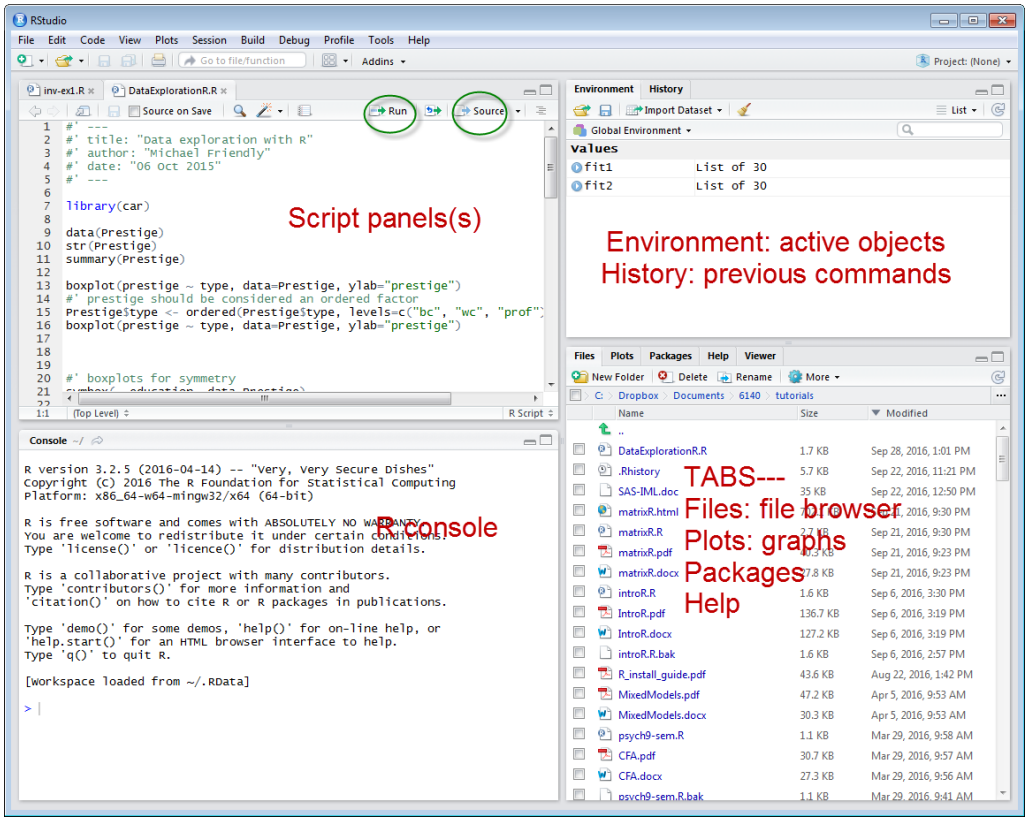
Data exploration & graphics with R Studio

In this exercise, we will work with data on Canadian occupational prestige from 1971, in relation to education, income and % women for 102 occupational categories. The data is in the `car` package, as `Prestige`. We will also learn to use R Studio somewhat more. You should be familiar with the layout of panels and what they do.

For this data set, we would be interested in predicting prestige from the other variables. The focus here is on using graphics to explore variable distributions and relations among them.

Start R Studio from the desktop icon or the Start menu.



In this exercise, you should create a new R script to save your work (File -> New file -> R script, or Ctrl+shift+N). Change the working directory to your home directory (X:) and save there.

1. Load the car package, and get a quick summary of the variables in the Prestige data set.

```
library(car)
data(Prestige)
str(Prestige)
summary(Prestige)
```

2. Boxplots are useful for showing the distributions of variables, often stratified by a factor. Note the difference between the two plots below.

```
boxplot(prestige ~ type, data=Prestige, ylab="prestige")
#' prestige should be considered an ordered factor
Prestige$type <- ordered(Prestige$type, levels=c("bc", "wc", "prof"))
boxplot(prestige ~ type, data=Prestige, ylab="prestige")
```

3. Other uses of boxplots include comparing the shapes of a distribution with various transformations, usually to make them more symmetric.

```
#' boxplots for symmetry
symbox(~ education, data=Prestige)
symbox(~ income, data=Prestige)
```

4. Examine the bivariate relations among the variables with a scatterplot matrix. The basic plot in R would be `plot(Prestige)`, but `car::scatterplotMatrix()` has many more options.

```
scatterplotMatrix(~prestige + education + income + women, data=Prestige)
scatterplotMatrix(~prestige + education + income + women | type,
data=Prestige)
```

5. You can create new variables or transform/recode existing ones in many ways. The simplest way is just to assign new variable names in the data frame.

```
Prestige$educ2 <- Prestige$education^2
Prestige$loginc <- log10(Prestige$income)
```

6. Let's fit a model predicting prestige from the original income and education (educ). The R function is lm(). When you have fit a model, there are lots of functions to explore it

```
mod1 <- lm(prestige ~ education + income, data=Prestige)
summary(mod1)
plot(mod1)
```

   Try fitting another regression model, this time using `loginc` and `educ2` as predictors.

7. The car package has a bunch of other functions that produces nicer versions of these plots.

```
qqPlot(mod1, id.n=2)
influencePlot(mod1, id.n=2)
```

The script for this exercise is on the N: drive, N:\psy6140\tutorials\DataExploration.R. You can also find it on the web, http://friendly.apps01.yorku.ca/psy6140/tutorials/DataExplorationR.R. It also includes sections on 3D plots and multivariate outliers.