iter 6 value -2.879409  
## iter 7 value -2.879555  
## iter 8 value -2.879561  
## iter 9 value -2.879564  
## iter 10 value -2.879564  
## iter 11 value -2.879564  
## iter 12 value -2.879564  
## iter 12 value -2.879564  
## iter 12 value -2.879564  
## final value -2.879564   
## converged  
## initial value -2.875113   
## iter 2 value -2.875115  
## iter 3 value -2.875124  
## iter 4 value -2.875125  
## iter 5 value -2.875126  
## iter 5 value -2.875126  
## iter 5 value -2.875126  
## final value -2.875126   
## converged



## $fit  
##   
## Call:  
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),   
## xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,   
## optim.control = list(trace = trc, REPORT = 1, reltol = tol))  
##   
## Coefficients:  
## ar1 ar2 xmean  
## -0.9608 -0.5319 0.0028  
## s.e. 0.0396 0.0400 0.0011  
##   
## sigma^2 estimated as 0.003174: log likelihood = 668.39, aic = -1328.78  
##   
## $degrees\_of\_freedom  
## [1] 456  
##   
## $ttable  
## Estimate SE t.value p.value  
## ar1 -0.9608 0.0396 -24.2530 0.0000  
## ar2 -0.5319 0.0400 -13.3045 0.0000  
## xmean 0.0028 0.0011 2.6006 0.0096  
##   
## $AIC  
## [1] -2.894945  
##   
## $AICc  
## [1] -2.89483  
##   
## $BIC  
## [1] -2.858962

sarima(diff(log(all\_p)), p=3, d=0, q=0) # AR(3)

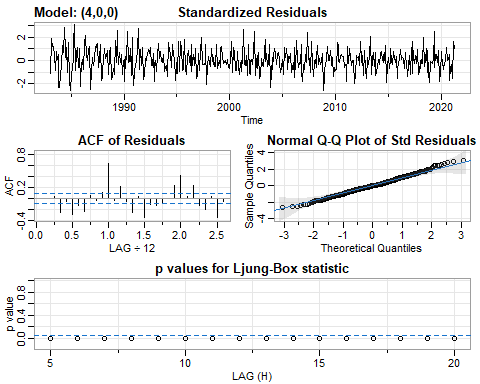
## initial value -2.464292   
## iter 2 value -2.851924  
## iter 3 value -2.876119  
## iter 4 value -2.894698  
## iter 5 value -2.904788  
## iter 6 value -2.907010  
## iter 7 value -2.907085  
## iter 8 value -2.907086  
## iter 9 value -2.907086  
## iter 9 value -2.907086  
## iter 9 value -2.907086  
## final value -2.907086   
## converged  
## initial value -2.901851   
## iter 2 value -2.901865  
## iter 3 value -2.901873  
## iter 4 value -2.901874  
## iter 5 value -2.901875  
## iter 5 value -2.901875  
## iter 5 value -2.901875  
## final value -2.901875   
## converged



## $fit  
##   
## Call:  
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),   
## xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,   
## optim.control = list(trace = trc, REPORT = 1, reltol = tol))  
##   
## Coefficients:  
## ar1 ar2 ar3 xmean  
## -0.8394 -0.3102 0.2308 0.0028  
## s.e. 0.0455 0.0588 0.0459 0.0013  
##   
## sigma^2 estimated as 0.003007: log likelihood = 680.67, aic = -1351.34  
##   
## $degrees\_of\_freedom  
## [1] 455  
##   
## $ttable  
## Estimate SE t.value p.value  
## ar1 -0.8394 0.0455 -18.4553 0.0000  
## ar2 -0.3102 0.0588 -5.2739 0.0000  
## ar3 0.2308 0.0459 5.0260 0.0000  
## xmean 0.0028 0.0013 2.1018 0.0361  
##   
## $AIC  
## [1] -2.944087  
##   
## $AICc  
## [1] -2.943895  
##   
## $BIC  
## [1] -2.899108

sarima(diff(log(all\_p)), p=4, d=0, q=0) # AR(4)

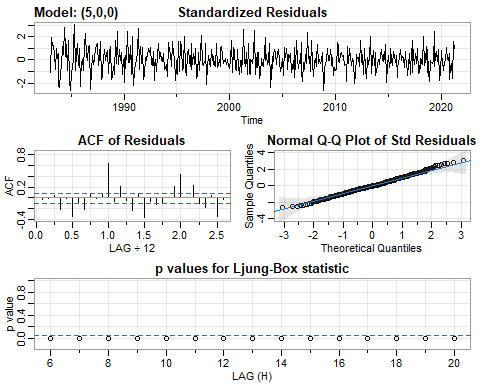
## initial value -2.463428   
## iter 2 value -2.673666  
## iter 3 value -2.848804  
## iter 4 value -2.889797  
## iter 5 value -2.903160  
## iter 6 value -2.907549  
## iter 7 value -2.908573  
## iter 8 value -2.909134  
## iter 9 value -2.909143  
## iter 9 value -2.909143  
## iter 9 value -2.909143  
## final value -2.909143   
## converged  
## initial value -2.903206   
## iter 2 value -2.903207  
## iter 3 value -2.903251  
## iter 4 value -2.903251  
## iter 5 value -2.903251  
## iter 6 value -2.903251  
## iter 6 value -2.903251  
## iter 6 value -2.903251  
## final value -2.903251   
## converged



## $fit  
##   
## Call:  
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),   
## xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,   
## optim.control = list(trace = trc, REPORT = 1, reltol = tol))  
##   
## Coefficients:  
## ar1 ar2 ar3 ar4 xmean  
## -0.8275 -0.3268 0.1861 -0.0531 0.0028  
## s.e. 0.0467 0.0606 0.0607 0.0472 0.0013  
##   
## sigma^2 estimated as 0.002999: log likelihood = 681.3, aic = -1350.6  
##   
## $degrees\_of\_freedom  
## [1] 454  
##   
## $ttable  
## Estimate SE t.value p.value  
## ar1 -0.8275 0.0467 -17.7357 0.0000  
## ar2 -0.3268 0.0606 -5.3966 0.0000  
## ar3 0.1861 0.0607 3.0679 0.0023  
## ar4 -0.0531 0.0472 -1.1248 0.2613  
## xmean 0.0028 0.0013 2.2021 0.0282  
##   
## $AIC  
## [1] -2.942482  
##   
## $AICc  
## [1] -2.942194  
##   
## $BIC  
## [1] -2.888508

sarima(diff(log(all\_p)), p=5, d=0, q=0) # AR(5)

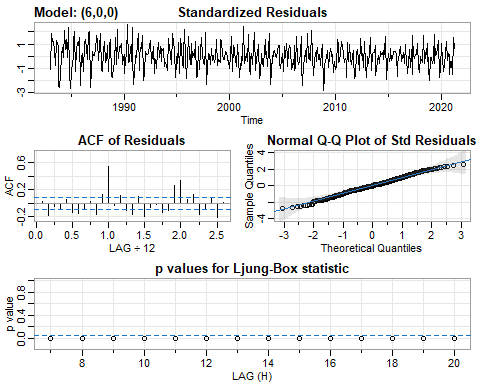
## initial value -2.462793   
## iter 2 value -2.478506  
## iter 3 value -2.854306  
## iter 4 value -2.873560  
## iter 5 value -2.875993  
## iter 6 value -2.899286  
## iter 7 value -2.907045  
## iter 8 value -2.910995  
## iter 9 value -2.911386  
## iter 10 value -2.911543  
## iter 11 value -2.911559  
## iter 12 value -2.911575  
## iter 13 value -2.911576  
## iter 13 value -2.911576  
## iter 13 value -2.911576  
## final value -2.911576   
## converged  
## initial value -2.906356   
## iter 2 value -2.906394  
## iter 3 value -2.906420  
## iter 4 value -2.906420  
## iter 5 value -2.906420  
## iter 6 value -2.906421  
## iter 6 value -2.906421  
## iter 6 value -2.906421  
## final value -2.906421   
## converged



## $fit  
##   
## Call:  
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),   
## xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,   
## optim.control = list(trace = trc, REPORT = 1, reltol = tol))  
##   
## Coefficients:  
## ar1 ar2 ar3 ar4 ar5 xmean  
## -0.8316 -0.3122 0.1596 -0.1200 -0.0804 0.0028  
## s.e. 0.0466 0.0610 0.0624 0.0612 0.0471 0.0012  
##   
## sigma^2 estimated as 0.00298: log likelihood = 682.75, aic = -1351.51  
##   
## $degrees\_of\_freedom  
## [1] 453  
##   
## $ttable  
## Estimate SE t.value p.value  
## ar1 -0.8316 0.0466 -17.8591 0.0000  
## ar2 -0.3122 0.0610 -5.1194 0.0000  
## ar3 0.1596 0.0624 2.5552 0.0109  
## ar4 -0.1200 0.0612 -1.9606 0.0505  
## ar5 -0.0804 0.0471 -1.7084 0.0882  
## xmean 0.0028 0.0012 2.3675 0.0183  
##   
## $AIC  
## [1] -2.944463  
##   
## $AICc  
## [1] -2.944058  
##   
## $BIC  
## [1] -2.881493

sarima(diff(log(all\_p)), p=6, d=0, q=0) # AR(6)

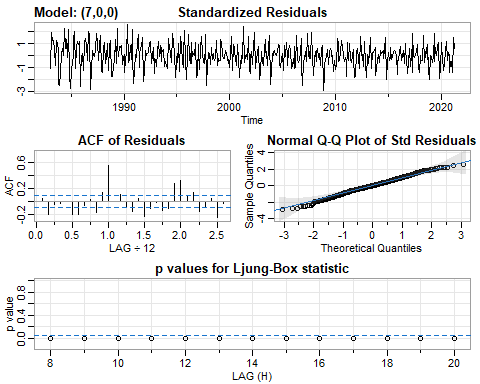
## initial value -2.464065   
## iter 2 value -2.627908  
## iter 3 value -2.824735  
## iter 4 value -2.865397  
## iter 5 value -2.939363  
## iter 6 value -2.966124  
## iter 7 value -2.974919  
## iter 8 value -2.976749  
## iter 9 value -2.977162  
## iter 10 value -2.977497  
## iter 11 value -2.977825  
## iter 12 value -2.977843  
## iter 13 value -2.977848  
## iter 13 value -2.977848  
## iter 13 value -2.977848  
## final value -2.977848   
## converged  
## initial value -2.972190   
## iter 2 value -2.972214  
## iter 3 value -2.972246  
## iter 4 value -2.972252  
## iter 5 value -2.972256  
## iter 6 value -2.972257  
## iter 7 value -2.972258  
## iter 7 value -2.972258  
## iter 7 value -2.972258  
## final value -2.972258   
## converged



## $fit  
##   
## Call:  
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),   
## xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,   
## optim.control = list(trace = trc, REPORT = 1, reltol = tol))  
##   
## Coefficients:  
## ar1 ar2 ar3 ar4 ar5 ar6 xmean  
## -0.8589 -0.3505 0.2167 -0.2316 -0.3777 -0.3554 0.0027  
## s.e. 0.0437 0.0572 0.0589 0.0589 0.0574 0.0441 0.0008  
##   
## sigma^2 estimated as 0.002608: log likelihood = 712.97, aic = -1409.95  
##   
## $degrees\_of\_freedom  
## [1] 452  
##   
## $ttable  
## Estimate SE t.value p.value  
## ar1 -0.8589 0.0437 -19.6738 0e+00  
## ar2 -0.3505 0.0572 -6.1269 0e+00  
## ar3 0.2167 0.0589 3.6770 3e-04  
## ar4 -0.2316 0.0589 -3.9343 1e-04  
## ar5 -0.3777 0.0574 -6.5759 0e+00  
## ar6 -0.3554 0.0441 -8.0533 0e+00  
## xmean 0.0027 0.0008 3.3595 8e-04  
##   
## $AIC  
## [1] -3.07178  
##   
## $AICc  
## [1] -3.071239  
##   
## $BIC  
## [1] -2.999814

sarima(diff(log(all\_p)), p=7, d=0, q=0) # AR(7)

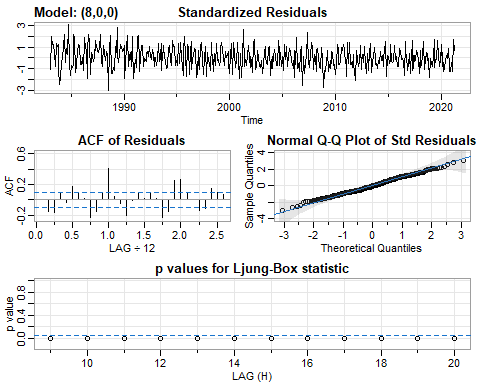
## initial value -2.465872   
## iter 2 value -2.629833  
## iter 3 value -2.844550  
## iter 4 value -2.863836  
## iter 5 value -2.876902  
## iter 6 value -2.968669  
## iter 7 value -2.975521  
## iter 8 value -2.981750  
## iter 9 value -2.982166  
## iter 10 value -2.982339  
## iter 11 value -2.982388  
## iter 12 value -2.982402  
## iter 13 value -2.982415  
## iter 14 value -2.982424  
## iter 14 value -2.982424  
## iter 14 value -2.982424  
## final value -2.982424   
## converged  
## initial value -2.975940   
## iter 2 value -2.975999  
## iter 3 value -2.976034  
## iter 4 value -2.976043  
## iter 5 value -2.976046  
## iter 6 value -2.976046  
## iter 7 value -2.976046  
## iter 8 value -2.976046  
## iter 8 value -2.976046  
## iter 8 value -2.976046  
## final value -2.976046   
## converged



## $fit  
##   
## Call:  
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),   
## xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,   
## optim.control = list(trace = trc, REPORT = 1, reltol = tol))  
##   
## Coefficients:  
## ar1 ar2 ar3 ar4 ar5 ar6 ar7 xmean  
## -0.8279 -0.3177 0.2364 -0.2502 -0.3462 -0.2793 0.0881 0.0027  
## s.e. 0.0465 0.0596 0.0596 0.0595 0.0597 0.0599 0.0471 0.0009  
##   
## sigma^2 estimated as 0.002588: log likelihood = 714.71, aic = -1411.42  
##   
## $degrees\_of\_freedom  
## [1] 451  
##   
## $ttable  
## Estimate SE t.value p.value  
## ar1 -0.8279 0.0465 -17.7923 0.0000  
## ar2 -0.3177 0.0596 -5.3265 0.0000  
## ar3 0.2364 0.0596 3.9648 0.0001  
## ar4 -0.2502 0.0595 -4.2073 0.0000  
## ar5 -0.3462 0.0597 -5.8015 0.0000  
## ar6 -0.2793 0.0599 -4.6608 0.0000  
## ar7 0.0881 0.0471 1.8689 0.0623  
## xmean 0.0027 0.0009 3.0886 0.0021  
##   
## $AIC  
## [1] -3.075  
##   
## $AICc  
## [1] -3.074303  
##   
## $BIC  
## [1] -2.994038

sarima(diff(log(all\_p)), p=8, d=0, q=0) # AR(8)

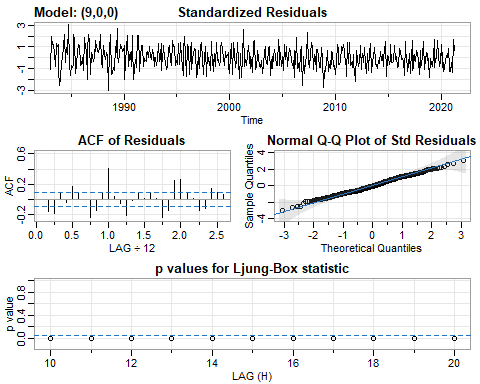
## initial value -2.465287   
## iter 2 value -2.570959  
## iter 3 value -3.007957  
## iter 4 value -3.042305  
## iter 5 value -3.077042  
## iter 6 value -3.101109  
## iter 7 value -3.143835  
## iter 8 value -3.165972  
## iter 9 value -3.168334  
## iter 10 value -3.173559  
## iter 11 value -3.173994  
## iter 12 value -3.174058  
## iter 13 value -3.174097  
## iter 14 value -3.174098  
## iter 14 value -3.174098  
## iter 14 value -3.174098  
## final value -3.174098   
## converged  
## initial value -3.164190   
## iter 2 value -3.164211  
## iter 3 value -3.164227  
## iter 4 value -3.164237  
## iter 5 value -3.164250  
## iter 6 value -3.164255  
## iter 7 value -3.164262  
## iter 8 value -3.164264  
## iter 9 value -3.164266  
## iter 10 value -3.164266  
## iter 11 value -3.164267  
## iter 12 value -3.164267  
## iter 12 value -3.164267  
## iter 12 value -3.164267  
## final value -3.164267   
## converged



## $fit  
##   
## Call:  
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),   
## xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,   
## optim.control = list(trace = trc, REPORT = 1, reltol = tol))  
##   
## Coefficients:  
## ar1 ar2 ar3 ar4 ar5 ar6 ar7 ar8  
## -0.7800 -0.4790 0.0389 -0.3891 -0.2156 -0.4640 -0.3856 -0.5668  
## s.e. 0.0384 0.0504 0.0510 0.0502 0.0500 0.0509 0.0506 0.0388  
## xmean  
## 0.0027  
## s.e. 0.0005  
##   
## sigma^2 estimated as 0.001764: log likelihood = 801.11, aic = -1582.21  
##   
## $degrees\_of\_freedom  
## [1] 450  
##   
## $ttable  
## Estimate SE t.value p.value  
## ar1 -0.7800 0.0384 -20.2901 0.0000  
## ar2 -0.4790 0.0504 -9.5134 0.0000  
## ar3 0.0389 0.0510 0.7636 0.4455  
## ar4 -0.3891 0.0502 -7.7557 0.0000  
## ar5 -0.2156 0.0500 -4.3133 0.0000  
## ar6 -0.4640 0.0509 -9.1169 0.0000  
## ar7 -0.3856 0.0506 -7.6210 0.0000  
## ar8 -0.5668 0.0388 -14.6212 0.0000  
## xmean 0.0027 0.0005 5.7115 0.0000  
##   
## $AIC  
## [1] -3.447084  
##   
## $AICc  
## [1] -3.44621  
##   
## $BIC  
## [1] -3.357126

sarima(diff(log(all\_p)), p=9, d=0, q=0) # AR(9)

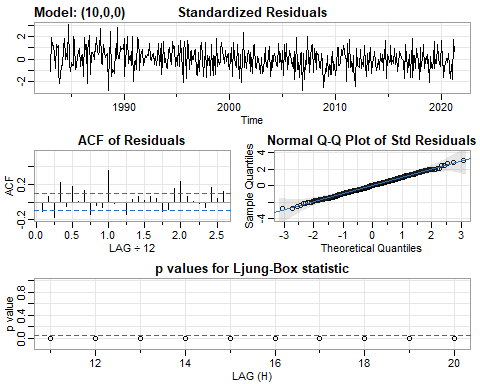
## initial value -2.464181   
## iter 2 value -2.672072  
## iter 3 value -2.802841  
## iter 4 value -2.843741  
## iter 5 value -3.038175  
## iter 6 value -3.091677  
## iter 7 value -3.125938  
## iter 8 value -3.133829  
## iter 9 value -3.155723  
## iter 10 value -3.165121  
## iter 11 value -3.169150  
## iter 12 value -3.172499  
## iter 13 value -3.174652  
## iter 14 value -3.174904  
## iter 15 value -3.174920  
## iter 16 value -3.174920  
## iter 16 value -3.174920  
## iter 16 value -3.174920  
## final value -3.174920   
## converged  
## initial value -3.164284   
## iter 2 value -3.164331  
## iter 3 value -3.164368  
## iter 4 value -3.164384  
## iter 5 value -3.164397  
## iter 6 value -3.164409  
## iter 7 value -3.164421  
## iter 8 value -3.164422  
## iter 9 value -3.164423  
## iter 10 value -3.164424  
## iter 11 value -3.164424  
## iter 12 value -3.164424  
## iter 12 value -3.164424  
## iter 12 value -3.164424  
## final value -3.164424   
## converged



## $fit  
##   
## Call:  
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),   
## xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,   
## optim.control = list(trace = trc, REPORT = 1, reltol = tol))  
##   
## Coefficients:  
## ar1 ar2 ar3 ar4 ar5 ar6 ar7 ar8  
## -0.7699 -0.4720 0.0474 -0.3852 -0.2087 -0.4646 -0.3768 -0.5526  
## s.e. 0.0468 0.0536 0.0556 0.0512 0.0532 0.0509 0.0556 0.0538  
## ar9 xmean  
## 0.0180 0.0027  
## s.e. 0.0473 0.0005  
##   
## sigma^2 estimated as 0.001764: log likelihood = 801.18, aic = -1580.36  
##   
## $degrees\_of\_freedom  
## [1] 449  
##   
## $ttable  
## Estimate SE t.value p.value  
## ar1 -0.7699 0.0468 -16.4496 0.0000  
## ar2 -0.4720 0.0536 -8.8026 0.0000  
## ar3 0.0474 0.0556 0.8525 0.3944  
## ar4 -0.3852 0.0512 -7.5214 0.0000  
## ar5 -0.2087 0.0532 -3.9233 0.0001  
## ar6 -0.4646 0.0509 -9.1254 0.0000  
## ar7 -0.3768 0.0556 -6.7771 0.0000  
## ar8 -0.5526 0.0538 -10.2680 0.0000  
## ar9 0.0180 0.0473 0.3804 0.7038  
## xmean 0.0027 0.0005 5.6127 0.0000  
##   
## $AIC  
## [1] -3.443041  
##   
## $AICc  
## [1] -3.441971  
##   
## $BIC  
## [1] -3.344088

sarima(diff(log(all\_p)), p=10, d=0, q=0) # AR(10)

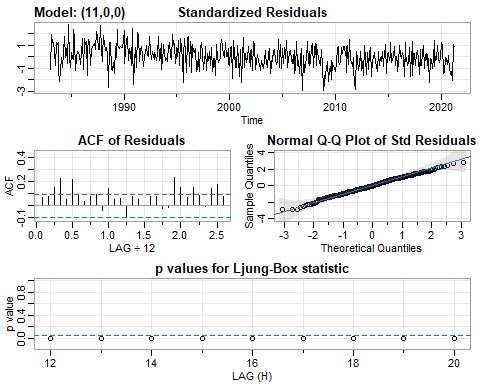
## initial value -2.464056   
## iter 2 value -2.659213  
## iter 3 value -2.883117  
## iter 4 value -2.984988  
## iter 5 value -3.080437  
## iter 6 value -3.102941  
## iter 7 value -3.125926  
## iter 8 value -3.148332  
## iter 9 value -3.181323  
## iter 10 value -3.183598  
## iter 11 value -3.210294  
## iter 12 value -3.217205  
## iter 13 value -3.218067  
## iter 14 value -3.218292  
## iter 15 value -3.218303  
## iter 16 value -3.218303  
## iter 17 value -3.218304  
## iter 17 value -3.218304  
## iter 17 value -3.218304  
## final value -3.218304   
## converged  
## initial value -3.206385   
## iter 2 value -3.206409  
## iter 3 value -3.206462  
## iter 4 value -3.206517  
## iter 5 value -3.206537  
## iter 6 value -3.206555  
## iter 7 value -3.206570  
## iter 8 value -3.206586  
## iter 9 value -3.206591  
## iter 10 value -3.206593  
## iter 11 value -3.206593  
## iter 12 value -3.206593  
## iter 13 value -3.206593  
## iter 13 value -3.206594  
## iter 13 value -3.206594  
## final value -3.206594   
## converged



## $fit  
##   
## Call:  
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),   
## xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,   
## optim.control = list(trace = trc, REPORT = 1, reltol = tol))  
##   
## Coefficients:  
## ar1 ar2 ar3 ar4 ar5 ar6 ar7 ar8  
## -0.7640 -0.6292 -0.0605 -0.5196 -0.2687 -0.5737 -0.3624 -0.6900  
## s.e. 0.0448 0.0570 0.0559 0.0534 0.0520 0.0516 0.0533 0.0558  
## ar9 ar10 xmean  
## -0.2058 -0.2885 0.0027  
## s.e. 0.0574 0.0453 0.0004  
##   
## sigma^2 estimated as 0.001618: log likelihood = 820.53, aic = -1617.07  
##   
## $degrees\_of\_freedom  
## [1] 448  
##   
## $ttable  
## Estimate SE t.value p.value  
## ar1 -0.7640 0.0448 -17.0463 0.0000  
## ar2 -0.6292 0.0570 -11.0433 0.0000  
## ar3 -0.0605 0.0559 -1.0823 0.2797  
## ar4 -0.5196 0.0534 -9.7338 0.0000  
## ar5 -0.2687 0.0520 -5.1711 0.0000  
## ar6 -0.5737 0.0516 -11.1090 0.0000  
## ar7 -0.3624 0.0533 -6.8049 0.0000  
## ar8 -0.6900 0.0558 -12.3745 0.0000  
## ar9 -0.2058 0.0574 -3.5887 0.0004  
## ar10 -0.2885 0.0453 -6.3690 0.0000  
## xmean 0.0027 0.0004 7.4592 0.0000  
##   
## $AIC  
## [1] -3.523022  
##   
## $AICc  
## [1] -3.521736  
##   
## $BIC  
## [1] -3.415073

sarima(diff(log(all\_p)), p=11, d=0, q=0) # AR(11)

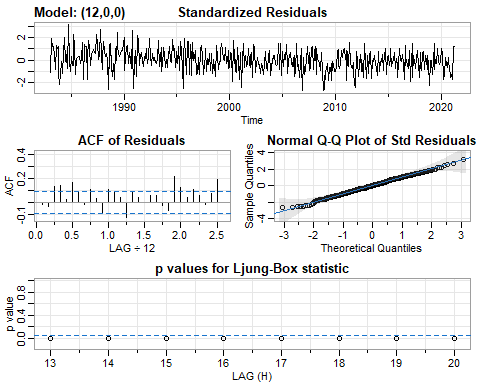
## initial value -2.463928   
## iter 2 value -2.673834  
## iter 3 value -2.815646  
## iter 4 value -2.938704  
## iter 5 value -3.095876  
## iter 6 value -3.120999  
## iter 7 value -3.165648  
## iter 8 value -3.180609  
## iter 9 value -3.236583  
## iter 10 value -3.240540  
## iter 11 value -3.281513  
## iter 12 value -3.296276  
## iter 13 value -3.298287  
## iter 14 value -3.299164  
## iter 15 value -3.299190  
## iter 16 value -3.299206  
## iter 17 value -3.299207  
## iter 18 value -3.299207  
## iter 18 value -3.299207  
## iter 18 value -3.299207  
## final value -3.299207   
## converged  
## initial value -3.280281   
## iter 2 value -3.280389  
## iter 3 value -3.280485  
## iter 4 value -3.280653  
## iter 5 value -3.280743  
## iter 6 value -3.280770  
## iter 7 value -3.280856  
## iter 8 value -3.280970  
## iter 9 value -3.281014  
## iter 10 value -3.281083  
## iter 11 value -3.281105  
## iter 12 value -3.281109  
## iter 13 value -3.281112  
## iter 14 value -3.281113  
## iter 15 value -3.281113  
## iter 15 value -3.281113  
## iter 15 value -3.281113  
## final value -3.281113   
## converged



## $fit  
##   
## Call:  
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),   
## xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,   
## optim.control = list(trace = trc, REPORT = 1, reltol = tol))  
##   
## Coefficients:  
## ar1 ar2 ar3 ar4 ar5 ar6 ar7 ar8  
## -0.8700 -0.7002 -0.3135 -0.6547 -0.4864 -0.6748 -0.5515 -0.7092  
## s.e. 0.0432 0.0533 0.0594 0.0518 0.0544 0.0493 0.0538 0.0515  
## ar9 ar10 ar11 xmean  
## -0.4453 -0.5825 -0.3790 0.0027  
## s.e. 0.0597 0.0538 0.0439 0.0002  
##   
## sigma^2 estimated as 0.001388: log likelihood = 854.74, aic = -1683.48  
##   
## $degrees\_of\_freedom  
## [1] 447  
##   
## $ttable  
## Estimate SE t.value p.value  
## ar1 -0.8700 0.0432 -20.1236 0  
## ar2 -0.7002 0.0533 -13.1370 0  
## ar3 -0.3135 0.0594 -5.2767 0  
## ar4 -0.6547 0.0518 -12.6490 0  
## ar5 -0.4864 0.0544 -8.9385 0  
## ar6 -0.6748 0.0493 -13.6999 0  
## ar7 -0.5515 0.0538 -10.2482 0  
## ar8 -0.7092 0.0515 -13.7716 0  
## ar9 -0.4453 0.0597 -7.4529 0  
## ar10 -0.5825 0.0538 -10.8220 0  
## ar11 -0.3790 0.0439 -8.6293 0  
## xmean 0.0027 0.0002 10.9275 0  
##   
## $AIC  
## [1] -3.667703  
##   
## $AICc  
## [1] -3.666179  
##   
## $BIC  
## [1] -3.550759

sarima(diff(log(all\_p)), p=12, d=0, q=0) # AR(12)

## initial value -2.462942   
## iter 2 value -2.784242  
## iter 3 value -3.100059  
## iter 4 value -3.139230  
## iter 5 value -3.181578  
## iter 6 value -3.218946  
## iter 7 value -3.229287  
## iter 8 value -3.230789  
## iter 9 value -3.255590  
## iter 10 value -3.273590  
## iter 11 value -3.280609  
## iter 12 value -3.294054  
## iter 13 value -3.315383  
## iter 14 value -3.316446  
## iter 15 value -3.317740  
## iter 16 value -3.317794  
## iter 17 value -3.317797  
## iter 18 value -3.317797  
## iter 18 value -3.317797  
## iter 18 value -3.317797  
## final value -3.317797   
## converged  
## initial value -3.297174   
## iter 2 value -3.297241  
## iter 3 value -3.297365  
## iter 4 value -3.297714  
## iter 5 value -3.297841  
## iter 6 value -3.297937  
## iter 7 value -3.298107  
## iter 8 value -3.298137  
## iter 9 value -3.298179  
## iter 10 value -3.298247  
## iter 11 value -3.298308  
## iter 12 value -3.298346  
## iter 13 value -3.298354  
## iter 14 value -3.298356  
## iter 15 value -3.298358  
## iter 16 value -3.298358  
## iter 16 value -3.298358  
## iter 16 value -3.298358  
## final value -3.298358   
## converged



## $fit  
##   
## Call:  
## arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D, Q), period = S),   
## xreg = xmean, include.mean = FALSE, transform.pars = trans, fixed = fixed,   
## optim.control = list(trace = trc, REPORT = 1, reltol = tol))  
##   
## Coefficients:  
## ar1 ar2 ar3 ar4 ar5 ar6 ar7 ar8  
## -0.7999 -0.5945 -0.2345 -0.5252 -0.3829 -0.5479 -0.4619 -0.5904  
## s.e. 0.0459 0.0586 0.0616 0.0602 0.0594 0.0577 0.0574 0.0587  
## ar9 ar10 ar11 ar12 xmean  
## -0.3865 -0.4476 -0.2118 0.1880 0.0027  
## s.e. 0.0606 0.0628 0.0600 0.0468 0.0003  
##   
## sigma^2 estimated as 0.00134: log likelihood = 862.65, aic = -1697.31  
##   
## $degrees\_of\_freedom  
## [1] 446  
##   
## $ttable  
## Estimate SE t.value p.value  
## ar1 -0.7999 0.0459 -17.4274 0e+00  
## ar2 -0.5945 0.0586 -10.1410 0e+00  
## ar3 -0.2345 0.0616 -3.8065 2e-04  
## ar4 -0.5252 0.0602 -8.7197 0e+00  
## ar5 -0.3829 0.0594 -6.4504 0e+00  
## ar6 -0.5479 0.0577 -9.4971 0e+00  
## ar7 -0.4619 0.0574 -8.0456 0e+00  
## ar8 -0.5904 0.0587 -10.0566 0e+00  
## ar9 -0.3865 0.0606 -6.3819 0e+00  
## ar10 -0.4476 0.0628 -7.1330 0e+00  
## ar11 -0.2118 0.0600 -3.5283 5e-04  
## ar12 0.1880 0.0468 4.0186 1e-04  
## xmean 0.0027 0.0003 9.1108 0e+00  
##   
## $AIC  
## [1] -3.697837  
##   
## $AICc  
## [1] -3.696054  
##   
## $BIC  
## [1] -3.571896

#### Yule-Walker Estimation

# all red meat  
rmp.yw = ar.yw(all\_rmp, order=12)  
rmp.yw$x.mean # mean estimate

## [1] 3738.141

rmp.yw$ar # phi parameter estimates

## [1] 0.44075593 0.09600863 0.41482443 -0.43528207 0.38147023 -0.29479448  
## [7] 0.19712510 -0.25245246 0.38504390 -0.30020108 0.15334287 0.19541801

sqrt(diag(rmp.yw$asy.var.coef)) # their standard errors

## [1] 0.04638647 0.05033586 0.04850472 0.04905097 0.05183909 0.05409165  
## [7] 0.05409165 0.05183909 0.04905097 0.04850472 0.05033586 0.04638647

# all poultry  
p.yw = ar.yw(all\_p, order=12)  
p.yw$x.mean # mean estimate

## [1] 2957.05

p.yw$ar # phi parameter estimates

## [1] 0.55792085 0.19755941 0.43744646 -0.47996689 0.34886169 -0.25032792  
## [7] 0.20226884 -0.26970524 0.39375872 -0.29757828 0.07303883 0.07698175

sqrt(diag(p.yw$asy.var.coef)) # their standard errors

## [1] 0.04715802 0.05392875 0.05289159 0.05365407 0.05684534 0.05841348  
## [7] 0.05841348 0.05684534 0.05365407 0.05289159 0.05392875 0.04715802