Pictorial Week 11: Time Series Week 2

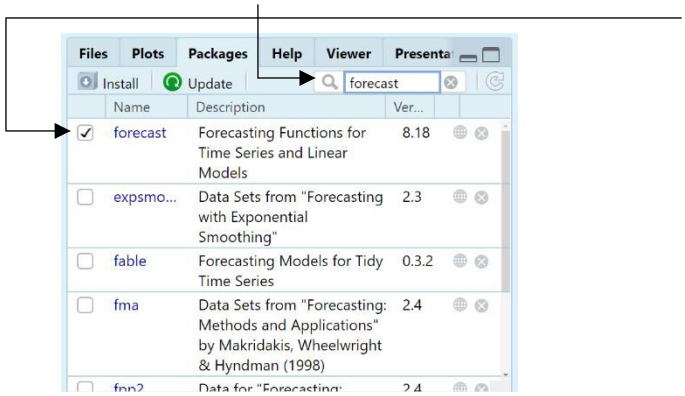
How to: Load Packages using the library() function

```
1 library(forecast)  # load the forecast package
2 library(fpp3)      # load the fpp3 package
3 library(ggplot2)   # load the ggplot2 package
```

OR

In the bottom right panel

Packages > Search for the package you want > Check the box of the package you want



How to: Assign a Time Series dataset coming from a package to an object

The woolyrnq dataset is used to demonstrate.

```
5 wool_TS <- forecast::woolyrnq   # Name_Of_Package::Name_Of_Dataset
```

How to: Apply the window() function to select the data we want

```
wool_TS_win <- window(wool_ts, start = c(1976,1), end = c(1983,4)) #This is quarterly data. We want to consider the data
#from the start of 1976 to the end of 1983
```

How to: Print a Time Series dataset to the console

```
9 wool_TS_win  #  View your Time Series data
```

How to: Apply the autoplot() function

```
11 autoplot(wool_TS_win) # Plot your data (it will appear on the bottom right panel)
12
13 autoplot(wool_TS_win) + ggtitle('Quarterly production of woollen yarn in Australia, 1970 - 1975') + ylab("Kilos")  # This
   option allows you to put your own title and also label the y-axis
```

TRANSFORMATIONS

How to: Apply the BoxCox.lamba() function if your time series needs a variance-stabilising transformation

```
15 lam <- BoxCox.lambda(wool_TS_win)  # Store your lambda value in an object called lam
16 lam                               # View the value of lam
```

How to: Apply the BoxCox () function to stabilise the variance

```
18 wool_TS_win_sta <- BoxCox(wool_TS_win,lambda = lam)  # Store your variance-stabalised Time Series in an object called
   wool_TS_win_sta
```

How to: Apply the nsdiffs() function to check if your data needs to be differenced at a seasonal lag

```
nsdiffs(wool_TS_win_sta)
#note that this would only be necessary if your time series displayed seasonality
```

What does the value you get from applying this function mean?

How to: Apply the ndiffs() function to check that if your data needs to be differenced at lag 1 to remove a trend

```
ndiffs(wool_TS_win_sta)
#note that this would only be necessary if your time series had a trend (without seasonality)
```

How to: Apply the diff() function

```
wool_TS_win_diffed <- diff(wool_TS_win_sta, lag = 4)
```

Why do we use a lag value of 4 for this time series?

How to: Apply the ndiffs() function to check that your differenced data does not need any more transformation via differencing

```
22 ndiffs(wool_TS_win_diffed)
```

What value would you want to see here?

How to: Apply the Acf() function

```
24 Acf(wool_TS_win_diffed)  # alternatively use ggAcf()
```

Why do we plot the ACF? What is its purpose?

How to use the Box.test() function to run a Ljung-Box test

```
 28 Box.test(wool_TS_win_diffed, type = "Ljung-Box", lag = 4)
```

Why would it make sense to test up to lag 4 for the transformed series?

FORECASTING

How to: Apply the window() function to create a training dataset and a test dataset

```
#we select the first 6 years worth of data as the training set
wool_TS_train <- window(wool_ts, start = c(1976,1), end = c(1981,4))
#we select the last two years of data as our test set
wool_TS_test <- window(wool_ts, start = c(1982,1), end = c(1983,4))
```

How to: Fit a simple forecasting model for this time series (fitted to the training data) with an appropriate forecast horizon 'h'.

```
mod_simp <- snaive(wool_TS_train,h = 8)
#note that we set h = 8 as there are 8 observations in the test set
```

How to: Plot the original time series and the forecasts from the fitted model together

```
autoplot(wool_TS_win) + autolayer(mod_simp)
```

What do the shaded blue regions represent?

How to: Plot the original time series and the forecasts from the fitted model together without prediction intervals

```
autoplot(wool_TS_win) + autolayer(mod_simp, PI = F)
```

Is this an appropriate model to forecast values for this time series? Why/why not?

Take time to familiarize yourself with the functions to fit the other simple forecasting models (in the lecture slides and the cheatsheet). You need to be able to fit these to time series for which they are appropriate.

How to: Apply the checkresiduals() function to the fitted model

```
checkresiduals(mod_simp)
```

Think about the output you get here - are there any concerns? Remember to consider each element of the output

How to: Assess the forecast accuracy of fitted model against the test set data

```
accuracy(mod_simp,wool_TS_test)
```

Which row and which columns do you look at?

If you get an error when trying to use this function, try using:

```
forecast::accuracy(mod_simp, wool_TS_test)
```