Project 1

Project 1

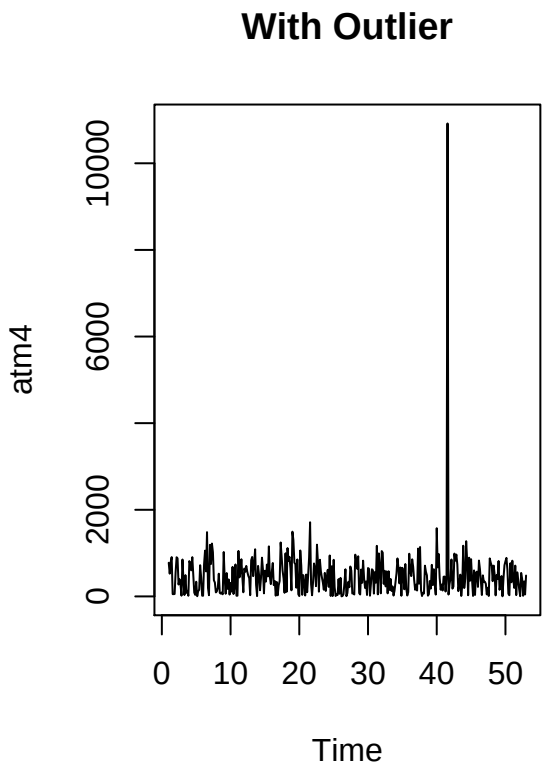
Part A

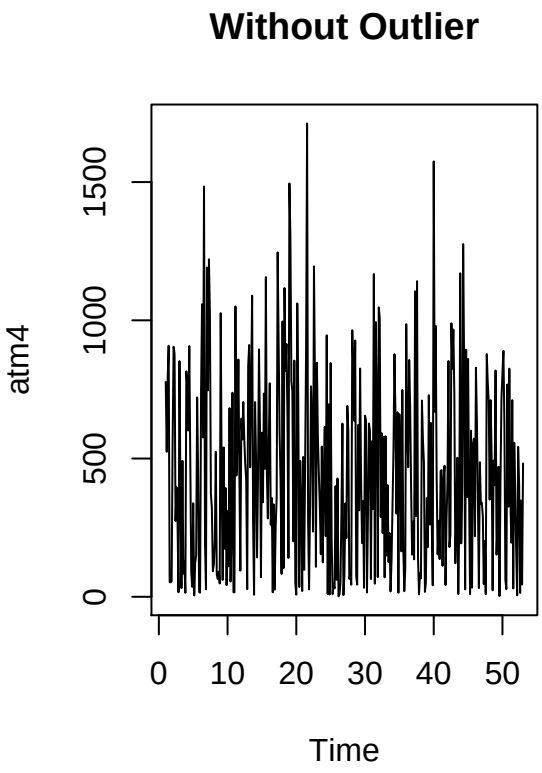
Reading Data

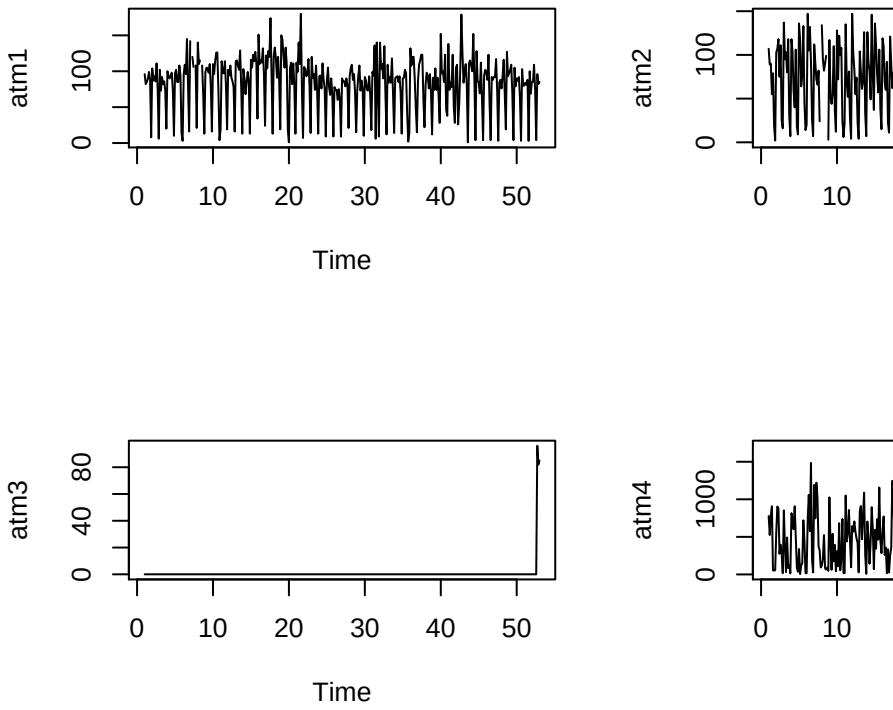
First, I imported the data using Rstudio and the `readxl` package. There are 365 entries for each atm, so I assumed that each is an independent dataset for one of 4 atms starting on May 1st 2009. I have been tasked with forecasting the next 30 days.

Cleaning

A single data point from ATM4 was five times the value of any of the others. I assume this is a data entry error. To minimize forecasting error, I replaced this value with the mean of the ATM4 data. This makes the data for ATM4 look like the data for ATMs one and two.

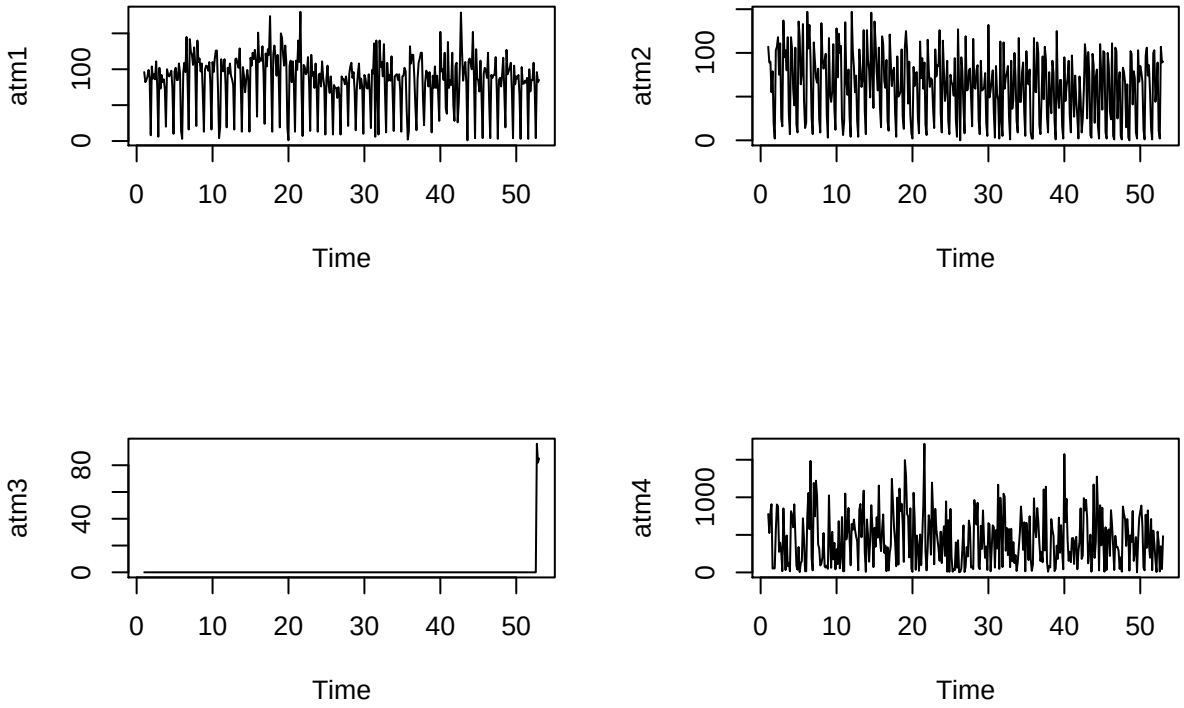






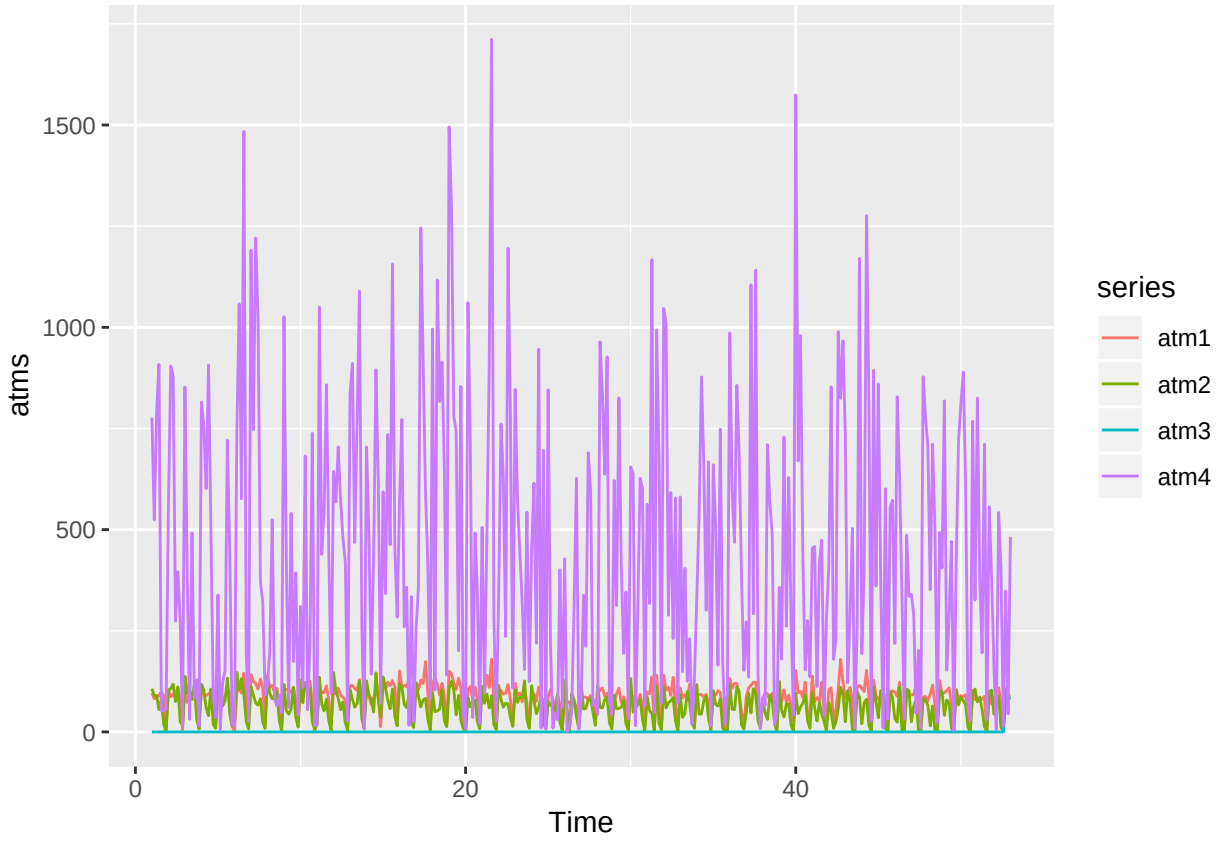
Atm 3 was installed recently, so it has less data.

Due to what I assume is reporting errors, there are several missing data points. To fill in the gaps, I averaged the day before and after to fill any null space. This technique is known as linear interpolation and is widely accepted as a simple method for interpolation. This step is necessary because forecasting models to not work



with null data points.

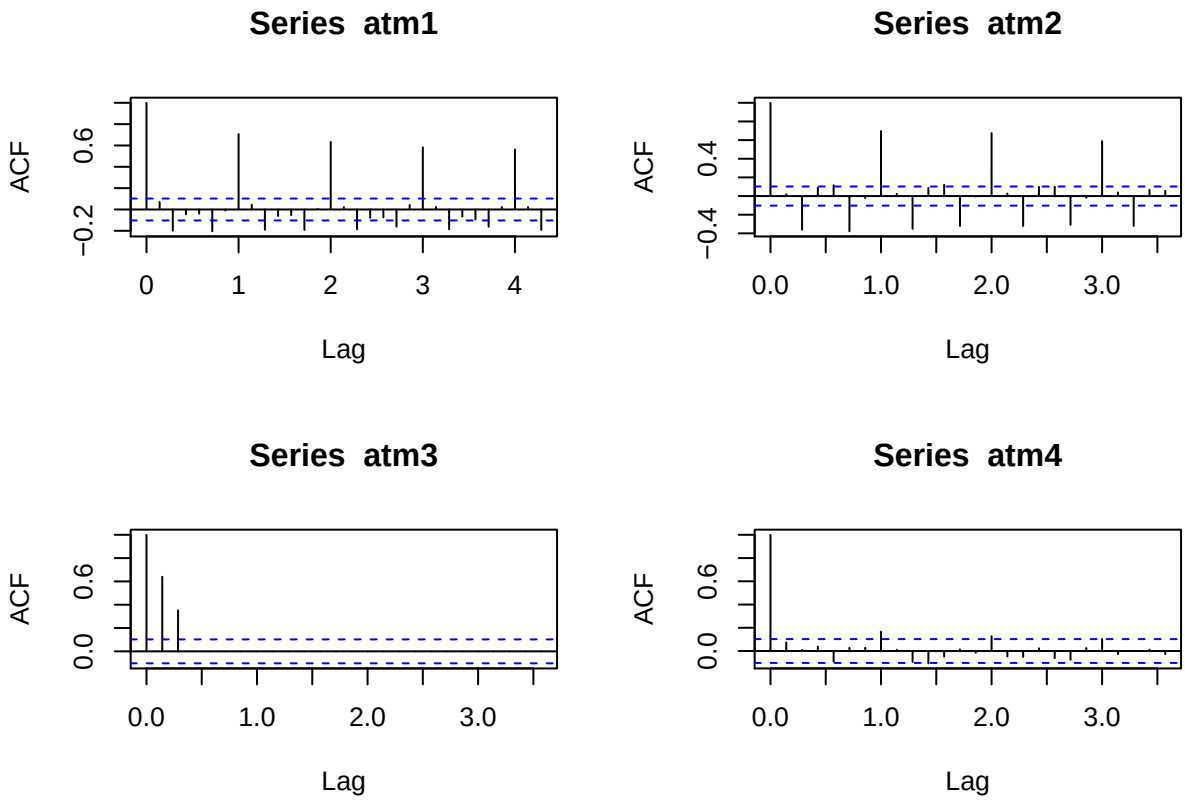
The data appears to be noisy, with random variance. However, we have to check for autocorrelation, trends,



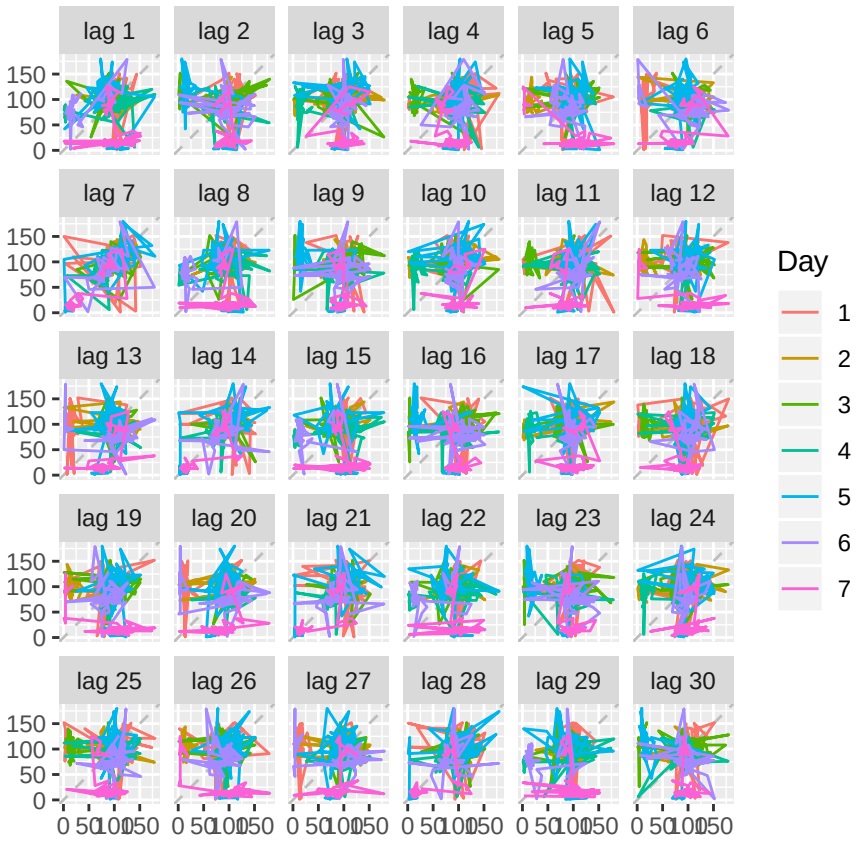
and seasonality.

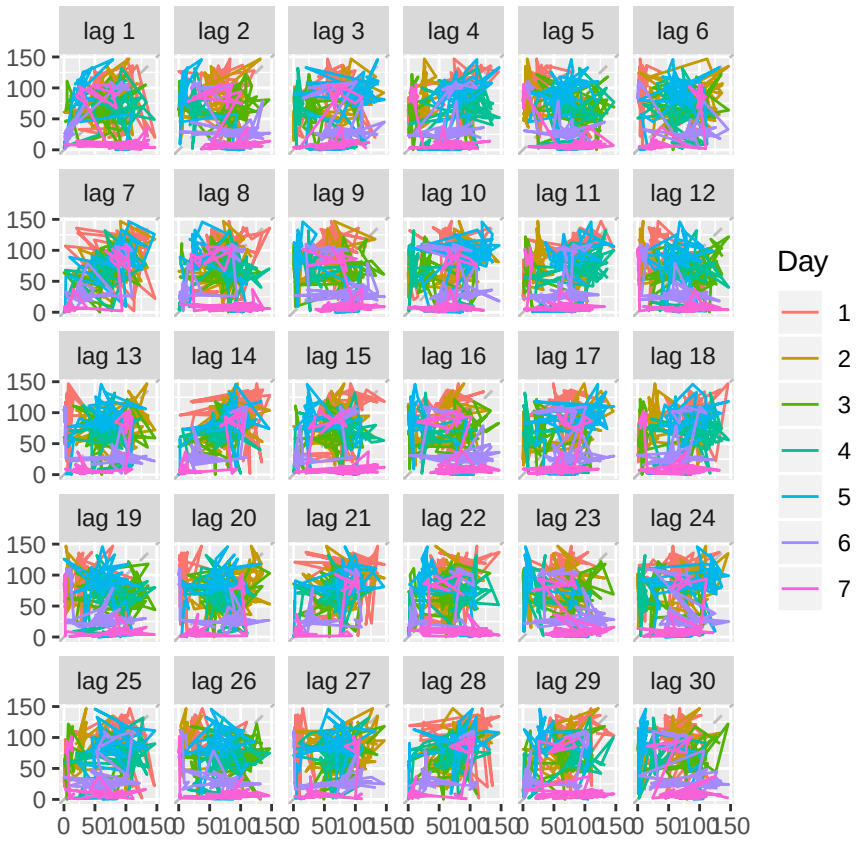
Preparing Data

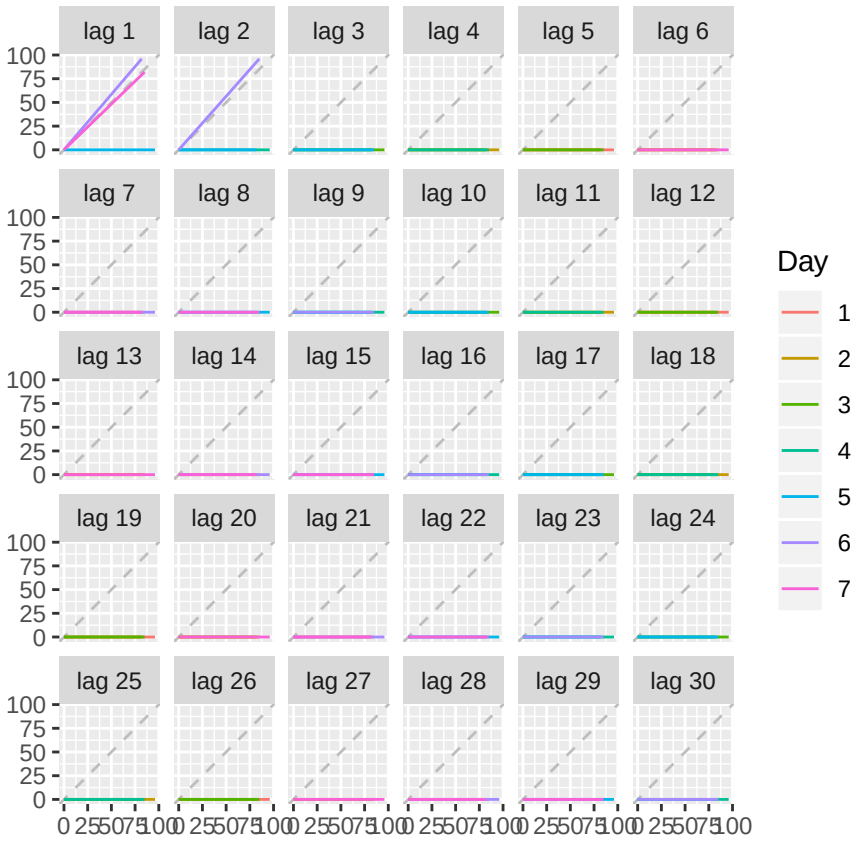
In order to perform a forecast on the data, it must be free of integer lags. The ACF plots below show the relationships between data points and the same data various times in the past (week-to-week or month-to-month, for example). These plots merely confirm the presence of lag rather than the size or length.

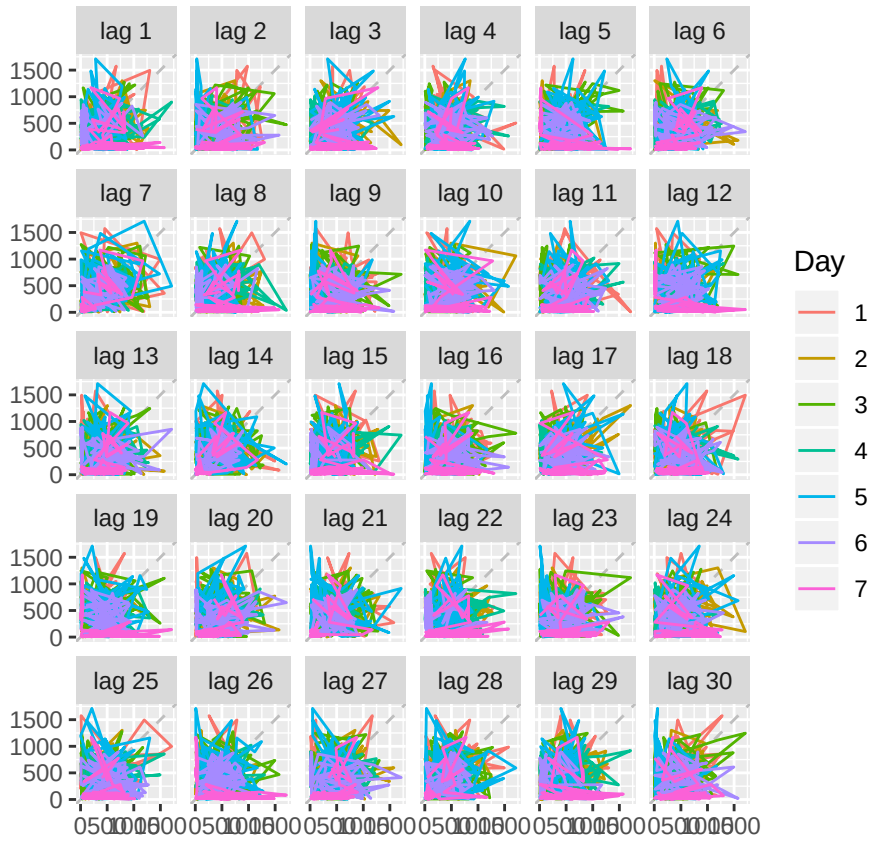


Luckily, we can use lag plots to see which lag is causing the issues. A plot that more strongly resembles a line corresponds to stronger lag at that number. As we can see across the datasets, lag 7 appears the strongest. Additionally, we can see a diminishing trend in each of the multiples of 7 (as we'd expect). Below are lags 1-30 for each of four atms.


lag 1
lag 2
lag 3
lag 4
lag 5
lag 6
lag 7
lag 8
lag 9
lag 10
lag 11
lag 12
lag 13
lag 14
lag 15
lag 16
lag 17
lag 18
lag 19
lag 20
lag 21
lag 22
lag 23
lag 24
lag 25
lag 26
lag 27
lag 28
lag 29
lag 30
150
100
50
0
Day
1
2
3
4
5
6
7


lag 1
lag 2
lag 3
lag 4
lag 5
lag 6
lag 7
lag 8
lag 9
lag 10
lag 11
lag 12
lag 13
lag 14
lag 15
lag 16
lag 17
lag 18
lag 19
lag 20
lag 21
lag 22
lag 23
lag 24
lag 25
lag 26
lag 27
lag 28
lag 29
lag 30
0
50
100
150
Day
1
2
3
4
5
6
7


lag 1
lag 2
lag 3
lag 4
lag 5
lag 6
lag 7
lag 8
lag 9
lag 10
lag 11
lag 12
lag 13
lag 14
lag 15
lag 16
lag 17
lag 18
lag 19
lag 20
lag 21
lag 22
lag 23
lag 24
lag 25
lag 26
lag 27
lag 28
lag 29
lag 30
0
25
50
75
100
Day
1
2
3
4
5
6
7



This cell shows the output of four different KPSS tests which determine whether or not a time series needs to be differenced in order to be stationary. Only ATM 2 has sifnificant autocorrelation.

```
## [1] 0
```

```
## [1] 1
```

```
## [1] 0
```

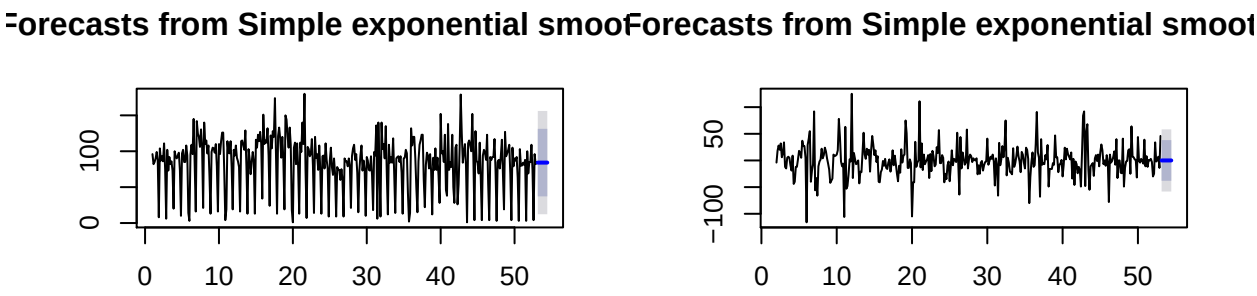
```
## [1] 0
```

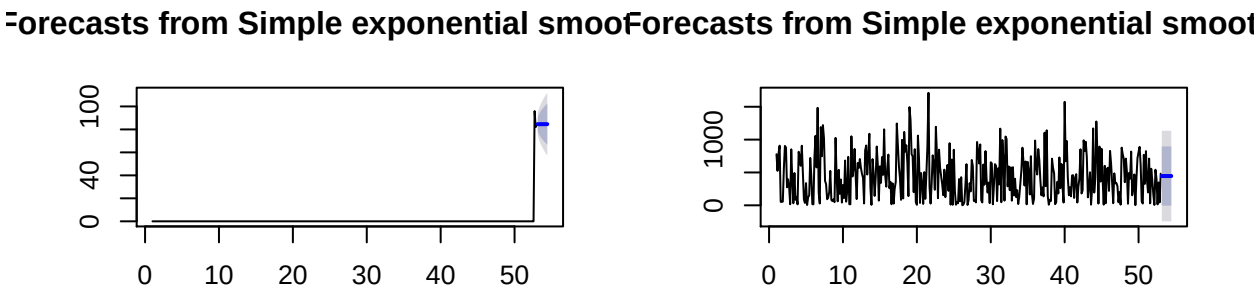
After a single differencing, we see that atm2 passes the KPSS test.

```
## [1] 1
```

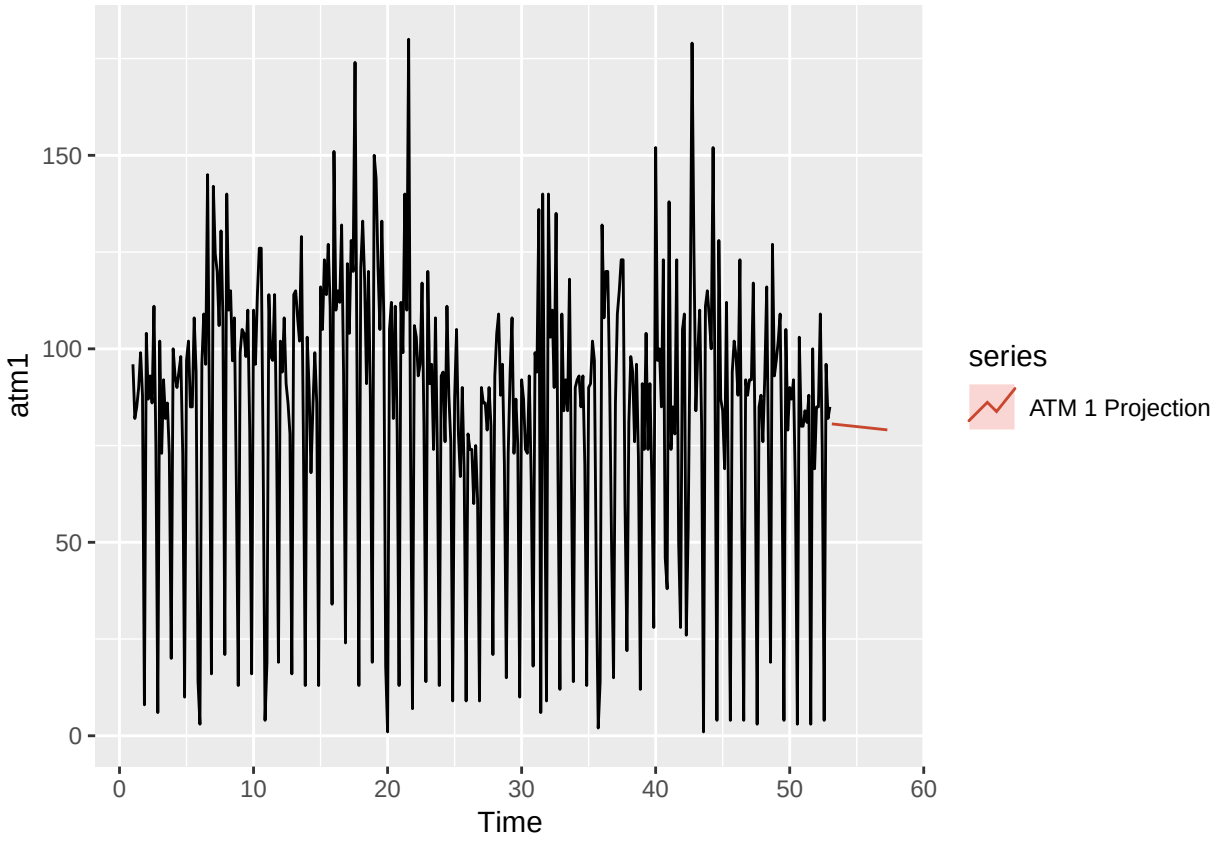
Modelling

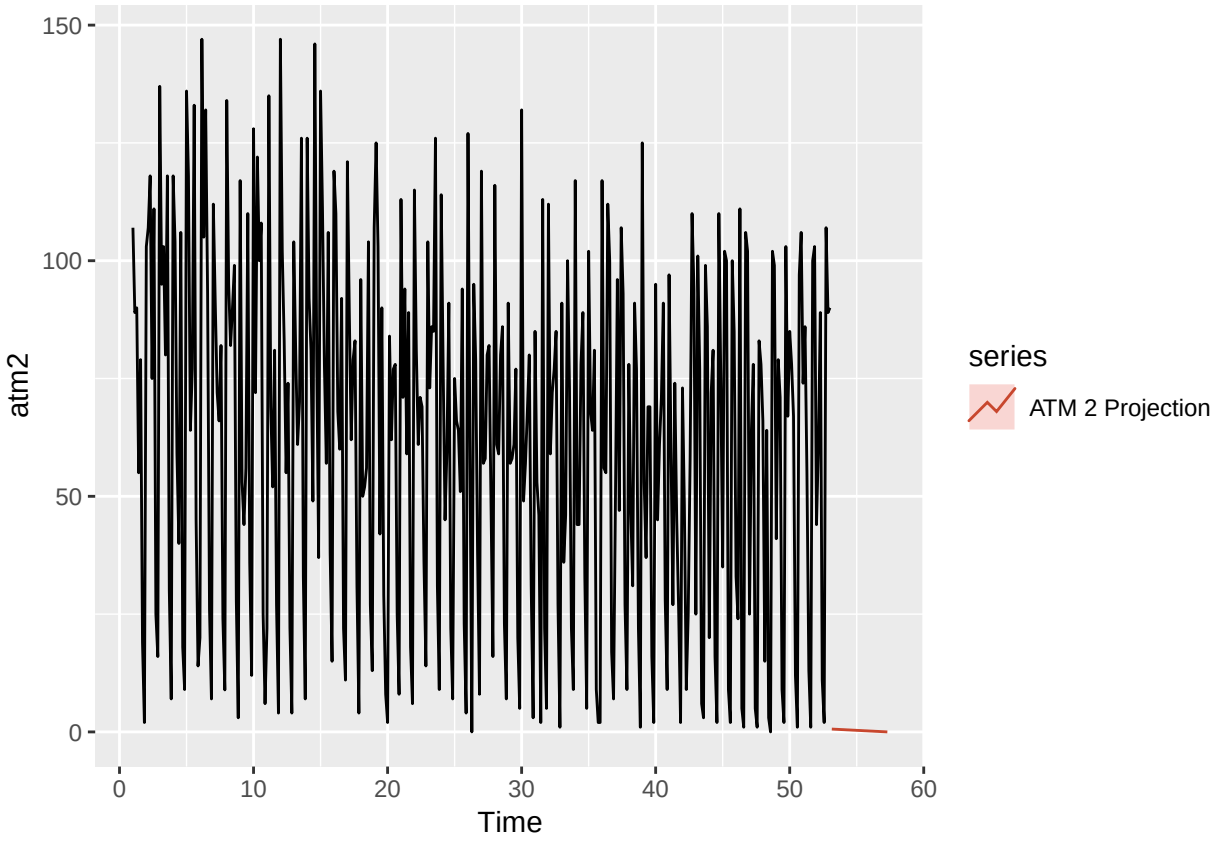
The simplest method is simple exponential smoothing. It is appropriate for forecasting data with no clear seasonal pattern or long-term trends.

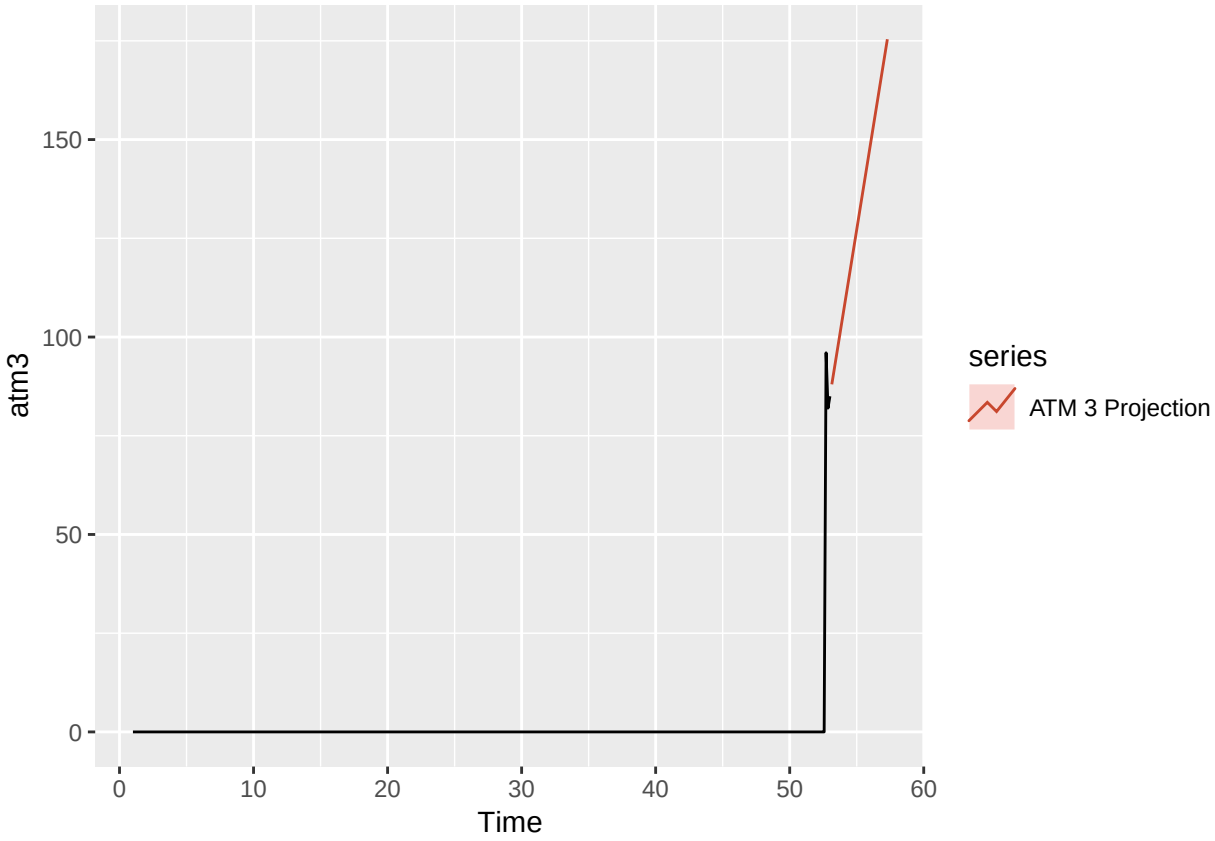


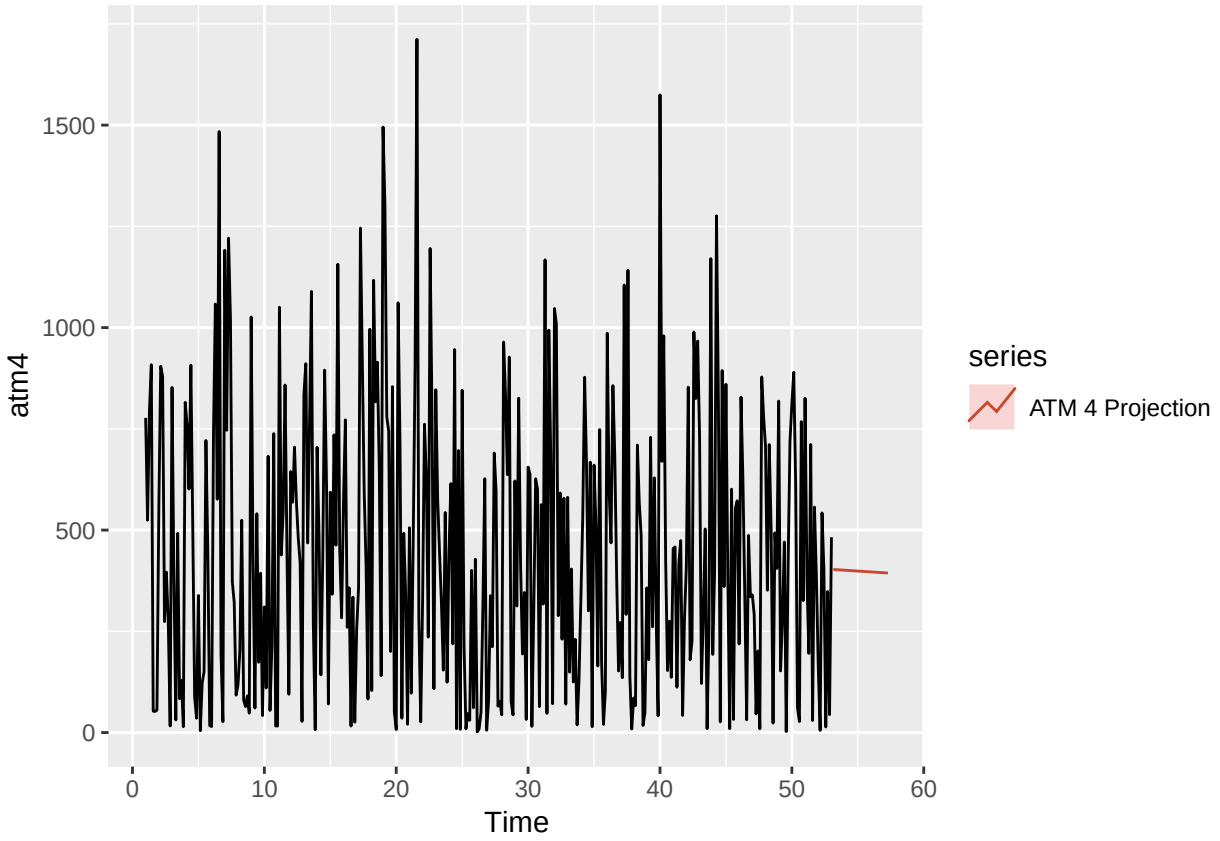


The Holt method is like the simple exponential smoothing method above, but includes a trend parameter. As you can see, each ATM is predicted to go up or down, depending on the preceding data.

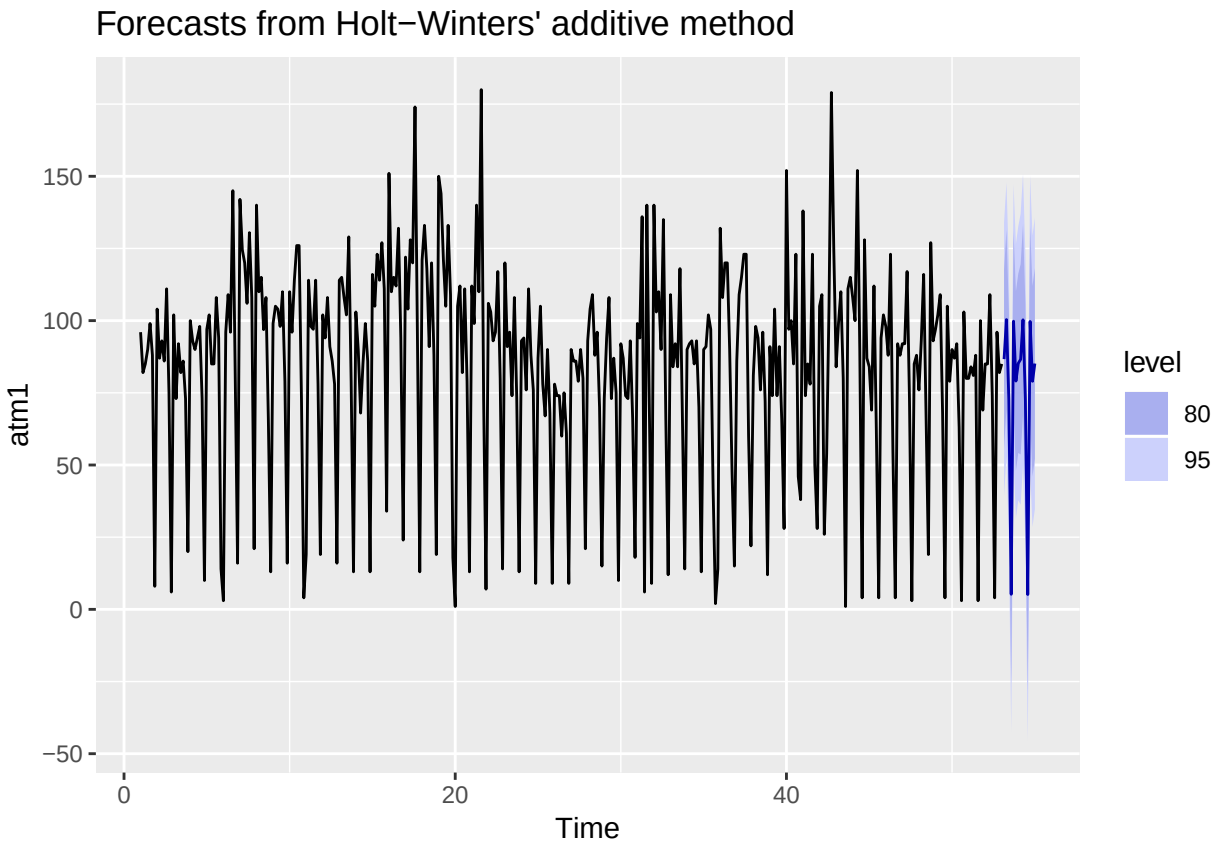

150
100
50
0
atm1
0
10
20
30
40
50
60
Time
series
ATM 1 Projection

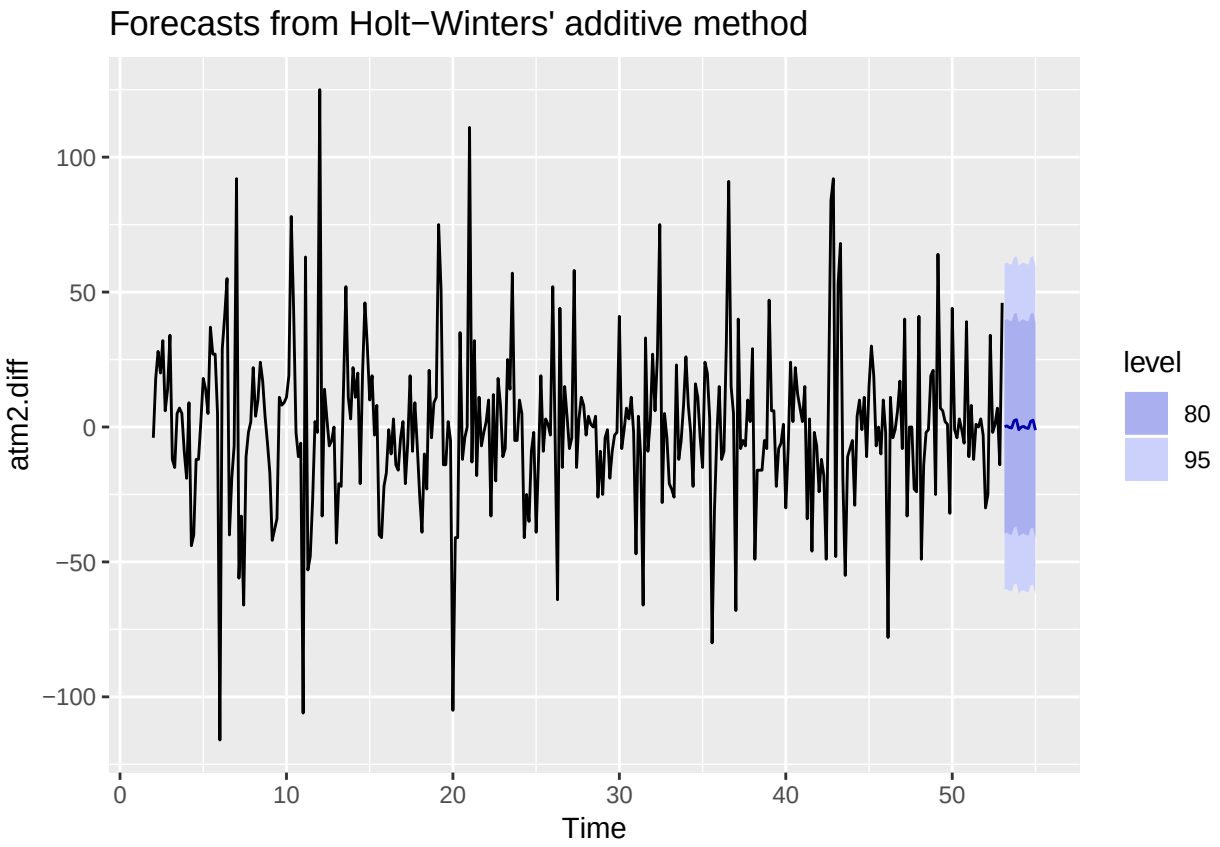

150
100
50
0
atm2
0
10
20
30
40
50
60
Time
series
ATM 2 Projection

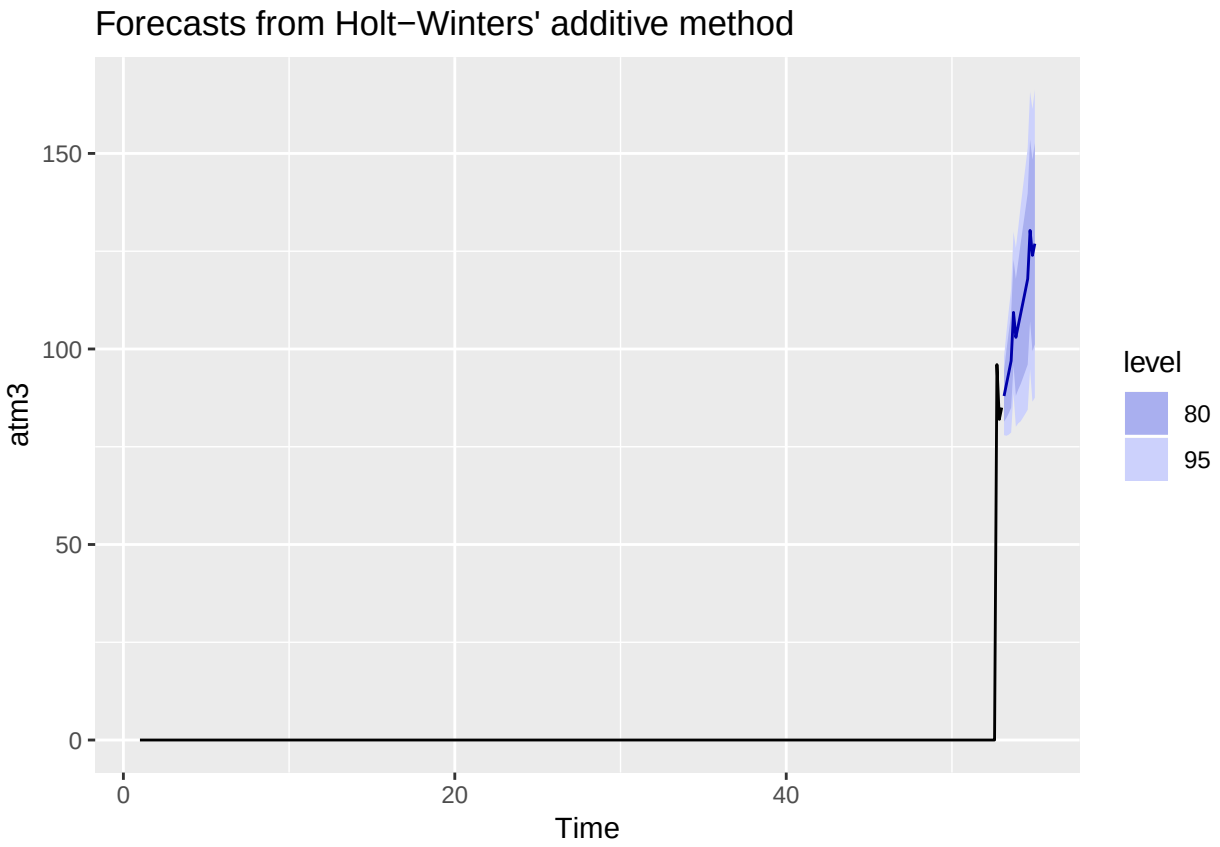

150
100
50
0
atm3
0
10
20
30
40
50
60
Time
series
ATM 3 Projection

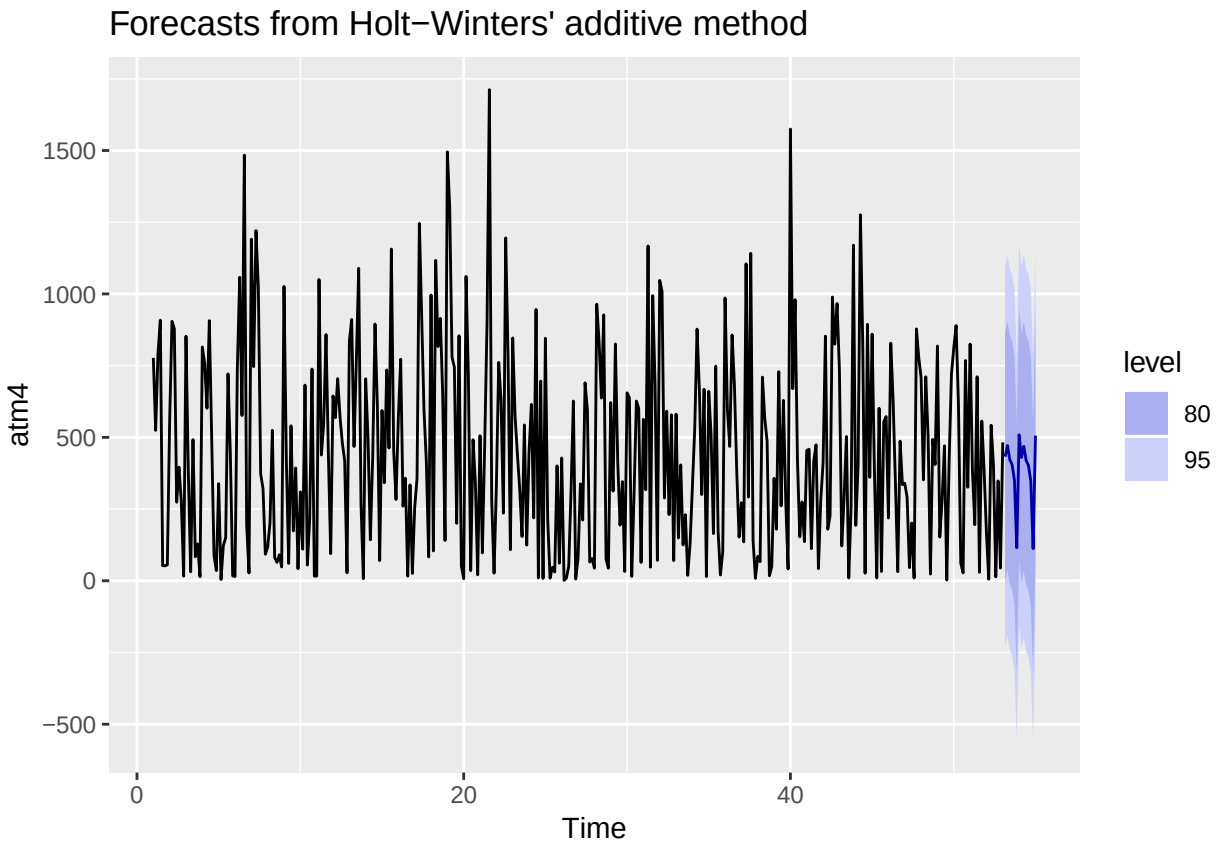


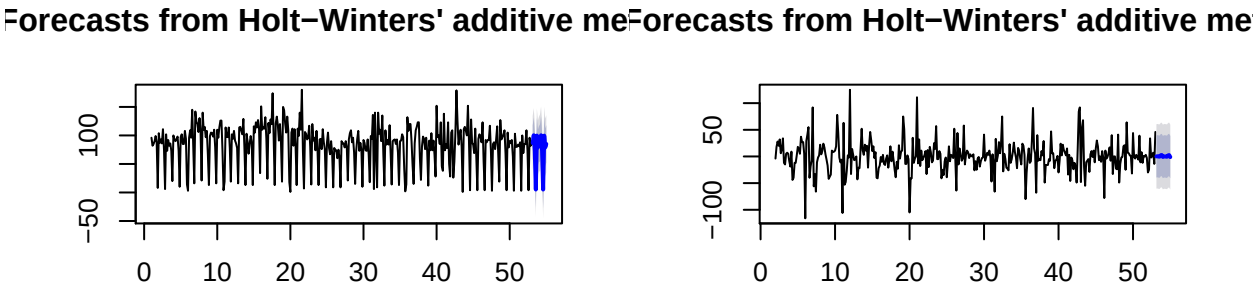
The Holt-Winters model uses differencing, trends, and seasonality to build a model. It chooses these parameters so as to minimize error.

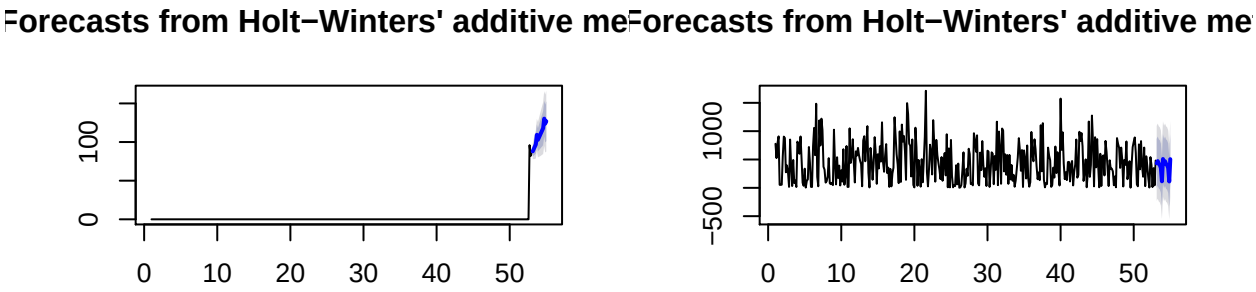

Forecasts from Holt−Winters' additive method
atm1
150
100
50
0
−50
0
20
40
Time
level
80
95


Forecasts from Holt−Winters' additive method
atm2.diff
100
50
0
−50
−100
0
10
20
30
40
50
Time
level
80
95

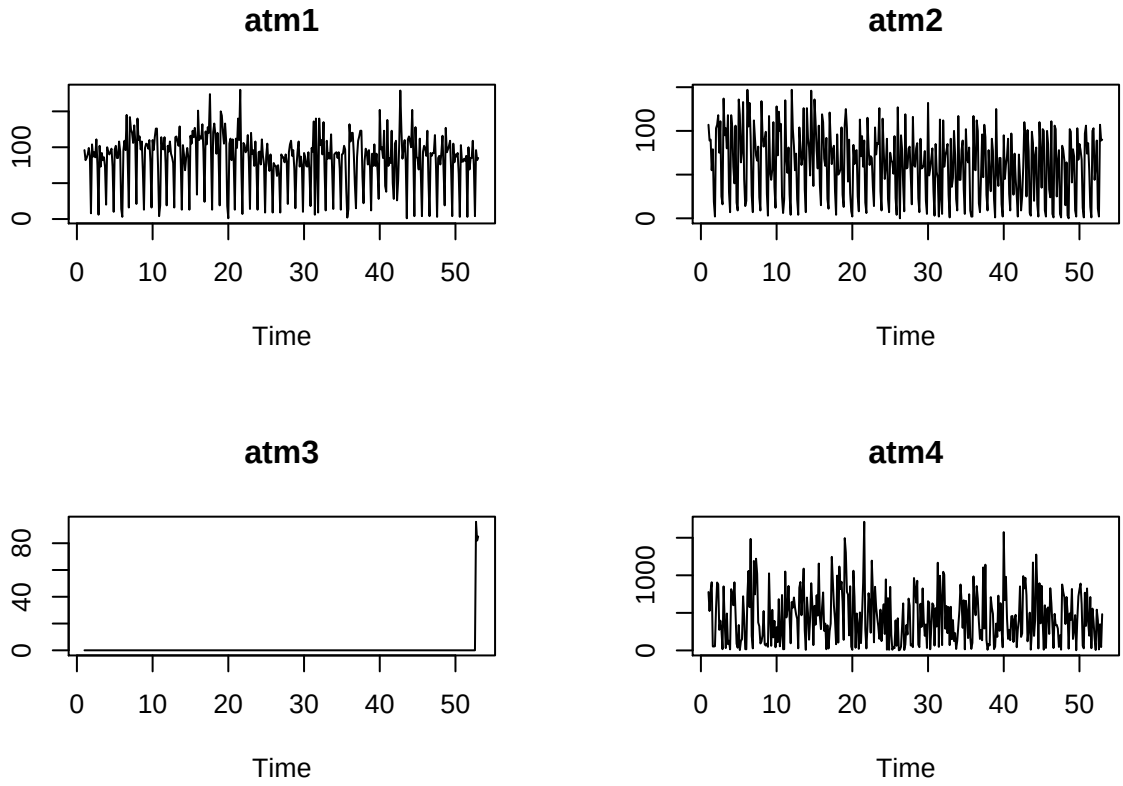

Forecasts from Holt−Winters' additive method
150
100
50
0
atm3
0
20
40
Time
level
80
95


Forecasts from Holt−Winters' additive method
atm4
1500
1000
500
0
−500
0
20
40
Time
level
80
95

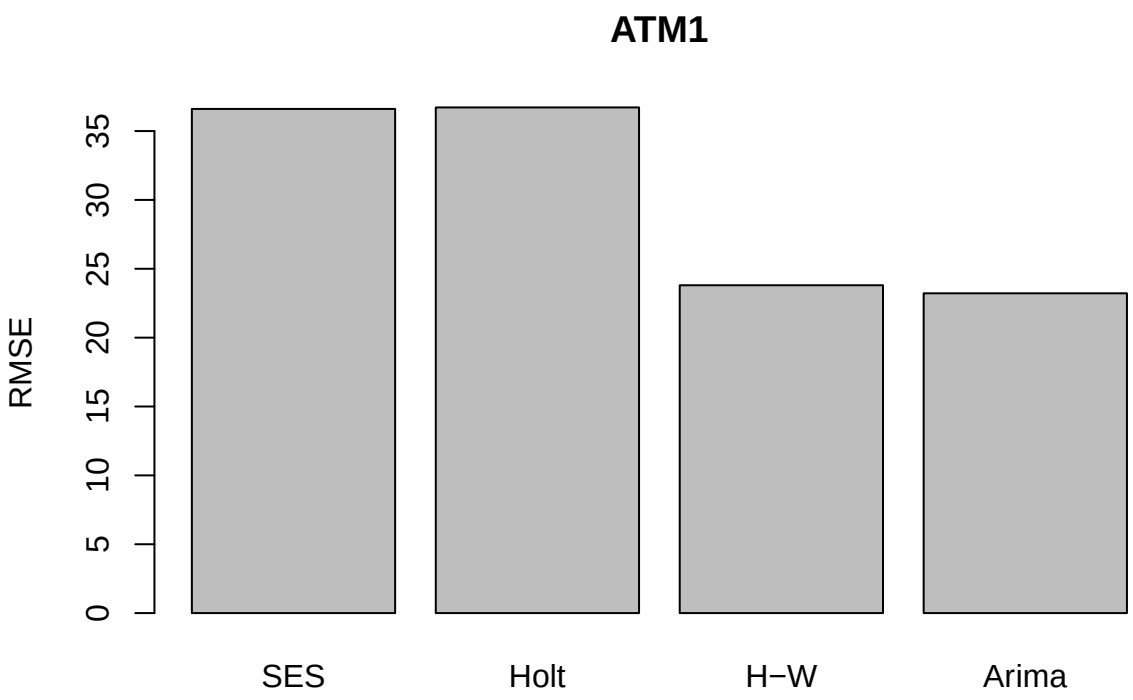


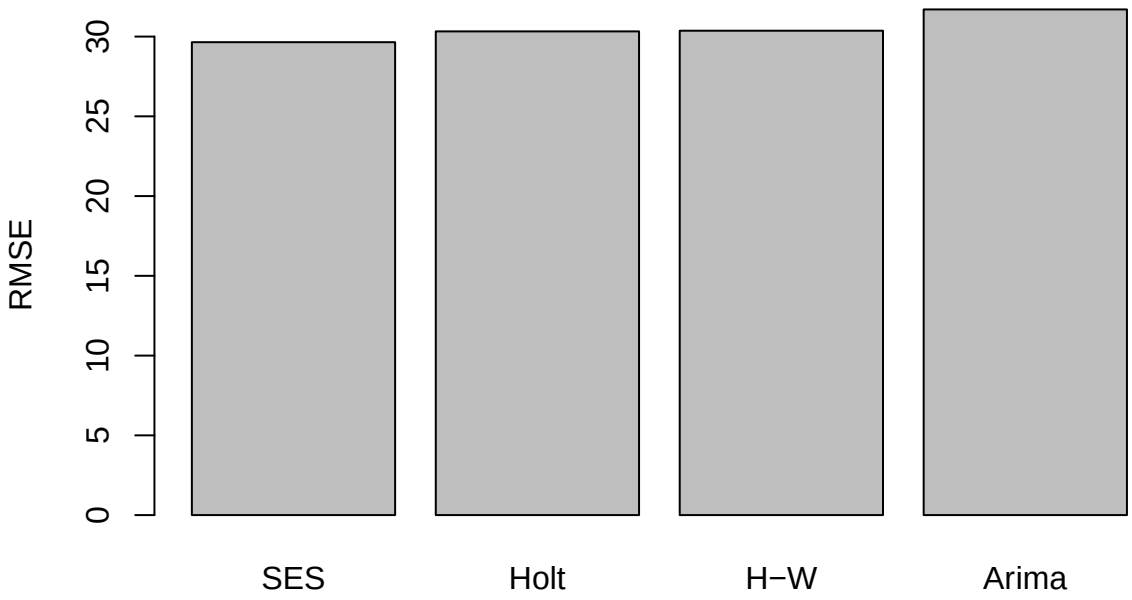


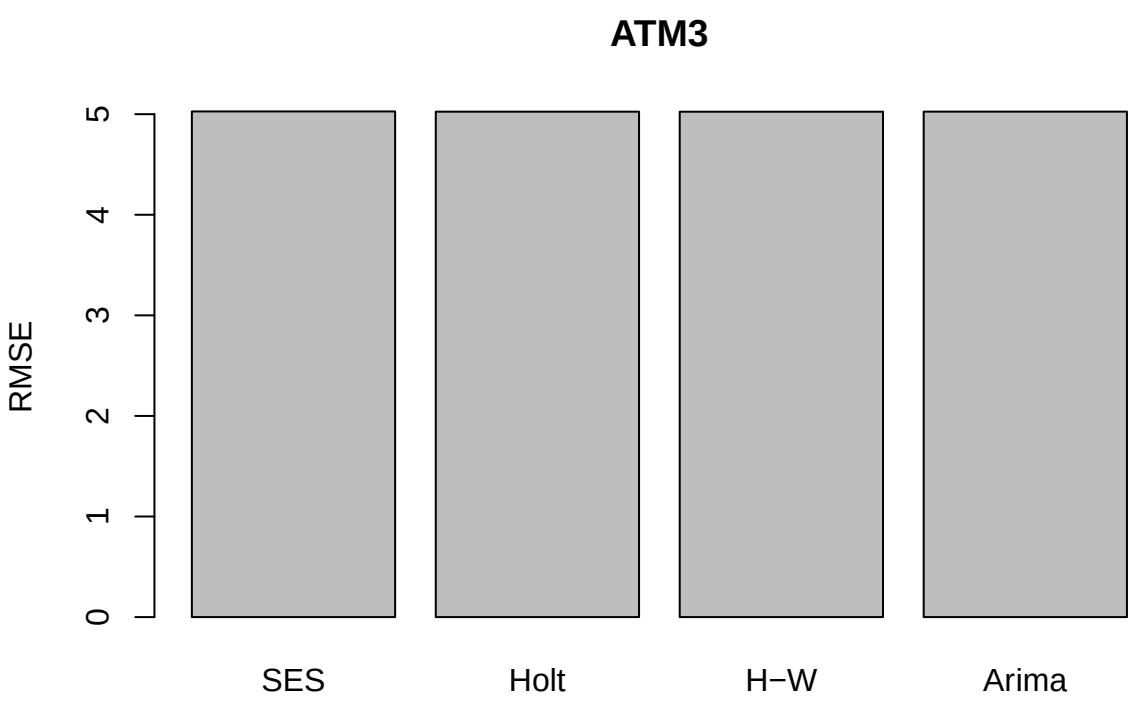
Finally, there is the arima model, which includes additive and multiplicative effects of data on itself. The `auto.arima()` function in R scans through several different models to find the one that minimizes the error between the fitted values and the observed ones.

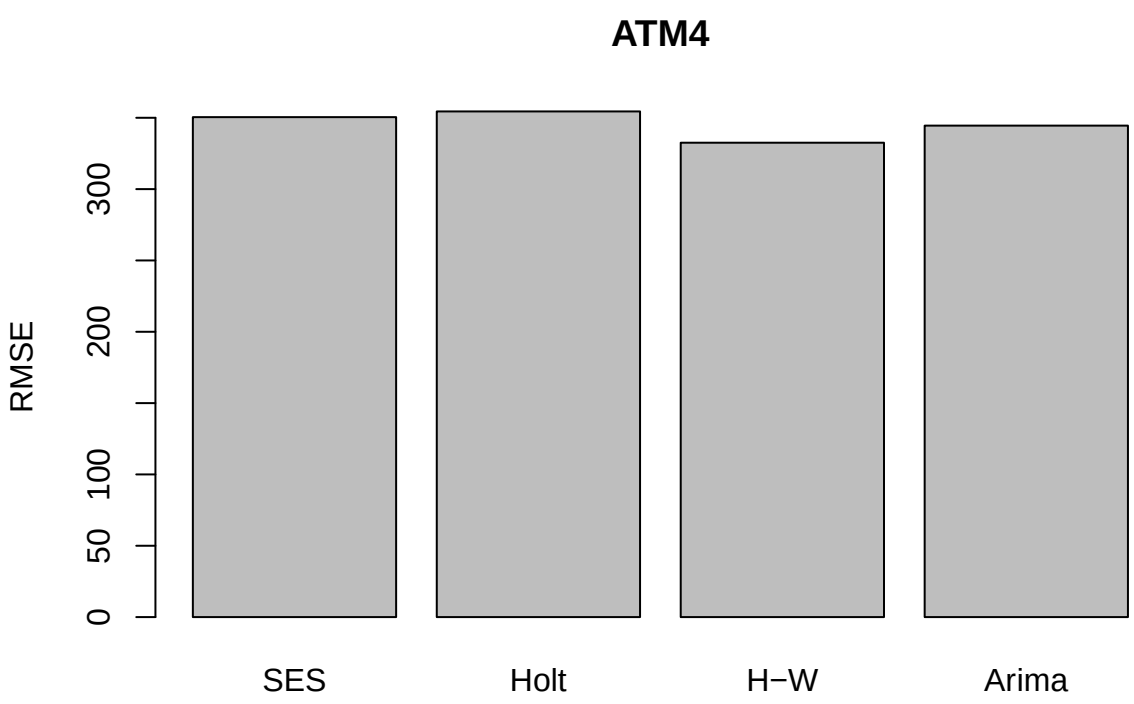


We can use the root mean square error to compare the results of our models. We square the residuals to amplify the effects of large errors and to remove any negative signs. Then we find the average and the square root to scale it. As we can see below, the Arima model resulted in the smallest RMSE for ATM 1 and a negligible increase for the other atms. However, the Holt-Winters model performed the best consistently.

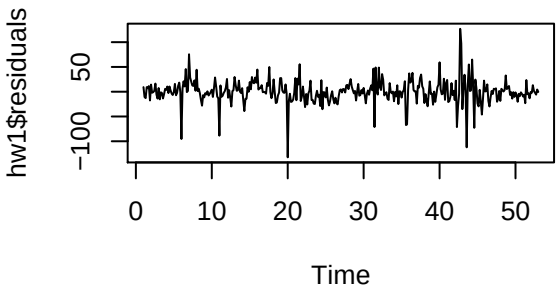

ATM1
RMSE
0
5
10
15
20
25
30
35
SES
Holt
H−W
Arima

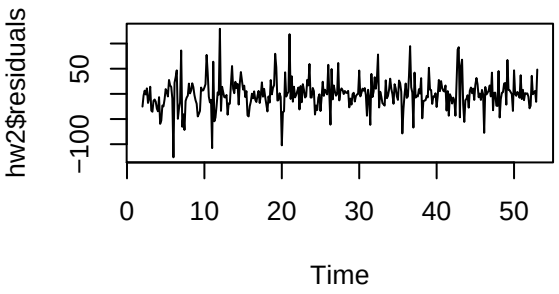

ATM2
RMSE
0
5
10
15
20
25
30
SES
Holt
H−W
Arima

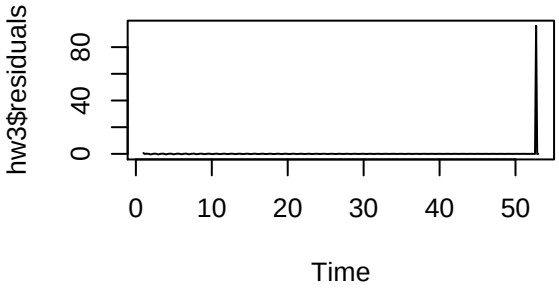

ATM3
RMSE
0
1
2
3
4
5
SES
Holt
H−W
Arima

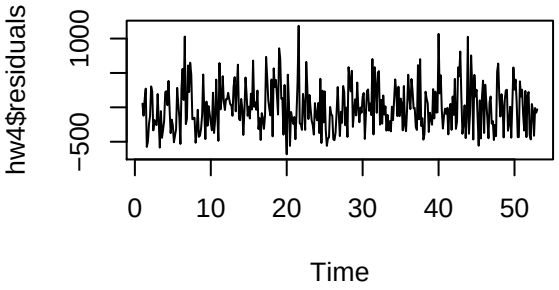


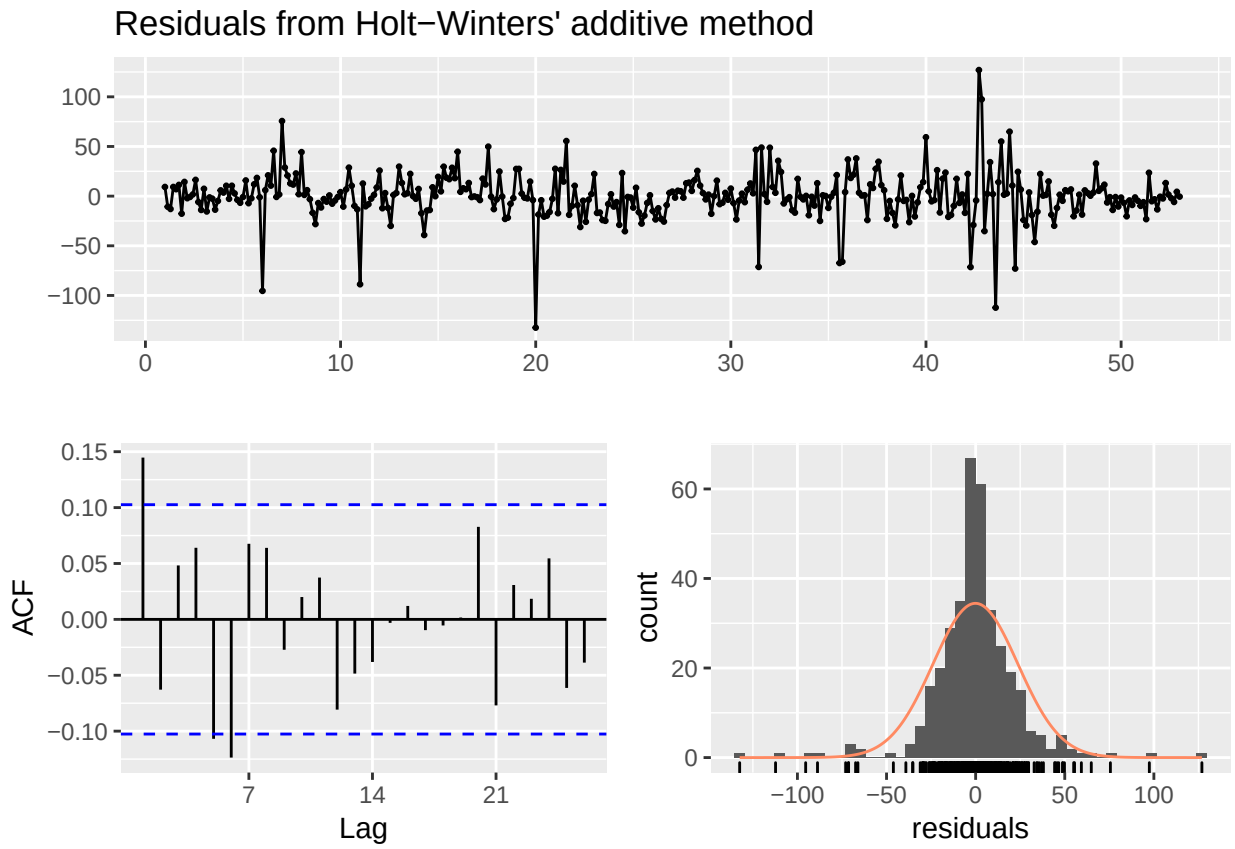
Finally, we check the residuals to ensure that there is a uniform variance across the model. We see that each ATMs one, two, and four were modelled adequately. For model3, the residuals are larger than expected because this model predicts seasonality that we don't see at this atm(yet). However, due to it success on the other atms, I suggest we keep it here. The larger than testable residuals are expected in this case.


hw1$residuals
−100
50
0
10
20
30
40
50
Time

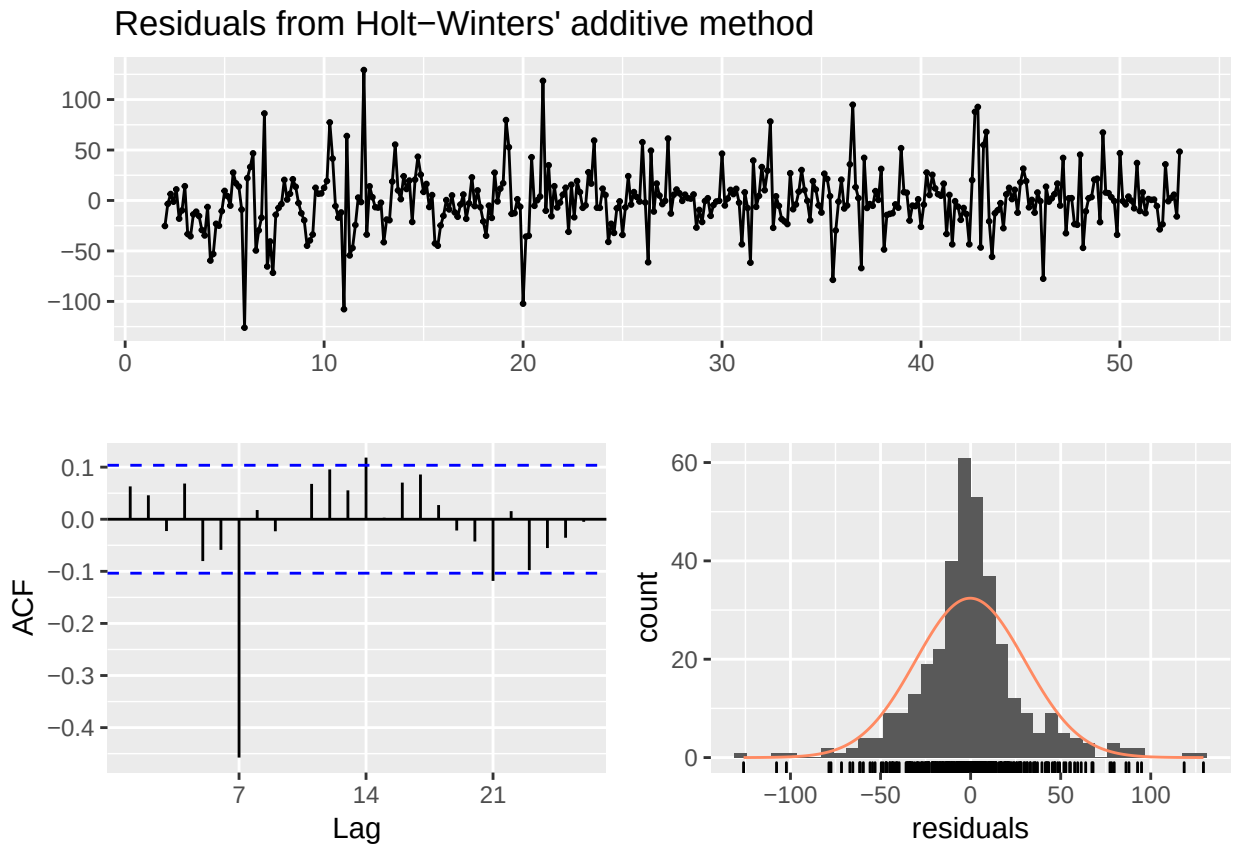

hw2$residuals
−100
50
0
10
20
30
40
50
Time


hw3$residuals
0
40
80
0
10
20
30
40
50
Time

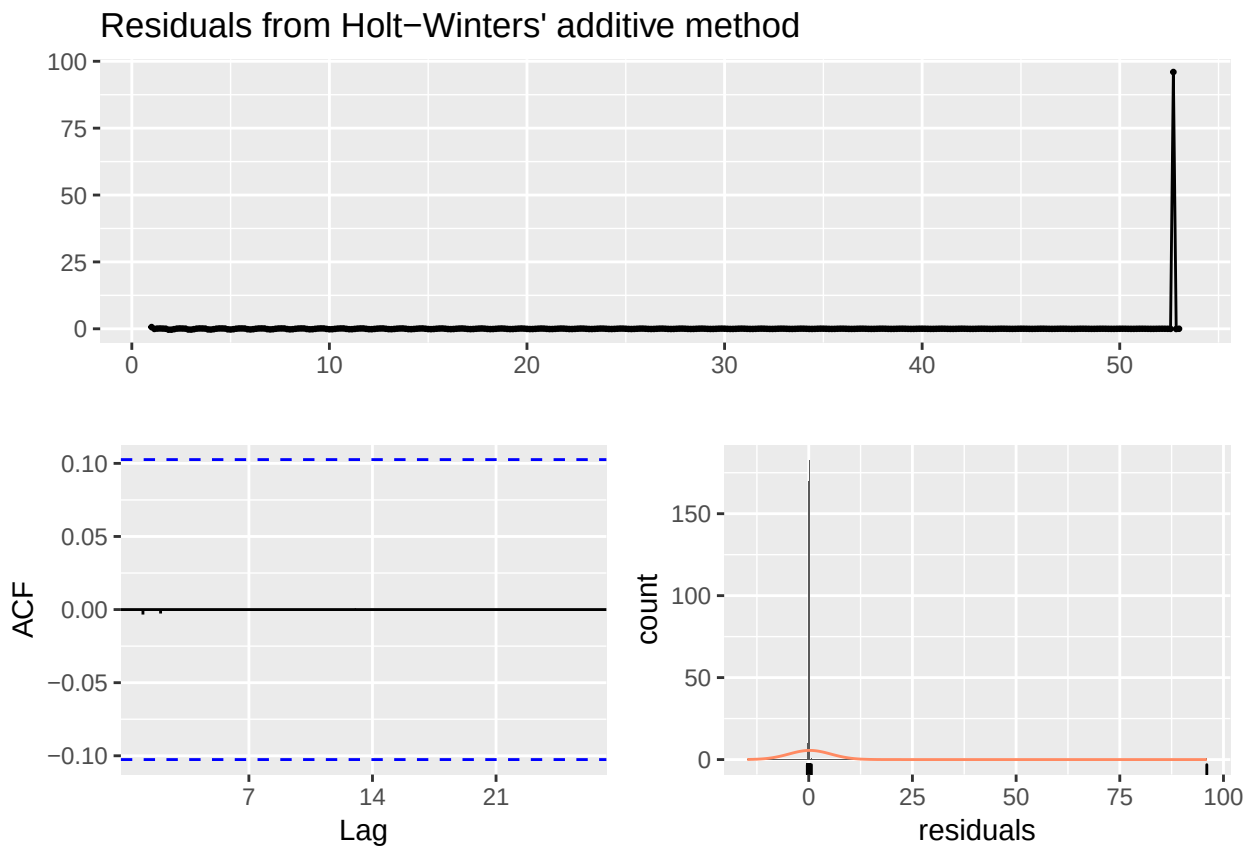

hw4$residuals
−500
1000
0
10
20
30
40
50
Time



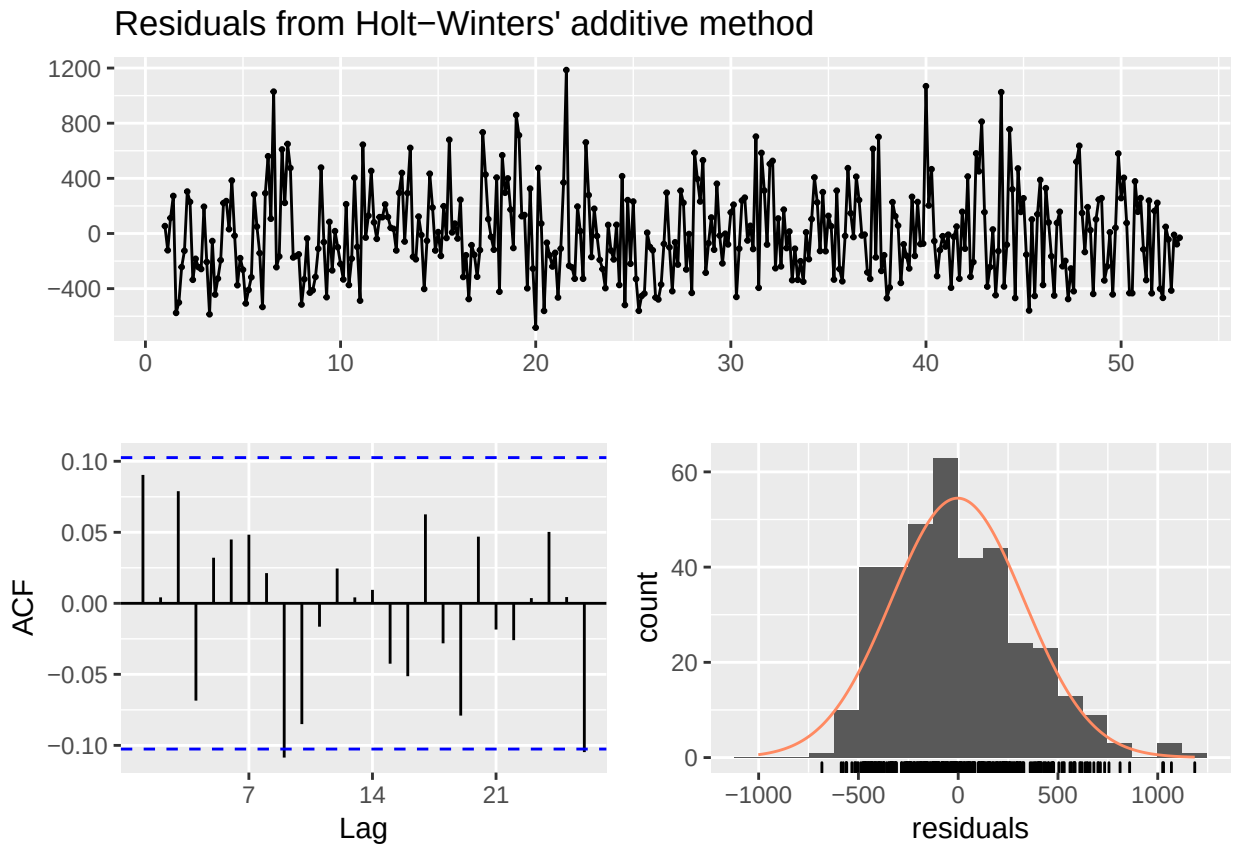

```
##
##  Ljung-Box test
##
## data:  Residuals from Holt-Winters' additive method
## Q* = 29.628, df = 3, p-value = 1.653e-06
##
## Model df: 11.   Total lags used: 14
```




```
##
##  Ljung-Box test
##
## data:  Residuals from Holt-Winters' additive method
## Q* = 96.381, df = 3, p-value < 2.2e-16
##
## Model df: 11.   Total lags used: 14
```




```
##
##  Ljung-Box test
##
## data:  Residuals from Holt-Winters' additive method
## Q* = 0.006946, df = 3, p-value = 0.9998
##
## Model df: 11.   Total lags used: 14
```

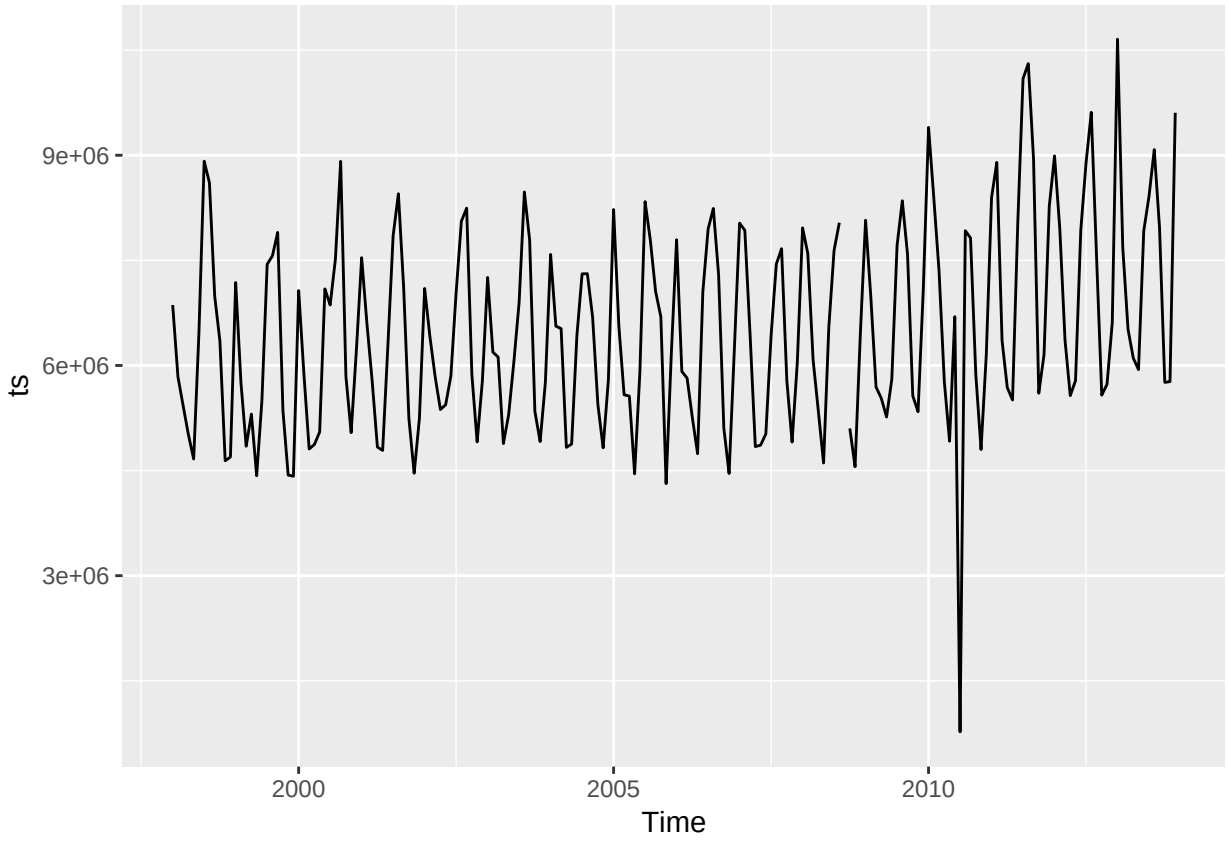



```
##
##  Ljung-Box test
##
## data:  Residuals from Holt-Winters' additive method
## Q* = 16.776, df = 3, p-value = 0.0007858
##
## Model df: 11.   Total lags used: 14
```

Even though our data passed the KPSS test, it is still not white noise. This is either due to the STL model not being the best fit, or the way in which I constructed the initial time series and windowed by week rather than month or quarter. This final model produces residuals with means near 0 and a forecasted variance that corresponds to the observed variance. This upper bounds of this model should give us a great idea of how much cash to have on hand in each atm. The forecasted values are saved in the `/predictions` folder. While I included atm3 for the sake of completion, please note that with such a small amount of data, the forecasts are fairly useless.

Part B

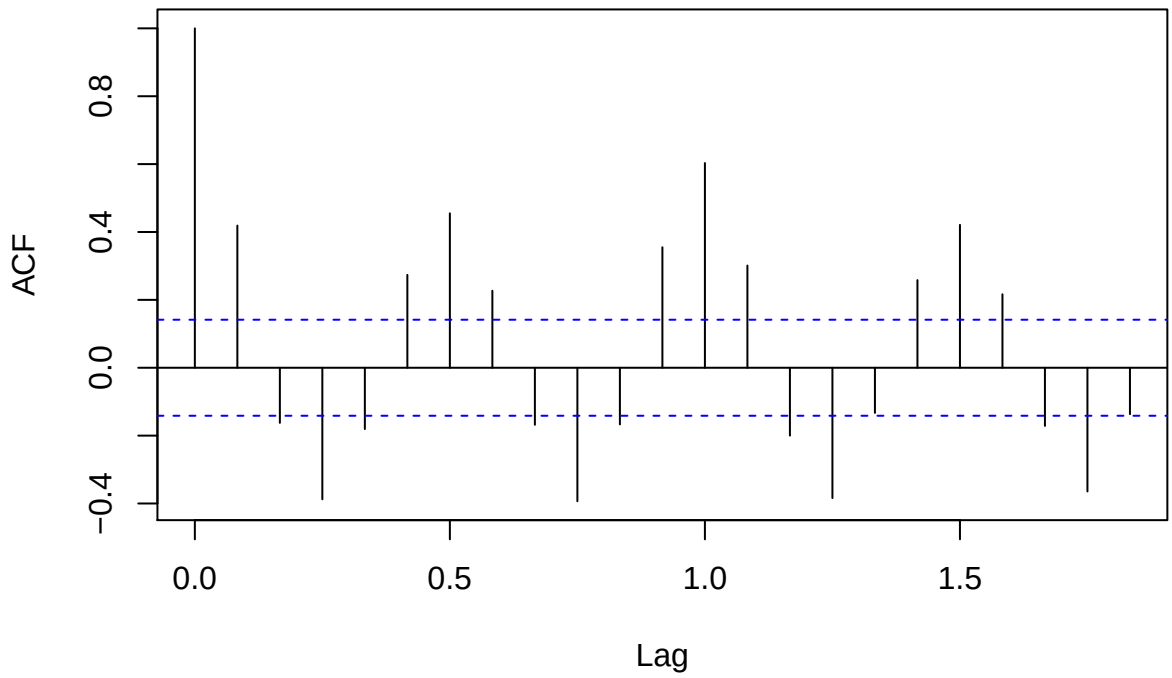
Part B consists of a simple dataset of residential power usage for January 1998 until December 2013. Your assignment is to model these data and a monthly forecast for 2014. The data is given in a single file. The variable 'KWH' is power consumption in Kilowatt hours, the rest is straight forward. Add this to your existing files above.



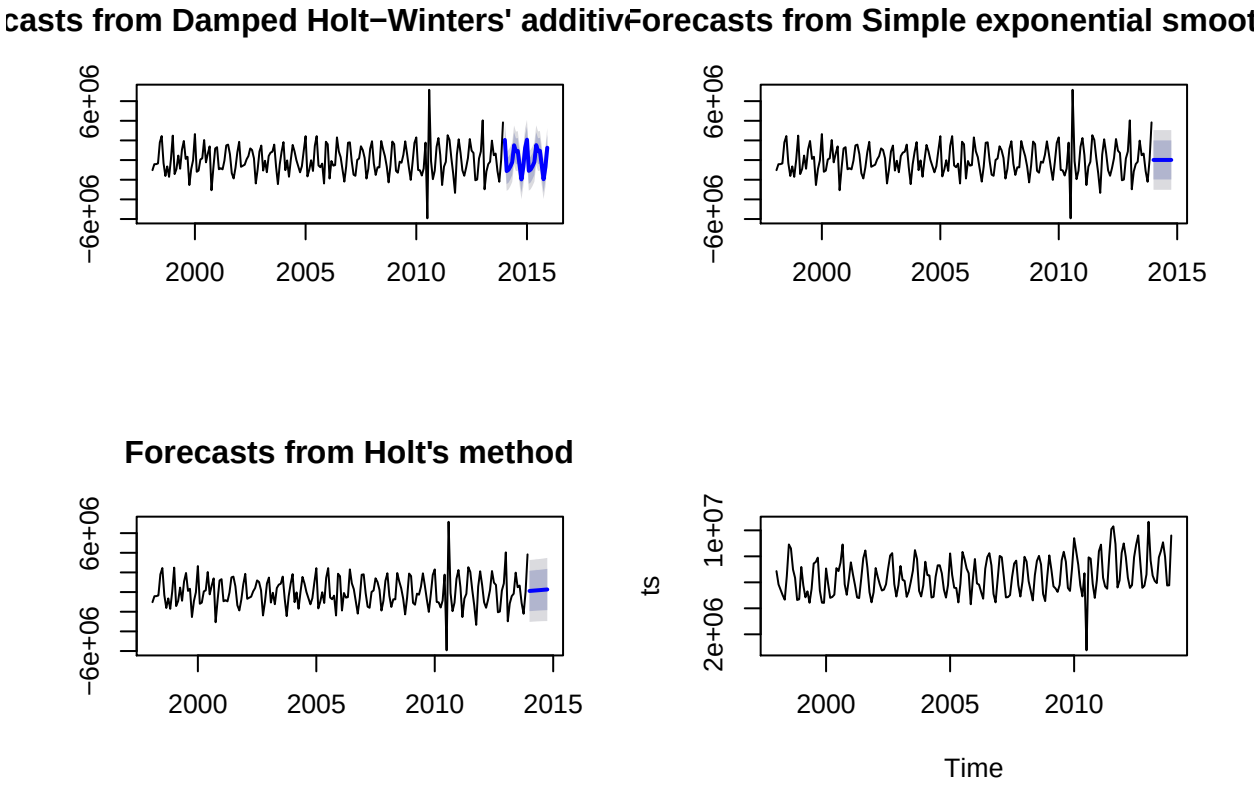
Then, we use STL interpolation to find the missing data.

Then, using the ndiff function, we find that we need to difference twice. Additionally, the ACF plot highlights the lag in question (at 1). After differencing, we run another kpss, which the data passes.

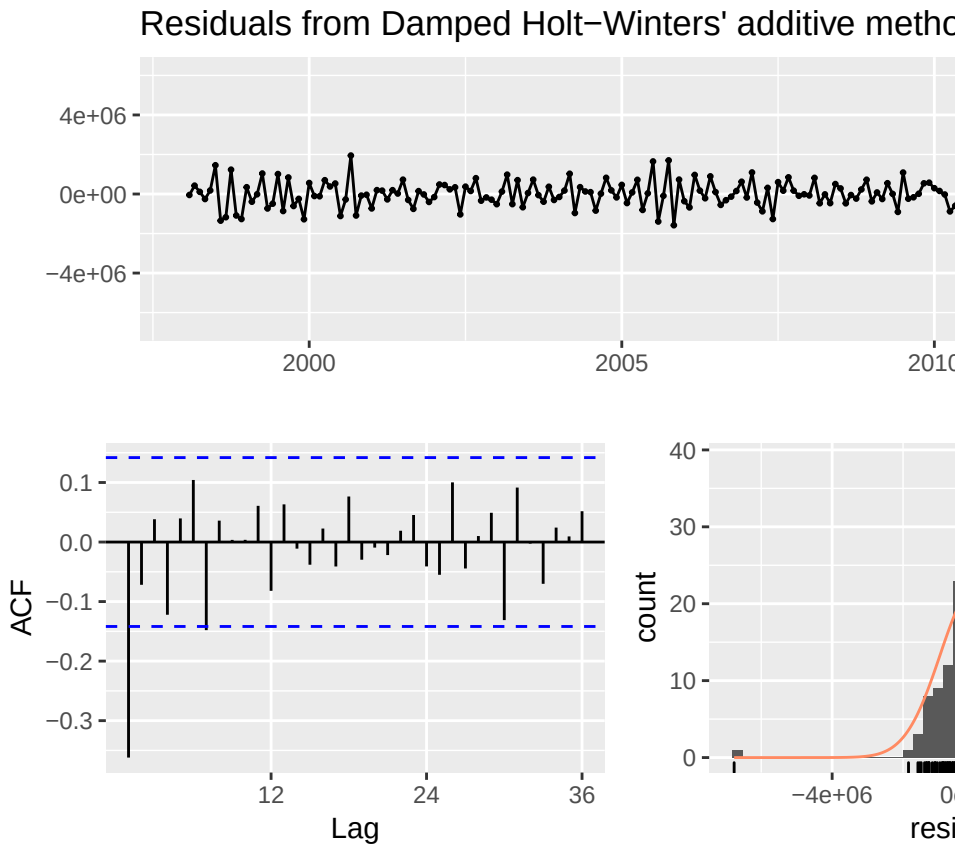
```
## [1] 1
```




```
## [1] 0
```

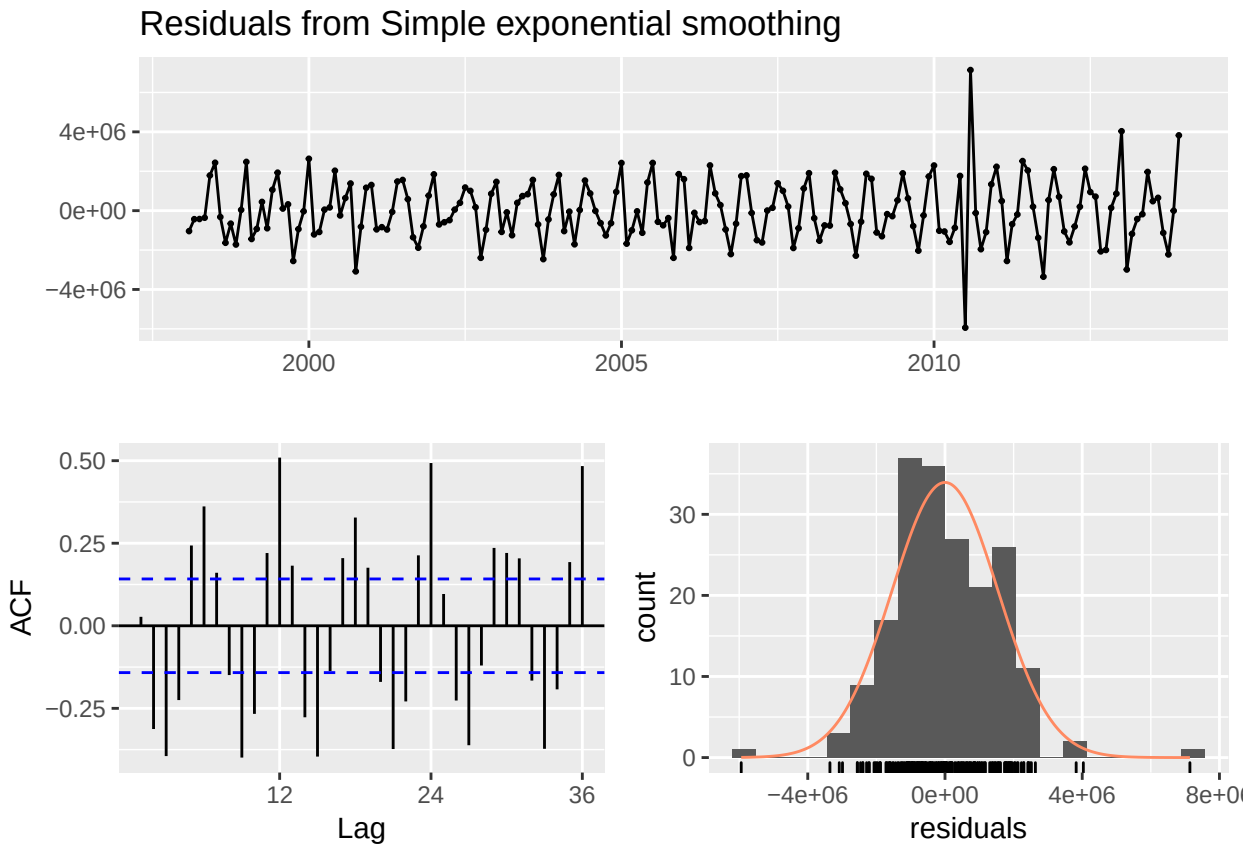


We can see below that Holt, Holt-Winters, and SES models all produce residuals that resemble white noise.

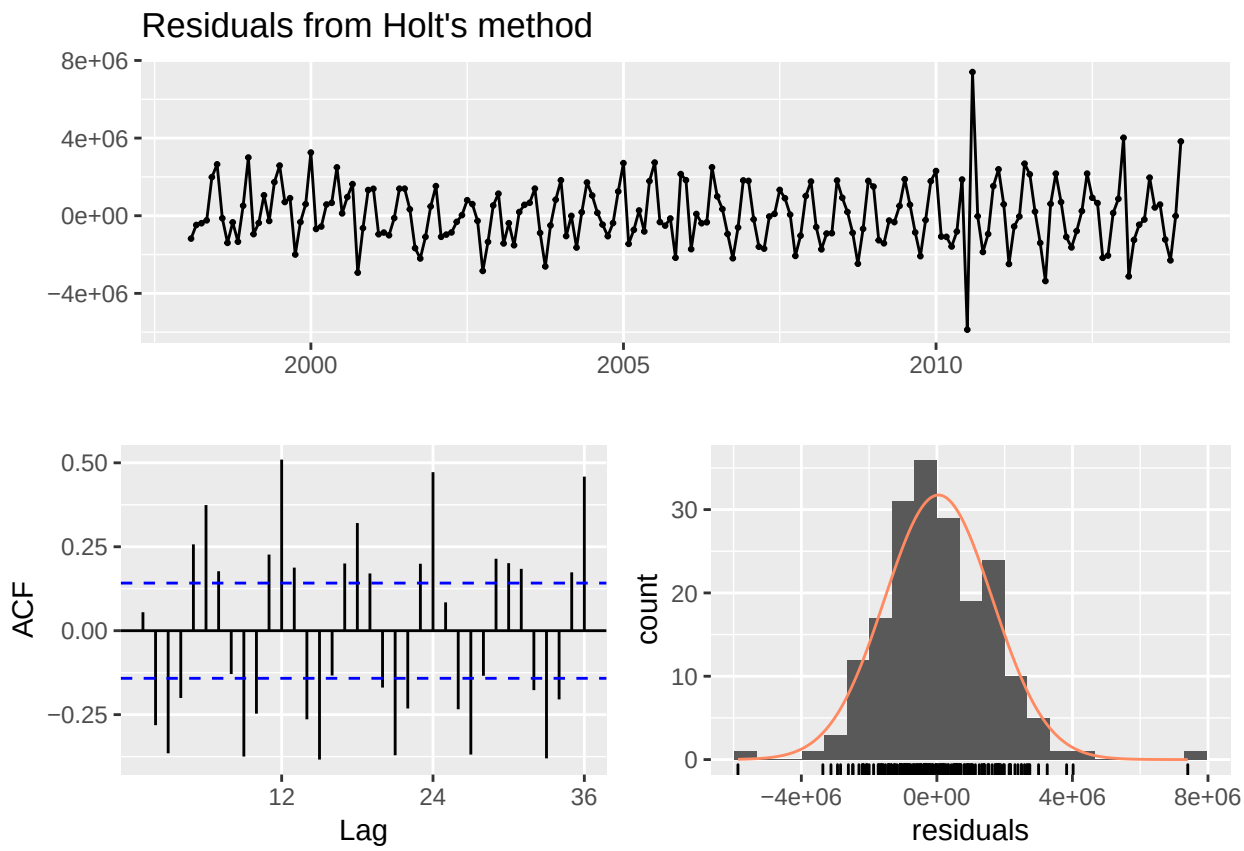


However, the Auto arima model does not.

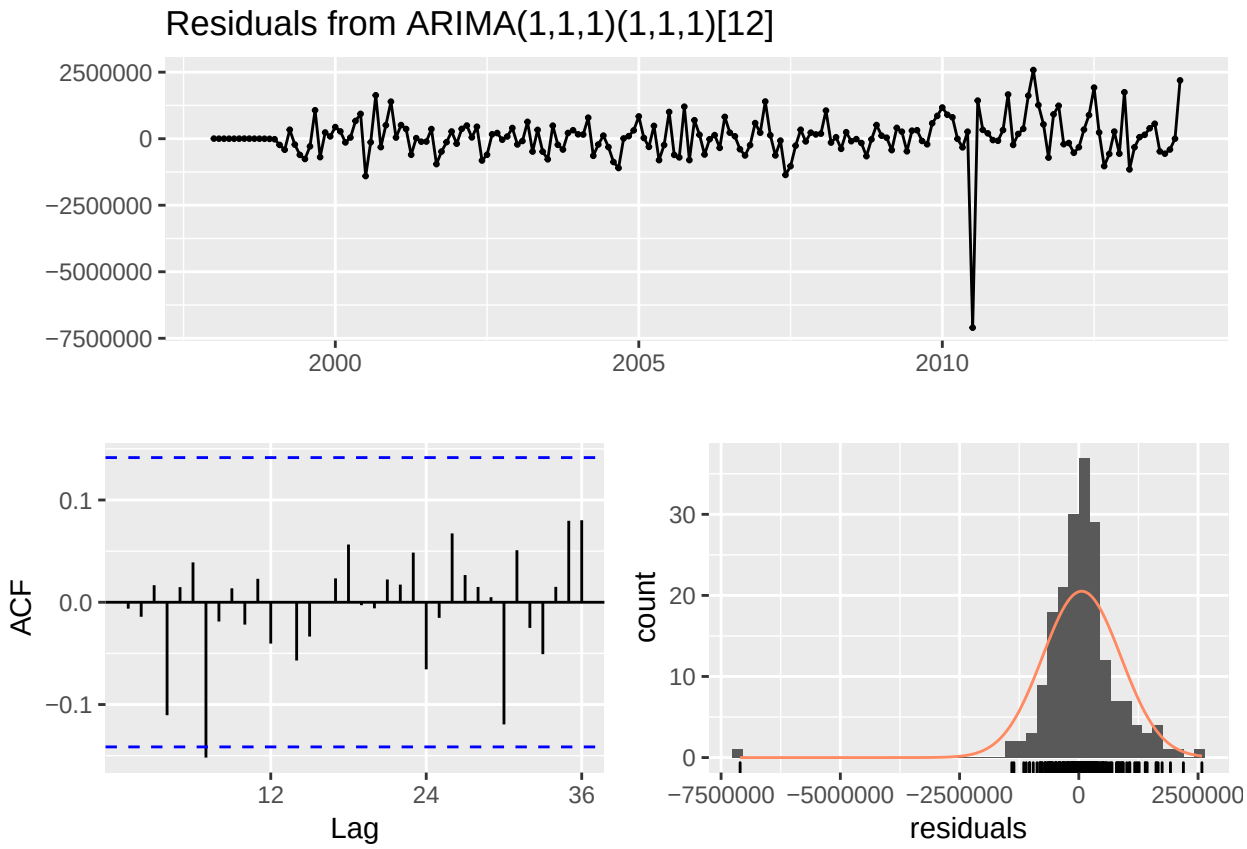
```
##
##  Ljung-Box test
##
## data:  Residuals from Damped Holt-Winters' additive method
## Q* = 43.003, df = 7, p-value = 3.331e-07
##
## Model df: 17.   Total lags used: 24
```




```
##
##  Ljung-Box test
##
## data:  Residuals from Simple exponential smoothing
## Q* = 426.78, df = 22, p-value < 2.2e-16
##
## Model df: 2.   Total lags used: 24
```




```
##
##  Ljung-Box test
##
## data:  Residuals from Holt's method
## Q* = 404.31, df = 20, p-value < 2.2e-16
##
## Model df: 4.   Total lags used: 24
```




```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(1,1,1)(1,1,1)[12]
## Q* = 11.534, df = 20, p-value = 0.9312
##
## Model df: 4.   Total lags used: 24
```

It is clearly not a good fit for the data. So, of the remaining models, I suggest the ses because it has the smallest RMSE.

```
##                     ME     RMSE      MAE       MPE     MAPE      MASE
## Training set 26507.29 980070.4 604331.4 -202.1775 479.0651 0.6969996
##                     ACF1
## Training set -0.3619677
```

```
##                     ME    RMSE     MAE      MPE     MAPE     MASE
## Training set 103.4049 1541560 1197139 95.88835 103.4718 1.380708
##                     ACF1
## Training set 0.02716413
```

```
##                     ME    RMSE     MAE      MPE     MAPE     MASE     ACF1
## Training set 47890.72 1589452 1227231 123.1649 186.4913 1.415414 0.054988
```

```
##                     ME     RMSE    MAE       MPE     MAPE    MASE
## Training set 60013.17 821444.8 495085 -4.049446 11.59297 0.71983
##                      ACF1
## Training set -0.006230607
```

Then, I saved the file as a .csv in the `/predictions` folder under the name partB.csv. We can see below that our expected values and observed values are within the same range and have uniform variance.