Project 2: Cal Parker and Jordan Youhanaie

2023-05-16

```
# Load libraries
library(tseries)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
library(tseries)
library(seasonal)
library(stats)
require(graphics)
library(dynlm)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
library(tidyverse)
```

```
## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2
## ──
```

```
## ✓ ggplot2 3.4.1     ✓ purrr   1.0.1
## ✓ tibble  3.1.8     ✓ dplyr   1.1.0
## ✓ tidyr   1.3.0     ✓ stringr 1.5.0
## ✓ readr   2.1.4     ✓ forcats 1.0.0
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✕ dplyr::filter() masks stats::filter()
## ✕ dplyr::lag()    masks stats::lag()
## ✕ tibble::view()  masks seasonal::view()
```

```
library(readxl)
library(tsibble)
```

```
##
## Attaching package: 'tsibble'
##
## The following object is masked from 'package:zoo':
##
##     index
##
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, union
```

```
library(tseries)
library(forecast)
library(fpp3)
```

```
## ── Attaching packages ──────────────────────────────────────────── fpp3 0.5 ──
## ✓ lubridate   1.9.2     ✓ fable       0.3.3
## ✓ tsibbledata 0.4.1     ✓ fabletools  0.3.3
## ✓ feasts      0.3.1
## ── Conflicts ─────────────────────────────────────────────── fpp3_conflicts ──
## ✗ lubridate::date()     masks base::date()
## ✗ dplyr::filter()       masks stats::filter()
## ✗ tsibble::index()      masks zoo::index()
## ✗ tsibble::intersect()  masks base::intersect()
## ✗ lubridate::interval() masks tsibble::interval()
## ✗ dplyr::lag()          masks stats::lag()
## ✗ tsibble::setdiff()    masks base::setdiff()
## ✗ tsibble::union()      masks base::union()
## ✗ tibble::view()        masks seasonal::view()
```

```
library(tseries)
library(seasonal)
library(fable)
library(stats)
require(graphics)
library(feasts)
library(vars)
```

```
## Loading required package: MASS
## 
## Attaching package: 'MASS'
## 
## The following object is masked from 'package:dplyr':
## 
##     select
## 
## Loading required package: strucchange
## Loading required package: sandwich
## 
## Attaching package: 'strucchange'
## 
## The following object is masked from 'package:stringr':
## 
##     boundary
## 
## Loading required package: urca
## Loading required package: lmtest
## 
## Attaching package: 'vars'
## 
## The following object is masked from 'package:fable':
## 
##     VAR
```

Section I

For this project, we used New Car Retail Sales data and New Home Supply from the United States. More specificially, the FRED database titles these datasets as “Retail Sales: New Car Dealers” and “Monthly Supply of New Houses in the United States” and the values that FRED presents come from the U.S. Census Bureau. Both housing and car sales data are vital to assessing the direction and size of an economy, and for the United States, they present massive portions of total transaction costs and economic activity for assessing the status of the economy. These datasets are updated monthly and begin in January 1992. The number of houses available for purchase in the U.S. was around 7.7 million homes to begin 2023 and the total value of retail sales for new cars was around 90 billion for the same moment.

Section II

```
#Read in the data and conver to time series
HousingData<-read_csv("MSACSRNSA (1).csv")
```

```
## Rows: 375 Columns: 2
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## dbl  (1): MSACSRNSA
## date (1): DATE
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

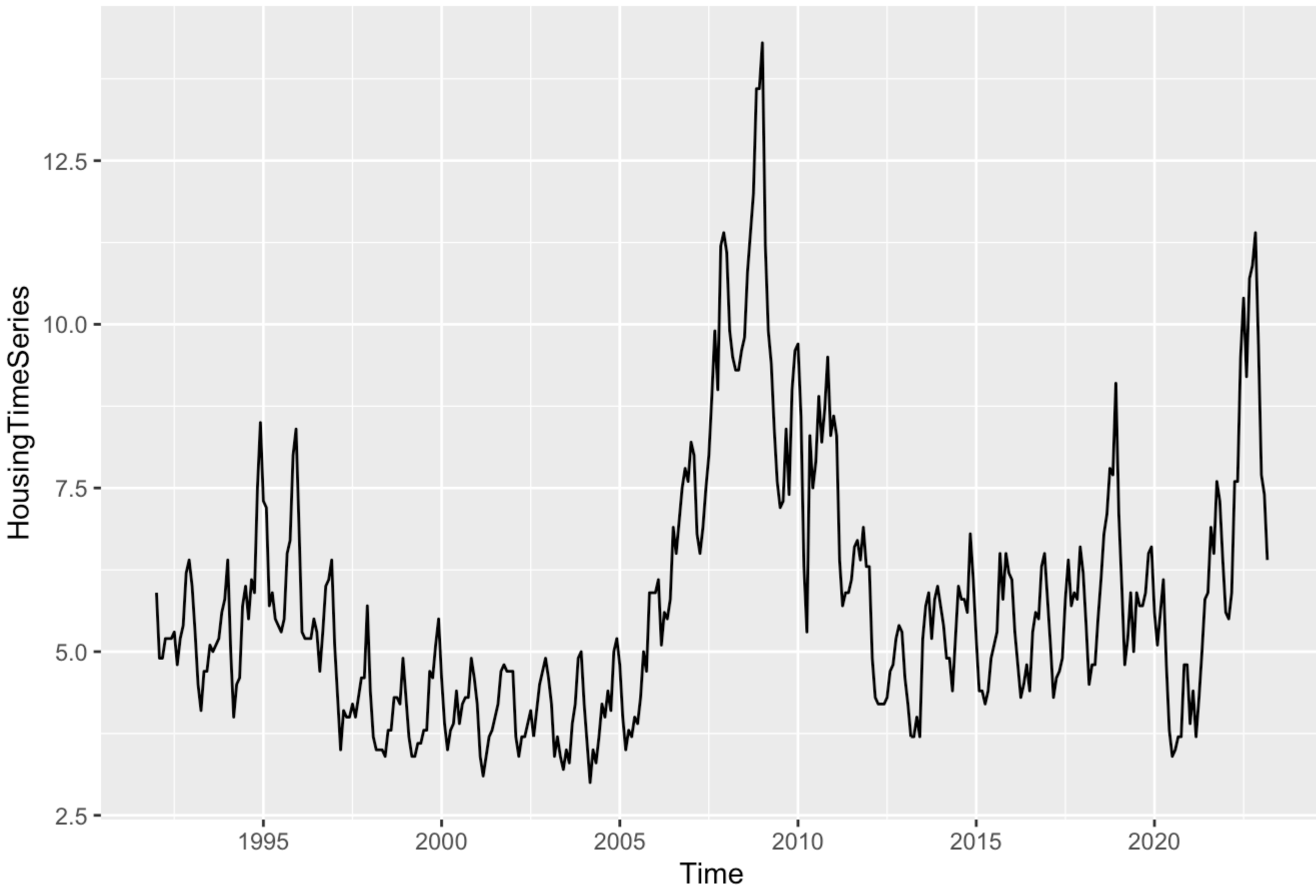
```
HousingTimeSeries<-ts(HousingData$MSACSRNSA, start = 1992, frequency = 12)

RetailNewCarsData<-read_csv("RetailNewCarsData (2).csv")
```

```
## Rows: 375 Columns: 2
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## dbl  (1): MRTSSM44111USN
## date (1): DATE
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

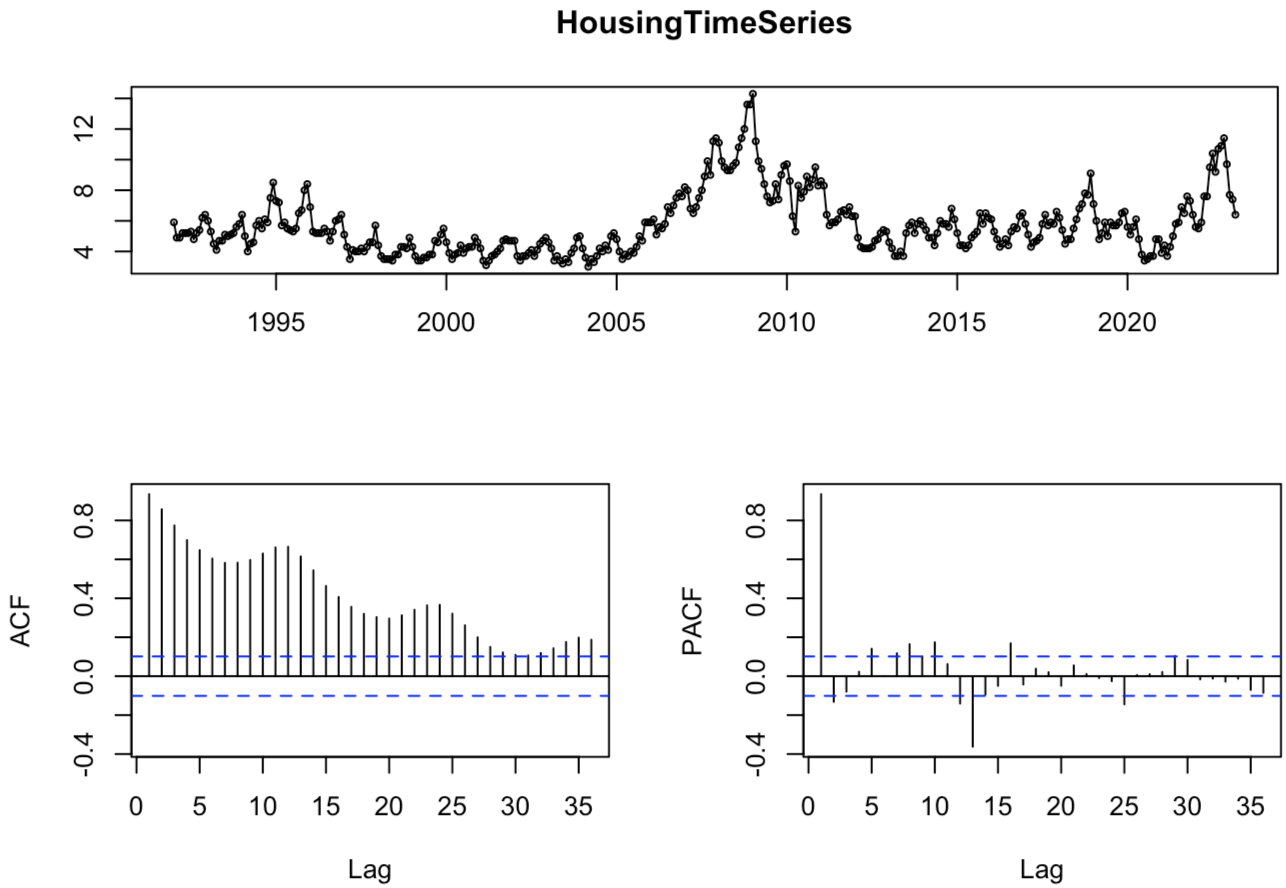
```
RetailNewCarsDataTS<-ts(RetailNewCarsData$MRTSSM44111USN, start = 1992, frequency = 12)
```

Part a

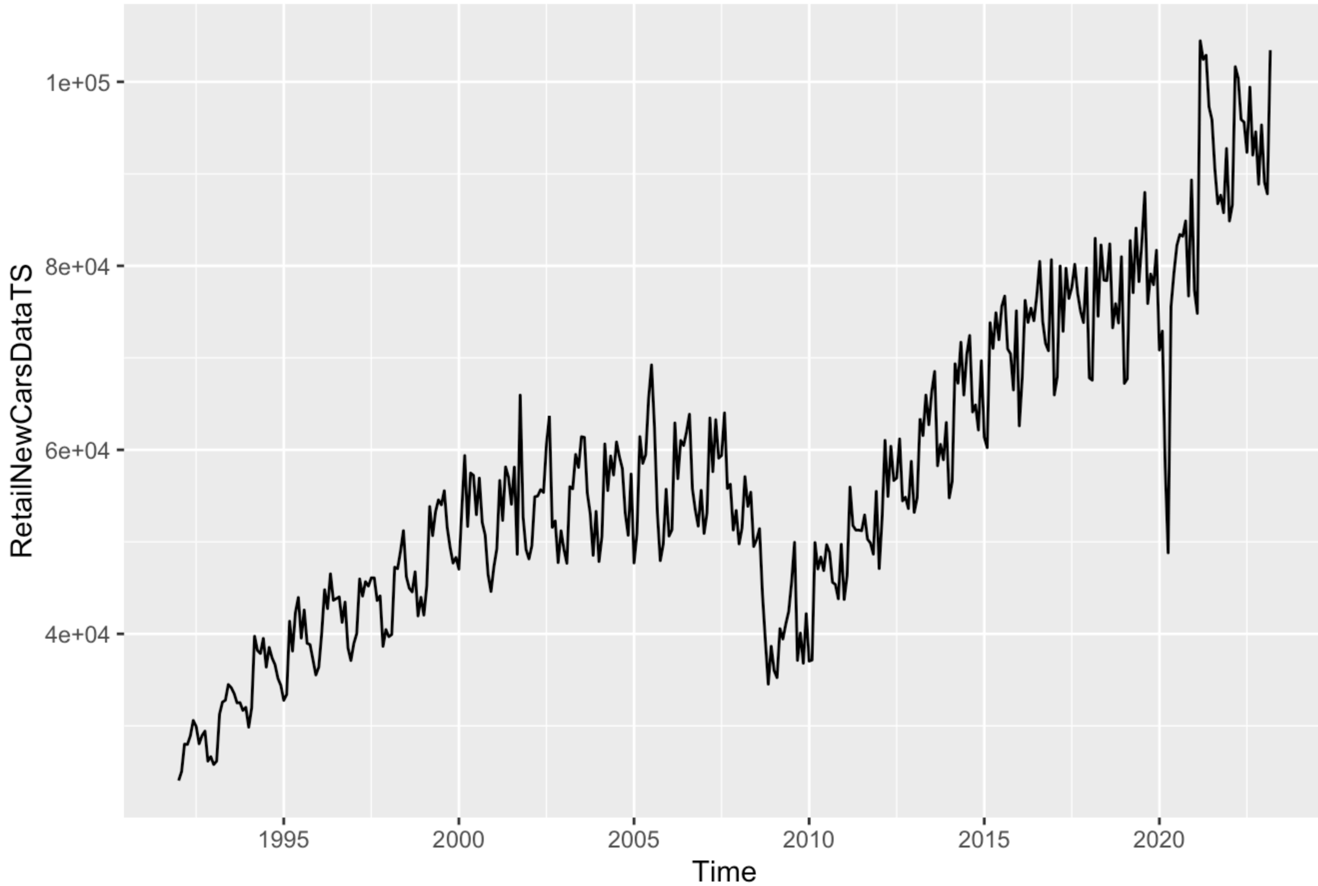
```
#plot time series individually and use tsdisplay() to examine ACF and PACF
autoplot(HousingTimeSeries)
```



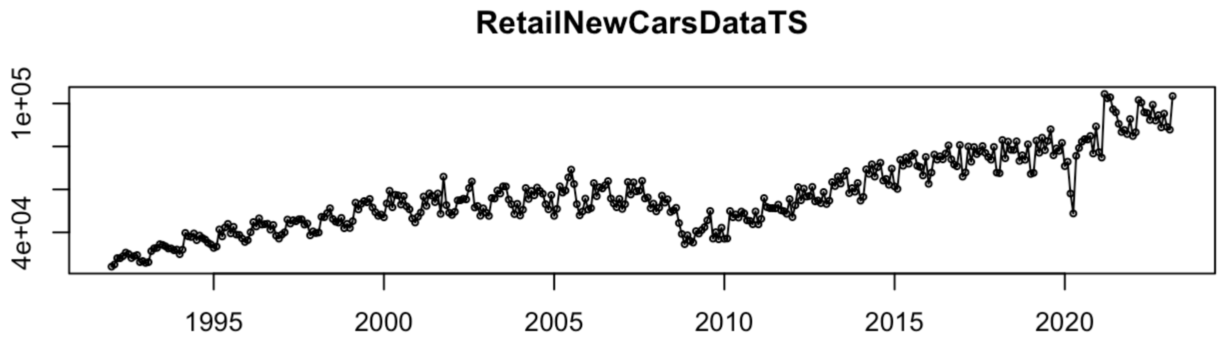

```
tsdisplay(HousingTimeSeries)
```

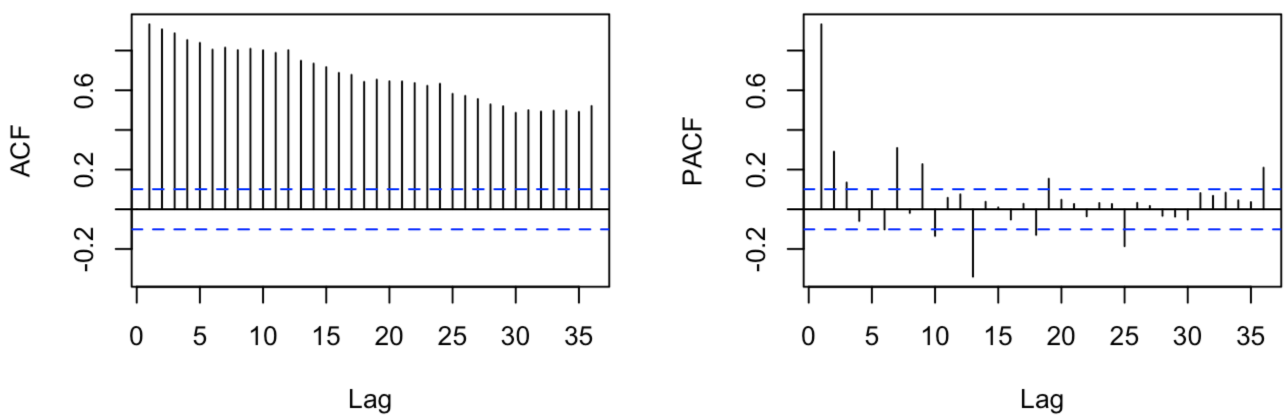



```
autoplot(RetailNewCarsDataTS)
```



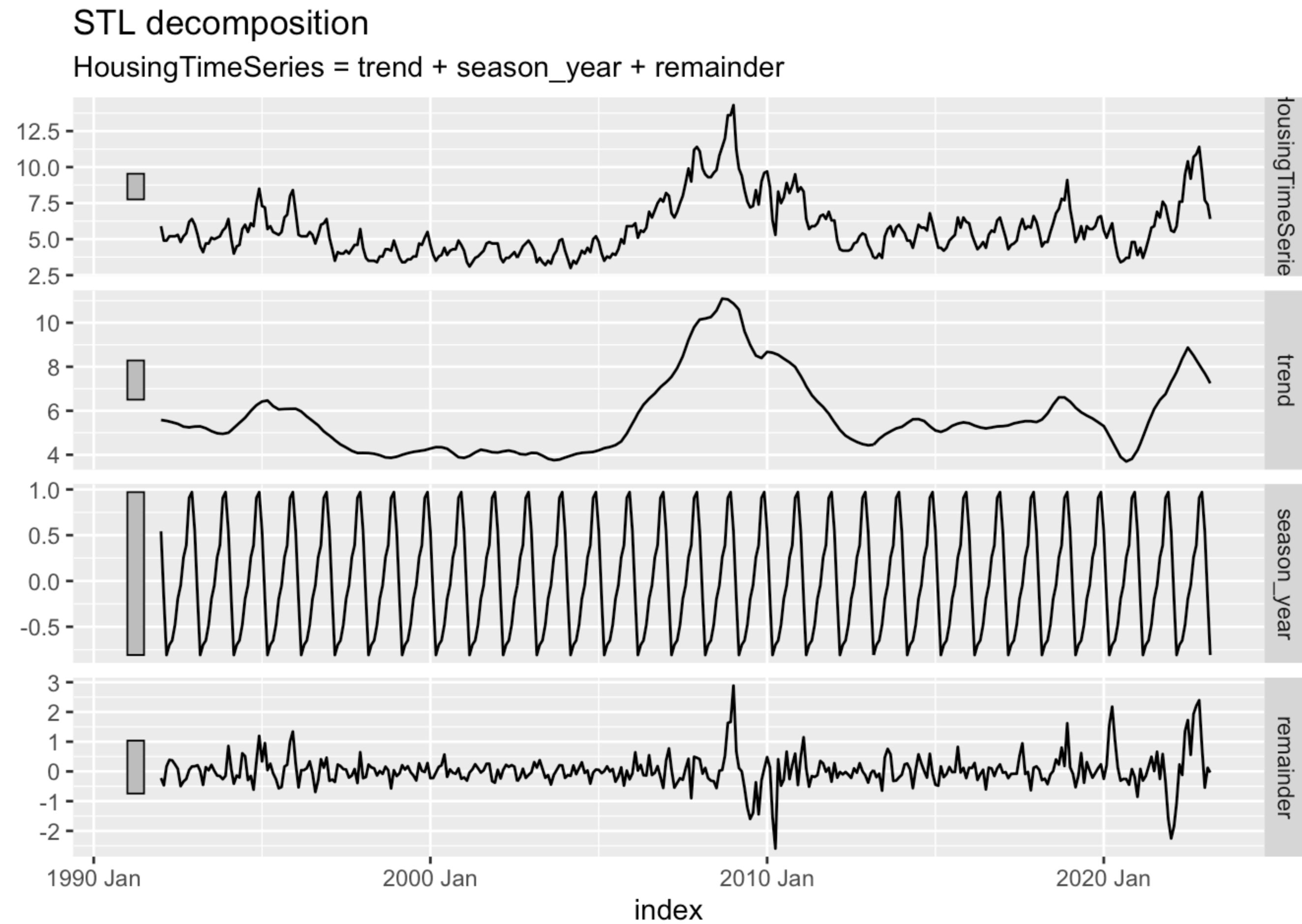

```
tsdisplay(RetailNewCarsDataTS)
```



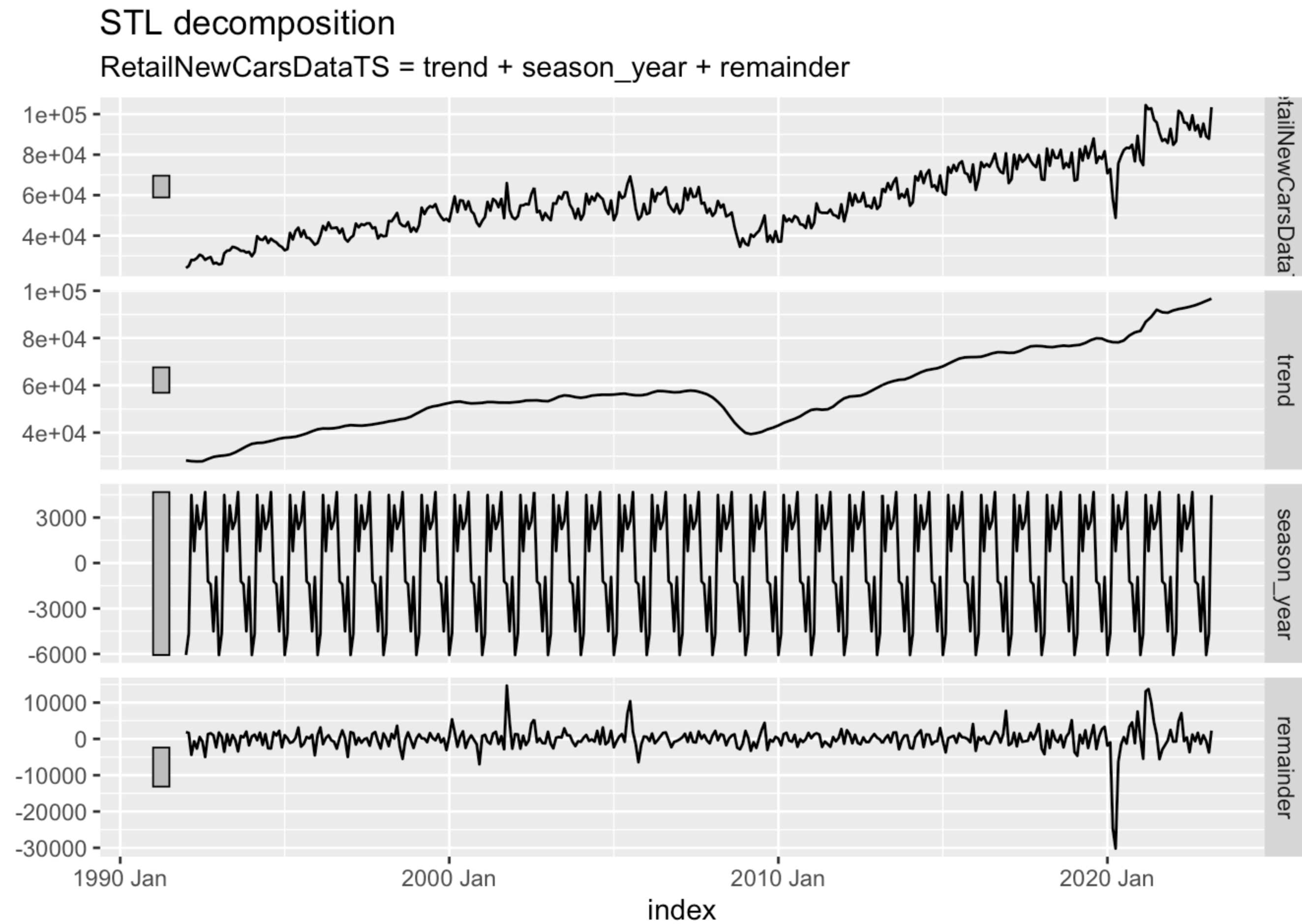


Part b

```
#Convert data to tsibble for decomposition
HousingTSTibble<-as_tsibble(HousingTimeSeries, key = origin)

#STL decomposition of Housing Supply data
HousingDecomp<-HousingTSTibble|>
  model(
    STL(HousingTimeSeries~ trend(window = 12) +
                      season(window = "periodic"),
    robust = TRUE)) |>
  components() |>
  autoplot()

#Provide graphical ouptput of decomposition
HousingDecomp
```



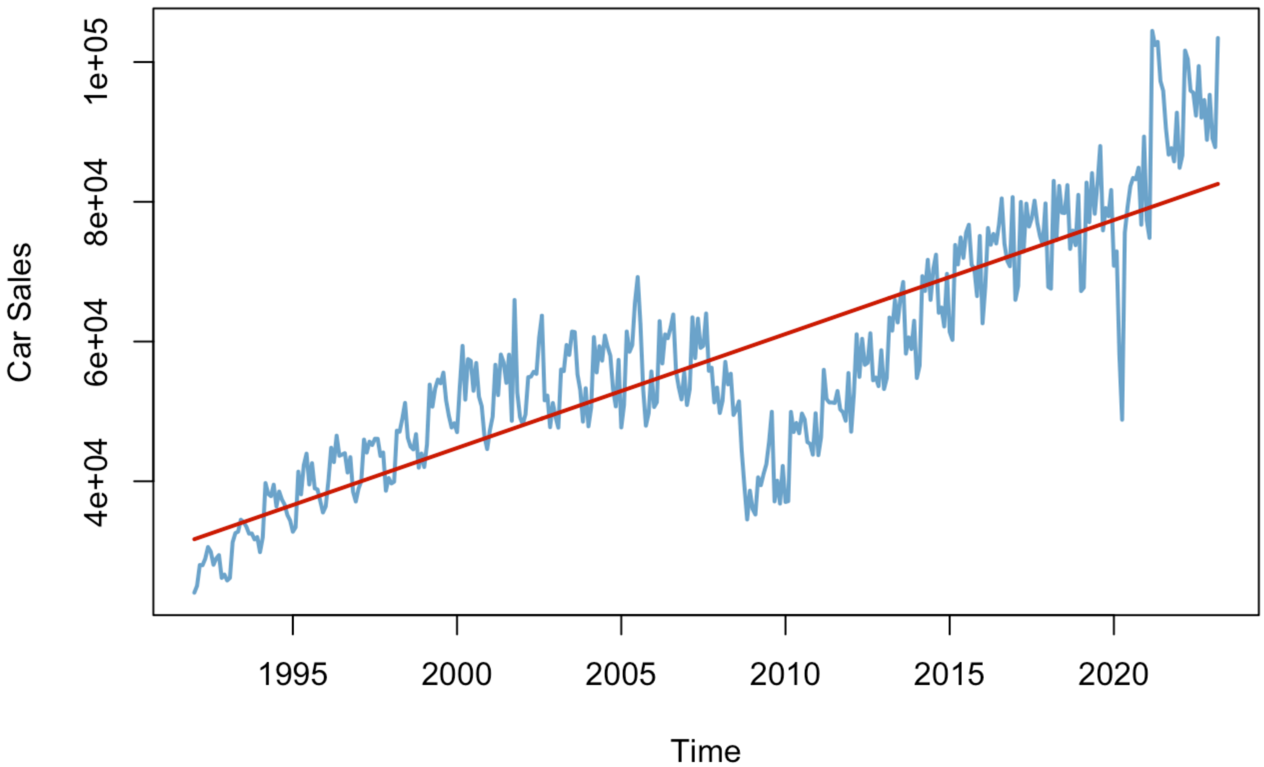

```
#same process as above but for New Car Sales data
RNCTSTibble<-as_tsibble(RetailNewCarsDataTS, key = origin)

RNCTSDecomp<-RNCTSTibble|>
  model(
    STL(RetailNewCarsDataTS~ trend(window = 12) +
                   season(window = "periodic"),
    robust = TRUE)) |>
  components() |>
  autoplot()

RNCTSDecomp
```



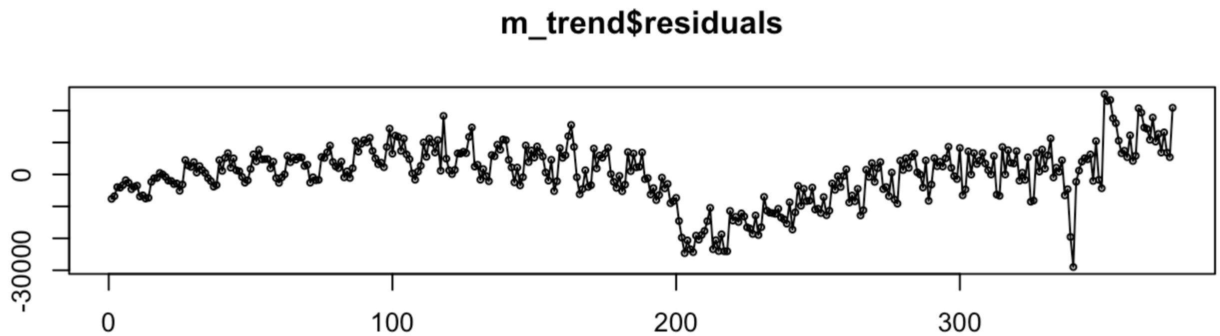
According to our stl decompositions for the new car retail sales and new housing supply data, there appears to be significant seasonal variation for the new car retail sales, and significant seasonal variation for the new housing supply time series. The trend for new car retail sales increases almost constantly while the new housing supply increases around 2008 and the levels out back to its previous levels before 2008. This change in new housing supply is most likely caused by the financial crisis since there were so many homes that were not accessible to creditor that could have previously bought the same houses. Also, there is massive variation around 2008 for the remainder portion of the decomposition of supply of housing.

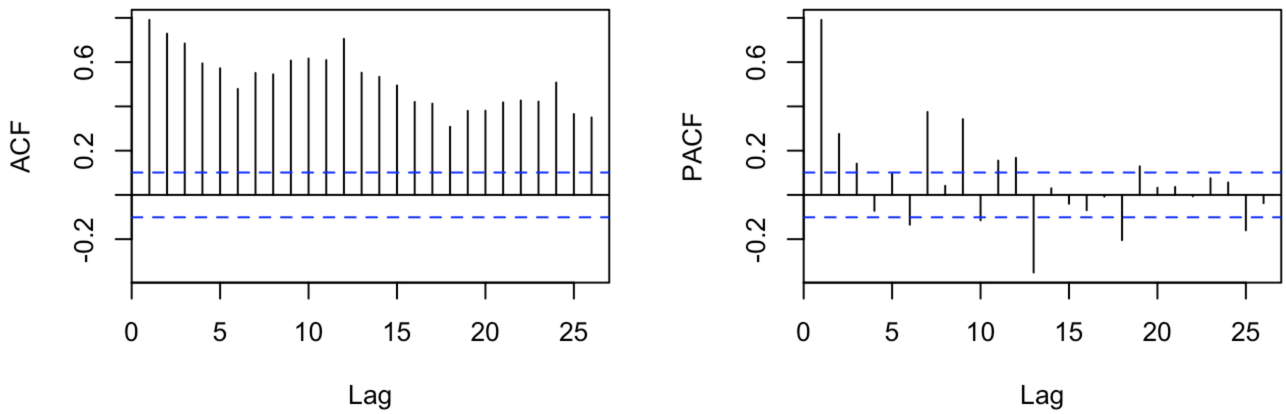
Part c

```
#Create time variable for trend fitting and plotting, can be used for both time series since the
cover the same time frame
t <- seq(1992, 2023 + 2/12,length=length(RetailNewCarsDataTS))

#linear model for trend of New Car Sales data
m_trend <- lm(RetailNewCarsDataTS ~ t)
plot(RetailNewCarsDataTS, ylab="Car Sales", xlab="Time", lwd=2, col='skyblue3', xlim=c(1992,2023
+ 2/12))
lines(t,m_trend$fit,col="red3",lwd=2)
```

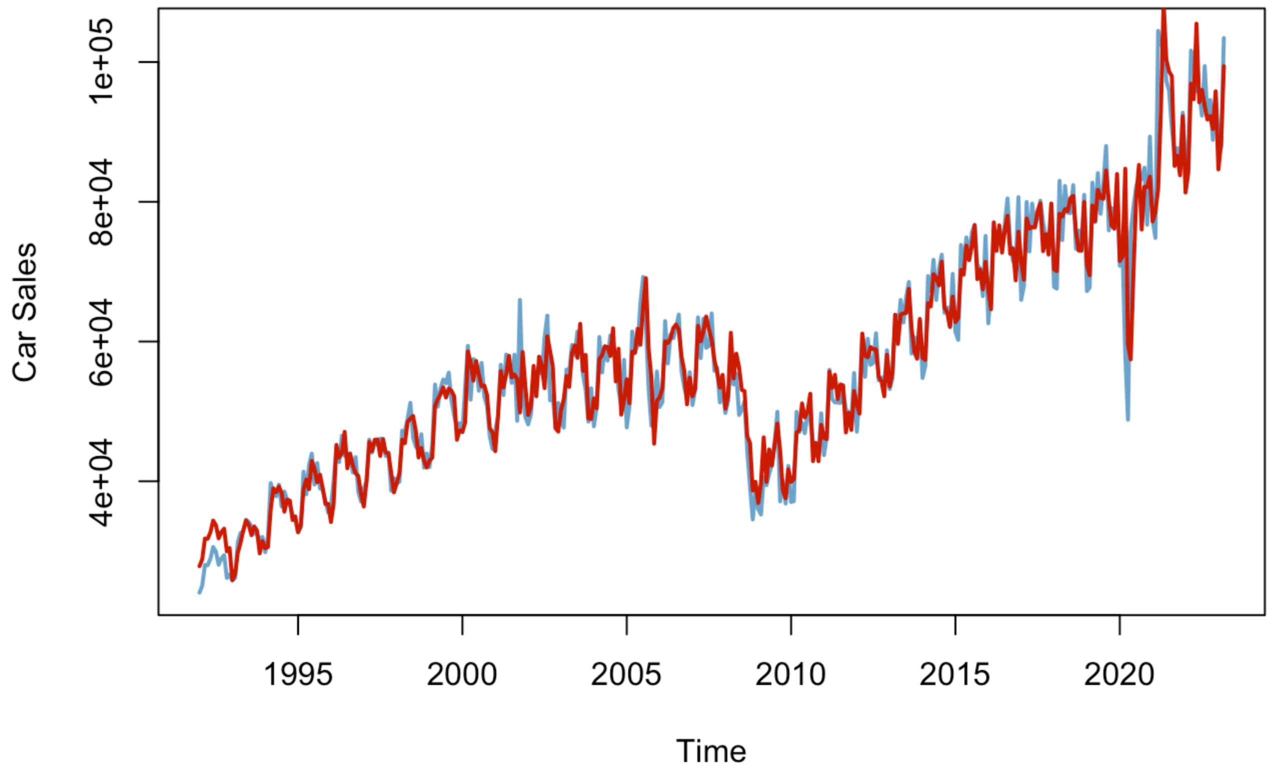



```
#ACF and PACF of residuals of linear trend to find season and cycles
tsdisplay(m_trend$residuals)
```

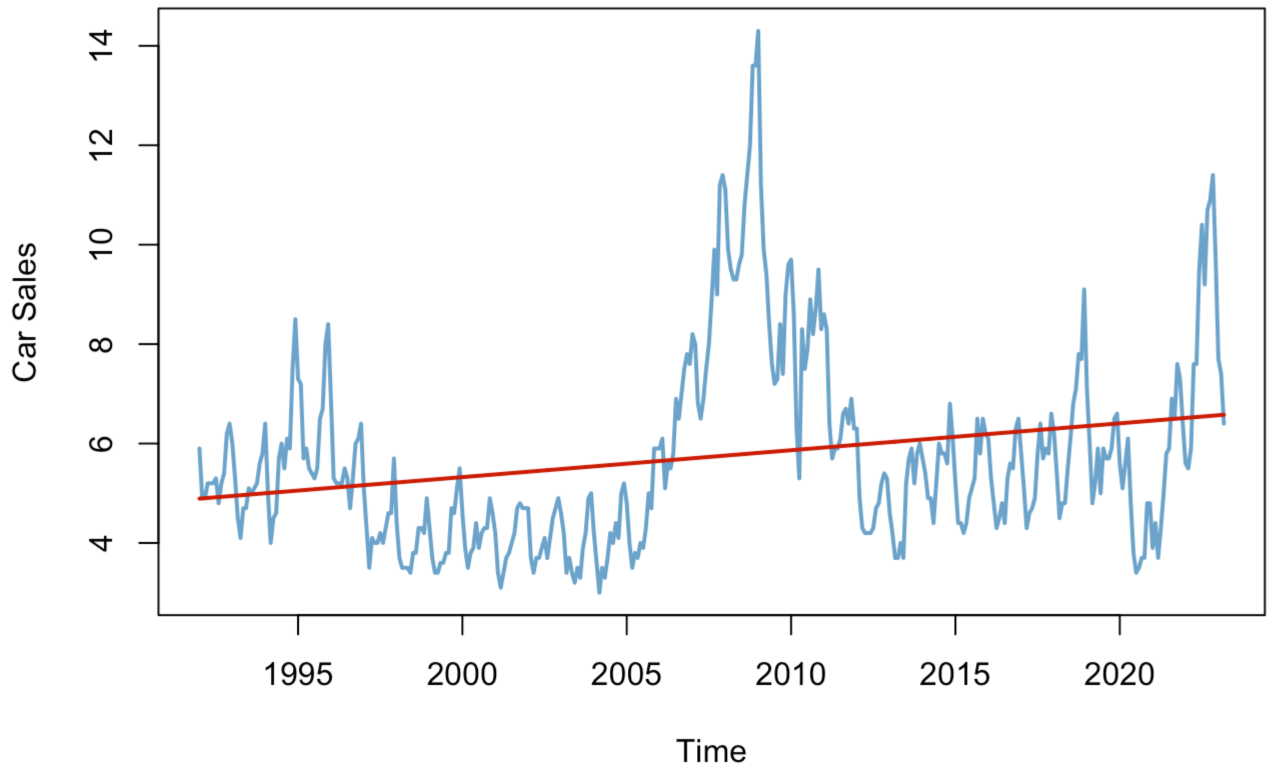




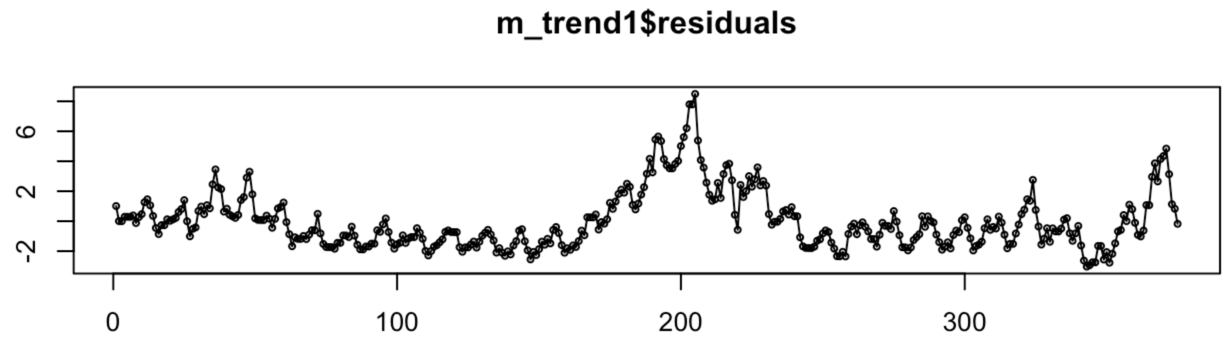

```
#Fit and plot Arima model based on information found above
fit <- Arima(RetailNewCarsDataTS, order=c(2,0,0), xreg = t, seasonal=list(order=c(1,1,1)))
plot(RetailNewCarsDataTS, ylab="Car Sales", xlab="Time", lwd=2, col='skyblue3', xlim=c(1992,2023
+ 2/12))
lines(t, fit$fit,col="red3",lwd=2)
```

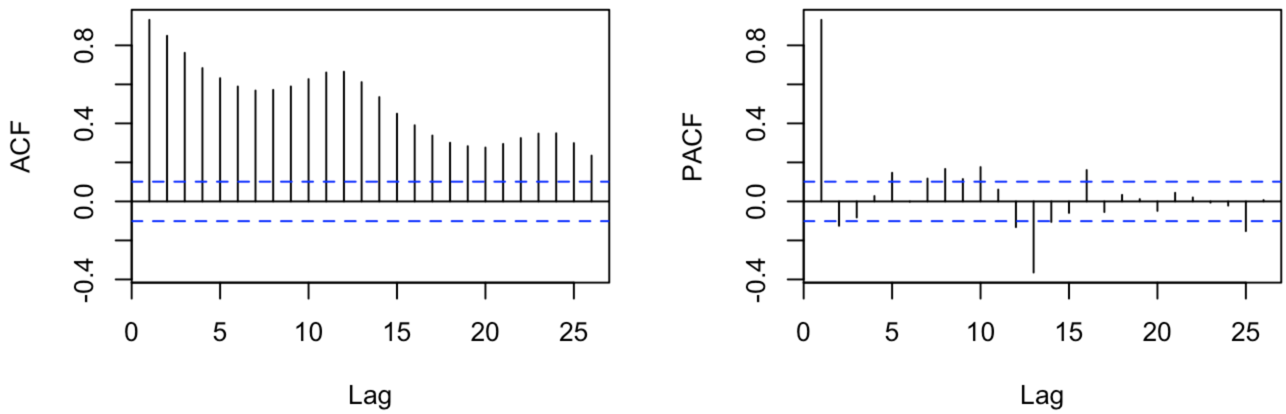



```
#linear model for trend of Housing Supply data
m_trend1 <- lm(HousingTimeSeries ~ t)
plot(HousingTimeSeries, ylab="Car Sales", xlab="Time", lwd=2, col='skyblue3', xlim=c(1992,2023 +
2/12))
lines(t,m_trend1$fit,col="red3",lwd=2)
```

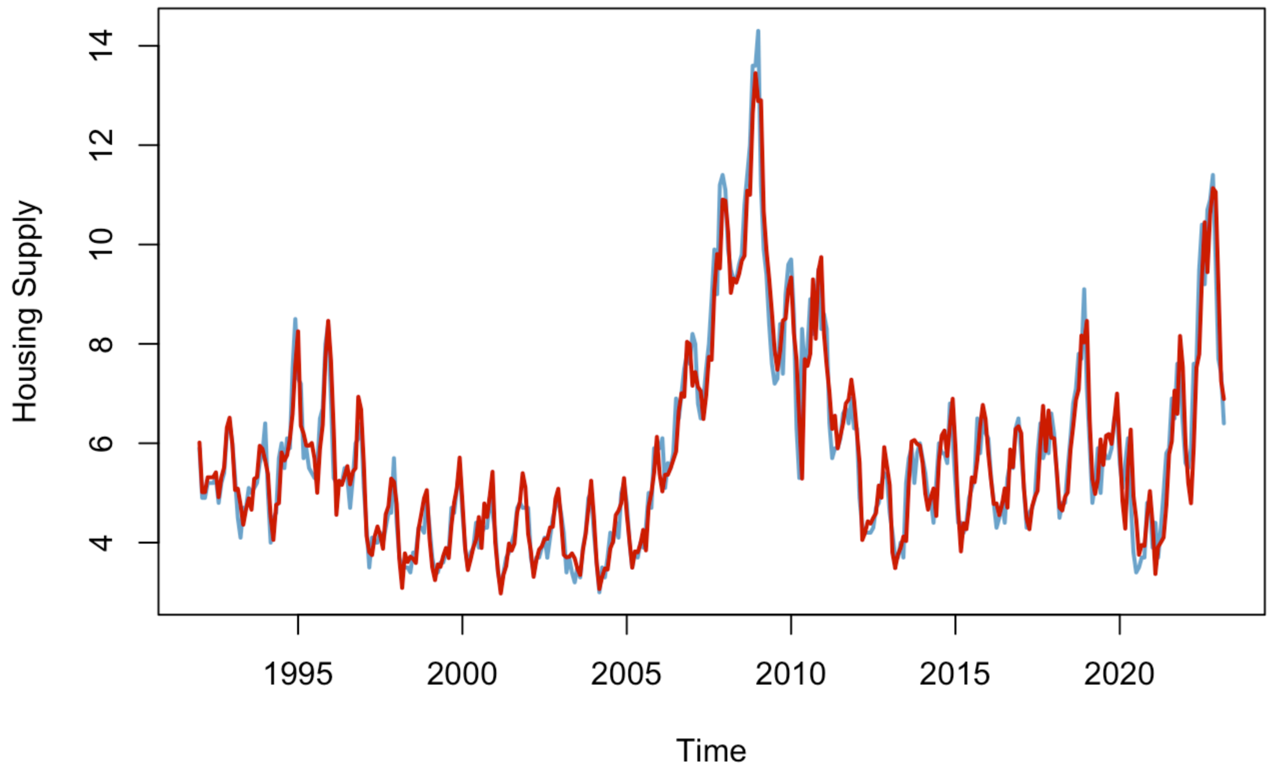



```
#ACF and PACF of residuals of linear trend to find season and cycles
tsdisplay(m_trend1$residuals)
```





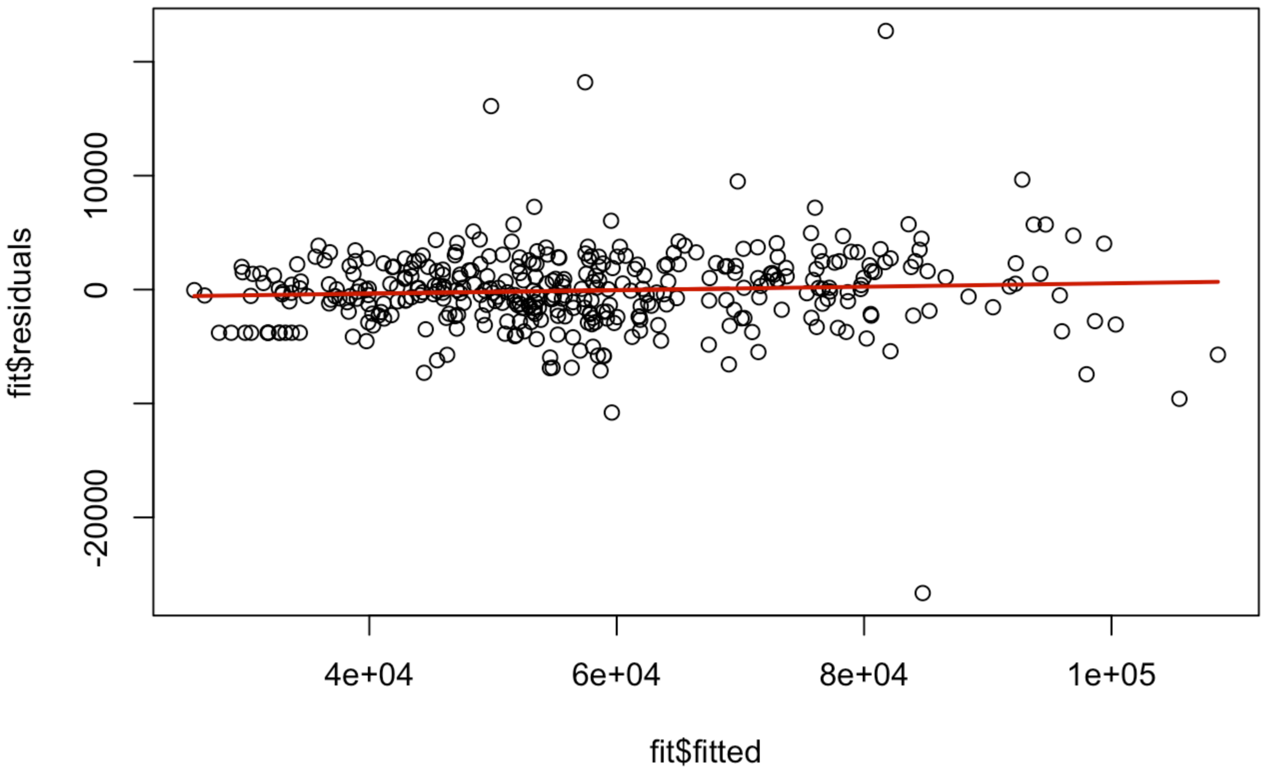

```
#Fit and plot Arima model based on information found above
fit2 = Arima(HousingTimeSeries, order=c(2,0,1), xreg = t, seasonal=list(order=c(1,1,1)))
plot(HousingTimeSeries, ylab="Housing Supply", xlab="Time", lwd=2, col='skyblue3', xlim=c(1992,2
023 + 2/12))
lines(t, fit2$fit,col="red3",lwd=2)
```



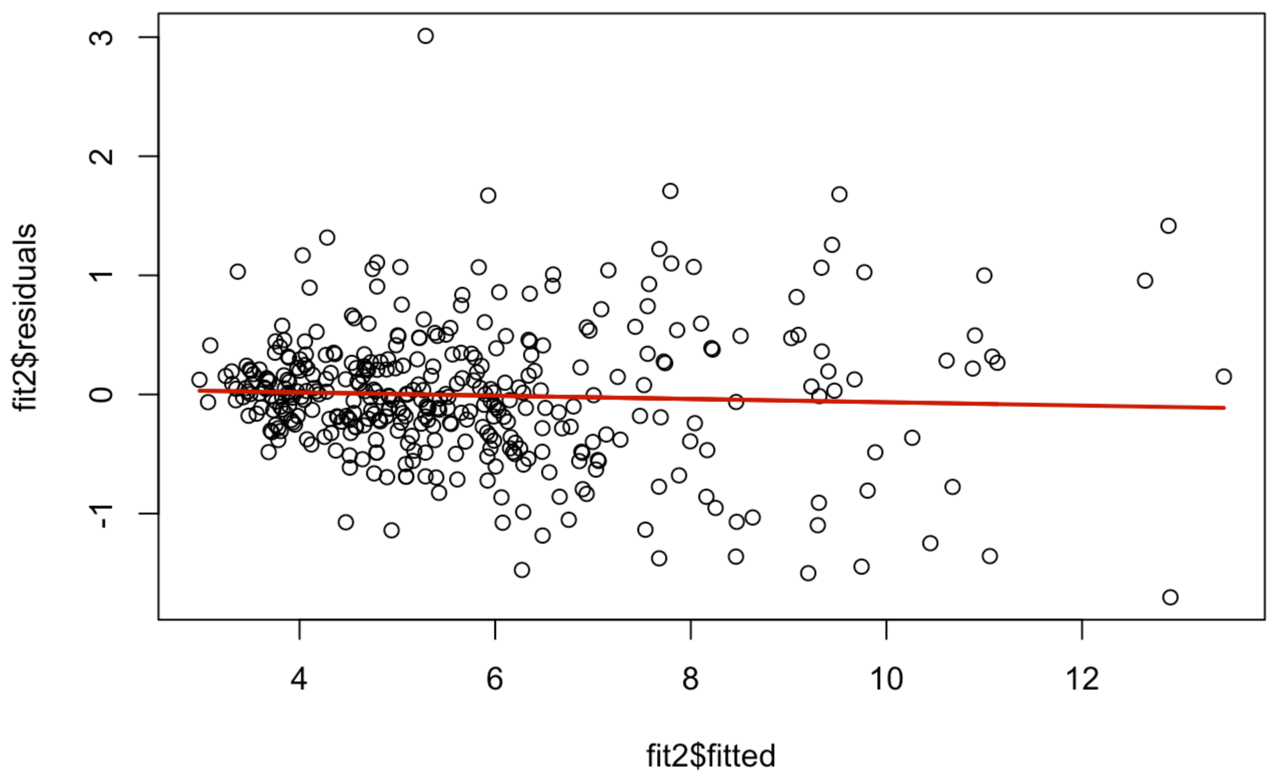
To be able to determine the seasonality and cyclical components of each model, we must first identify the trend and create a basic trend model. We must then analyze the ACF and PACF of the residuals for our trend models to determine the seasonal and cyclical nature of the data and create an appropriate Arima model that accounts for these things as well as trend we had previously determined. For both of our time series, we concluded that a simple linear trend was best, and we created and plotted these models accordingly. We then looked at the tsdisplay() outputs for the residuals of these simple linear models to find our Arima components. The ACF and PACF plots of the residulas for housing supply suggest that there is seasonal variation of residuals between both the autoregressive and moving average components of the charts because of the increasing and decaying components of the ACF and alternating PACF lag directions with significant spikes around 12 months apart from each other. The ACF and PACF plots of the residuals for the new car retail sales model suggests that there is also seasonal variation of residuals for the autoregressive and moving average components of the charts for the same reasons as the as the new housing supply ACF and PACF charts. There are common relationships among the seasonal variation between the two time series plots. We then fit Arima models for each time series based on the information described above, reincorporating the trend components with the xreg = t argument. Upon plotting the model fits against the time series, we can see that both models do a very good job of explaining the data.

Part e

```
#Residuals vs. Fitted Values plots for both models (must be done manually)
plot(fit$fitted, fit$residuals)
lines(as.double(fit$fitted), lm(fit$residuals ~ fit$fitted)$fit,col="red3",lwd=2)
```



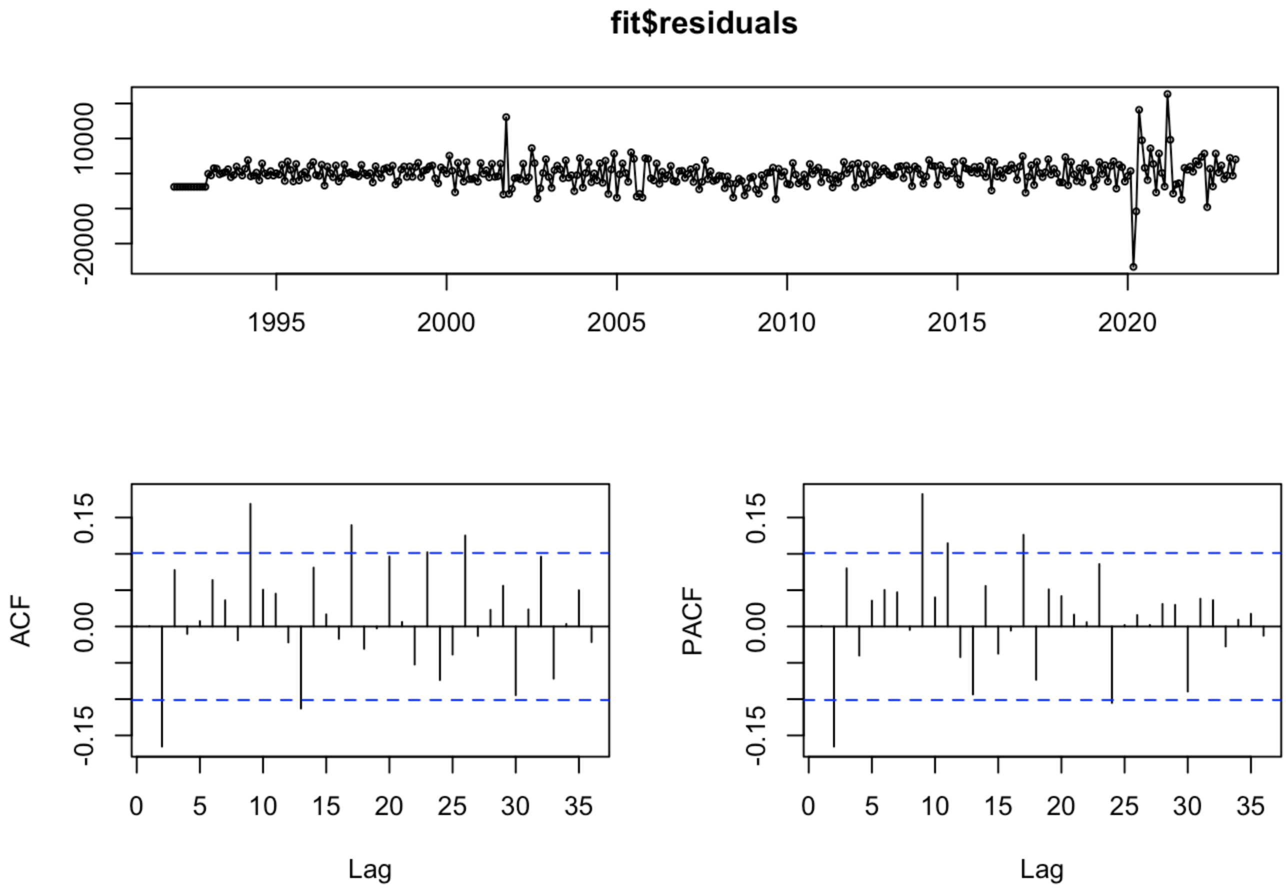

```
plot(fit2$fitted, fit2$residuals)
lines(as.double(fit2$fitted), lm(fit2$residuals ~ fit2$fitted)$fit,col="red3",lwd=2)
```



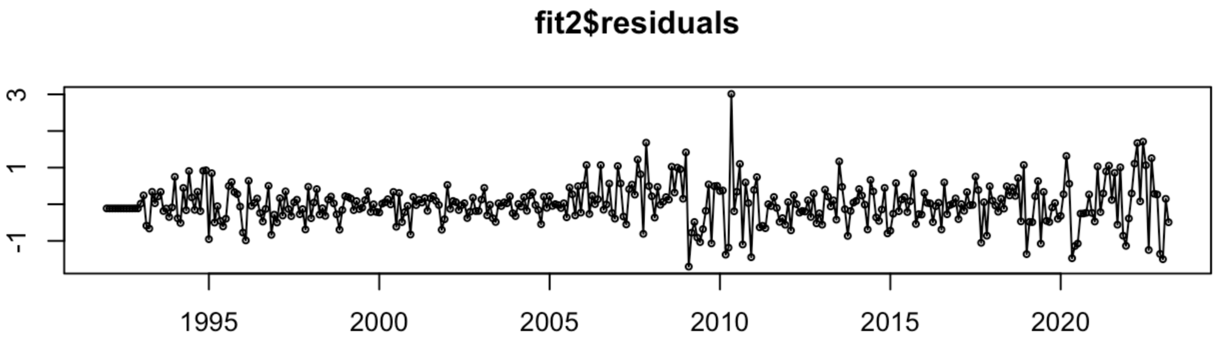
After plotting the residuals against the fitted values of the data, we can see that the new car retail sales error variances for different values of the fits are mostly constant. This indicates that for increases in the values of the fitted values, we can expect our forecasts to be equally reliable and consistent for increasing values as the economy for new cars grows. On the other hand, for new housing supply, we see a slight and sudden increase in the error variances for higher fitted values, possibly indicating the presence of heteroskedasticity.

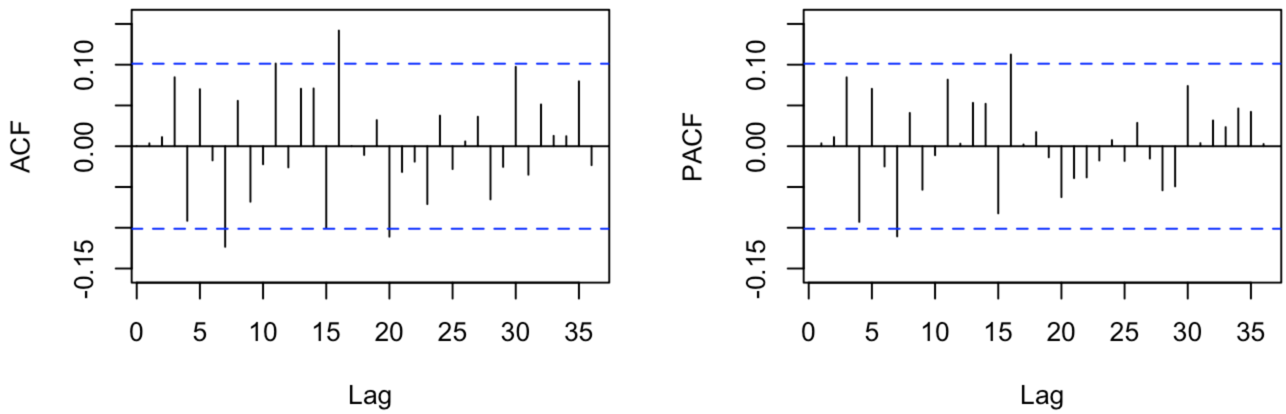
Part f

```
#ACF and PACF of residuals for Arima models
tsdisplay(fit$residuals)
```




```
tsdisplay(fit2$residuals)
```

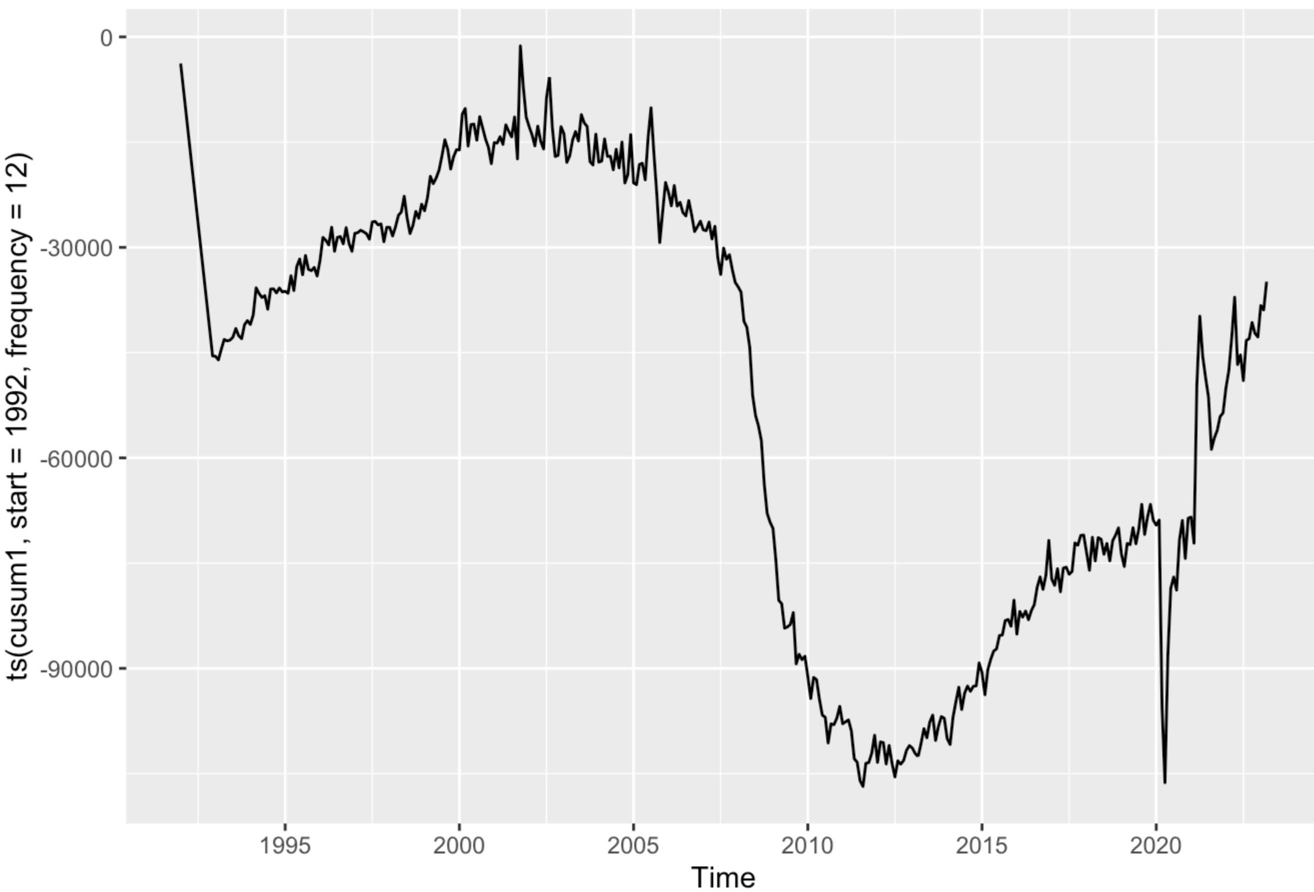




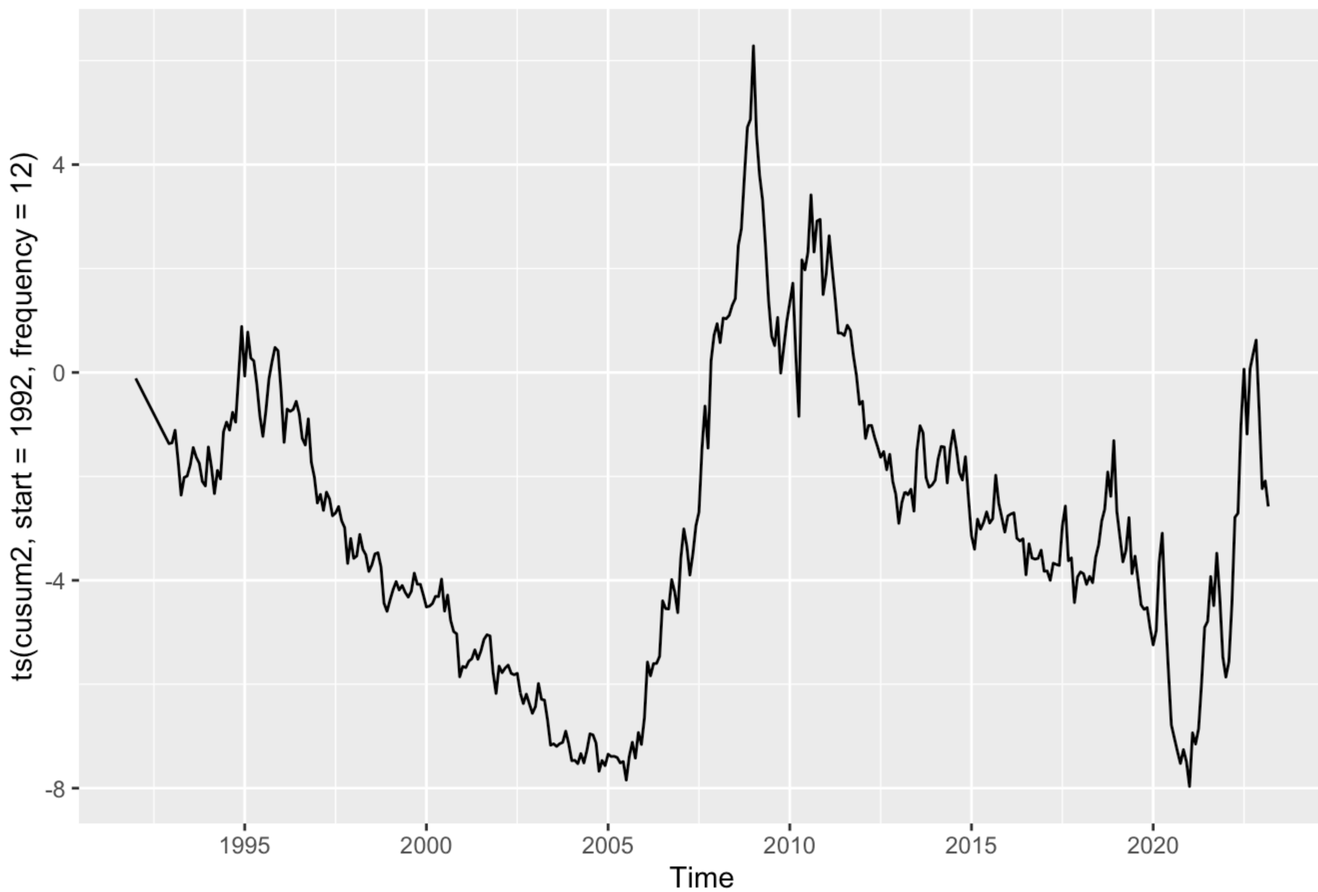
After specifying our model to include a trend and various lags for the values and seasonal values of the data, we can confidently assume that most of the dynamics of the model have been accounted for, leaving us with white noise of our fitted models according to our ACF and PACF plots.

Part g

```
#Use for loop to calculate cumulative sum of residuals for both models, plot time series of resu
lts to reference dates
cusum1<-c(fit$residuals[1])
for (i in 2:length(fit$residuals))
{
  cusum1[i]<-(cusum1[i-1]+fit$residuals[i])
}
ts(cusum1, start = 1992, frequency=12)|>autoplot()
```




```
cusum2<-c(fit2$residuals[1])
for (i in 2:length(fit2$residuals))
{
  cusum2[i]<-(cusum2[i-1]+fit2$residuals[i])
}
ts(cusum2, start = 1992, frequency=12)|>autoplot()
```



Because stability() does not have an Arima method (it is designed for VAR models), we cannot properly analyze the CUSUM of our models, as it would be too difficult to manually recalculate the standardized recursive residuals, but Professor Rojas instructed us to simply examine the cumulative sum of the regular residuals, so we have doe this here. We do not have confidence bands to examine, but it appears that there are multiple structural breaks in our Housing Supply data, as there are several significant spikes, and the cumulative sum does not revert to zero. Our New Car Sales Data, however, looks more promising. There is significant fluctuation in the cumulative sum, so it is very possible that the data would cross over a confidence band if we had one, but the cumulative sum does stay within a relatively small range, approximately (-8, 6), and it generally reverts to zero. Additionally, we might anticipate that a cumulative sum of the standardized recursive residuals would fluctuate less, thus staying within the confidence bands, which would be indicative of no structural breaks and a good model.

Part h

```
#Examiine descriptive statistics of both Arima models
summary(fit)
```

```
## Series: RetailNewCarsDataTS
## Regression with ARIMA(2,0,0)(1,1,1)[12] errors
##
## Coefficients:
##          ar1     ar2    sar1     sma1       xreg
##       0.7540  0.1459  0.0115  -0.8031  1915.8022
## s.e.  0.0521  0.0536  0.0672   0.0399   429.4758
##
## sigma^2 = 13785059:  log likelihood = -3496.37
## AIC=7004.74   AICc=7004.98   BIC=7028.11
##
## Training set error measures:
##                     ME    RMSE      MAE        MPE     MAPE      MASE
## Training set -93.00761 3627.69 2444.964 -0.7300106 4.461093 0.5632439
##                     ACF1
## Training set 0.000627569
```

```
summary(fit2)
```

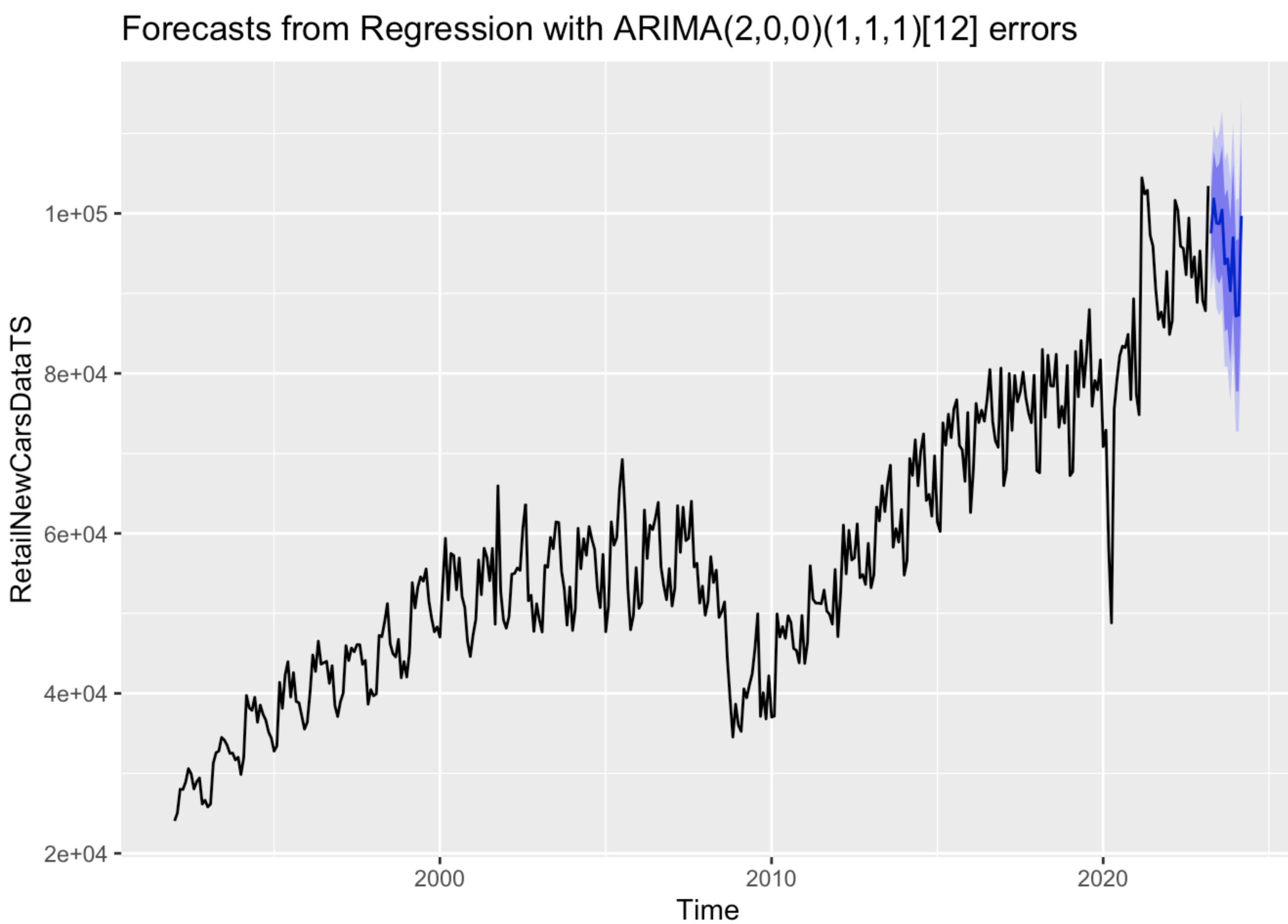
```
## Series: HousingTimeSeries
## Regression with ARIMA(2,0,1)(1,1,1)[12] errors
##
## Coefficients:
##          ar1     ar2     ma1    sar1     sma1    xreg
##       0.8457  0.1007  -0.032  0.2188  -0.9735  0.0600
## s.e.  0.2631  0.2469   0.261  0.0694   0.1189  0.0651
##
## sigma^2 = 0.3279:  log likelihood = -323.77
## AIC=661.54   AICc=661.86   BIC=688.8
##
## Training set error measures:
##                        ME      RMSE       MAE        MPE     MAPE      MASE
## Training set -0.006877906 0.5586981 0.4043412 -0.7202912 6.859597 0.3750981
##                     ACF1
## Training set 0.003542885
```

The diagnostic statistics suggest that the first 2 AR lags, first seasonal MA lag, and Trend values for the new cars retail sales fit that we suggested are statistically significant since the standard errors are 2 standard deviations away from the expected coefficient of the lags. However, the first season autoregressive parameter coefficient is within 2 standard deviations of the expected value indicating that this coefficient is not statistically significant and worth discarding.

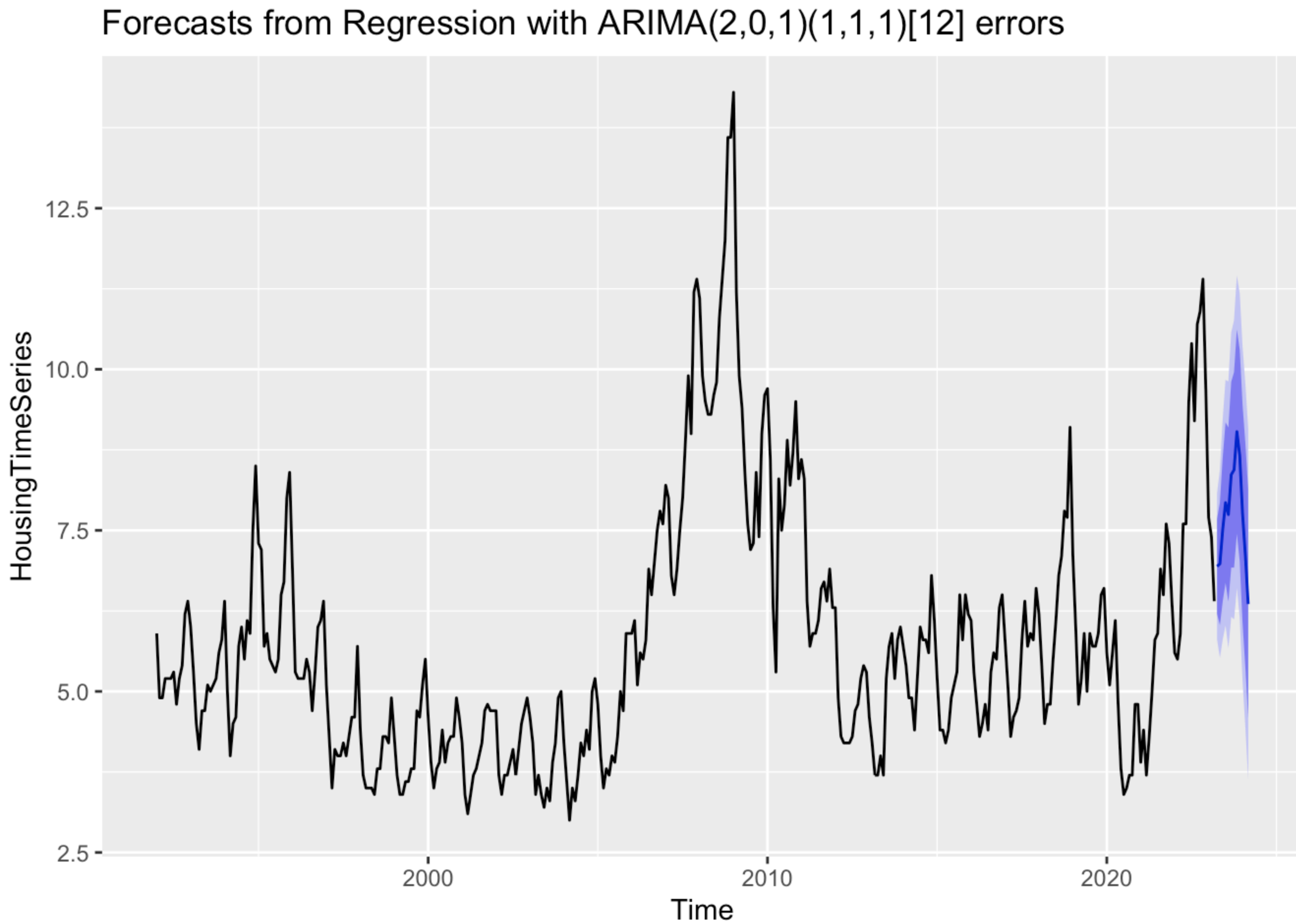
For the new homes supply model fit, our diagnostic statistics suggest that only the first AR lag, seasonal AR lag, and seasonal MA lag is worth keeping while the other parameter coefficients (the second AR lag, first MA lag, and trend) are statistically insignificant accroding to their standard errors and worth discarding to use when forecasting a model.

Part i

```
#selected models forecasts
forecast1<-forecast(fit, xreg = seq(2023 + 3/12, 2024 + 2/12, length = 12))
autoplot(forecast1)
```

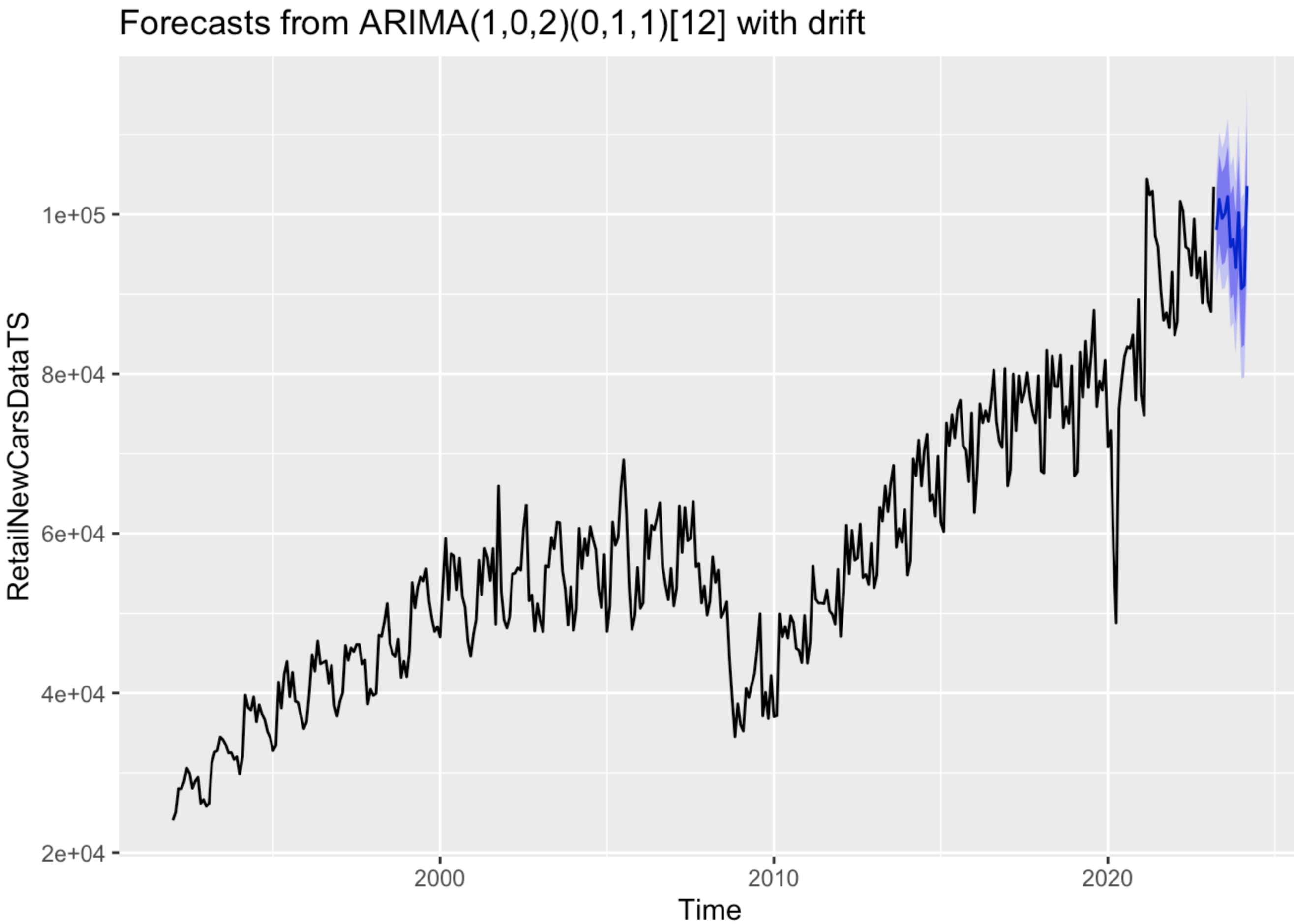



```
forecast3<-forecast(fit2, xreg = seq(2023 + 3/12, 2024 + 2/12, length = 12))
autoplot(forecast3)
```



Part j

```
#Define auto.arima() models, forecast them, and look at their summaries
autoArima1<-auto.arima(RetailNewCarsDataTS)
forecast2<-forecast(autoArima1, h=12)
autoplot(forecast2)
```



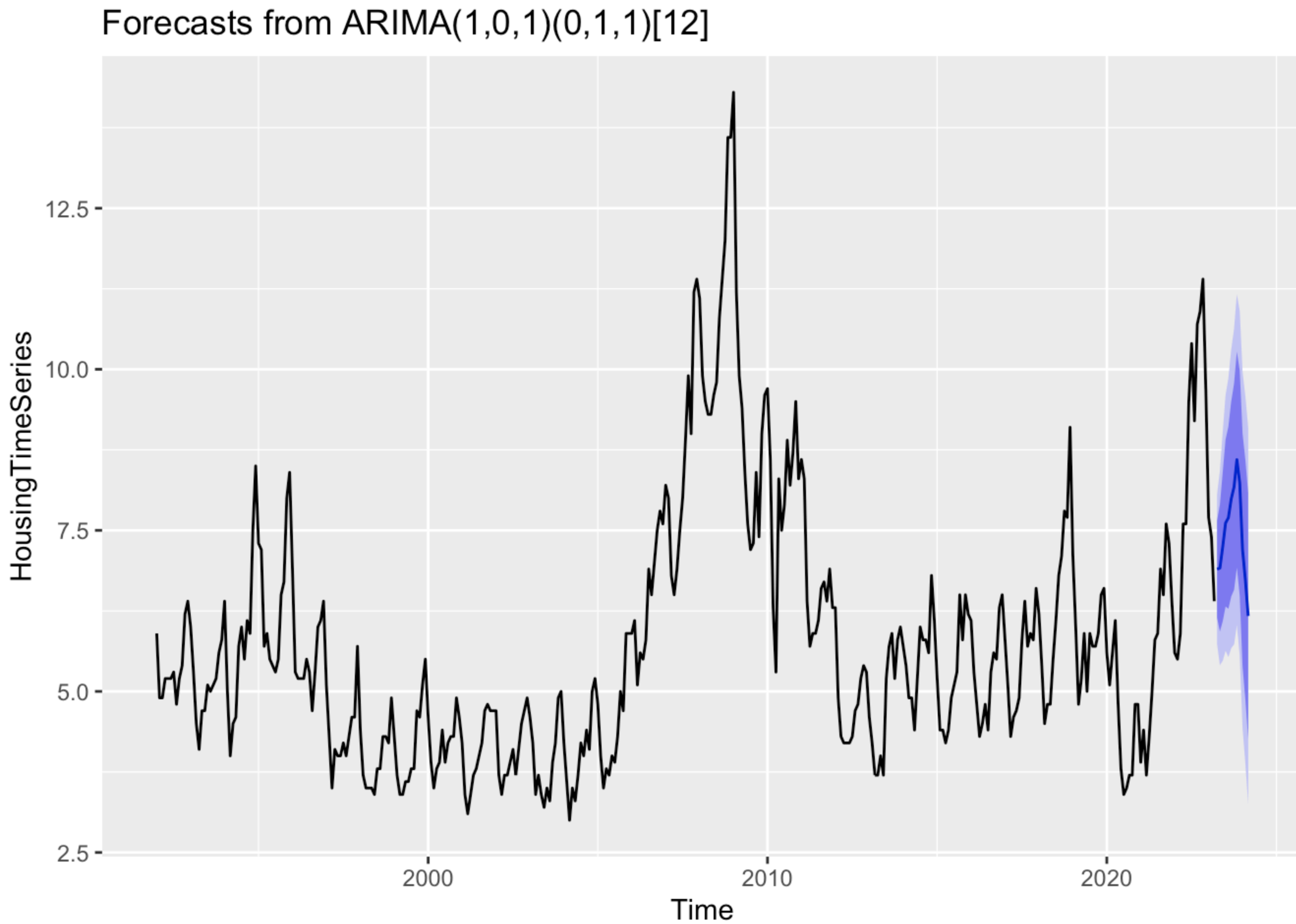

```
summary(forecast1)
```

```
##
## Forecast method: Regression with ARIMA(2,0,0)(1,1,1)[12] errors
##
## Model Information:
## Series: RetailNewCarsDataTS
## Regression with ARIMA(2,0,0)(1,1,1)[12] errors
##
## Coefficients:
##          ar1     ar2    sar1     sma1       xreg
##       0.7540  0.1459  0.0115  -0.8031  1915.8022
## s.e.  0.0521  0.0536  0.0672   0.0399   429.4758
##
## sigma^2 = 13785059:  log likelihood = -3496.37
## AIC=7004.74   AICc=7004.98   BIC=7028.11
##
## Error measures:
##                     ME    RMSE      MAE        MPE     MAPE      MASE
## Training set -93.00761 3627.69 2444.964 -0.7300106 4.461093 0.5632439
##                     ACF1
## Training set 0.000627569
##
## Forecasts:
##          Point Forecast    Lo 80     Hi 80    Lo 95    Hi 95
## Apr 2023       97509.45 92751.27 102267.62 90232.45 104786.5
## May 2023      101820.82 95861.70 107779.95 92707.13 110934.5
## Jun 2023       98793.48 91932.95 105654.01 88301.20 109285.8
## Jul 2023       98704.29 91181.39 106227.20 87199.01 110209.6
## Aug 2023      100428.19 92392.94 108463.44 88139.34 112717.0
## Sep 2023       93687.92 85248.88 102126.96 80781.52 106594.3
## Oct 2023       94317.79 85555.86 103079.71 80917.58 107718.0
## Nov 2023       90325.39 81302.76  99348.02 76526.46 104124.3
## Dec 2023       96963.69 87729.04 106198.34 82840.51 111086.9
## Jan 2024       87183.79 77775.80  96591.78 72795.51 101572.1
## Feb 2024       87278.37 77728.08  96828.67 72672.45 101884.3
## Mar 2024       99698.51 90031.02 109366.01 84913.35 114483.7
```

```
summary(forecast2)
```

```
##
## Forecast method: ARIMA(1,0,2)(0,1,1)[12] with drift
##
## Model Information:
## Series: RetailNewCarsDataTS
## ARIMA(1,0,2)(0,1,1)[12] with drift
##
## Coefficients:
##          ar1      ma1      ma2     sma1     drift
##       0.9793  -0.2799  -0.2699  -0.8180  180.8180
## s.e.  0.0129   0.0546   0.0567   0.0322   63.6304
##
## sigma^2 = 12325719:  log likelihood = -3482.57
## AIC=6977.13   AICc=6977.37   BIC=7000.5
##
## Error measures:
##                   ME     RMSE      MAE        MPE     MAPE      MASE
## Training set 5.98564 3430.299 2259.572 -0.2902616 3.931946 0.5205352
##                     ACF1
## Training set -0.00398984
##
## Forecasts:
##          Point Forecast    Lo 80     Hi 80    Lo 95    Hi 95
## Apr 2023       98068.23 93568.95 102567.50 91187.18 104949.3
## May 2023      101895.35 96404.95 107385.75 93498.51 110292.2
## Jun 2023       99480.86 93681.77 105279.95 90611.91 108349.8
## Jul 2023      100097.65 94017.23 106178.07 90798.45 109396.8
## Aug 2023      102231.78 95893.29 108570.26 92537.90 111925.6
## Sep 2023       95924.78 89348.32 102501.23 85866.96 105982.6
## Oct 2023       96862.41 90065.58 103659.24 86467.55 107257.3
## Nov 2023       93315.34 86313.68 100317.00 82607.23 104023.5
## Dec 2023      100194.36 93001.75 107386.96 89194.22 111194.5
## Jan 2024       90701.53 83330.46  98072.59 79428.46 101974.6
## Feb 2024       91132.65 83594.42  98670.89 79603.91 102661.4
## Mar 2024      103535.12 95839.98 111230.26 91766.42 115303.8
```

```
autoArima2<-auto.arima(HousingTimeSeries)
forecast4<-forecast(autoArima2, h=12)
autoplot(forecast4)
```




```
summary(forecast3)
```

```
##
## Forecast method: Regression with ARIMA(2,0,1)(1,1,1)[12] errors
##
## Model Information:
## Series: HousingTimeSeries
## Regression with ARIMA(2,0,1)(1,1,1)[12] errors
##
## Coefficients:
##          ar1     ar2     ma1    sar1     sma1    xreg
##       0.8457  0.1007  -0.032  0.2188  -0.9735  0.0600
## s.e.  0.2631  0.2469   0.261  0.0694   0.1189  0.0651
##
## sigma^2 = 0.3279:  log likelihood = -323.77
## AIC=661.54   AICc=661.86   BIC=688.8
##
## Error measures:
##                        ME      RMSE       MAE        MPE     MAPE      MASE
## Training set -0.006877906 0.5586981 0.4043412 -0.7202912 6.859597 0.3750981
##                     ACF1
## Training set 0.003542885
##
## Forecasts:
##          Point Forecast    Lo 80     Hi 80    Lo 95     Hi 95
## Apr 2023       6.940953 6.202874  7.679032 5.812159  8.069747
## May 2023       6.981179 6.029701  7.932656 5.526020  8.436337
## Jun 2023       7.517969 6.402589  8.633348 5.812143  9.223794
## Jul 2023       7.930492 6.685742  9.175242 6.026811  9.834173
## Aug 2023       7.745018 6.393781  9.096255 5.678480  9.811557
## Sep 2023       8.363915 6.923095  9.804735 6.160371 10.567459
## Oct 2023       8.438926 6.921609  9.956243 6.118390 10.759461
## Nov 2023       9.032075 7.448744 10.615406 6.610579 11.453570
## Dec 2023       8.664817 7.024068 10.305566 6.155507 11.174126
## Jan 2024       7.757063 6.066120  9.448006 5.170989 10.343137
## Feb 2024       7.111965 5.376694  8.847235 4.458098  9.765832
## Mar 2024       6.355801 4.581374  8.130229 3.642049  9.069554
```

```
summary(forecast4)
```

```
##
## Forecast method: ARIMA(1,0,1)(0,1,1)[12]
##
## Model Information:
## Series: HousingTimeSeries
## ARIMA(1,0,1)(0,1,1)[12]
##
## Coefficients:
##          ar1     ma1     sma1
##       0.9615  -0.123  -0.8123
## s.e.  0.0154   0.053   0.0576
##
## sigma^2 = 0.3462:  log likelihood = -327.98
## AIC=663.95   AICc=664.06   BIC=679.53
##
## Error measures:
##                      ME      RMSE      MAE        MPE     MAPE      MASE
## Training set 0.01182832 0.5764934 0.417959 -0.2808093 7.131522 0.3877309
##                       ACF1
## Training set -0.0007556274
##
## Forecasts:
##          Point Forecast    Lo 80     Hi 80    Lo 95     Hi 95
## Apr 2023       6.897532 6.143491  7.651573 5.744326  8.050738
## May 2023       6.914576 5.930554  7.898597 5.409645  8.419506
## Jun 2023       7.247036 6.090395  8.403676 5.478107  9.015964
## Jul 2023       7.614120 6.318196  8.910044 5.632176  9.596064
## Aug 2023       7.695881 6.283362  9.108399 5.535619  9.856142
## Sep 2023       7.986775 6.474445  9.499105 5.673865 10.299684
## Oct 2023       8.175271 6.576203  9.774338 5.729707 10.620834
## Nov 2023       8.597426 6.922164 10.272689 6.035334 11.159519
## Dec 2023       8.236471 6.493731  9.979210 5.571181 10.901760
## Jan 2024       7.198560 5.395687  9.001433 4.441304  9.955817
## Feb 2024       6.732834 4.876102  8.589566 3.893208  9.572461
## Mar 2024       6.171011 4.265844  8.076179 3.257309  9.084714
```

Visually, the forecasts for our models and the auto.arima() models look very similar for both time series, the only difference being that our models' error bands appear to be slightly larger than the ones in the forecasts of the auto.arima() models. The Mean Absolute Percentage Errors (MAPE) were lowest suggested by the auto arima functions for the New Car Retail Sales but was lower using our model for the New Housing Supply data. For future forecasting methods, it would be best to use the auto arima functions for the most accurate point estimate for the new car sales time series' but may be best to use our arima function for the new housing supply data if the criteria is sufficient to produce an optimal forecast.

Part k

```
#Combine our Arima and auto.arima() models for each time series, forecast them, and look at new MAPE in summary
cars_combined <- (fit$fitted + autoArima1$fitted) / 2
cars_combined_model <- fit
cars_combined_model$fitted <- cars_combined
cars_combined_forecast <- forecast(cars_combined_model, xreg = seq(2023 + 3/12, 2024 + 2/12, length = 12))
summary(cars_combined_forecast)
```

```
##
## Forecast method: Regression with ARIMA(2,0,0)(1,1,1)[12] errors
##
## Model Information:
## Series: RetailNewCarsDataTS
## Regression with ARIMA(2,0,0)(1,1,1)[12] errors
##
## Coefficients:
##          ar1     ar2    sar1     sma1       xreg
##       0.7540  0.1459  0.0115  -0.8031  1915.8022
## s.e.  0.0521  0.0536  0.0672   0.0399   429.4758
##
## sigma^2 = 13785059:  log likelihood = -3496.37
## AIC=7004.74   AICc=7004.98   BIC=7028.11
##
## Error measures:
##                     ME     RMSE      MAE        MPE    MAPE      MASE
## Training set -43.51099 3476.496 2327.752 -0.5101361 4.15624 0.5362419
##                     ACF1
## Training set -0.01860936
##
## Forecasts:
##          Point Forecast    Lo 80     Hi 80    Lo 95    Hi 95
## Apr 2023       97509.45 92751.27 102267.62 90232.45 104786.5
## May 2023      101820.82 95861.70 107779.95 92707.13 110934.5
## Jun 2023       98793.48 91932.95 105654.01 88301.20 109285.8
## Jul 2023       98704.29 91181.39 106227.20 87199.01 110209.6
## Aug 2023      100428.19 92392.94 108463.44 88139.34 112717.0
## Sep 2023       93687.92 85248.88 102126.96 80781.52 106594.3
## Oct 2023       94317.79 85555.86 103079.71 80917.58 107718.0
## Nov 2023       90325.39 81302.76  99348.02 76526.46 104124.3
## Dec 2023       96963.69 87729.04 106198.34 82840.51 111086.9
## Jan 2024       87183.79 77775.80  96591.78 72795.51 101572.1
## Feb 2024       87278.37 77728.08  96828.67 72672.45 101884.3
## Mar 2024       99698.51 90031.02 109366.01 84913.35 114483.7
```

```
housing_combined <- (fit2$fitted + autoArima2$fitted) / 2
housing_combined_model <- fit2
housing_combined_model$fitted <- housing_combined
housing_combined_forecast <- forecast(housing_combined_model, xreg = seq(2023 + 3/12, 2024 + 2/1
2, length = 12))
summary(housing_combined_forecast)
```

```
##
## Forecast method: Regression with ARIMA(2,0,1)(1,1,1)[12] errors
##
## Model Information:
## Series: HousingTimeSeries
## Regression with ARIMA(2,0,1)(1,1,1)[12] errors
##
## Coefficients:
##          ar1     ar2     ma1    sar1     sma1    xreg
##       0.8457  0.1007  -0.032  0.2188  -0.9735  0.0600
## s.e.  0.2631  0.2469   0.261  0.0694   0.1189  0.0651
##
## sigma^2 = 0.3279:  log likelihood = -323.77
## AIC=661.54   AICc=661.86   BIC=688.8
##
## Error measures:
##                       ME      RMSE      MAE        MPE     MAPE      MASE
## Training set 0.002475205 0.5649086 0.408318 -0.5005502 6.941833 0.3787872
##                     ACF1
## Training set 0.001343904
##
## Forecasts:
##          Point Forecast    Lo 80     Hi 80    Lo 95     Hi 95
## Apr 2023       6.940953 6.202874  7.679032 5.812159  8.069747
## May 2023       6.981179 6.029701  7.932656 5.526020  8.436337
## Jun 2023       7.517969 6.402589  8.633348 5.812143  9.223794
## Jul 2023       7.930492 6.685742  9.175242 6.026811  9.834173
## Aug 2023       7.745018 6.393781  9.096255 5.678480  9.811557
## Sep 2023       8.363915 6.923095  9.804735 6.160371 10.567459
## Oct 2023       8.438926 6.921609  9.956243 6.118390 10.759461
## Nov 2023       9.032075 7.448744 10.615406 6.610579 11.453570
## Dec 2023       8.664817 7.024068 10.305566 6.155507 11.174126
## Jan 2024       7.757063 6.066120  9.448006 5.170989 10.343137
## Feb 2024       7.111965 5.376694  8.847235 4.458098  9.765832
## Mar 2024       6.355801 4.581374  8.130229 3.642049  9.069554
```

As found in part j, our Arima fit for New Car Sales has a MAPE of 4.461093, our auto.arima() fit for New Car Sales has a MAPE of 3.931946, our Arima fit for Housing Supply has a MAPE of 6.859597, our auto.arima() fit for Housing Supply has a MAPE of 7.131522. By contrast, our combined model for New Car Sales has a MAPE of 4.15624, and our combined model for Housing Supply has a MAPE of 6.941833. For both time series, the MAPE of the combined model is in between those of the two individual models. This likely indicates that our models and the auto.arima() models make similar errors in predicting the data, so averaging them does not produce a combined model that is a better fit for the time series.

Part I

```
#Combine data into onoe data frame
Var1 <- cbind(RetailNewCarsDataTS, HousingTimeSeries)
Var_tot <- data.frame(Var1)

#Find optimal p value for VAR model
VARselect(Var_tot, lag.max = 10, type = "trend", season = 12)
```

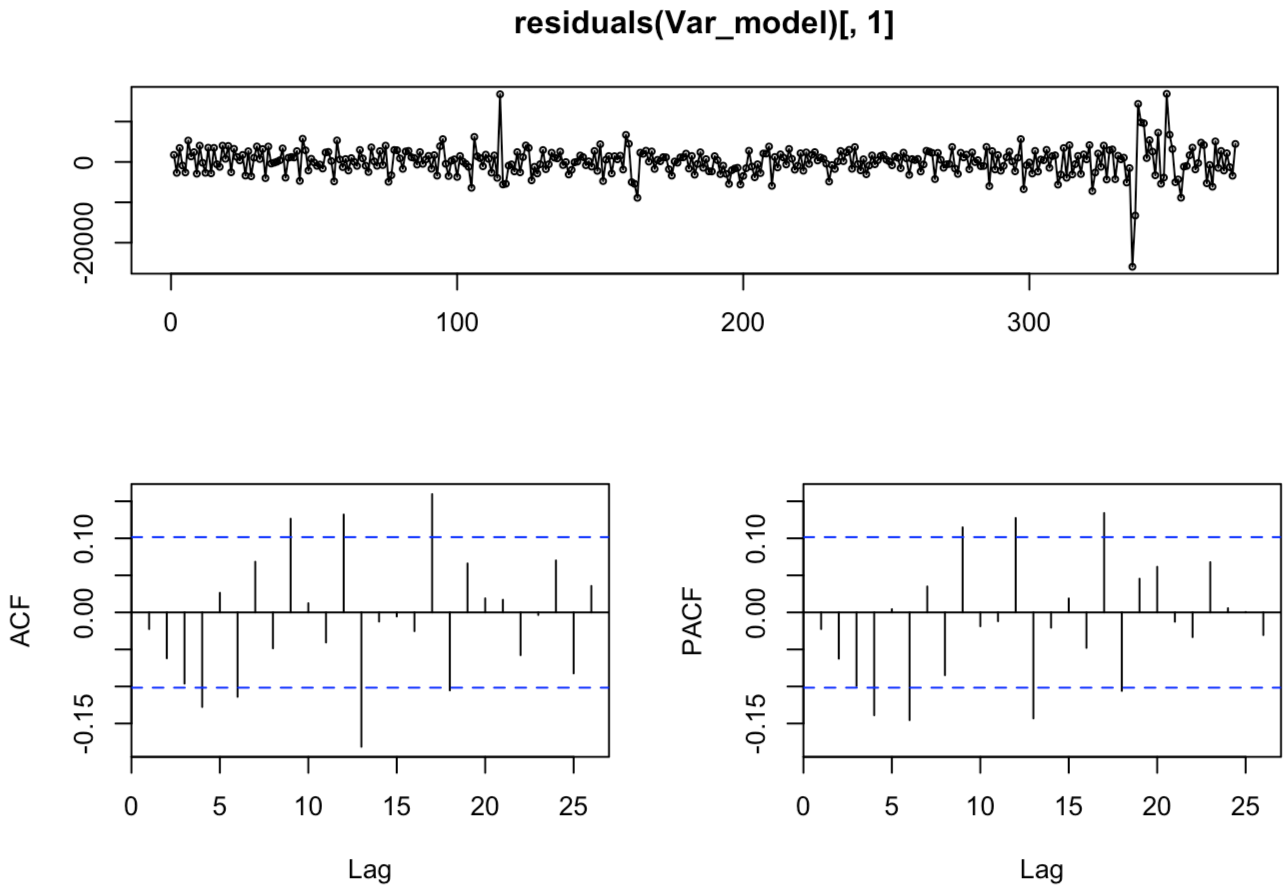
```
## $selection
## AIC(n)  HQ(n)  SC(n) FPE(n)
##      3      3      3      3
##
## $criteria
##                   1            2            3            4            5
## AIC(n) 1.548314e+01 1.542220e+01 1.535471e+01 1.536645e+01 1.536148e+01
## HQ(n)  1.560203e+01 1.555808e+01 1.550758e+01 1.553630e+01 1.554831e+01
## SC(n)  1.578231e+01 1.576411e+01 1.573936e+01 1.579384e+01 1.583160e+01
## FPE(n) 5.299993e+06 4.986857e+06 4.661611e+06 4.716944e+06 4.693882e+06
##                   6            7            8            9           10
## AIC(n) 1.537989e+01 1.537659e+01 1.537418e+01 1.535728e+01 1.535585e+01
## HQ(n)  1.558371e+01 1.559739e+01 1.561197e+01 1.561205e+01 1.562761e+01
## SC(n)  1.589275e+01 1.593219e+01 1.597252e+01 1.599835e+01 1.603967e+01
## FPE(n) 4.781533e+06 4.766272e+06 4.755358e+06 4.676306e+06 4.670406e+06
```

```
#VARselect() unanimously recommends p=3, so we use this in our VAR model
Var_model <- VAR(Var_tot,p=3, type = "trend", season = 12)
summary(Var_model)
```

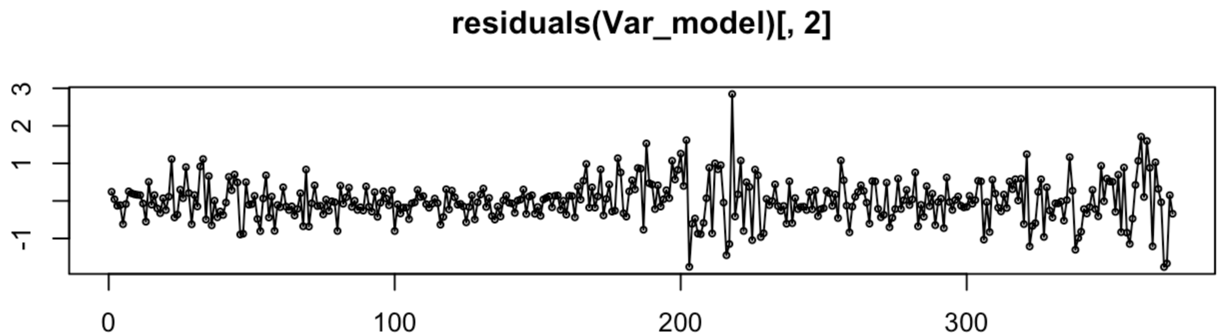
```
##
## VAR Estimation Results:
## =========================
## Endogenous variables: RetailNewCarsDataTS, HousingTimeSeries
## Deterministic variables: trend
## Sample size: 372
## Log Likelihood: -3871.572
## Roots of the characteristic polynomial:
## 0.9958 0.9656 0.5468 0.5468 0.0756 0.0756
## Call:
## VAR(y = Var_tot, p = 3, type = "trend", season = 12L)
##
##
## Estimation results for equation RetailNewCarsDataTS:
## ====================================================
## RetailNewCarsDataTS = RetailNewCarsDataTS.l1 + HousingTimeSeries.l1 + RetailNewCarsDataTS.l2
+ HousingTimeSeries.l2 + RetailNewCarsDataTS.l3 + HousingTimeSeries.l3 + trend + sd1 + sd2 + sd3
+ sd4 + sd5 + sd6 + sd7 + sd8 + sd9 + sd10 + sd11
##
##                         Estimate Std. Error t value Pr(>|t|)
## RetailNewCarsDataTS.l1  6.851e-01  5.148e-02  13.309  < 2e-16 ***
## HousingTimeSeries.l1   -1.672e+02  3.407e+02  -0.491   0.6240
## RetailNewCarsDataTS.l2  1.395e-02  6.317e-02   0.221   0.8253
## HousingTimeSeries.l2   -2.064e+02  4.398e+02  -0.469   0.6392
## RetailNewCarsDataTS.l3  2.959e-01  5.235e-02   5.652 3.26e-08 ***
## HousingTimeSeries.l3    3.619e+02  3.354e+02   1.079   0.2813
## trend                   2.997e+00  2.969e+00   1.009   0.3135
## sd1                    -8.116e+03  1.012e+03  -8.016 1.60e-14 ***
## sd2                    -2.279e+03  1.021e+03  -2.233   0.0262 *
## sd3                     4.663e+03  1.121e+03   4.159 4.01e-05 ***
## sd4                    -3.642e+03  1.210e+03  -3.010   0.0028 **
## sd5                     1.855e+03  1.049e+03   1.768   0.0780 .
## sd6                    -4.104e+03  1.011e+03  -4.059 6.08e-05 ***
## sd7                    -1.676e+03  9.631e+02  -1.740   0.0828 .
## sd8                    -1.678e+03  9.560e+02  -1.756   0.0800 .
## sd9                    -7.949e+03  9.706e+02  -8.190 4.81e-15 ***
## sd10                   -4.002e+03  9.378e+02  -4.267 2.54e-05 ***
## sd11                   -7.797e+03  9.967e+02  -7.823 5.99e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 3618 on 354 degrees of freedom
## Multiple R-Squared: 0.9965,  Adjusted R-squared: 0.9963
## F-statistic:  5636 on 18 and 354 DF,  p-value: < 2.2e-16
##
##
## Estimation results for equation HousingTimeSeries:
## ==================================================
## HousingTimeSeries = RetailNewCarsDataTS.l1 + HousingTimeSeries.l1 + RetailNewCarsDataTS.l2 +
HousingTimeSeries.l2 + RetailNewCarsDataTS.l3 + HousingTimeSeries.l3 + trend + sd1 + sd2 + sd3 +
sd4 + sd5 + sd6 + sd7 + sd8 + sd9 + sd10 + sd11
```

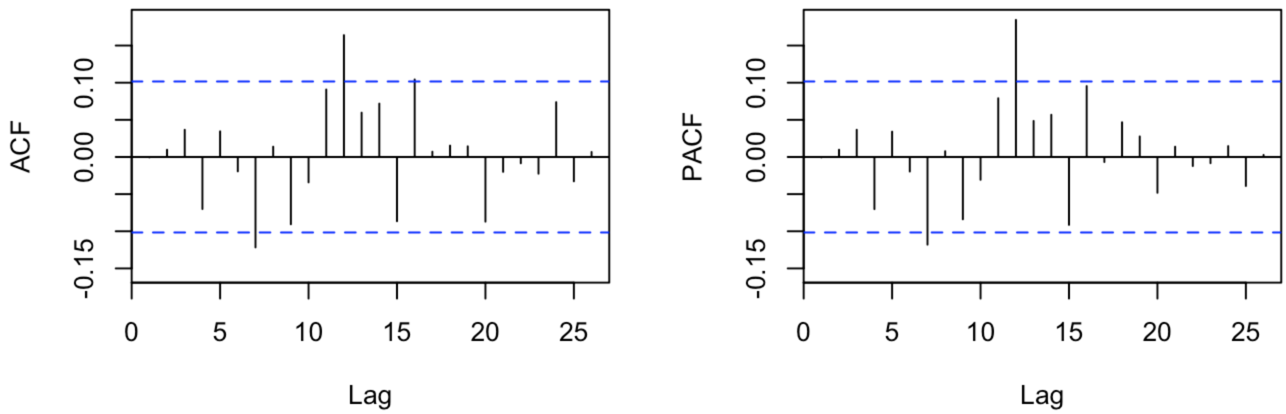
```
##
##                          Estimate Std. Error t value Pr(>|t|)
## RetailNewCarsDataTS.l1 -1.030e-05  8.111e-06  -1.269 0.205171
## HousingTimeSeries.l1    8.308e-01  5.368e-02  15.477  < 2e-16 ***
## RetailNewCarsDataTS.l2  1.447e-05  9.955e-06   1.453 0.147056
## HousingTimeSeries.l2    1.230e-01  6.931e-02   1.774 0.076877 .
## RetailNewCarsDataTS.l3  1.835e-06  8.249e-06   0.222 0.824094
## HousingTimeSeries.l3    7.797e-03  5.285e-02   0.148 0.882796
## trend                  -5.963e-04  4.679e-04  -1.274 0.203379
## sd1                    -6.005e-01  1.595e-01  -3.764 0.000196 ***
## sd2                    -9.980e-01  1.608e-01  -6.205 1.52e-09 ***
## sd3                    -9.315e-01  1.767e-01  -5.273 2.34e-07 ***
## sd4                    -4.445e-02  1.906e-01  -0.233 0.815767
## sd5                    -1.305e-01  1.654e-01  -0.789 0.430432
## sd6                     1.023e-02  1.593e-01   0.064 0.948857
## sd7                     9.592e-02  1.518e-01   0.632 0.527809
## sd8                    -1.262e-02  1.506e-01  -0.084 0.933288
## sd9                     2.426e-01  1.529e-01   1.586 0.113594
## sd10                   -1.279e-01  1.478e-01  -0.865 0.387455
## sd11                    5.081e-01  1.571e-01   3.235 0.001329 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.5701 on 354 degrees of freedom
## Multiple R-Squared: 0.9916,  Adjusted R-squared: 0.9912
## F-statistic:  2320 on 18 and 354 DF,  p-value: < 2.2e-16
##
##
##
## Covariance matrix of residuals:
##                     RetailNewCarsDataTS HousingTimeSeries
## RetailNewCarsDataTS          13081588.6         -326.5779
## HousingTimeSeries               -326.6            0.3251
##
## Correlation matrix of residuals:
##                     RetailNewCarsDataTS HousingTimeSeries
## RetailNewCarsDataTS              1.0000           -0.1584
## HousingTimeSeries               -0.1584            1.0000
```

```
#Check residuals of VAR model
tsdisplay(residuals(Var_model)[,1])
```




```
tsdisplay(residuals(Var_model)[,2])
```





Our VAR model suggests that the values for the time series of new car retail sales and new housing supply is inversely correlated as shown by the correlation matrix of residuals. A unit change in new car retail sales corresponds to a 15% reduction in new housing supply. Intuitively, for a growing economy, it makes sense for car sales and housing supplies to be inversely correlated; when national economies are thriving, people are more likely to buy cars and houses, which increases the amount of cars and homes that are purchased, resulting in higher sales for cars at the same time as a lower inventory of houses for sale, which corresponds with higher prices for houses also. The reverse is generally true for when economies are deteriorating, like the US economy did in 2008. This correlation does not indicate the direction of causality so we will have to run further tests to determine which variable is resposible for changes in the other.

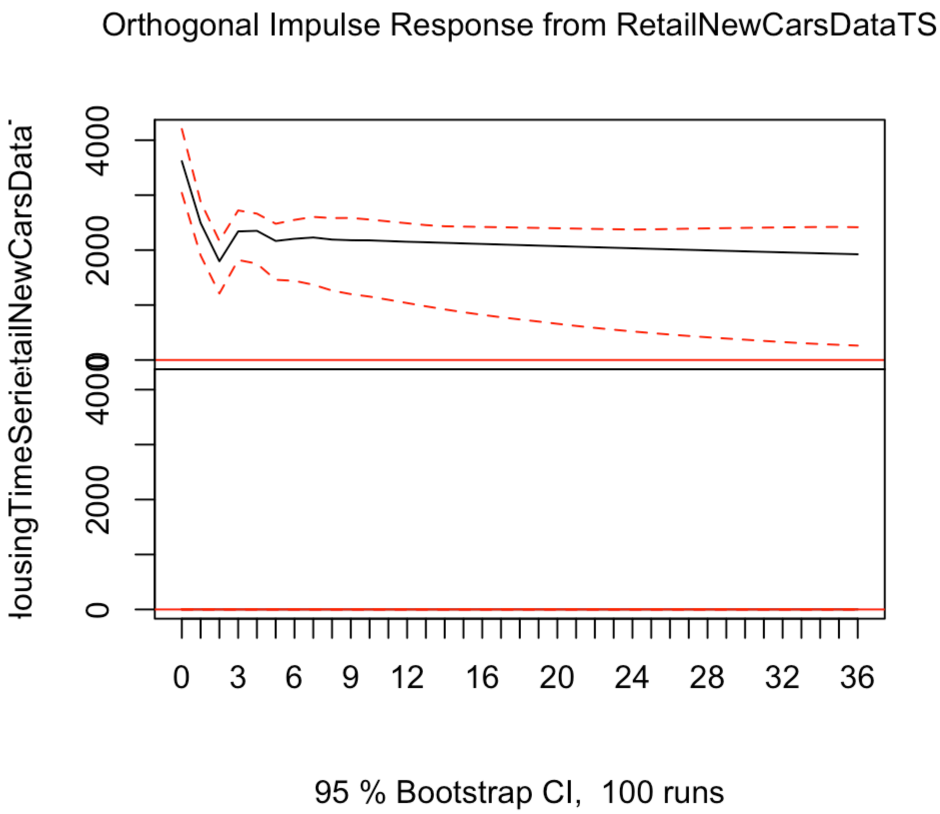
Part m

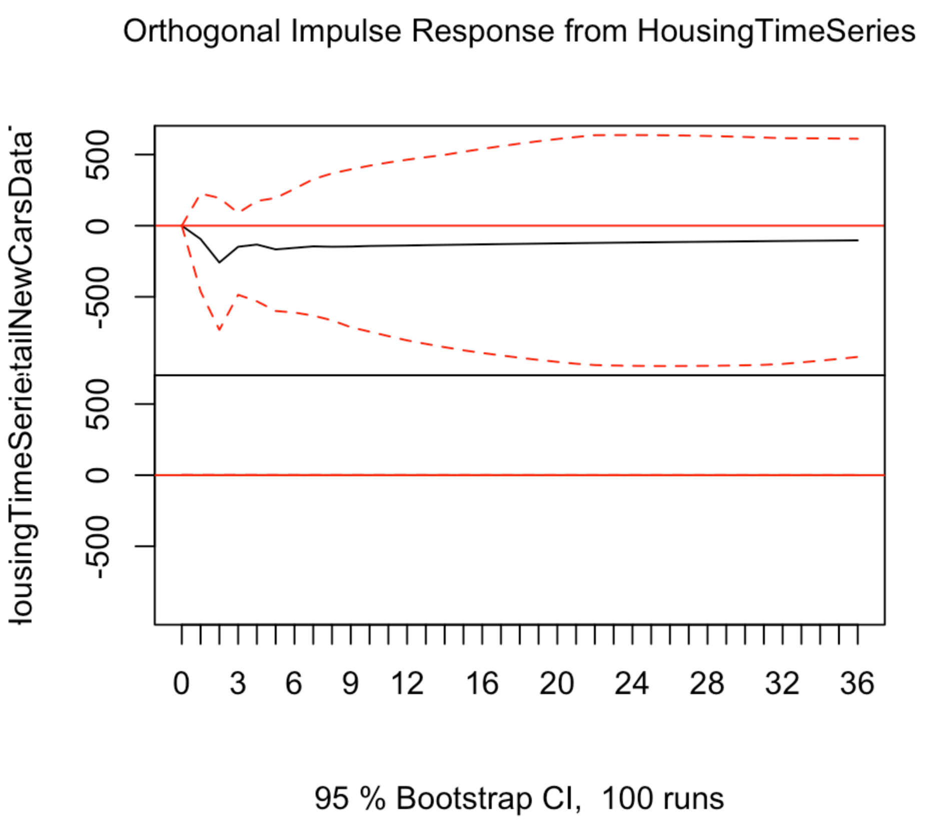
```
#Compute IRF of VAR model
irf(Var_model)
```

```
##
## Impulse response coefficients
## $RetailNewCarsDataTS
##       RetailNewCarsDataTS HousingTimeSeries
##  [1,]            3618.213      -0.090323390
##  [2,]            2493.897      -0.112296435
##  [3,]            1796.435      -0.077741322
##  [4,]            2339.670      -0.054882145
##  [5,]            2350.500      -0.049557792
##  [6,]            2166.009      -0.035586368
##  [7,]            2205.356      -0.020092336
##  [8,]            2229.396      -0.008513797
##  [9,]            2191.745       0.003103537
## [10,]            2179.205       0.015107967
## [11,]            2176.989       0.026229175
##
## $HousingTimeSeries
##       RetailNewCarsDataTS HousingTimeSeries
##  [1,]             0.00000         0.5629500
##  [2,]           -94.10151         0.4677231
##  [3,]          -258.82175         0.4587994
##  [4,]          -148.11682         0.4443992
##  [5,]          -132.63070         0.4268996
##  [6,]          -166.55197         0.4116596
##  [7,]          -155.86860         0.3975097
##  [8,]          -145.26295         0.3831706
##  [9,]          -148.08408         0.3693813
## [10,]          -146.56411         0.3562531
## [11,]          -142.57288         0.3435011
##
##
## Lower Band, CI= 0.95
## $RetailNewCarsDataTS
##       RetailNewCarsDataTS HousingTimeSeries
##  [1,]            2901.765       -0.15476187
##  [2,]            1890.053       -0.18816548
##  [3,]            1244.974       -0.15309272
##  [4,]            1688.645       -0.11208822
##  [5,]            1658.608       -0.10867833
##  [6,]            1429.745       -0.10044310
##  [7,]            1415.955       -0.08218724
##  [8,]            1398.796       -0.07018652
##  [9,]            1300.078       -0.06291335
## [10,]            1233.247       -0.05495829
## [11,]            1177.179       -0.04188155
##
## $HousingTimeSeries
##       RetailNewCarsDataTS HousingTimeSeries
##  [1,]              0.0000         0.4871636
##  [2,]           -376.8249         0.3712441
##  [3,]           -638.4297         0.3655622
##  [4,]           -376.4825         0.3483716
```

```
## [5,]          -365.0902        0.3344531
## [6,]          -454.5513        0.3119352
## [7,]          -455.8388        0.2917650
## [8,]          -466.1408        0.2723702
## [9,]          -496.4987        0.2540151
## [10,]         -532.6715        0.2357480
## [11,]         -546.6019        0.2176194
##
##
## Upper Band, CI= 0.95
## $RetailNewCarsDataTS
##       RetailNewCarsDataTS HousingTimeSeries
##  [1,]            4113.009       -0.02071238
##  [2,]            2929.629       -0.03466358
##  [3,]            2144.022        0.01312125
##  [4,]            2658.997        0.02726389
##  [5,]            2710.626        0.04088380
##  [6,]            2503.166        0.06403843
##  [7,]            2542.016        0.08793869
##  [8,]            2581.578        0.10508465
##  [9,]            2550.703        0.11718366
## [10,]            2534.411        0.13174677
## [11,]            2532.658        0.14562177
##
## $HousingTimeSeries
##       RetailNewCarsDataTS HousingTimeSeries
##  [1,]              0.0000         0.5963750
##  [2,]            361.0022         0.5106556
##  [3,]            142.2966         0.5147187
##  [4,]            133.5725         0.4942044
##  [5,]            199.7440         0.4793551
##  [6,]            239.2832         0.4666756
##  [7,]            268.5232         0.4539039
##  [8,]            293.2370         0.4405626
##  [9,]            312.7228         0.4282210
## [10,]            327.5945         0.4181800
## [11,]            343.3730         0.4047938
```

```
#Plot IRF of VAR model
plot(irf(Var_model, n.ahead = 36, type="trend"))
```


Orthogonal Impulse Response from RetailNewCarsDataTS
HousingTimeSerie:tailNewCarsData
4000
2000
0
4000
2000
0
0
3
6
9
12
16
20
24
28
32
36
95 % Bootstrap CI, 100 runs


Orthogonal Impulse Response from HousingTimeSeries
HousingTimeSerie:tailNewCarsData
500
0
-500
500
0
-500
0
3
6
9
12
16
20
24
28
32
36
95 % Bootstrap CI, 100 runs

From observing the relationship between the impulse response functions (IRF) between new car sales and new housing supply, the housing time series presents charts that suggest there strong behavior for following the new car time series. Also, there appears to be no causal relationship for the new car time series to follow the housing time series. These findings present us with data consistent to our previous section that measured the VAR model and suggested that we have strong correlation between the two variables with significan dependence on each other, but now we can see that new home supply follows car sales, but car sales is not determined or influenced by the supply of new homes.

Part n

```
#Conduct Granger-Causality test in both directions to determine which variable explains the othe
r
grangertest(RetailNewCarsDataTS~HousingTimeSeries)
```

```
## Granger causality test
##
## Model 1: RetailNewCarsDataTS ~ Lags(RetailNewCarsDataTS, 1:1) + Lags(HousingTimeSeries, 1:1)
## Model 2: RetailNewCarsDataTS ~ Lags(RetailNewCarsDataTS, 1:1)
##   Res.Df Df      F Pr(>F)
## 1    371
## 2    372 -1 0.6846 0.4085
```

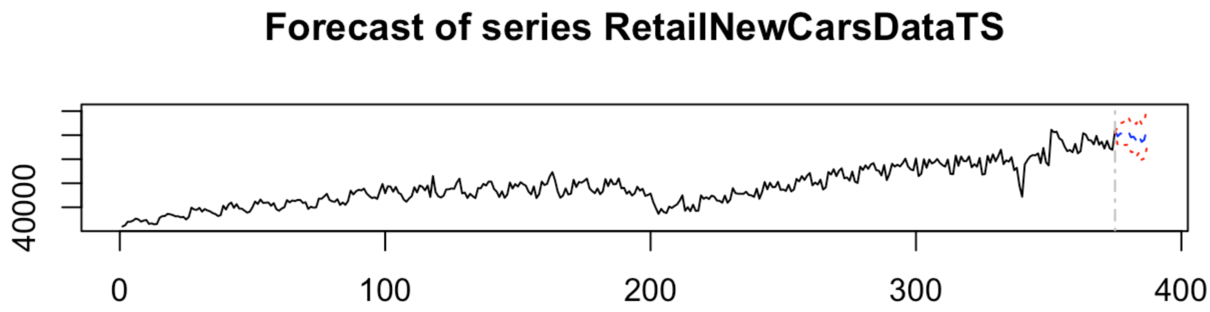
```
grangertest(HousingTimeSeries~RetailNewCarsDataTS)
```

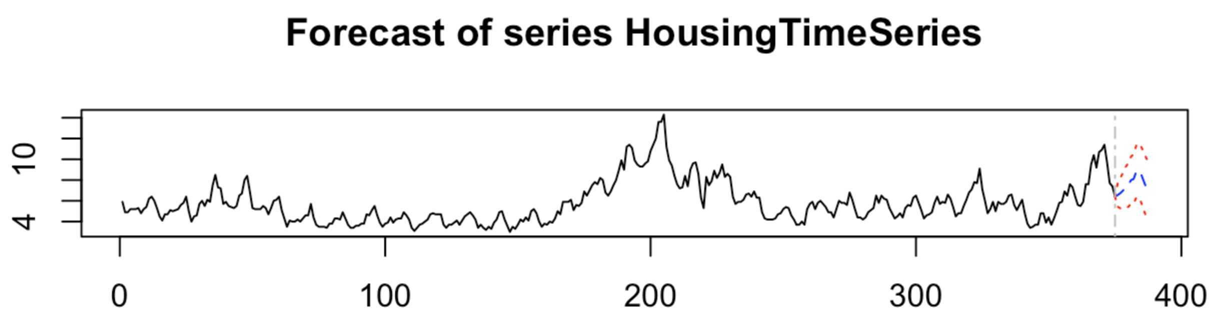
```
## Granger causality test
##
## Model 1: HousingTimeSeries ~ Lags(HousingTimeSeries, 1:1) + Lags(RetailNewCarsDataTS, 1:1)
## Model 2: HousingTimeSeries ~ Lags(HousingTimeSeries, 1:1)
##   Res.Df Df      F  Pr(>F)
## 1    371
## 2    372 -1 5.9897 0.01485 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Once again, we can see more consistent habits for the data that suggest that there is a strong causal relationship for new car sales to lead new housing supply demonstrated by our granger causality tests (GCT). The GCT measuring the influence that new housing supply has on new car sales presented us with insignificant findings, shown by the p-value of the test which was 0.4094. However, the GCT measuring the influence that new car sale has on new housing supply presented us with statistically significant findings, shown by the p-value of the test to be 0.02106. From the last 3 tests for causal significance, we have enough to conclude that there is a supporting evidence that the value of new car sales can be used to indicate and predict the number of new housing supply for the US economy. This is the only direction of causality that we can assess from the data.

Part o

```
#Plot VAR model
plot(predict(object = Var_model, n.ahead = 12))
```





It is difficult to confidently compare the VAR forecast to other forecasts because the output is in a different style, but they appear to be fairly similar to our Arima and auto.arima() forecasts in terms of general direction and the width of the error bands. It appears as if the VAR forecasts might be slightly smoother, but again, it is hard to confidently conclude this due to the different style of output.

Section III

In conclusion, we believe that from our the tests used to identify a causal relationship between new car retail sales and new housing supply that there exists a pattern of the direction of new car sales leading the direction of new housing supply for the US economy. For the US economy, we can expect that the retail sales of new cars can offer economists a gauge for how the new housing production performs in the months following an observation of new car sales. In times of recession, we expect new car sales to deteriorate faster than new housing supply, and during times of expansion, we expect car sales to improve more quickly than new housing. These findings make sense considering the frequency that cars and houses are produced and financial costs and laborious efforts associated with each purchase and production of the goods.

Some areas that we could have improved our project is by using data for these two categories of the economy with longer periods. Also, we were unable to find seasonal data for new housing retail sales, which could have given us more thorough observations during recessions or other shocks to the US economy that might better explain the relationship better because of price stability or instability.

Section IV

Our CSV files contain values for each month's values for retail sales for new cars and the total supply of newly constructed houses collected by the US Census Bureau and stored by FRED over the last few decades. The US Census Bureau releases the value of Retail Sales every month, which includes the value of Car Retail Sales for new and used cars, but only records the values for the last few months. The Dept. of Housing and Urban Development collects the supply of new houses and presents them to the U.S. C.B. and also only presents the last few months with each monthly report. FRED, on the other hand, has a much longer period of records for both datasets and contains the last 30 years of the Car Retail Sales data for both used and new car sales separately instead of only including them in a single table or value.

Data References: New Car Retail Sales FRED Data: https://fred.stlouisfed.org/series/MRTSSM44111USN (https://fred.stlouisfed.org/series/MRTSSM44111USN) Monthly Supply of New Houses in the United States: https://fred.stlouisfed.org/series/MSACSRNSA (https://fred.stlouisfed.org/series/MSACSRNSA)

US Census Bureau: https://www.census.gov/retail/marts/www/marts_current.pdf (https://www.census.gov/retail/marts/www/marts_current.pdf) Housing and Urban Development: https://www.census.gov/construction/nrs/index.html (https://www.census.gov/construction/nrs/index.html)