ADS-506 Assignment 3.1

Gabi Rivera

```
-- Attaching packages --------------------------------------- tidyverse 1.3.2 --
v ggplot2 3.3.6      v purrr   0.3.4
v tibble  3.1.8      v dplyr   1.0.10
v tidyr   1.2.1      v stringr 1.4.1
v readr   2.1.2      v forcats 0.5.2
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
-- Attaching packages ---------------------------------------------- fpp3 0.5 --

v lubridate   1.8.0     v feasts      0.3.1
v tsibble     1.1.3     v fable       0.3.3
v tsibbledata 0.4.1     v fabletools  0.3.4

-- Conflicts ------------------------------------------------- fpp3_conflicts --
x lubridate::date()    masks base::date()
x dplyr::filter()      masks stats::filter()
x tsibble::intersect() masks base::intersect()
x tsibble::interval()  masks lubridate::interval()
x dplyr::lag()         masks stats::lag()
x tsibble::setdiff()   masks base::setdiff()
x tsibble::union()     masks base::union()


Attaching package: 'zoo'


The following object is masked from 'package:tsibble':

    index


The following objects are masked from 'package:base':
```

```
    as.Date, as.Date.numeric


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


Attaching package: 'gridExtra'


The following object is masked from 'package:dplyr':

    combine
```

ADS 506 Module 4 Exercises: Chapter 7 This assignment is due on Day 7 of the learning week. The assignment for this module is a mixture of programming and written work. Complete this entire assignment in R Markdown. You will need to include the question and number that you are answering within your submitted assignment. Once completed, you will knit your deliverable to a Word/PDF file.

Chapter 7: Regression Models: Autocorrelation & External Info (Pages 170-178): #1, 2, & 6

Note: The homework file has an incorrect time scale … what's above is correct (and matches the text book)

1. Analysis of Canadian Manufacturing Workers Work-Hours: The time series plot in Figure 7.7 describes the average annual number of weekly hours spent by Canadian manufacturing workers. The data is available in CanadianWorkHours.csv.

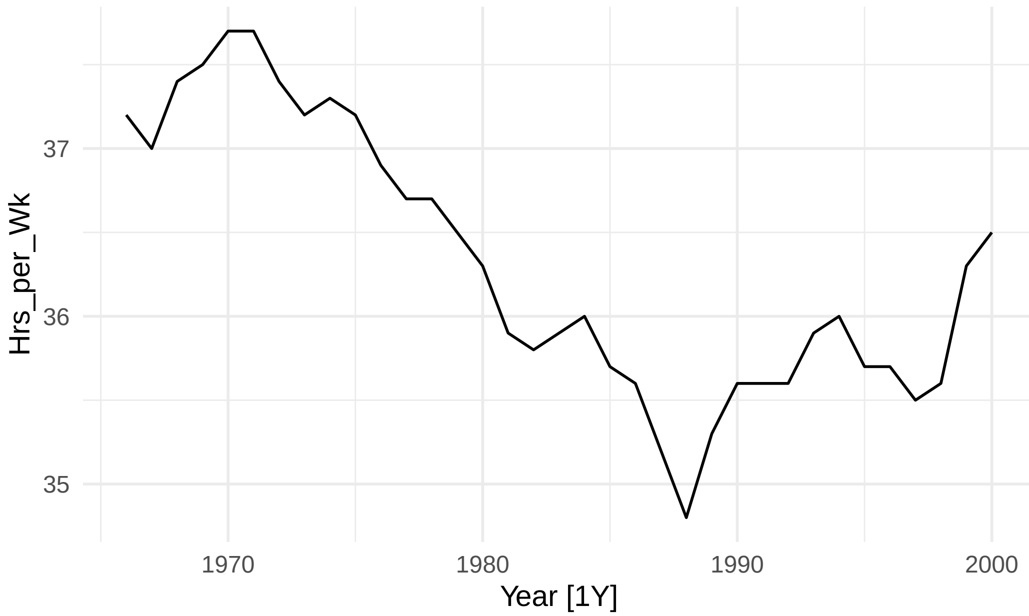


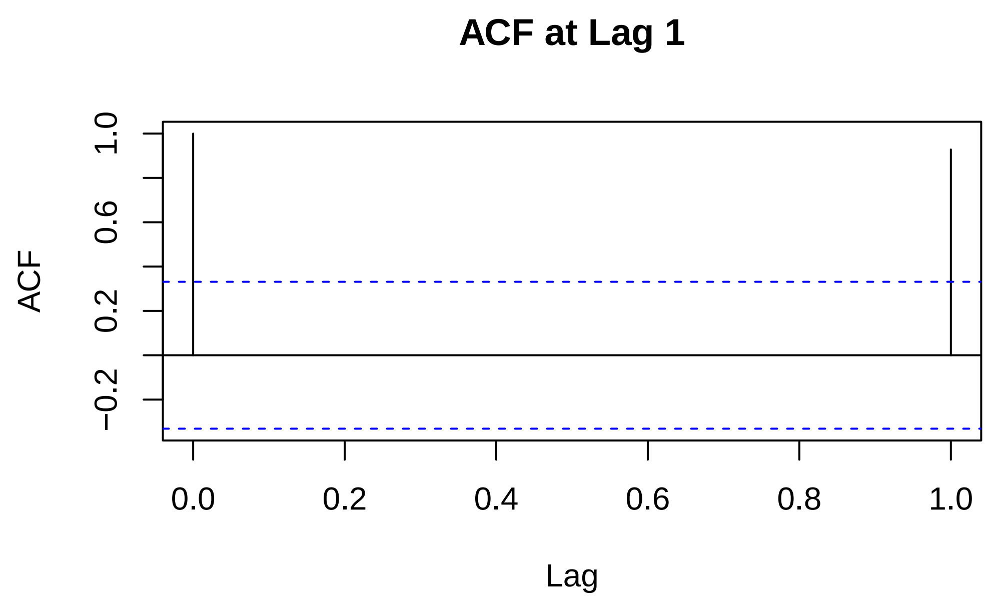
Figure 7.7

a. If we computed the autocorrelation of this series, would the lag-1 autocorrelation exhibit negative, positive, or no autocorrelation? How can you see this from the plot?

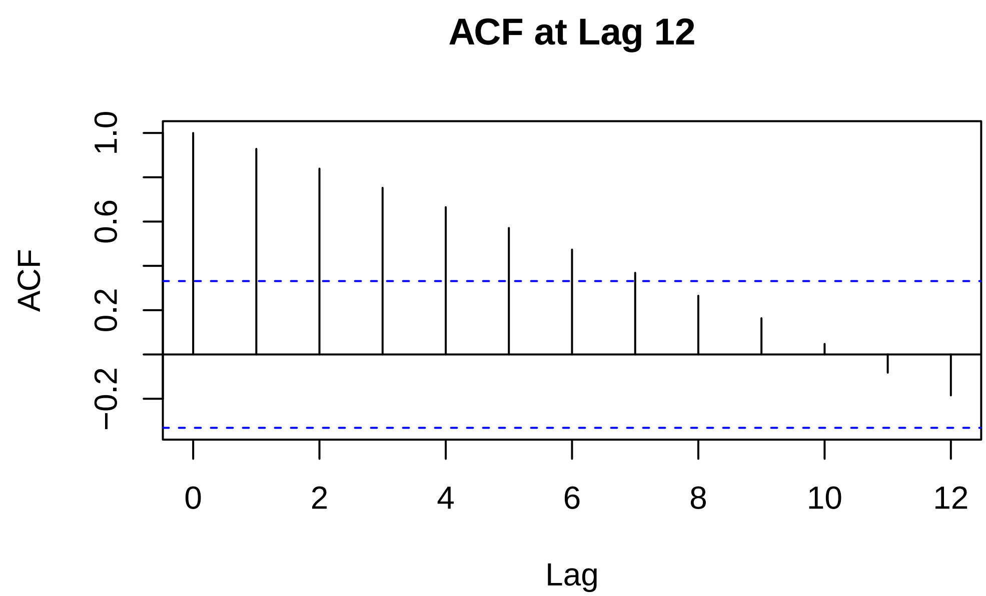
It will most likely exhibits a positive or stickiness autocorrelation at lag-1 because the original plot has a declining linear trend up until ~1988. Looking at the plot and knowing that lag-1 series is the original data moved one time period forward, the autocorrelation will most likely behave in a descending order.

b. Compute the autocorrelation and produce an ACF plot. Verify your answer to the previous question.

```
#code that produces an ACF plot should go here.

manu_wh = ts(work_hrs$Hrs_per_Wk, start = '1966', end = '2000', frequency = 1)
acf(manu_wh, lag.max = 1, main = 'ACF at Lag 1')
```




```r
acf(manu_wh, lag.max = 12, main = 'ACF at Lag 12')
```




```r
# The positive autocorrelation is more noticable at lag =12.
```

2. Forecasting Walmart Stock: Figure 7.10 shows a time plot of Wal-Mart daily closing prices between February 2001 and February 2002. The data is available at finance.yahoo.com and in

WalmartStock.csv.

The ACF plots of these daily closing prices and its lag-1 differenced series are in Figure 7.11. Table 7.4 shows the output from fitting an AR(1) model to the series of closing prices and to the series of differences. Use all the information to answer the following questions.

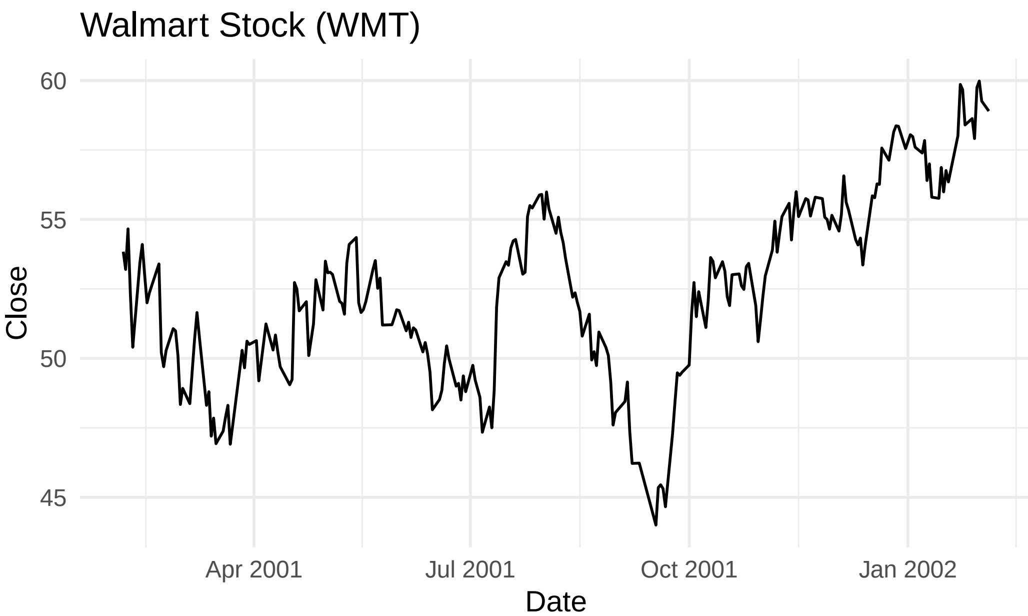


Figure 7.10

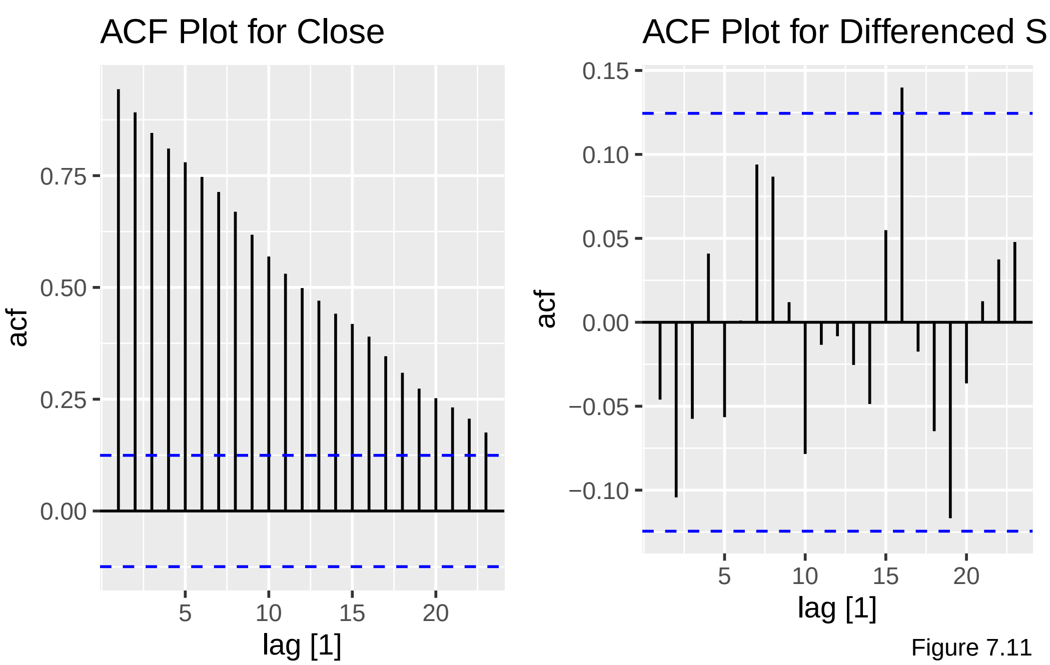
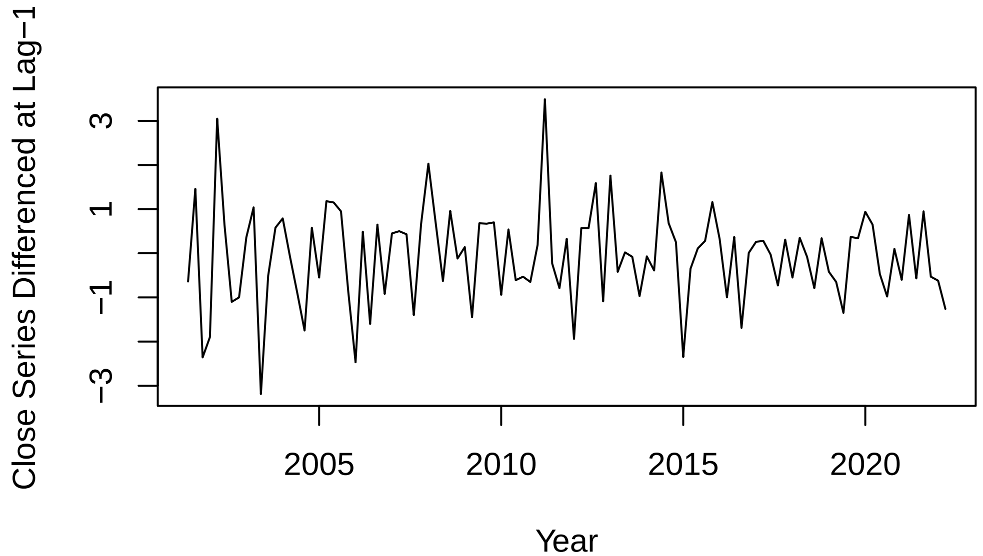


Figure 7.11

a. Create a time plot of the differenced series.

```
#code that produces a plot should go here

wmt_c = ts(wmt$Close, start = c(2001, 02), end = c(2022, 02), frequency = 5)
plot(diff(wmt_c, lag=1), ylab = 'Close Series Differenced at Lag-1', xlab = 'Year', main =
```



b. Which of the following is/are relevant for testing whether this stock is a random walk?

(enter "Yes" *OR* "No" in column 1 below)

Question	Answer
a) The autocorrelations of the closing price series.	No
b) The AR(1) slope coefficient for the closing price series.	Yes
c) The AR(1) constant coefficient for the closing price series.	No
d) The autocorrelations of the differenced series.	Yes
e) The AR(1) slope coefficient for the differenced series.	No
f) The AR(1) constant coefficient for the differenced series.	No

c. Recreate the AR(1) model output for the Close price series shown in the left panel of Table 7.4.

```
#HINT: to match the textbook we use the forecast::Arima() function
# output from fable::ARIMA() is also acceptable
```

```r
ar1_fit = arima(wmt_c, order = c(1, 0, 0))
summary(ar1_fit)
```

```

Call:
arima(x = wmt_c, order = c(1, 0, 0))

Coefficients:
         ar1  intercept
      0.8396    50.7033
s.e.  0.0551     0.6008

sigma^2 estimated as 1.082:  log likelihood = -155.21,  aic = 316.41

Training set error measures:
                      ME     RMSE       MAE        MPE     MAPE      MASE
Training set -0.03835921 1.040298 0.7907029 -0.1169893 1.556731 0.9451709
                    ACF1
Training set -0.02318388
```

```r
adf.test(wmt_c) # random walk test
```

```

    Augmented Dickey-Fuller Test

data:  wmt_c
Dickey-Fuller = -1.4783, Lag order = 4, p-value = 0.793
alternative hypothesis: stationary
```

Does the AR model indicate that this is a random walk? Explain how you reached your conclusion.

The augmented Dickey-Fuller test of the time series indicates that it is non-stationary meaning that it is random walk. The p value of 0.793 is higher than 0.05 sigsnificance level and so the null hypothesis that the series is non-stationary can't be rejected. Looking at the AR model, the slope coefficient is 0.8396 is not equals to 1.

d. What are the implications of finding that a time series is a random walk? Indicate the correct statement(s) below with 'Yes' OR 'No':

Question	Answer
a) It is impossible to obtain useful forecasts of the series.	Yes
b) The series is random.	No
c) The changes in the series from one period to the other are random.	Yes

6. Forecasting Weekly Sales at Walmart: The data in WalmartStore1Dept72.csv is a subset from a larger datasets on weekly department-wise sales at 45 Walmart stores, which were released by Walmart as part of a hiring contest hosted on kaggle.com. The file includes data on a single department at one specific store.

The fields include:

- Date - the week
- Weekly_Sales - sales for the given department in the given store
- IsHoliday - whether the week is a special holiday week
- Temperature - average temperature in the region
- Fuel_Price - cost of fuel in the region
- MarkDown1-5 - anonymized data related to promotional markdowns that
- Walmart is running. MarkDown data is only available after Nov 2011, and is not available for all stores all the time.
- CPI - the consumer price index
  Unemployment - the unemployment rate

Figure 7.15 shows a time plot of weekly sales in this department. We are interested in creating a forecasting model for weekly sales for the next 26 weeks.

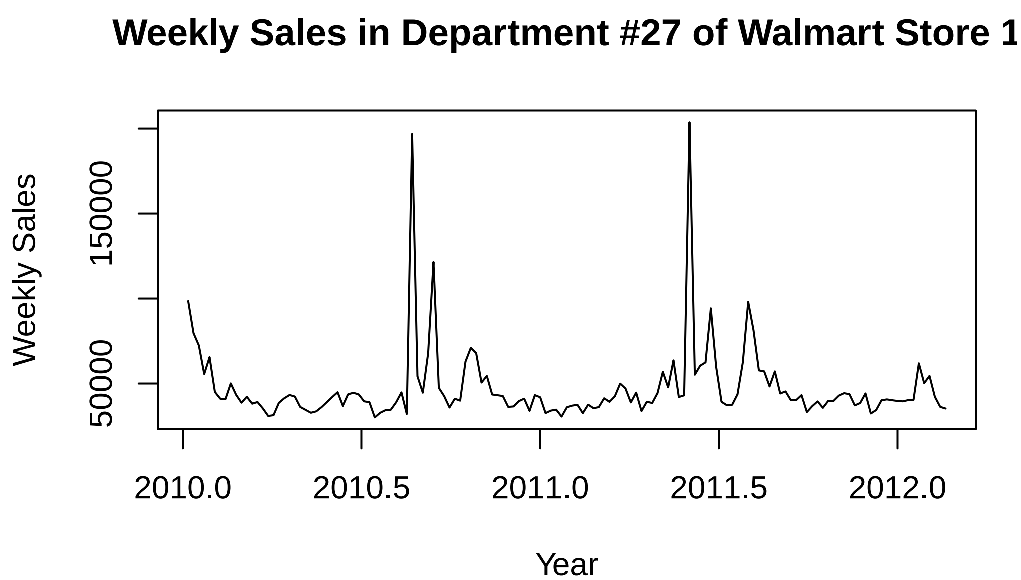
a. Recreate the time plot of the weekly sales data.

```
#code that produces a plot should go here

wmt_s1 = read_csv("WalmartStore1Dept72.csv", show_col_types = FALSE)
wmt_s1 = wmt_s1[order(as.Date(wmt_s1$Date, format = "%m/%d/%Y")),]
ws_ts = ts(wmt_s1$Weekly_Sales, start = c(2010, 2), end = c(2012, 10), frequency = 67)
ws_ts
```

```
Time Series:
Start = c(2010, 2)
End = c(2012, 10)
Frequency = 67
  [1]  98499.12  79636.32  72377.79  55590.73  65525.74  45046.81  41069.55
  [8]  40796.33  50079.67  43346.92  38711.22  42222.53  38131.53  39104.10
 [15]  35355.34  30922.75  31378.88  38638.60  41297.84  43244.02  42363.96
```

```
 [22]  36242.12  34523.84  32796.05  33638.52  36106.41  39070.70  42051.15
 [29]  44878.45  36690.76  43630.10  44555.16  43582.89  39528.86  39017.88
 [36]  30051.47  32775.14  34301.02  34596.86  38960.78  44777.94  32072.93
 [43] 196810.42  54268.03  44605.69  67832.13 121470.12  47448.91  42581.15
 [50]  35873.69  41051.86  39918.87  62891.35  71013.56  67900.90  50577.46
 [57]  54450.94  43535.72  43128.73  42627.73  36258.07  36512.94  39638.92
 [64]  41104.42  33941.57  43167.81  41898.67  32615.01  34073.97  34638.69
 [71]  30561.01  36041.85  36991.99  37493.58  32577.03  37570.26  35464.49
 [78]  36132.57  41315.33  39234.13  42481.55  49944.34  47054.71  38861.38
 [85]  44656.76  33809.02  39319.17  38528.36  44370.76  56909.42  47743.78
 [92]  63620.54  42060.48  43050.78 203670.47  55188.08  60494.84  62424.90
 [99]  94243.47  59732.61  39235.39  37248.54  37526.80  43660.60  62524.58
[106]  98104.80  81287.05  57718.19  57117.62  48275.27  57135.91  44132.05
[113]  45350.26  40174.72  40241.21  43176.95  33229.49  36683.63  39483.03
[120]  35718.85  39801.69  39791.94  42900.32  44313.56  43726.70  37125.37
[127]  38528.40  44166.90  32363.17  34394.82  40143.72  40663.95  40176.27
[134]  39729.78  39515.27  40231.04  40346.48  61883.75  50209.37  54480.13
[141]  42221.07  36267.08  35282.73
```

```
plot(ws_ts, xlab = 'Year', ylab = 'Weekly Sales',
     main = 'Weekly Sales in Department #27 of Walmart Store 1')
```

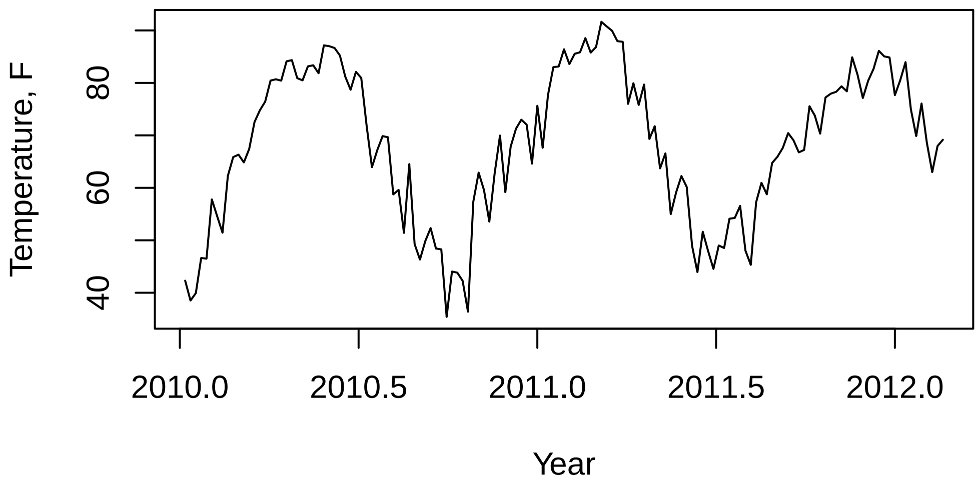


Which systematic patterns appear in this series?

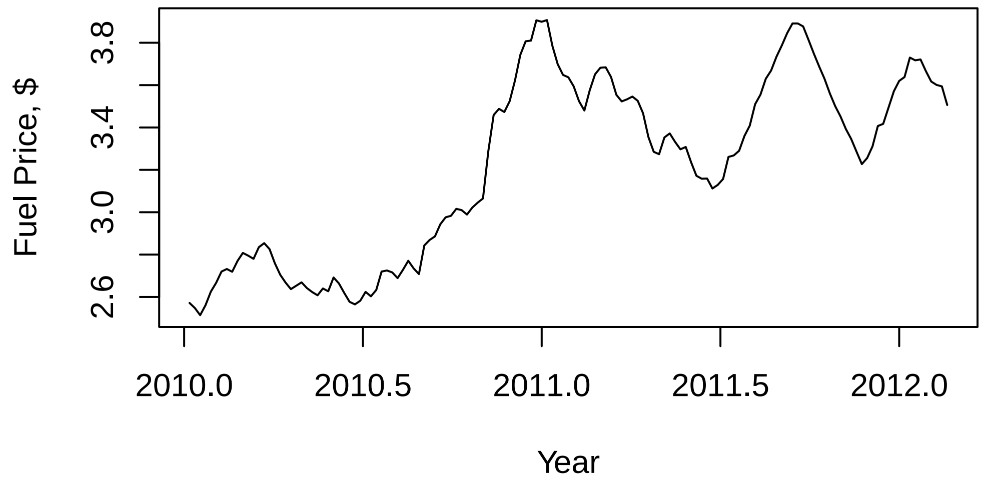
There seems to be a lack of trend as well as seasonality in the weekly sales time series. There is a cycle however that occurs around late 2010 and early 2011 with the two speak events.

b. Create time plots of the other numerical series (Temperature, Fuel_Price, CPI, and Unemployment). Also create scatter plots of the sales series against each of these four series (each point in the scatter plot will be a week).

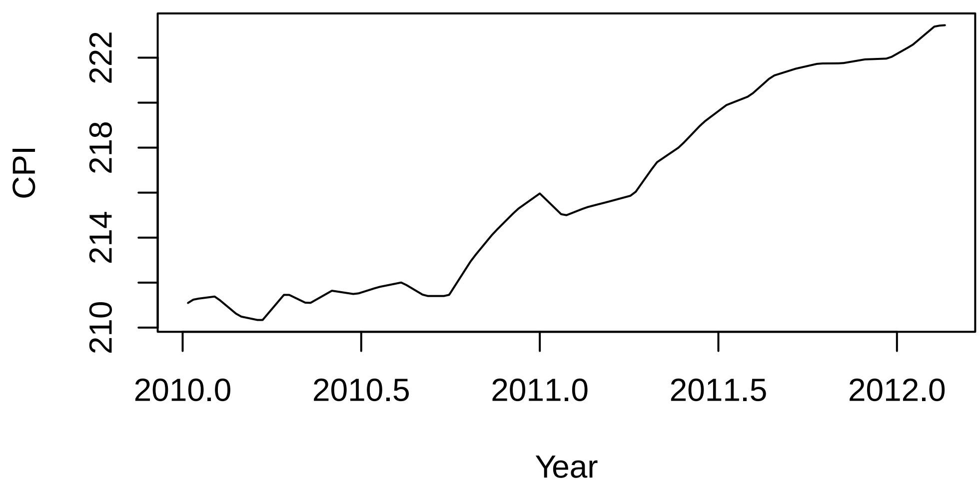
```
# code for external variables time series plots

ws_temp = ts(wmt_s1$Temperature, start = c(2010, 2), end = c(2012, 10), frequency = 67)
ws_fp = ts(wmt_s1$Fuel_Price, start = c(2010, 2), end = c(2012, 10), frequency = 67)
ws_cpi = ts(wmt_s1$CPI, start = c(2010, 2), end = c(2012, 10), frequency = 67)
ws_unem = ts(wmt_s1$Unemployment, start = c(2010, 2), end = c(2012, 10), frequency = 67)

plot(ws_temp, xlab = 'Year', ylab = 'Temperature, F')
```



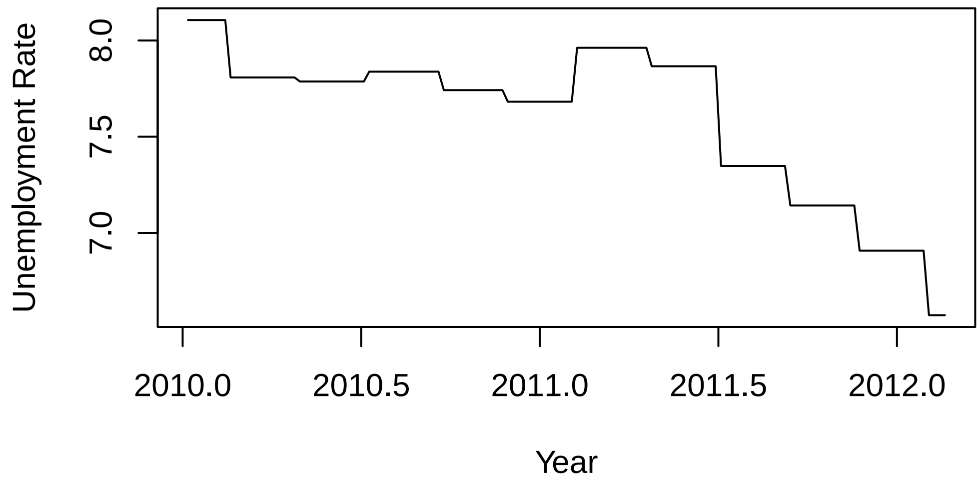

```
plot(ws_fp, xlab = 'Year', ylab = 'Fuel Price, $')
```



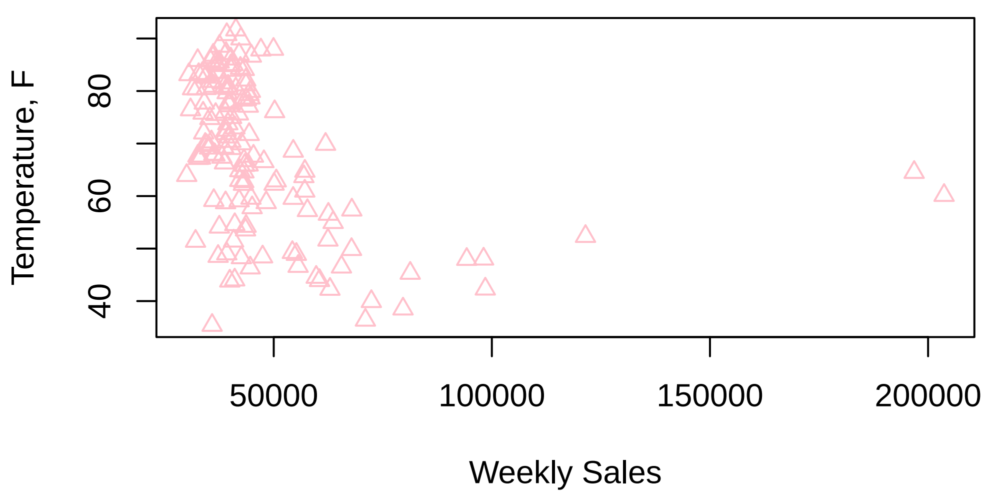

```
plot(ws_cpi, xlab = 'Year', ylab = 'CPI')
```



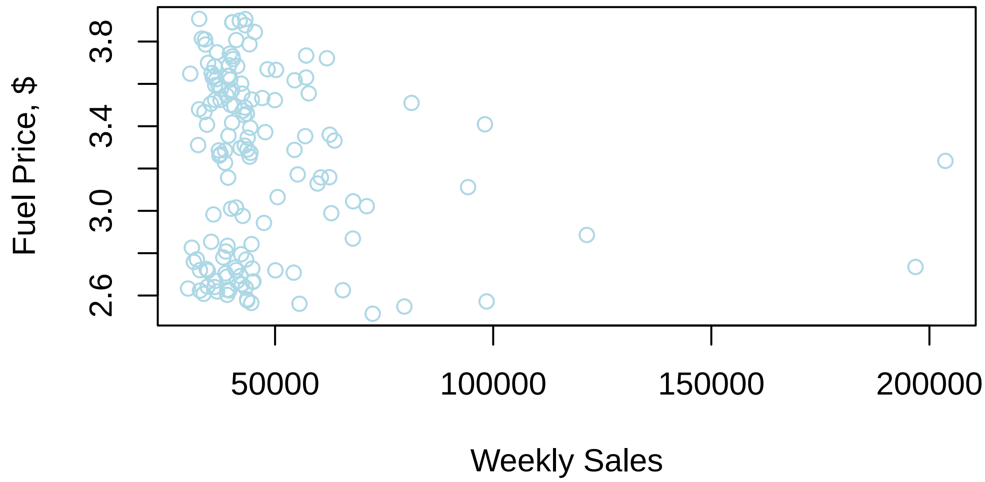

```
plot(ws_unem, xlab = 'Year', ylab = 'Unemployment Rate')
```



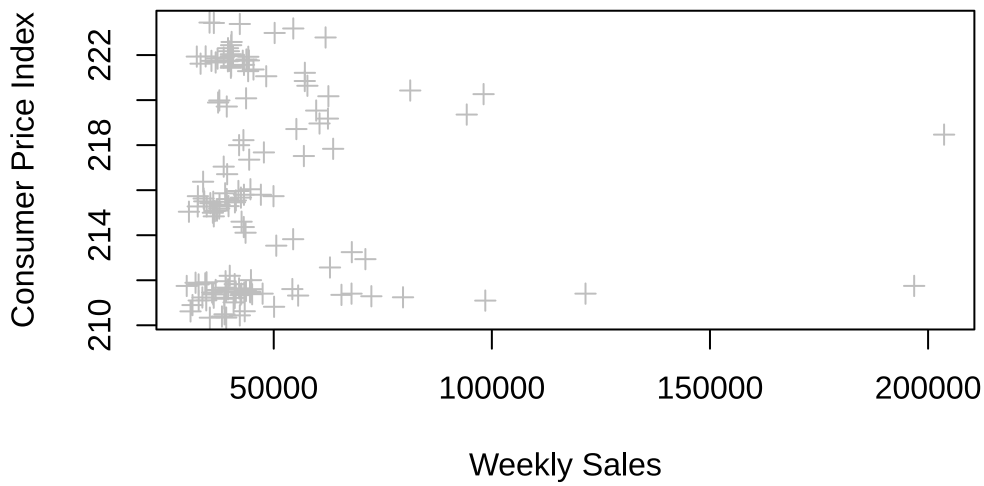

```
# code for external variables scatter plots

scat_ws = wmt_s1$Weekly_Sales
scat_temp = wmt_s1$Temperature
scat_fp = wmt_s1$Fuel_Price
scat_cpi = wmt_s1$CPI
scat_Unem = wmt_s1$Unemployment


plot(scat_ws,scat_temp, col='pink', pch=2, xlab = 'Weekly Sales',
     ylab = 'Temperature, F')
```



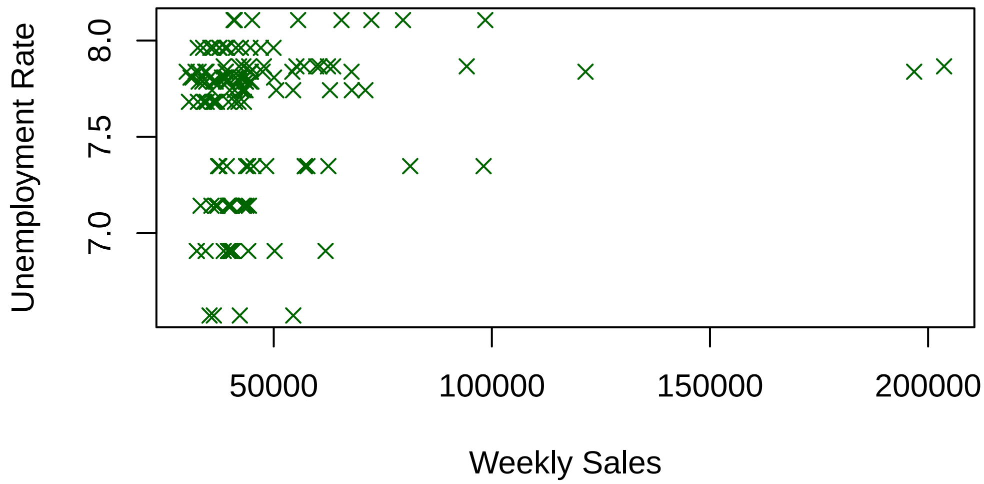
```
plot(scat_ws,scat_fp, col='lightblue', pch=1, xlab = 'Weekly Sales',
     ylab = 'Fuel Price, $')
```




```
plot(scat_ws,scat_cpi, col='gray', pch=3, xlab = 'Weekly Sales',
     ylab = 'Consumer Price Index')
```




```
plot(scat_ws,scat_Unem, col='darkgreen', pch=4, xlab = 'Weekly Sales',
     ylab = 'Unemployment Rate')
```



From the charts, which of the four series would potentially be useful as external predictors in a regression model for forecasting sales?

The fuel price seems to exhibit the less trend and seasonality among the rest of the series. CPI and Unemployment rate have clear trend while temperature is both seasonal and cyclical. Also, the scatter plot show similar spread across the four series against weekly sales. Fuel price might be the better choice as the external predictor.

The following questions are not in your textbook. You will need to also complete these programming questions in your submission notebook.

c. Fit an ARIMA model with 1 lag and external predictors for Weekly_Sales that treats Nov 4, 2011 to Oct 26, 2012 as the training period, and the next 26 weeks as the test period.

```
train_set = wmt_s1[92:143, ]

outcome = train_set$Weekly_Sales
predictors = as.matrix(train_set[c("Temperature", "Fuel_Price",
                                   "CPI", "Unemployment")])

fp = as.matrix(train_set[c("Fuel_Price")])
```

Compute the RMSE for the training period.

```
# model selection, fit and accuracy code goes here

arima_train_fit.a = arima(outcome, order=c(1,1,0), xreg =predictors)
summary(arima_train_fit.a)
```

```
Call:
arima(x = outcome, order = c(1, 1, 0), xreg = predictors)

Coefficients:
          ar1  Temperature  Fuel_Price        CPI  Unemployment
      -0.4661     297.1330   -15423.86   3432.071      43751.59
s.e.   0.1231     790.0336    56111.58  22545.953      41047.43

sigma^2 estimated as 834828427:  log likelihood = -596.33,  aic = 1204.66

Training set error measures:
                   ME     RMSE      MAE       MPE    MAPE     MASE       ACF1
Training set 268.9029 28614.56 13557.51 -5.987626 21.4786 1.000304 -0.1703106
```

```
arima_train_fit.fp = arima(outcome, order=c(1,1,0), xreg = fp)
summary(arima_train_fit.fp)
```

```
Call:
arima(x = outcome, order = c(1, 1, 0), xreg = fp)

Coefficients:
          ar1  Fuel_Price
      -0.4457   -13599.71
s.e.   0.1237    56429.93

sigma^2 estimated as 857485301:  log likelihood = -597,  aic = 1200

Training set error measures:
                    ME     RMSE      MAE       MPE     MAPE      MASE
Training set -656.8596 28999.92 13472.05 -8.219491 21.49947 0.9939992
                   ACF1
Training set -0.1609225
```

```
arima_train_fit = arima(outcome, order=c(1,1,0))
summary(arima_train_fit)
```

```
Call:
arima(x = outcome, order = c(1, 1, 0))

Coefficients:
          ar1
      -0.4458
s.e.   0.1237

sigma^2 estimated as 858462125:  log likelihood = -597.03,  aic = 1198.06

Training set error measures:
                    ME     RMSE      MAE      MPE     MAPE      MASE       ACF1
Training set -734.7419 29016.43 13464.48 -8.35883 21.47435 0.9934411 -0.1604928
```

d. Create a mean forecasts for the test period. Create a time plot of the fitted values and a plot of the model residuals. Compute the RMSE for the training period.

arima_sforecast

```r
# mean model code goes here

wmt_val = read.csv('WalmartStore1Dept72_validation.csv',
                   stringsAsFactors = FALSE)
wmt_valt = as.matrix(wmt_val[c("Temperature", "Fuel_Price",
                               "CPI", "Unemployment")])


arima_forecast = predict(arima_train_fit.a, newxreg=wmt_valt) # no residual
summary(arima_forecast)
```

```
     Length Class Mode
pred 26     ts    numeric
se    1     ts    numeric
```

```r
arima_sforecast = forecast(arima_train_fit) # just to test forecast
summary(arima_sforecast)
```

```
Forecast method: ARIMA(1,1,0)
```

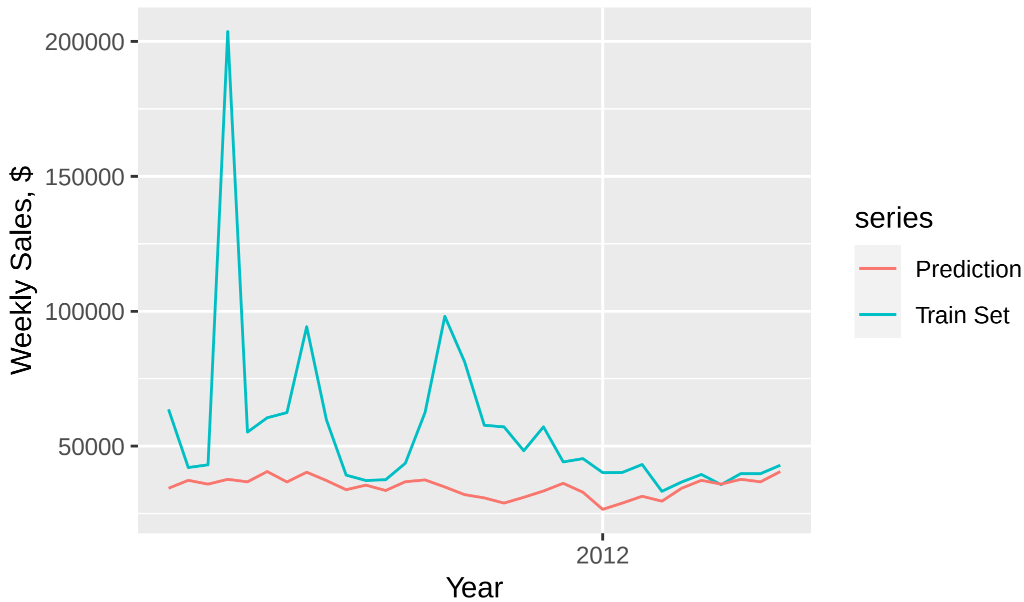
```
Model Information:

Call:
arima(x = outcome, order = c(1, 1, 0))

Coefficients:
          ar1
      -0.4458
s.e.   0.1237

sigma^2 estimated as 858462125:  log likelihood = -597.03,  aic = 1198.06

Error measures:
                    ME     RMSE      MAE      MPE     MAPE      MASE       ACF1
Training set -734.7419 29016.43 13464.48 -8.35883 21.47435 0.9934411 -0.1604928

Forecasts:
   Point Forecast      Lo 80     Hi 80     Lo 95     Hi 95
53       35721.51  -1827.346  73270.36 -21704.51  93147.52
54       35525.92  -7404.607  78456.45 -30130.66 101182.50
55       35613.10 -15790.634  87016.84 -43002.13 114228.33
56       35574.24 -21563.067  92711.55 -51809.73 122958.21
57       35591.56 -27362.064  98545.19 -60687.70 131870.83
58       35583.84 -32439.355 103607.04 -68448.66 139616.35
59       35587.28 -37257.881 108432.45 -75819.78 146994.35
60       35585.75 -41737.302 112908.80 -82669.65 153841.15
61       35586.43 -45987.638 117160.51 -89170.34 160343.21
62       35586.13 -50020.230 121192.49 -95337.50 166509.76
```

```r
#arima_mforecast = forecast(arima_train_fit.a, xreg = wmt_valt) # error message
#summary(arima_mforecast)
```

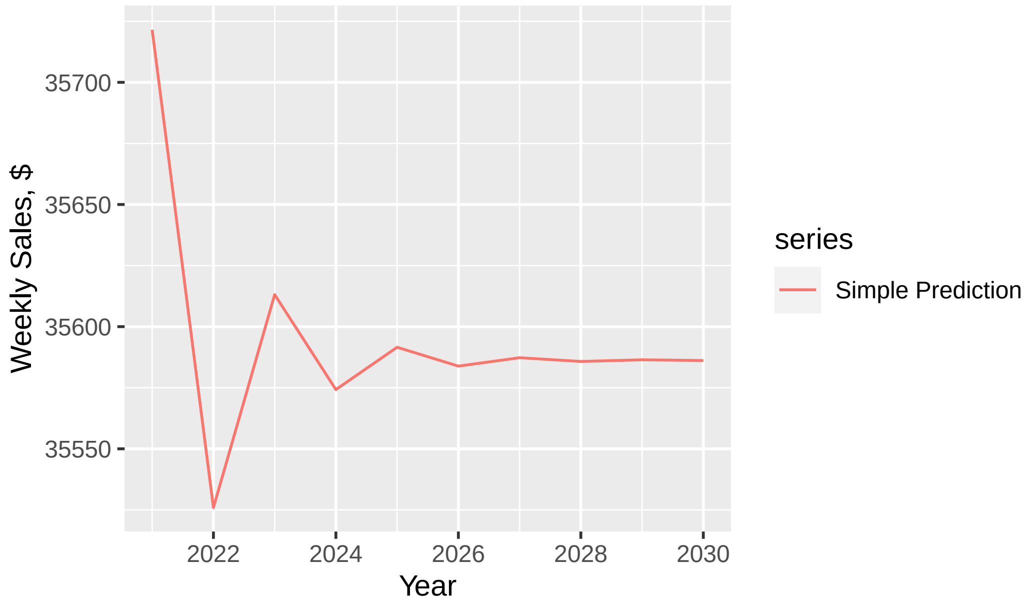
```r
# mean and arima model plots  code goes here

ws_train = ts(train_set$Weekly_Sales, start = c(2011, 11),
              end = c(2012, 10), frequency = 32)

arima_pred = ts(data.frame(arima_forecast$pred), start = c(2011, 11),
                end = c(2012, 10), frequency = 32)

sf = data.frame(arima_sforecast)
```

```
arima_sf = ts(sf$Point.Forecast, start = c(2011, 11))
```

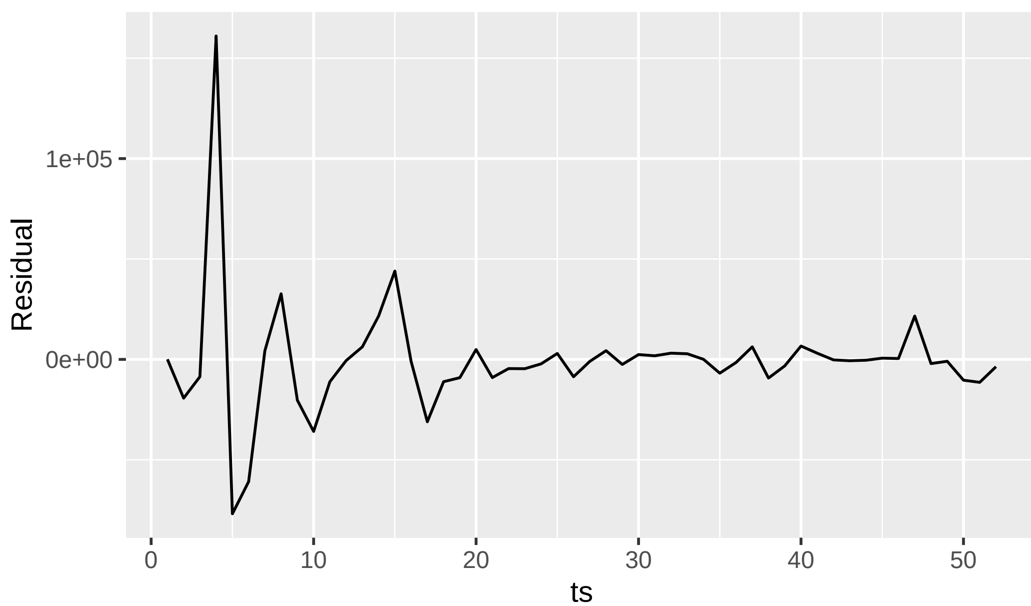
```
autoplot(ws_train, ylab = "Weekly Sales, $", xlab = "Year", series = 'Train Set') +
   autolayer(arima_pred, series = "Prediction")
```




```
autoplot(arima_sf, ylab = "Weekly Sales, $", xlab = "Year", series = 'Simple Prediction')
```




```
# arima model residuals plot code goes here

autoplot(arima_sforecast$residuals, ylab = "Residual", xlab = "ts")
```




```
#  model accuracy comparison code goes here
```

e. Compare the performance of the ARIMA model to the mean over the training period. Which one performs better?

Was not successful to get forecast function to run properly with the xreg. No comparison.

f. Plot the ARIMA model forecasted values. Use WalmartStore1Dept72_validation.csv for your regression model data.

```
# code for ARIMA model forecasted values plot goes here
```