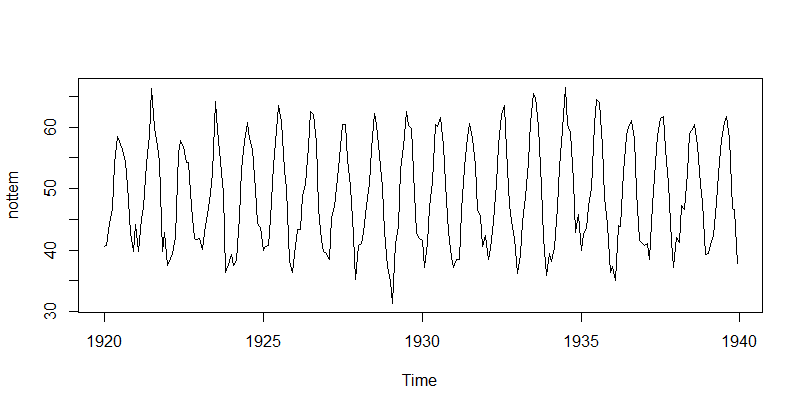
|  |  |
| --- | --- |
|  | Time Series Analysis |
|  |  |
|  | Rebecca Leu  Northeastern University |

Part 1 - Decompose seasonal time series data

For decomposing I used the nottem dataset. This dataset is the average monthly temperatures at Nottingham Castle from 1920 to 1939. They are in degrees Fahrenheit. The are 240 averages over the 20 year span.

This is a plot of the data which of course shows the seasonality of the temperatures. This confirms we have a seasonal time series.



To begin we are going to decompose nottem. Since the time series is seasonal, we first estimate the trend, seasonal and irregular components. The estimated values of the seasonal components is saved as nottemtimeseries$seasonal.

> nottemtimeseries <-decompose(nottem)

> nottemtimeseries$seasonal

Jan Feb Mar Apr May Jun Jul

1920 -9.3393640 -9.8998904 -6.9466009 -2.7573465 3.4533991 8.9865132 12.9672149

1921 -9.3393640 -9.8998904 -6.9466009 -2.7573465 3.4533991 8.9865132 12.9672149

1922 -9.3393640 -9.8998904 -6.9466009 -2.7573465 3.4533991 8.9865132 12.9672149

1923 -9.3393640 -9.8998904 -6.9466009 -2.7573465 3.4533991 8.9865132 12.9672149

1924 -9.3393640 -9.8998904 -6.9466009 -2.7573465 3.4533991 8.9865132 12.9672149

1925 -9.3393640 -9.8998904 -6.9466009 -2.7573465 3.4533991 8.9865132 12.9672149

1926 -9.3393640 -9.8998904 -6.9466009 -2.7573465 3.4533991 8.9865132 12.9672149

1927 -9.3393640 -9.8998904 -6.9466009 -2.7573465 3.4533991 8.9865132 12.9672149

1928 -9.3393640 -9.8998904 -6.9466009 -2.7573465 3.4533991 8.9865132 12.9672149

1929 -9.3393640 -9.8998904 -6.9466009 -2.7573465 3.4533991 8.9865132 12.9672149

1930 -9.3393640 -9.8998904 -6.9466009 -2.7573465 3.4533991 8.9865132 12.9672149

1931 -9.3393640 -9.8998904 -6.9466009 -2.7573465 3.4533991 8.9865132 12.9672149

1932 -9.3393640 -9.8998904 -6.9466009 -2.7573465 3.4533991 8.9865132 12.9672149

1933 -9.3393640 -9.8998904 -6.9466009 -2.7573465 3.4533991 8.9865132 12.9672149

1934 -9.3393640 -9.8998904 -6.9466009 -2.7573465 3.4533991 8.9865132 12.9672149

1935 -9.3393640 -9.8998904 -6.9466009 -2.7573465 3.4533991 8.9865132 12.9672149

1936 -9.3393640 -9.8998904 -6.9466009 -2.7573465 3.4533991 8.9865132 12.9672149

1937 -9.3393640 -9.8998904 -6.9466009 -2.7573465 3.4533991 8.9865132 12.9672149

1938 -9.3393640 -9.8998904 -6.9466009 -2.7573465 3.4533991 8.9865132 12.9672149

1939 -9.3393640 -9.8998904 -6.9466009 -2.7573465 3.4533991 8.9865132 12.9672149

Aug Sep Oct Nov Dec

1920 11.4591009 7.4001096 0.6547149 -6.6176535 -9.3601974

1921 11.4591009 7.4001096 0.6547149 -6.6176535 -9.3601974

1922 11.4591009 7.4001096 0.6547149 -6.6176535 -9.3601974

1923 11.4591009 7.4001096 0.6547149 -6.6176535 -9.3601974

1924 11.4591009 7.4001096 0.6547149 -6.6176535 -9.3601974

1925 11.4591009 7.4001096 0.6547149 -6.6176535 -9.3601974

1926 11.4591009 7.4001096 0.6547149 -6.6176535 -9.3601974

1927 11.4591009 7.4001096 0.6547149 -6.6176535 -9.3601974

1928 11.4591009 7.4001096 0.6547149 -6.6176535 -9.3601974

1929 11.4591009 7.4001096 0.6547149 -6.6176535 -9.3601974

1930 11.4591009 7.4001096 0.6547149 -6.6176535 -9.3601974

1931 11.4591009 7.4001096 0.6547149 -6.6176535 -9.3601974

1932 11.4591009 7.4001096 0.6547149 -6.6176535 -9.3601974

1933 11.4591009 7.4001096 0.6547149 -6.6176535 -9.3601974

1934 11.4591009 7.4001096 0.6547149 -6.6176535 -9.3601974

1935 11.4591009 7.4001096 0.6547149 -6.6176535 -9.3601974

1936 11.4591009 7.4001096 0.6547149 -6.6176535 -9.3601974

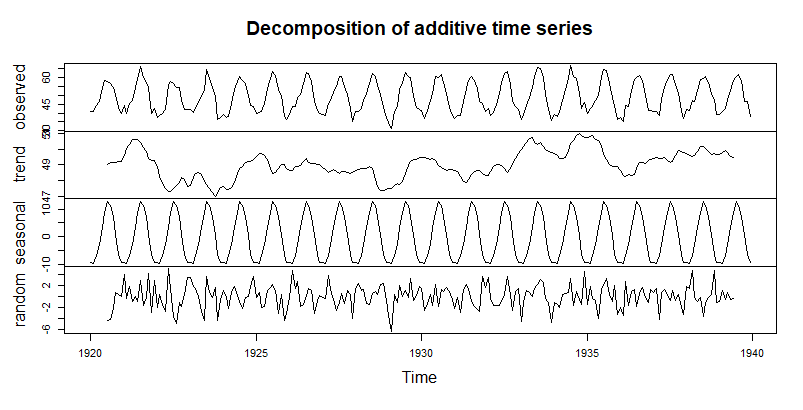
1937 11.4591009 7.4001096 0.6547149 -6.6176535 -9.3601974

1938 11.4591009 7.4001096 0.6547149 -6.6176535 -9.3601974

1939 11.4591009 7.4001096 0.6547149 -6.6176535 -9.3601974

Once we have gotten the estimated values, we plot the estimated trend, seasonal, and irregular components of the time series.

> plot(nottemtimeseries)

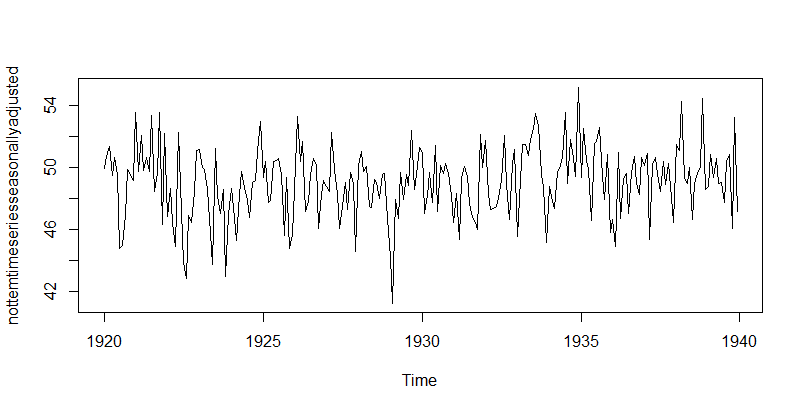


From the plot we can see the trend data does show that some years were hotter and colder on average. It appears that around 1922 appears to be quite hot overall along with the years 1934 and 1935. We can also see from this data that the seasonal decomposition is very smooth which would make sense this data set is the definition of seasonal.

We finish by removing the seasonal variation. The below graph just contains the trend component and an irregular component.

> nottemtimeseriesseasonallyadjusted <- nottem - nottemtimeseries$seasonal

> plot(nottemtimeseriesseasonallyadjusted)



Part 2 –Time series forecasting

For the second part we are looking at the Johnson and Johnson company quarterly earnings per share from 1960 to 1980. From a simple graph of the earnings we see a positive increase in earnings as time goes on. There are however some ebbs and flows which could be seasonal.

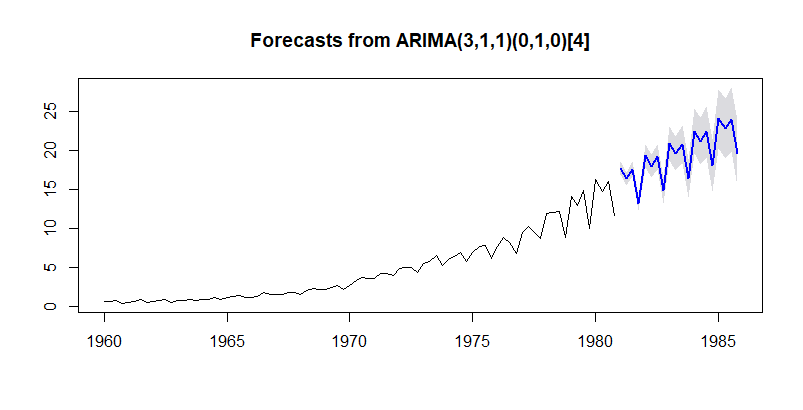
After we install the needed packages, we begin by asking R to calculate our ARIMA values for us. The easiest way to do this is to use the auto arima function, which automatically gives us the best values of P, D and Q. The values it gives us are 3,1,1.

|  |
| --- |
| > plot(JohnsonJohnson)    > install.packages("tseries")  > install.packages("forecast")  package ‘forecast’ successfully unpacked and MD5 sums checked  The downloaded binary packages are in  C:\Users\rebec\AppData\Local\Temp\RtmpQdVy8w\downloaded\_packages  > install.packages("ggplot2")  package ‘ggplot2’ successfully unpacked and MD5 sums checked  The downloaded binary packages are in  C:\Users\rebec\AppData\Local\Temp\RtmpQdVy8w\downloaded\_packages  > install.packages("tseries")  package ‘tseries’ successfully unpacked and MD5 sums checked  The downloaded binary packages are in  C:\Users\rebec\AppData\Local\Temp\RtmpQdVy8w\downloaded\_packages  > library(tseries)  > library(forecast)  > library(ggplot2)  > JohnsonJohnsonarima <- auto.arima(JohnsonJohnson)  > JohnsonJohnsonarima  Series: JohnsonJohnson  ARIMA(3,1,1)(0,1,0)[4]  Coefficients:  ar1 ar2 ar3 ma1  -0.1712 0.1387 -0.208 -0.6636  s.e. 0.1769 0.1701 0.121 0.1542  sigma^2 estimated as 0.1808: log likelihood=-43.01  AIC=96.02 AICc=96.84 BIC=107.86  > auto.arima(JohnsonJohnson, ic="aic", trace = TRUE)  ARIMA(2,1,2)(1,1,1)[4] : Inf  ARIMA(0,1,0)(0,1,0)[4] : 145.5812  ARIMA(1,1,0)(1,1,0)[4] : 102.1611  ARIMA(0,1,1)(0,1,1)[4] : 100.7145  ARIMA(0,1,1)(0,1,0)[4] : 100.126  ARIMA(0,1,1)(1,1,0)[4] : 100.7138  ARIMA(0,1,1)(1,1,1)[4] : Inf  ARIMA(1,1,1)(0,1,0)[4] : 98.20777  ARIMA(1,1,1)(1,1,0)[4] : 100.1669  ARIMA(1,1,1)(0,1,1)[4] : 100.1706  ARIMA(1,1,1)(1,1,1)[4] : 102.1609  ARIMA(1,1,0)(0,1,0)[4] : 100.2065  ARIMA(2,1,1)(0,1,0)[4] : 96.63415  ARIMA(2,1,1)(1,1,0)[4] : 98.58372  ARIMA(2,1,1)(0,1,1)[4] : 98.58468  ARIMA(2,1,1)(1,1,1)[4] : Inf  ARIMA(2,1,0)(0,1,0)[4] : 101.5208  ARIMA(3,1,1)(0,1,0)[4] : 96.01524  ARIMA(3,1,1)(1,1,0)[4] : 98.00528  ARIMA(3,1,1)(0,1,1)[4] : 98.00368  ARIMA(3,1,1)(1,1,1)[4] : Inf  ARIMA(3,1,0)(0,1,0)[4] : 98.94887  ARIMA(3,1,2)(0,1,0)[4] : Inf  ARIMA(2,1,2)(0,1,0)[4] : 97.03223  Best model: ARIMA(3,1,1)(0,1,0)[4]  Series: JohnsonJohnson  ARIMA(3,1,1)(0,1,0)[4]  Coefficients:  ar1 ar2 ar3 ma1  -0.1712 0.1387 -0.208 -0.6636  s.e. 0.1769 0.1701 0.121 0.1542  sigma^2 estimated as 0.1808: log likelihood=-43.01  AIC=96.02 AICc=96.84 BIC=107.86 |
|  |
| |  | | --- | |  | |

After we install the needed packages, we begin by asking R to calculate our ARIMA values for us. The easiest way to do this is to use the auto arima function, which automatically gives us the best values of P, D and Q. The values it gives us are 3,1,1. We can also run the auto arima function with trace to have the program show what it is computing to find those numbers. When we do this we can see that the 3,1,1 line (highlighted) gives us the lowest value. It gave us the same model as without trace, but with the trace option turned on we can see all the values it is going through. Once we have our model, we can forecast.

JohnsonJohnsonforecast <-forecast(JohnsonJohnsonarima, level=c(95),h=5\*4)

> plot(JohnsonJohnsonforecast)



The forecast I chose was for the next 5 years after that data ends in 1980. The Arima has found the patterns in the original data and has used that information to predict or forecast for the specified amount of time. The forecast plot here shows the forecasted line is in blue with the grey around it, that is the confidence interval. I specified a 95% confidence for this plot.