

R Notebook

Code ▼

This is an R Markdown (<http://rmarkdown.rstudio.com>) Notebook. When you execute code within the notebook, the results appear beneath the code.

Try executing this chunk by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Cmd+Shift+Enter*.

Hide

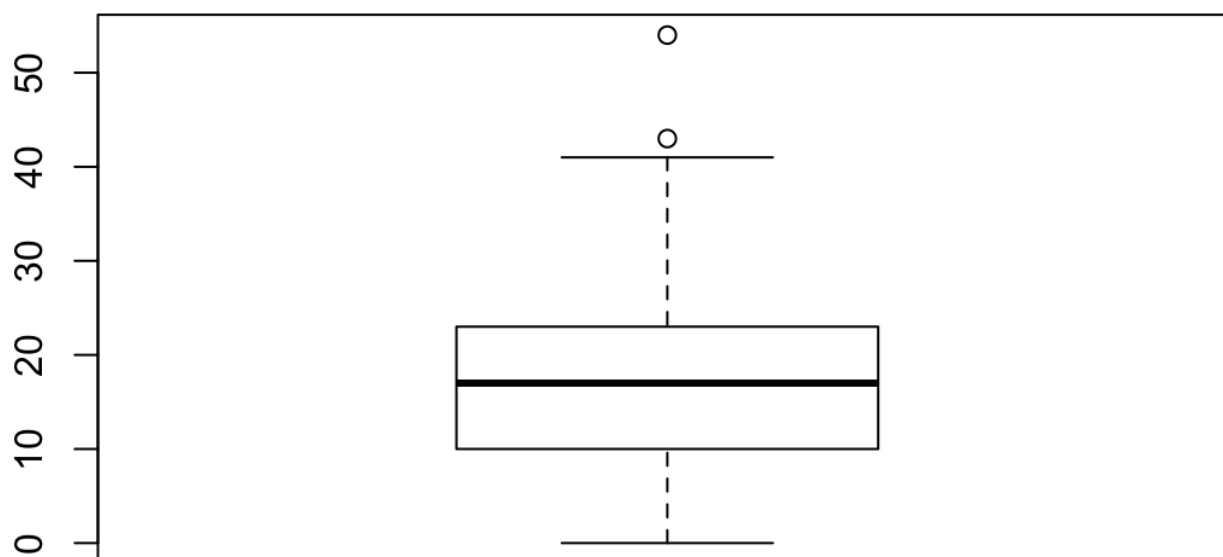
```
plot(cars)
```

Add a new chunk by clicking the *Insert Chunk* button on the toolbar or by pressing *Cmd+Option+I*.

When you save the notebook, an HTML file containing the code and output will be saved alongside it (click the *Preview* button or press *Cmd+Shift+K* to preview the HTML file).

Advanced Extension for those who want to go further

Here are some advanced things they could try for self study. Try applying some of this code to your dataset. Below is a Boxplot for the *beck* variable. Try to do a boxplot for the *age* variable. Boxplots are one of the most economical and simple visualizations we have. They show you the mean, median, IQR, and outliers (and, optionally, other distributional states) in one simple display.



Now, you can save documents to a PDF like this (of course you'd need to use your directory. It will just say "null device" in the output but it should create a PDF file in the appointed directory. Here I will create two plots: 1) a boxplot and 2) another kind of plot that we will work with later in the course called a scatterplot. Both will be saved to a PDF file.

Hide

```
pdf("beckBoxplot")  
# name the figure. As we already create the working directory at the beginning, we do not  
# need to write a full path code to save the figures.  
boxplot(drugtreatment$beck)  
pairs(~ drugtreatment$age+drugtreatment$beck)  
dev.off()
```

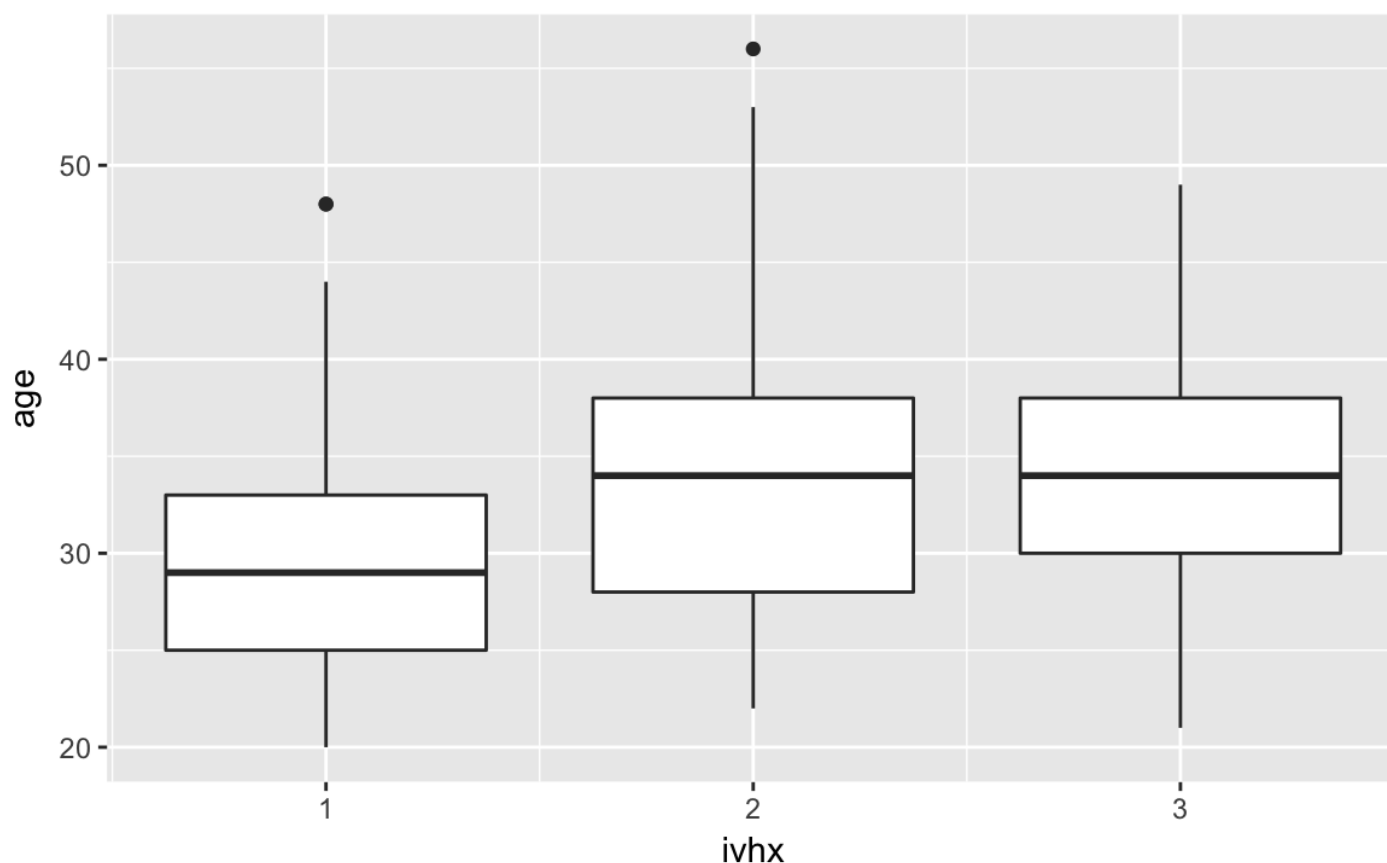
```
null device  
1
```

GGplot is a very powerful R procedure. First you have to install it which I do below. Then I show an example. Maybe change "beck" to "age" to get some familiarity. First we have to install the package. We only have to do this once. But we always have to add the package to the library.

Hide

```
install.packages("ggplot2")
```

```
Error in install.packages : Updating loaded packages
```



Here is how you create a function. This function is called `double.num`. A function is a set of code that you want to call again and again. You need to declare a function with `function(x)`. Then, in the curly brackets you list the code that you want to apply to the variable (which you declared as “x”). Then you use the function by plugging in a number for x. I realize this is advanced but see if you can follow the logic. First see if you can double another number like 6. Then see if you can create a function that multiplies by 3 and call it “triple”.

Hide

```
double.num(5)
```

```
[1] 10
```

Hide

```
triple.num <- function(x){  
  x*3  
}  
triple.num(4)
```

```
[1] 12
```

Hide

```
square.num <- function(x){  
  x*x  
}  
square.num(12)
```

```
[1] 144
```

Here I want to give you a little taste of what it is like to reshape data. The data now are at the person level of analysis. There is a package called `DPLYR` which, like SQL, can help you change the unit of analysis. There are two levels of the site variable. The following lines collapse the dataset from the person level to the site level of analysis. The code then saves a new file called “sitelevel”. If you follow this try to create a file called “IV level” which shows average age by levels of the IV drughistory variable which is called `ivhx`

Hide

```
sitelevel<-aggregate(age~SITE,drugtreatment, mean)
```

```
Error in eval(expr, envir, enclos) : object 'SITE' not found
```

`DPLYr` has some other nice features like filtering. Here we only select certain records and save to a dataset. We select individuals who are older than 40 and in drugtreatment. We save this file to a new file called “oldtreatment”. This is a common data reshaping technique called “subsetting”. Note the obs count under the “data” tab. Try to select individuals who are 25 or younger and call the new file “youngtreatment”

Hide

```
oldtreatment <- filter(drugtreatment,AGE > 40)
```

```
Error in filter_impl(.data, quo) :  
  Evaluation error: object 'AGE' not found.
```

selecting individuals 25 and younger: “youngtreatment”

[Hide](#)

```
youngtreatment <- filter(drugtreatment, AGE <= 25)  
View(youngtreatment)
```

When you save the notebook, an HTML file containing the code and output will be saved alongside it (click the *Preview* button or press *Ctrl+Shift+K* to preview the HTML file).

Save your work in both HTML file and PDF file (open the HTML file and choose print then save as PDF).