Data Science

EDA Course Activity

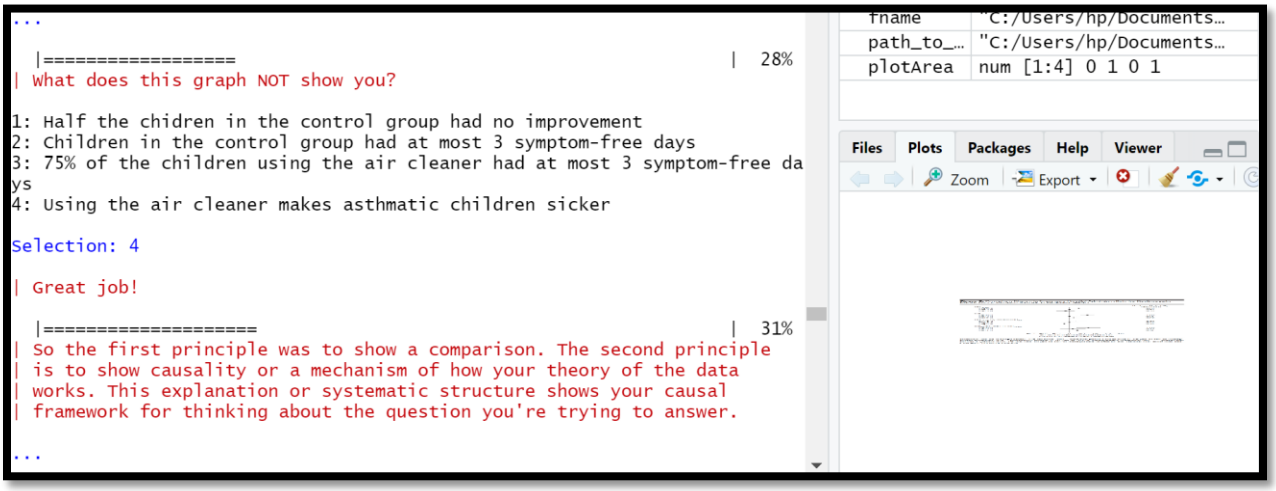
Sania Kashif **F17-8148**

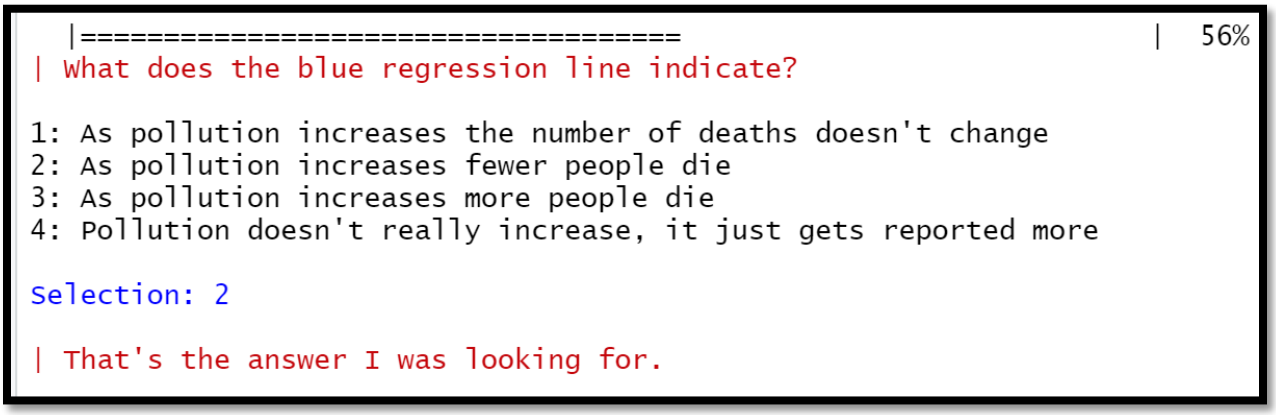
Submitted to

Sir Usman Joyia

Department of Computer Science Fast National University of Computer and Emerging Sciences

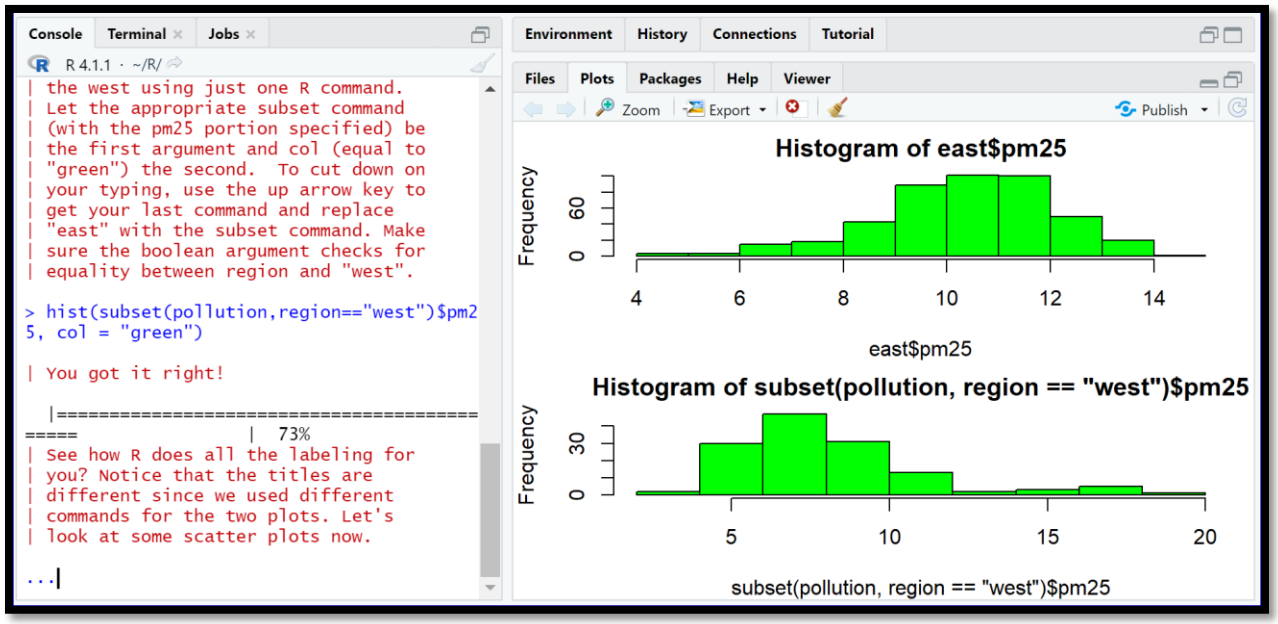
1: Principles of Analytic Graphs



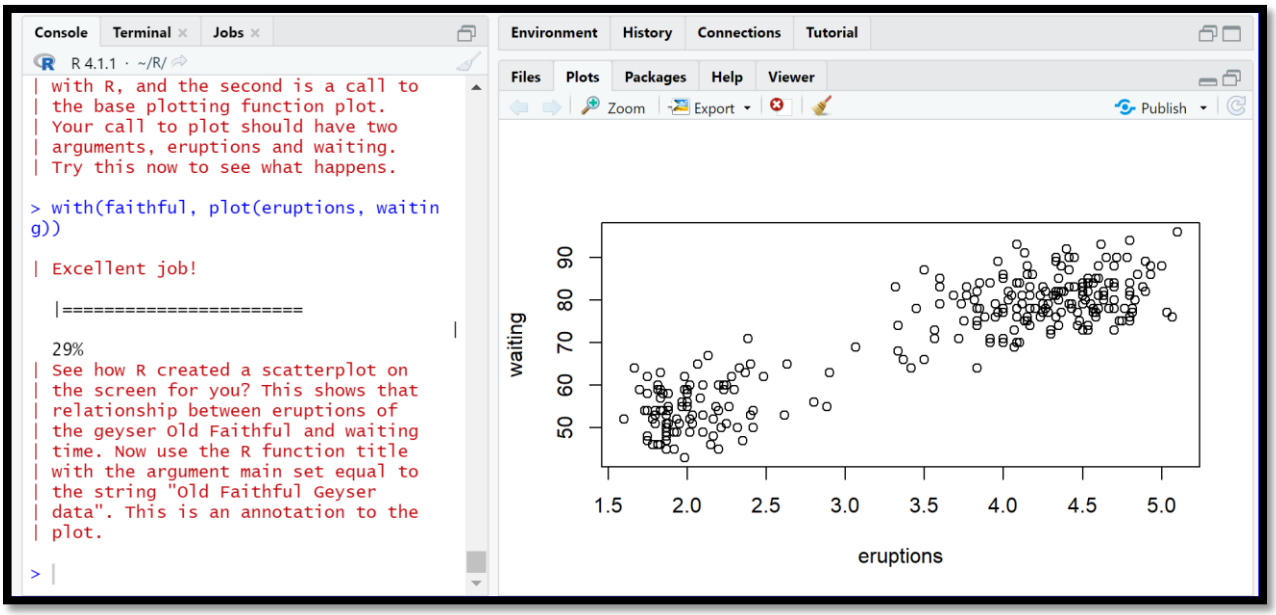


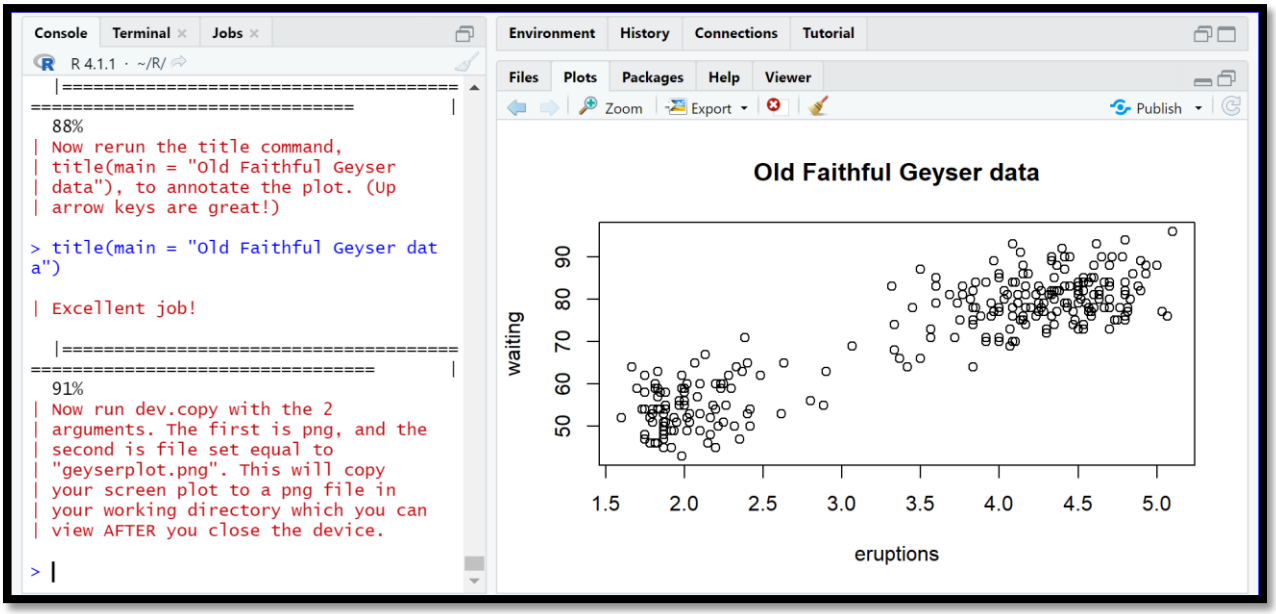
```
Selection: 4

| Great job!

  |=============================================================     |  94%
| True or False? Color is king.

1: True
2: False

Selection: 2

| Great job!

  |================================================================  |  97%
| Congrats! You've concluded exploring this lesson on principles of
| graphing. We hope you found it principally principled.

...

  |================================================================| 100%
| Would you like to receive credit for completing this course on
| Coursera.org?
```

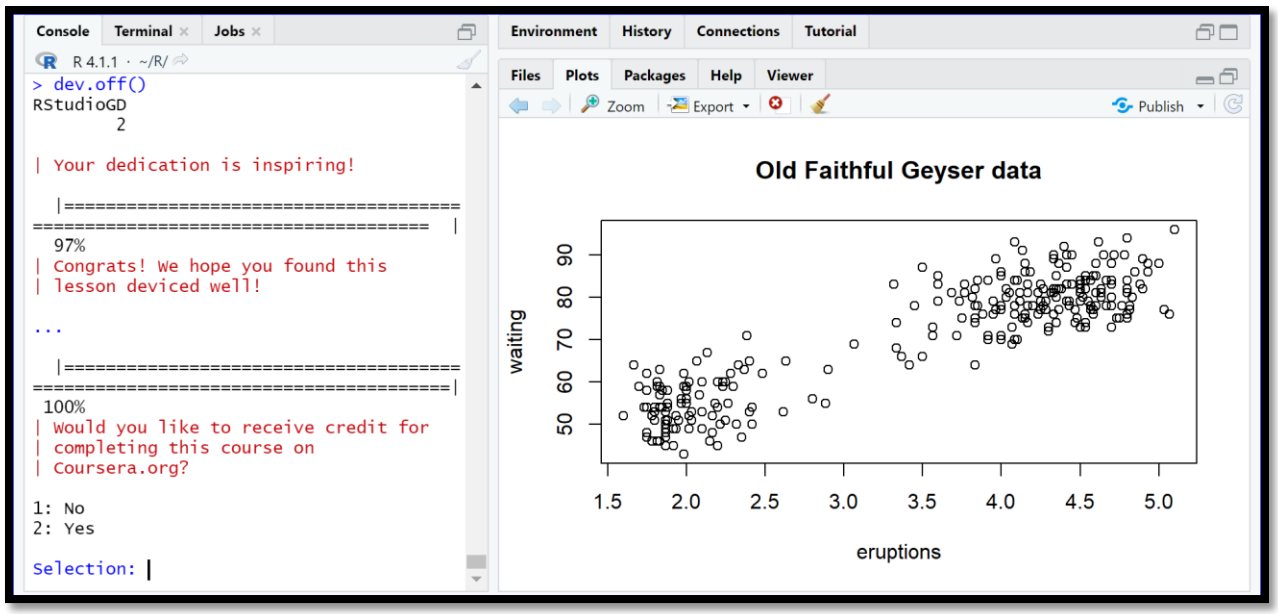
2: Exploratory Graphs



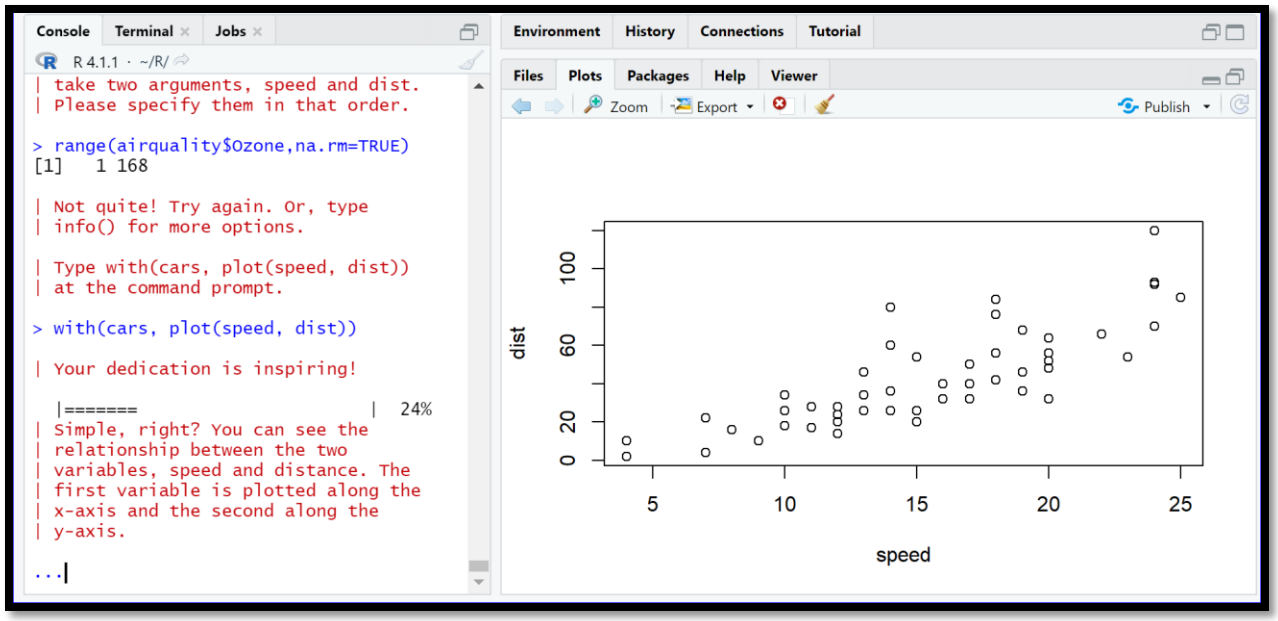
3: Graphics Devices in R



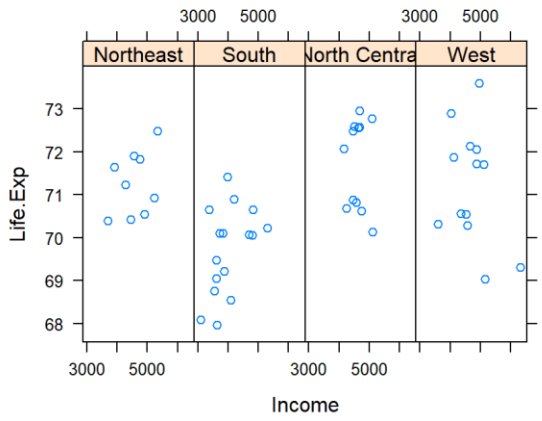




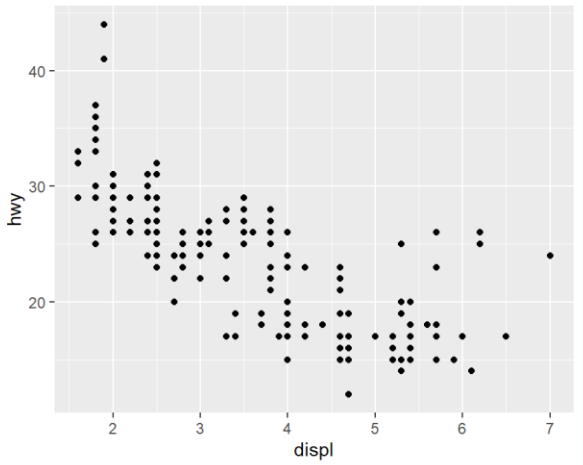
4: Plotting Systems

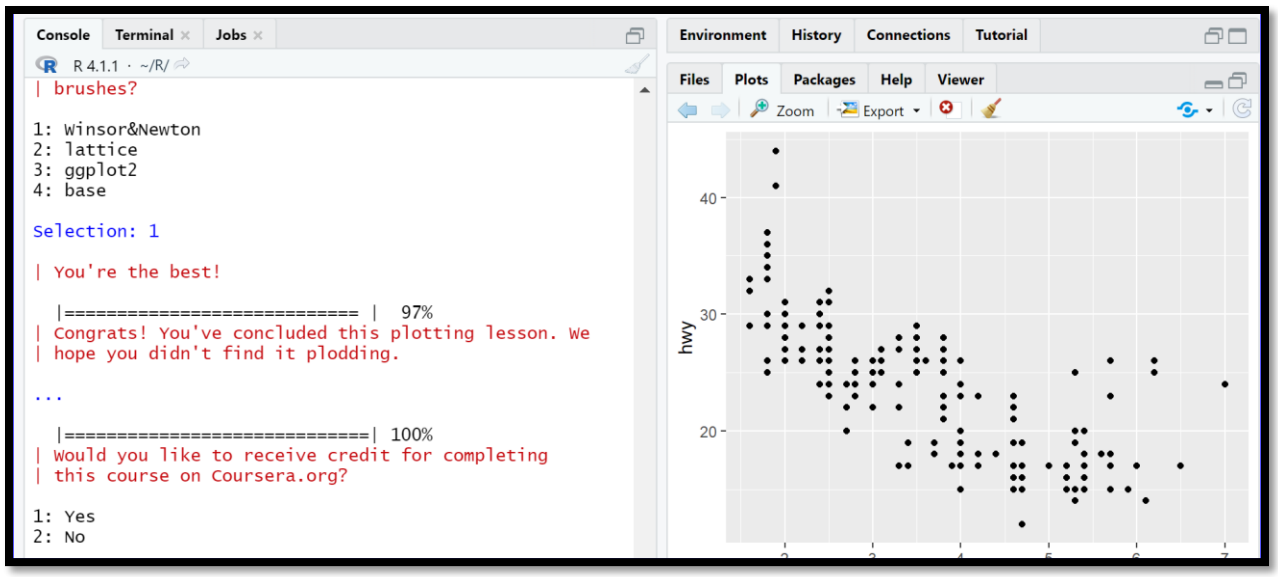


```
| plotting life expectancy as it depends on income
| for each region. The second argument, data, is set
| equal to state. This allows us to use "Life.Exp"
| and "Income" in the formula instead of specifying
| the dataset state for each term (as in
| state$Income). The third argument, layout, is set
| equal to the two-long vector c(4,1). Run xyplot now
| with these three arguments.

> xyplot(Life.Exp ~ Income | region, data = state, layout
 = c(4, 1))

| Keep up the great work!

  |===============                |  51%
| We see the data for each of the 4 regions plotted
| in one row. Based on this plot, which region of the
| U.S. seems to have the shortest life expectancy?

1: Northeast
2: North Central
3: West
4: South

Selection:
```

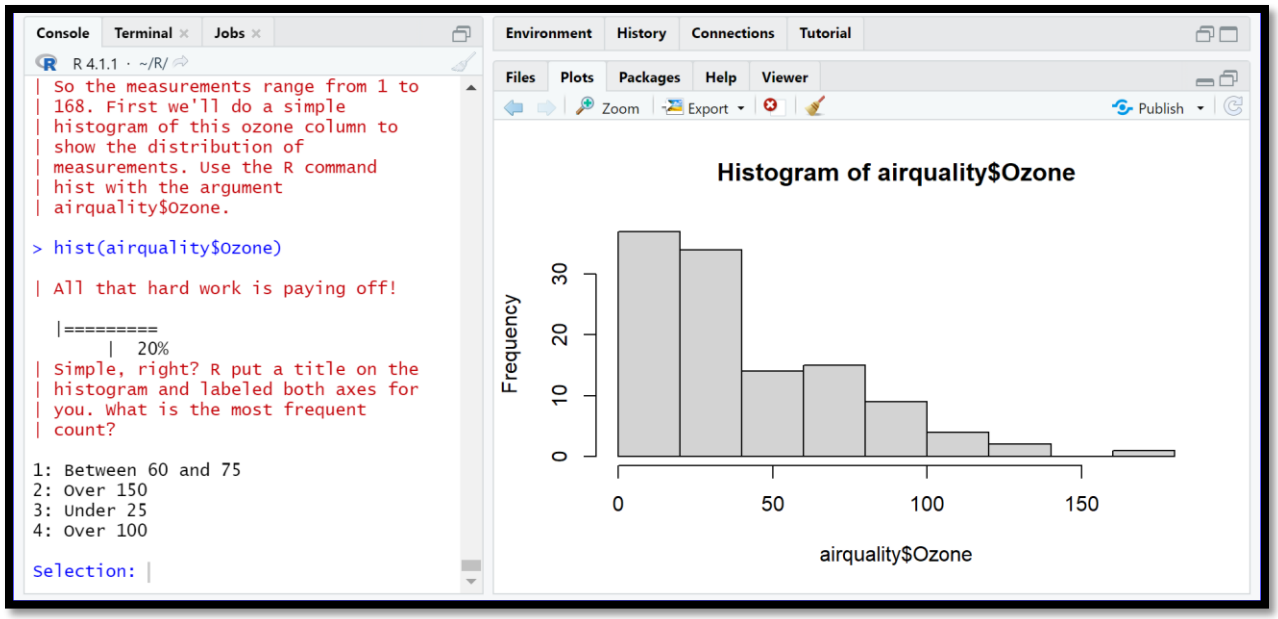


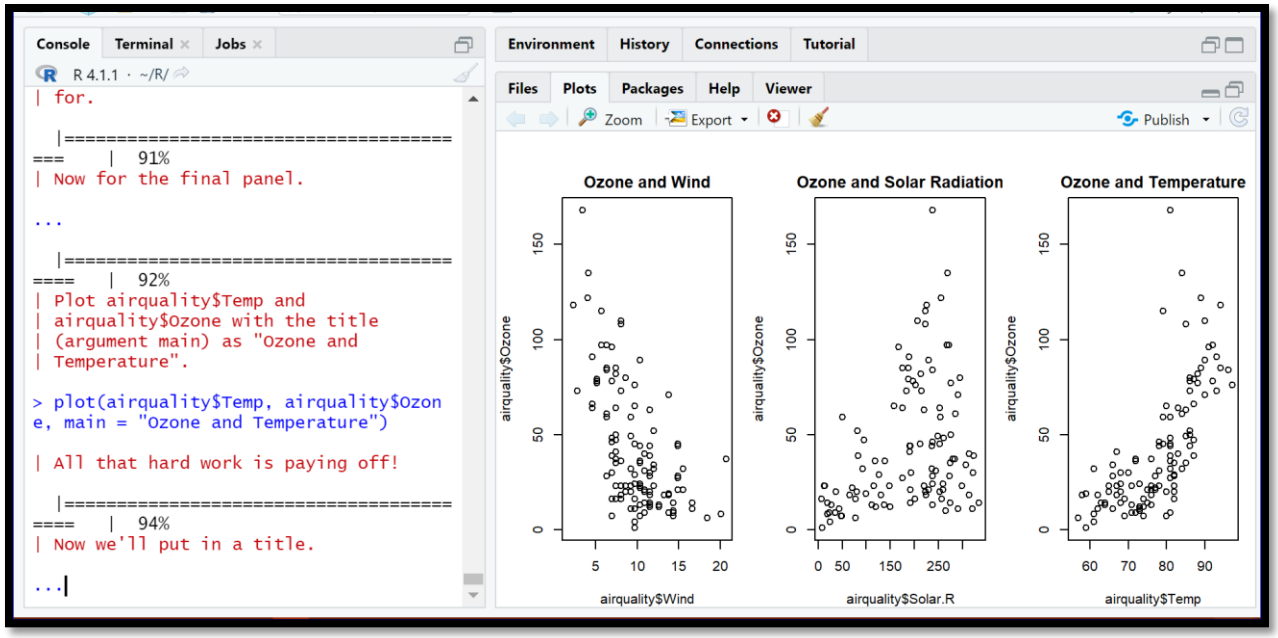

```
                tiburon      toyota tacoma 4wd
                      7                      7

| That's correct!

  |=====================          |  73%
| Oh, there are 38 models. We're interested in the
| effect engine displacement (displ) has on highway
| gas mileage (hwy), so we'll use the ggplot2 command
| qplot to display this relationship. Run qplot now
| with three arguments. The first two are the
| variables displ and hwy we want to plot, and the
| third is the argument data set equal to mpg. As
| before, this allows us to avoid using the
| mpg$variable notation for the first two arguments.

> qplot(displ, hwy, data = mpg)

| You're the best!

  |=====================          |  76%
| Not surprisingly we see that the bigger the engine
| displacement the lower the gas mileage.

...
```



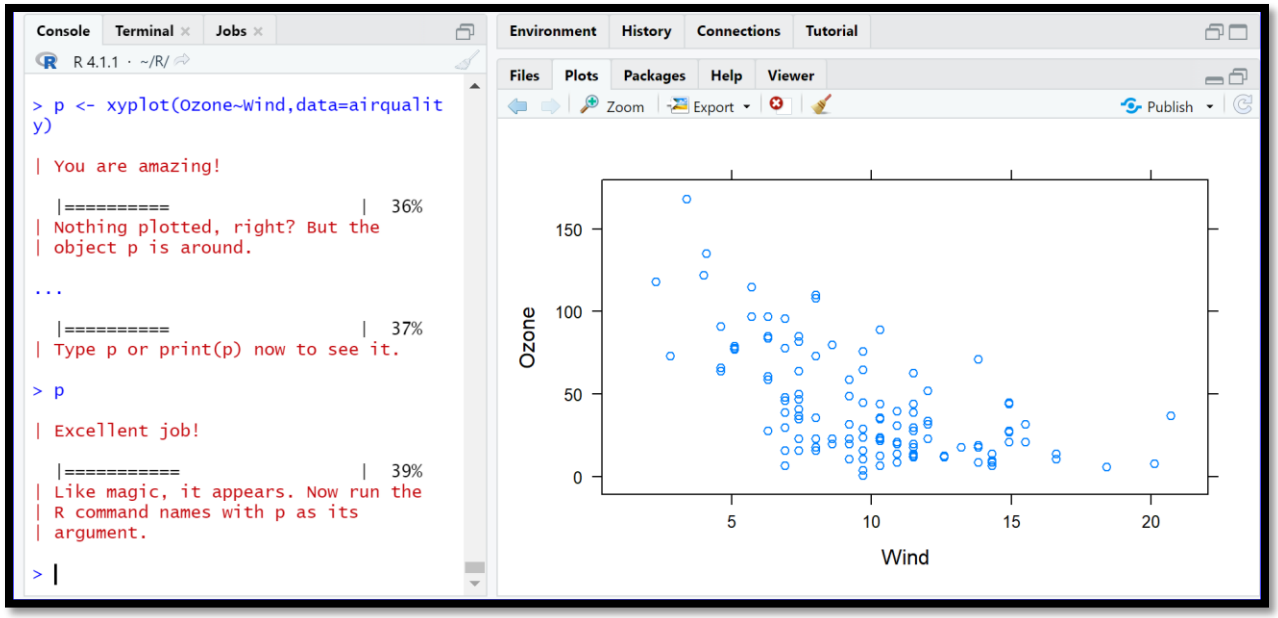


5: Base Plotting System

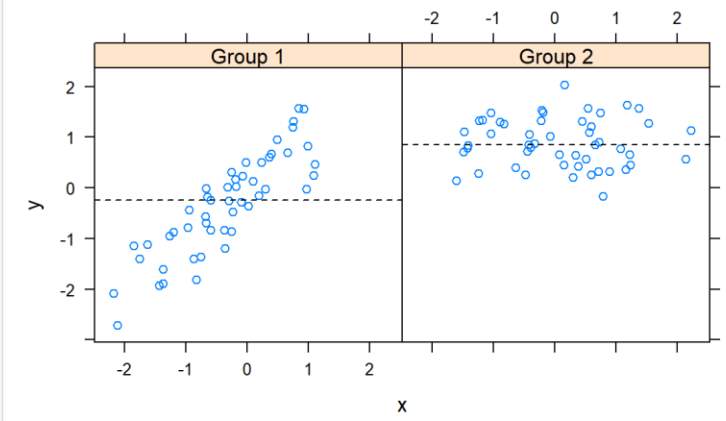




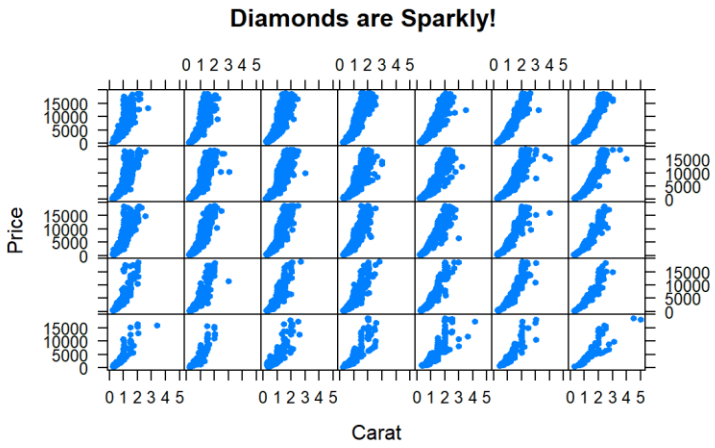
6: Lattice Plotting System

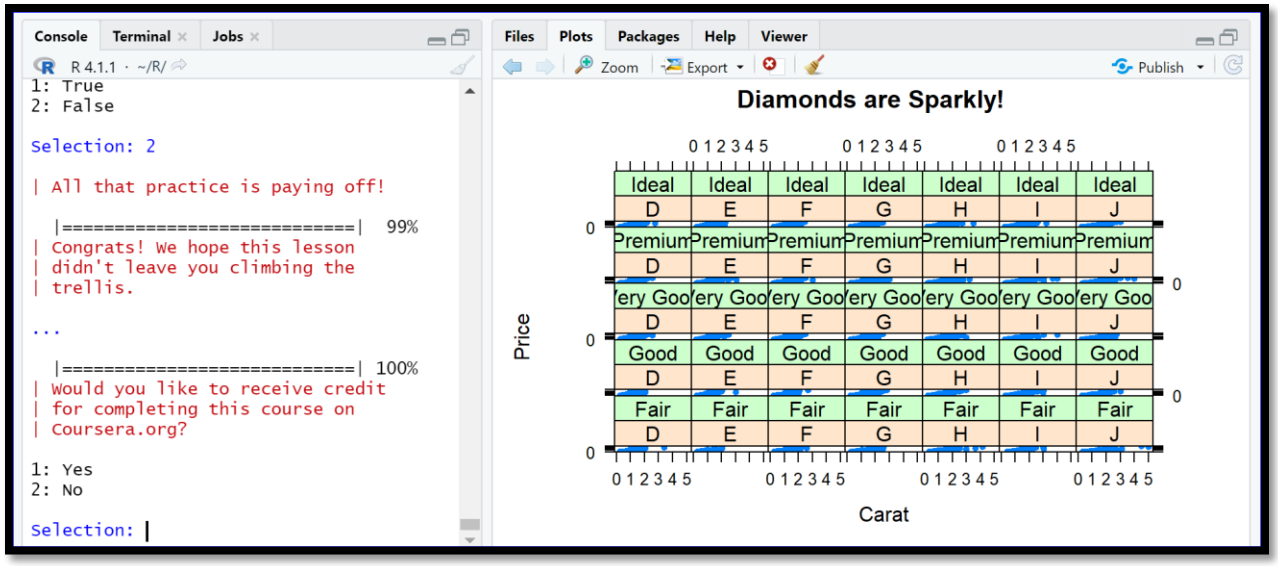


```
1: f contains 2 factors
2: there are 2 calls to panel methods
3: there are 2 variables
4: lattice can handle at most 2 panels

Selection: 1

| Perseverance, that's the answer.

  |==================            |  63%
| We've copied another piece of
| similar code, i.e., a call to
| xyplot with a custom panel
| function, from the slides. To see
| it, type myedit("plot2.R"). This
| will open your editor and display
| the R code in it.

>
```

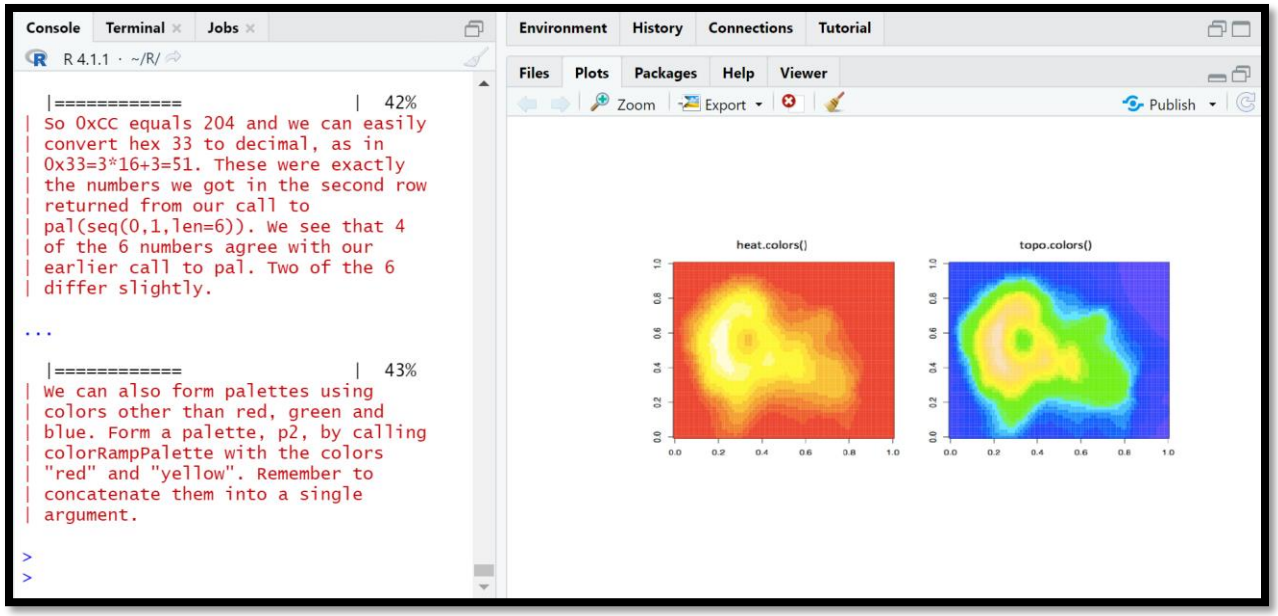



```
=myylab,main=mymain)

| That's the answer I was looking
| for.

  |=======================       |  81%
| Pretty cool, right? 35 panels, one
| for each combination of color and
| cut. The dots (pch=20) show how
| prices for the diamonds in each
| category (panel) vary depending on
| carat.

...

  |=======================       |  82%
| Are colors defining the rows or
| columns of the plot?

1: columns
2: rows

Selection:
```





7: Working with Colors



Console Terminal Jobs

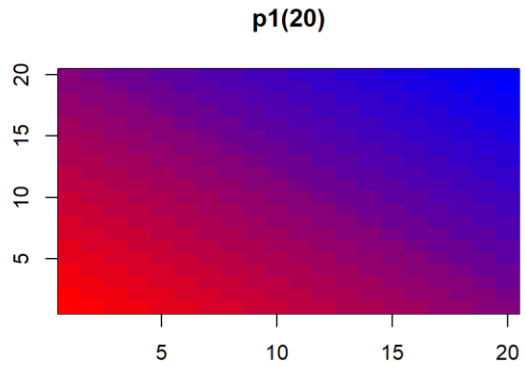
R 4.1.1 · ~/R/

```
...

  |==============                    |  51%
| This is all fine and dandy but you're probably
| wondering when you can see how all these colors
| show up in a display. We copied some code from
| the R documentation pages (color.scale if you're
| interested) and created a function for you,
| showMe. This takes as an argument, a color
| vector, which as you know, is precisely what
| calls to p1 and p2 return to you. Call showMe now
| with p1(20).

> showMe(p1(20))

| All that practice is paying off!

  |===============                   |  52%
| We see the interpolated palette here. Low values
| in the lower left corner are red and as you move
| to the upper right, the colors move toward blue.
| Now call showMe with p2(20) as its argument.

>
```

Environment History Connections Tutorial

Files Plots Packages Help Viewer

Zoom Export



Console Terminal Jobs

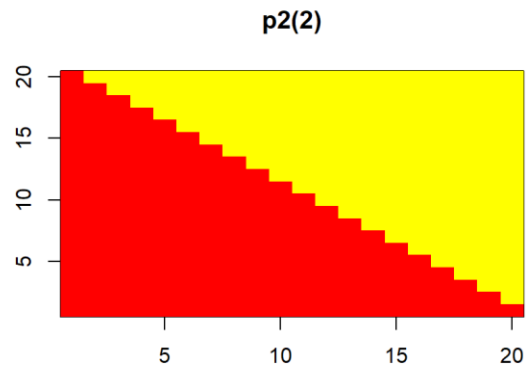
R 4.1.1 · ~/R/

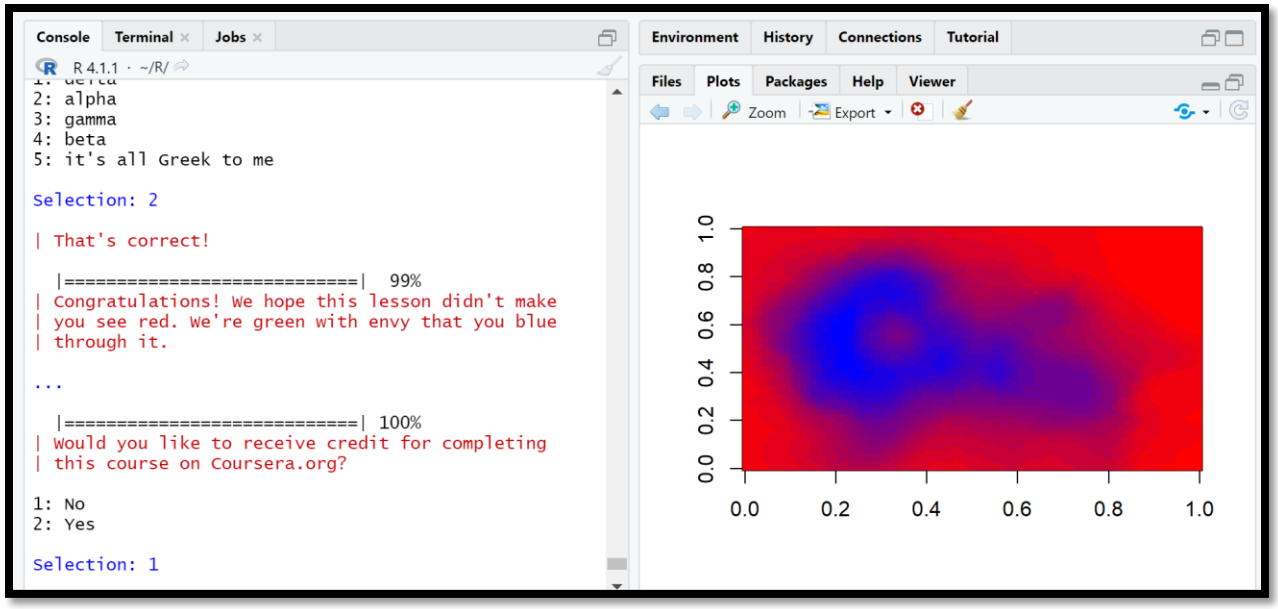
```
> showMe(p2(2))

| You got it right!

  |================                  |  55%
| A much more basic pattern, simple but elegant.

...

  |================                  |  57%
| We mentioned before that colorRamp (and
| colorRampPalette) could return a 3 or 4 long
| vector of colors. We saw 3-long vectors returned
| indicating red, green, and blue intensities. What
| would the 4th entry be?

...

  |================                  |  58%
| We'll answer this indirectly. First, look at the
| function p1 that colorRampPalette returned to
| you. Just type p1 at the command prompt.

>
```

Environment History Connections Tutorial

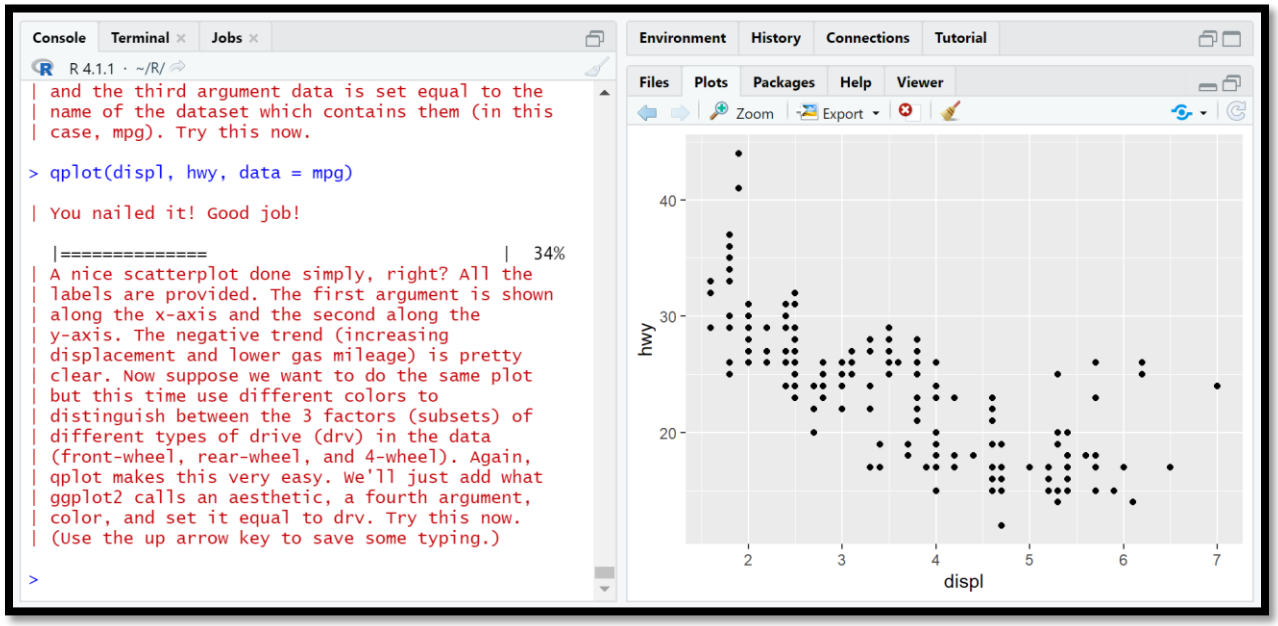
Files Plots Packages Help Viewer

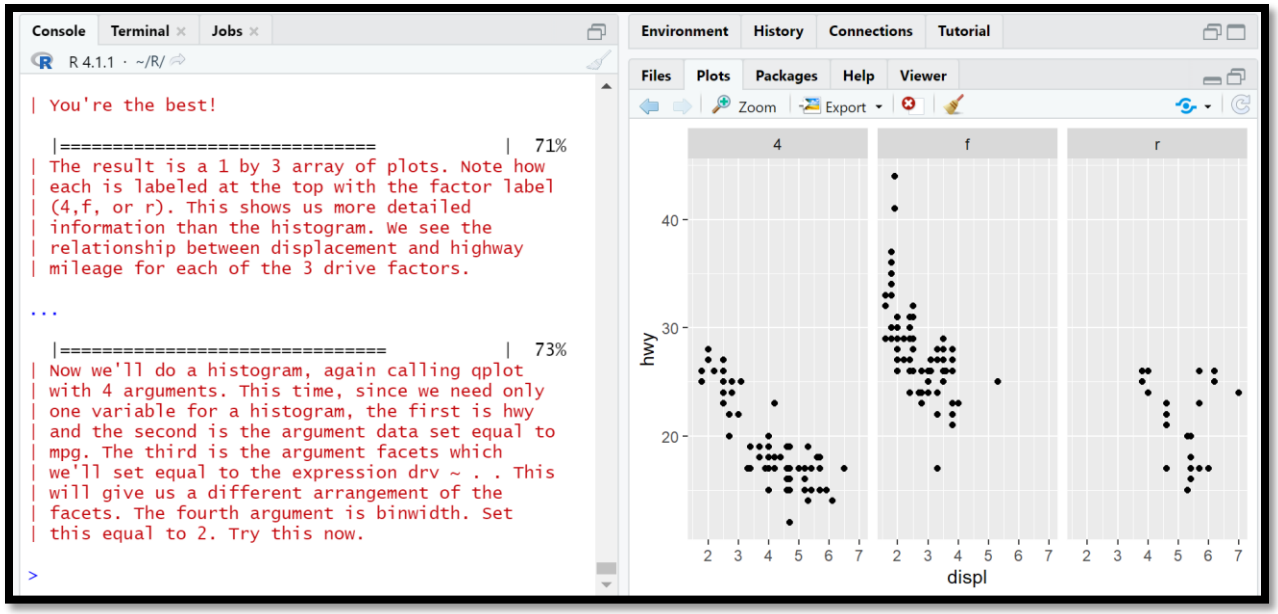
Zoom Export

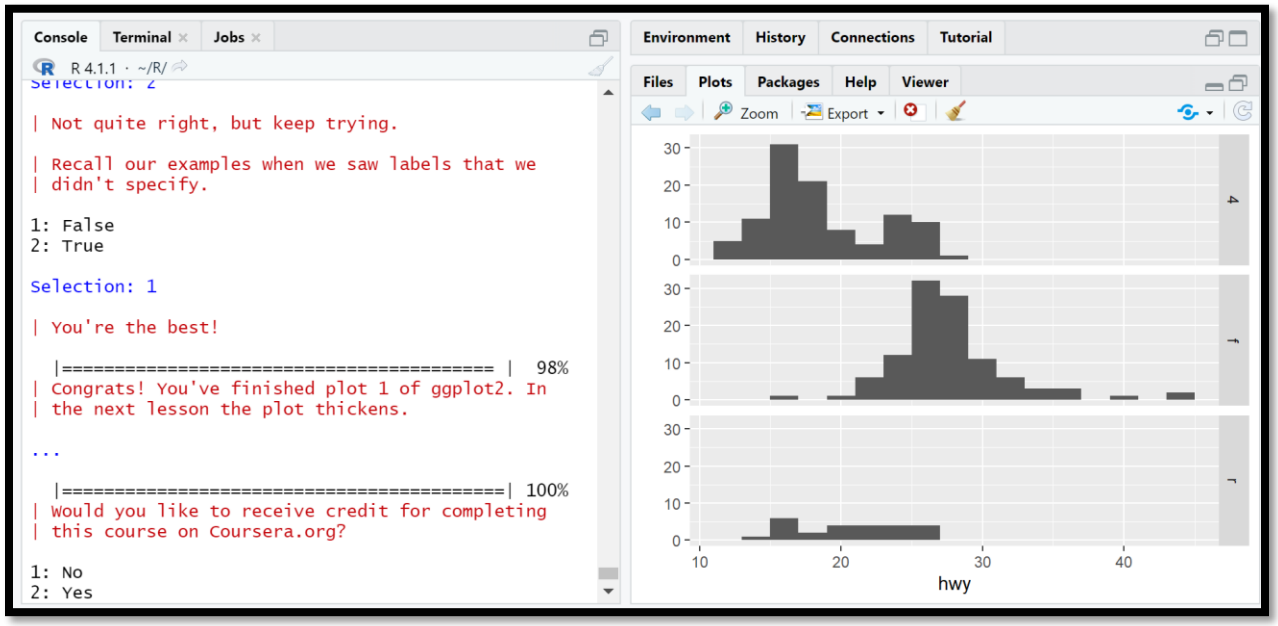




8: GGPlot2 Part1

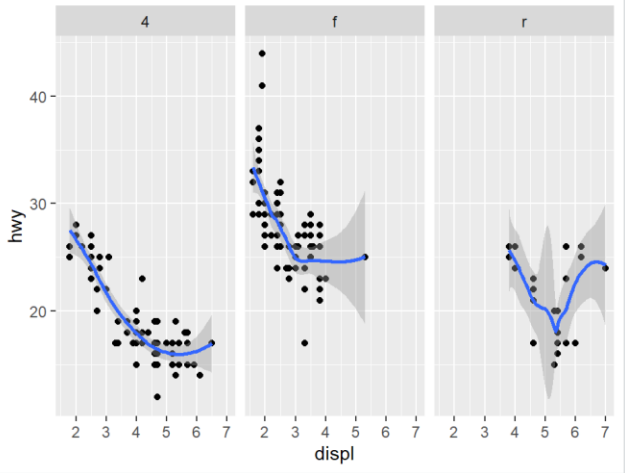




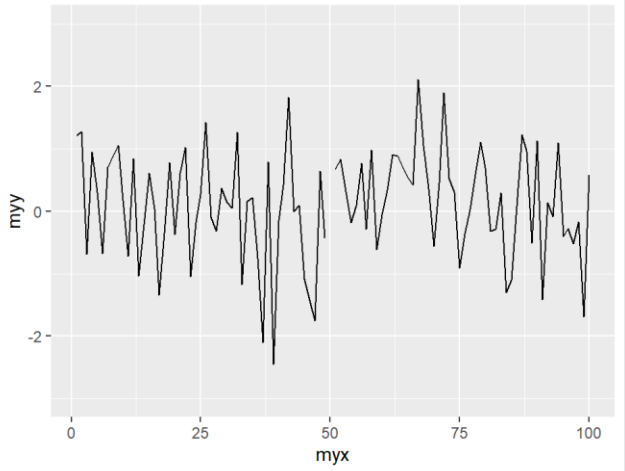


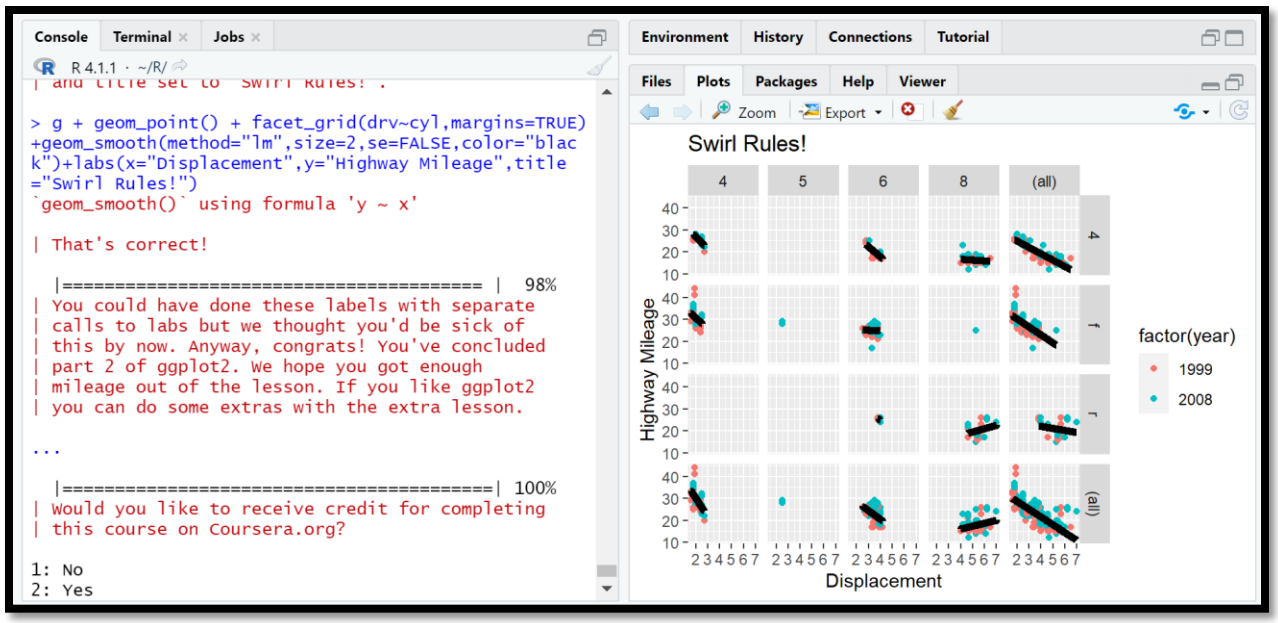
9: GGPlot2 Part2

```
`geom_smooth()` using method = 'loess' and formula 'y
 ~ x'

| You got it right!

  |========                                      |  19%
| We see a 3 facet plot, one for each drive type
| (4, f, and r). Now we'll see how ggplot works.
| We'll build up a similar plot using the basic
| components of the package. We'll do this in a
| series of steps.

...

  |=========                                     |  21%
| First we'll create a variable g by assigning to
| it the output of a call to ggplot with 2
| arguments. The first is mpg (our dataset) and
| the second will tell ggplot what we want to
| plot, in this case, displ and hwy. These are
| what we want our aesthetics to represent so we
| enclose these as two arguments to the function
| aes. Try this now.

>
```

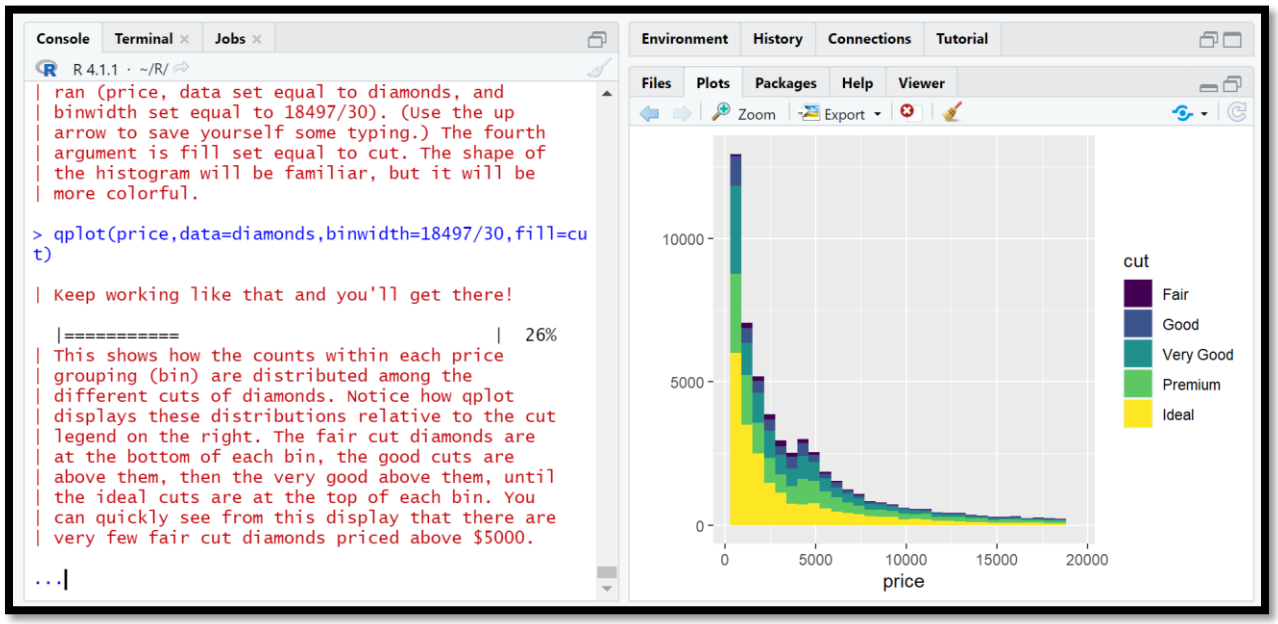


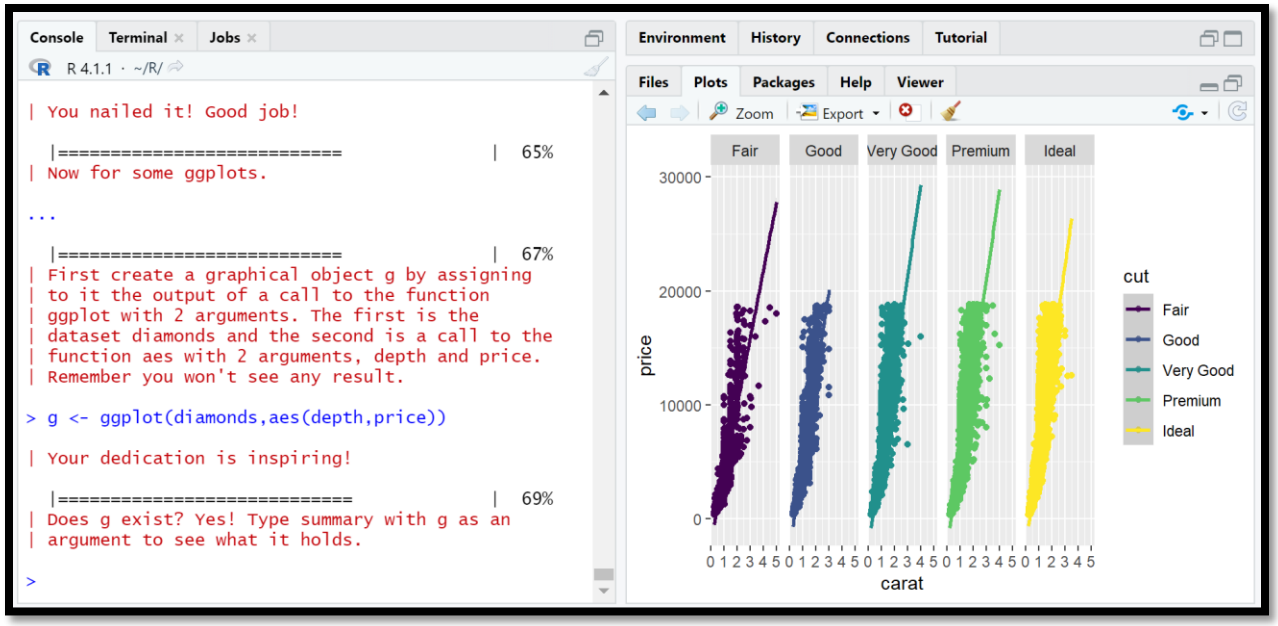

```
  |==============================                |  73%
| Let's take a guess that in addition to adding
| geom_line() to g we also just have to add
| ylim(-3,3) to it as we did with the call to
| plot. Try this now to see what happens.

> g + geom_line() + ylim(-3,3)

| Keep working like that and you'll get there!

  |===============================               |  75%
| Notice that by doing this, ggplot simply ignored
| the outlier point at (50,100). There's a break
| in the line which isn't very noticeable. Now
| recall that at the beginning of the lesson we
| mentioned 7 components of a ggplot plot, one of
| which was a coordinate system. This is a
| situation where using a coordinate system would
| be helpful. Instead of adding ylim(-3,3) to the
| expression g+geom_line(), add a call to the
| function coord_cartesian with the argument ylim
| set equal to c(-3,3).

>
```

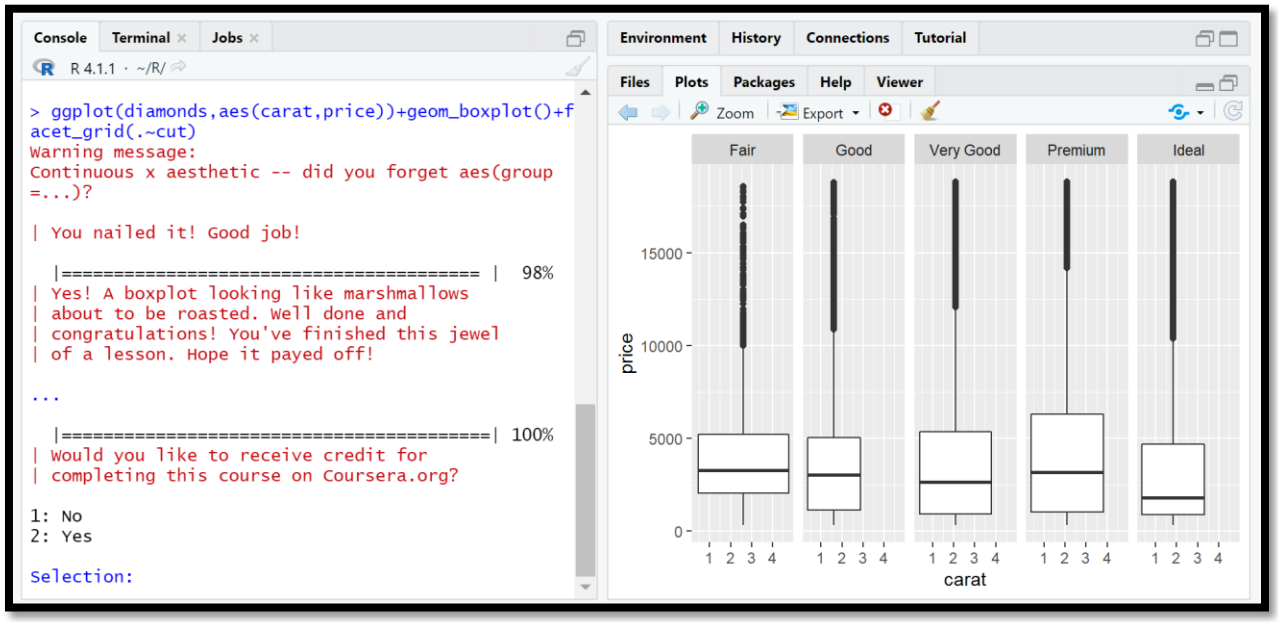




10: GGPlot2 Extras







11: Hierarchical Clustering

Console Terminal Jobs

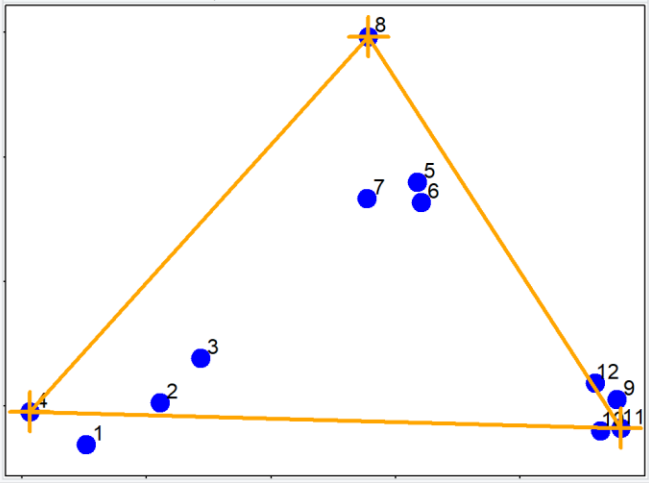
R 4.1.1 · ~/R/

```
| The distance between the two clusters of
| points (9 through 12) and (5 through 8), using
| complete linkage as the metric, is the
| distance between points 11 and 8 since this is
| the largest distance between the pairs of
| those groups.

...

  |======================                  |  58%
| As luck would have it, the distance between
| the two clusters of points (9 through 12) and
| (1 through 4), using complete linkage as the
| metric, is the distance between points 11 and
| 4.

...

  |======================                  |  60%
| We've created the dataframe dFsm for you
| containing these 3 points, 4, 8, and 11. Run
| dist on dFsm to see what the smallest distance
| between these 3 points is.

>
```

Environment History Connections Tutorial

Files Plots Packages Help Viewer

Zoom Export



Console Terminal Jobs

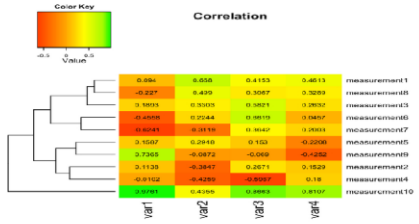
R 4.1.1 · ~/R/

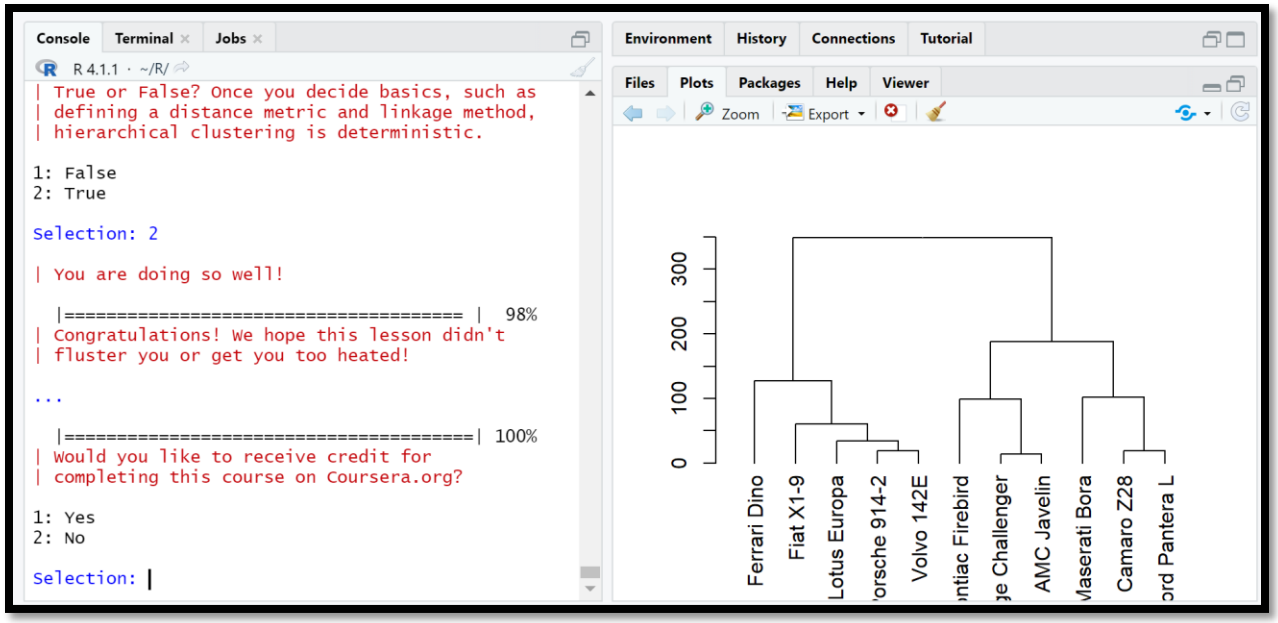
```
| We won't say too much on this topic, but a
| very nice concise tutorial on creating
| heatmaps in R exists at
| http://sebastianraschka.com/Articles/heatmaps_in_
r.html#clustering.
| Here's an image from the tutorial to start you
| thinking about the topic. It shows a sample
| heat map with a dendrogram on the left edge
| mapping the relationship between the rows. The
| legend at the top shows how colors relate to
| values.

...

  |============================            |  76%
| R provides a handy function to produce heat
| maps. It's called heatmap. We've put the point
| data we've been using throughout this lesson
| in a matrix. Call heatmap now with 2
| arguments. The first is dataMatrix and the
| second is col set equal to cm.colors(25). This
| last is optional, but we like the colors
| better than the default ones.

>
```

Environment History Connections Tutorial

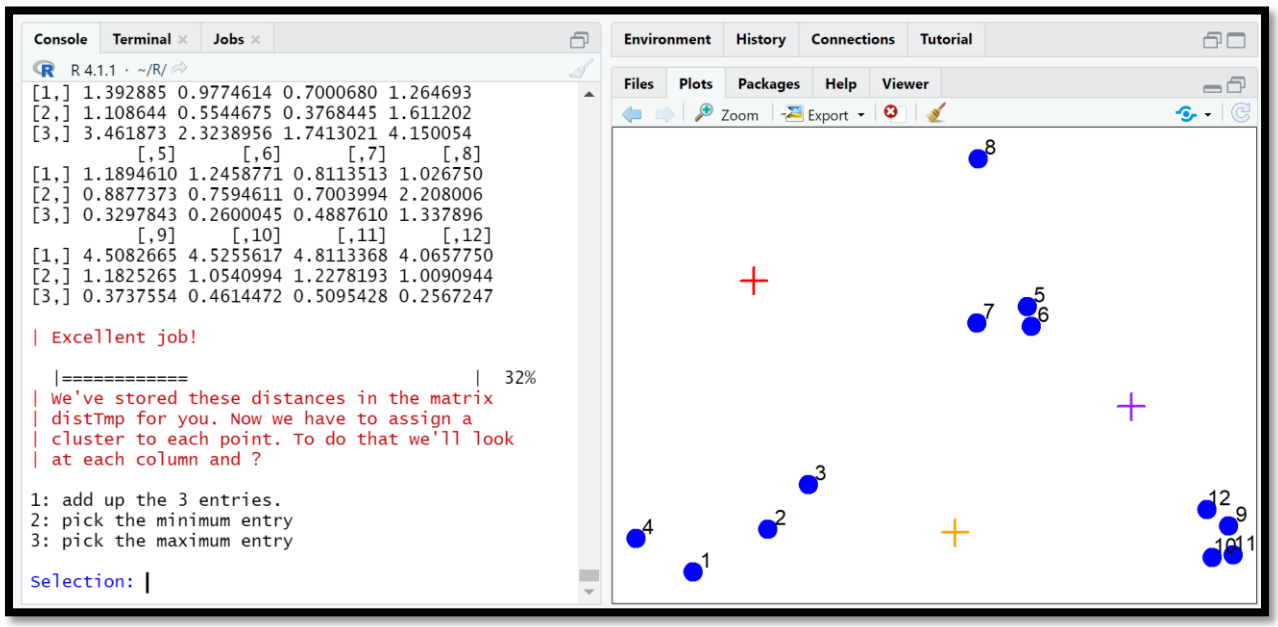
Files Plots Packages Help Viewer

Zoom Export





12: K Means Clustering



13: Dimension Reduction

addPatt.R*

```
1  set.seed(678910)
2  for(i in 1:40){
3    # flip a coin
4    coinFlip <- rbinom(1,size=1,prob=0.5)
5    # if coin is heads add a common pattern to that
6
```

1:1 (Top Level) R Script

Console Terminal Jobs

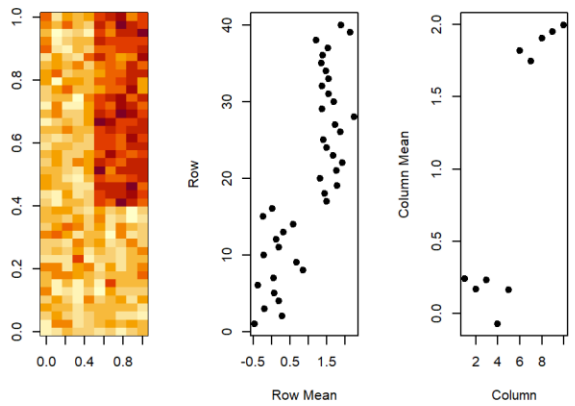
R 4.1.1 · ~/R/

```
| vectors of X and V's columns are the right
| singular vectors of X.  D is a diagonal matrix,
| by which we mean that all of its entries not on
| the diagonal are 0. The diagonal entries of D
| are the singular values of X.

...

  |==============
  |  28%
| To illustrate this idea we created a simple
| example matrix called mat. Look at it now.

>
```

Environment History Connections Tutorial

Files Plots Packages Help Viewer

Zoom Export



addPatt.R*

```
1  set.seed(678910)
2  for(i in 1:40){
3    # flip a coin
4    coinFlip <- rbinom(1,size=1,prob=0.5)
5    # if coin is heads add a common pattern to that
6
```

1:1 (Top Level) R Script

Console Terminal Jobs

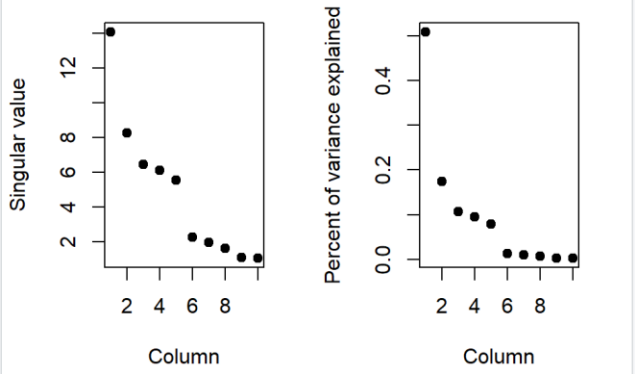
R 4.1.1 · ~/R/

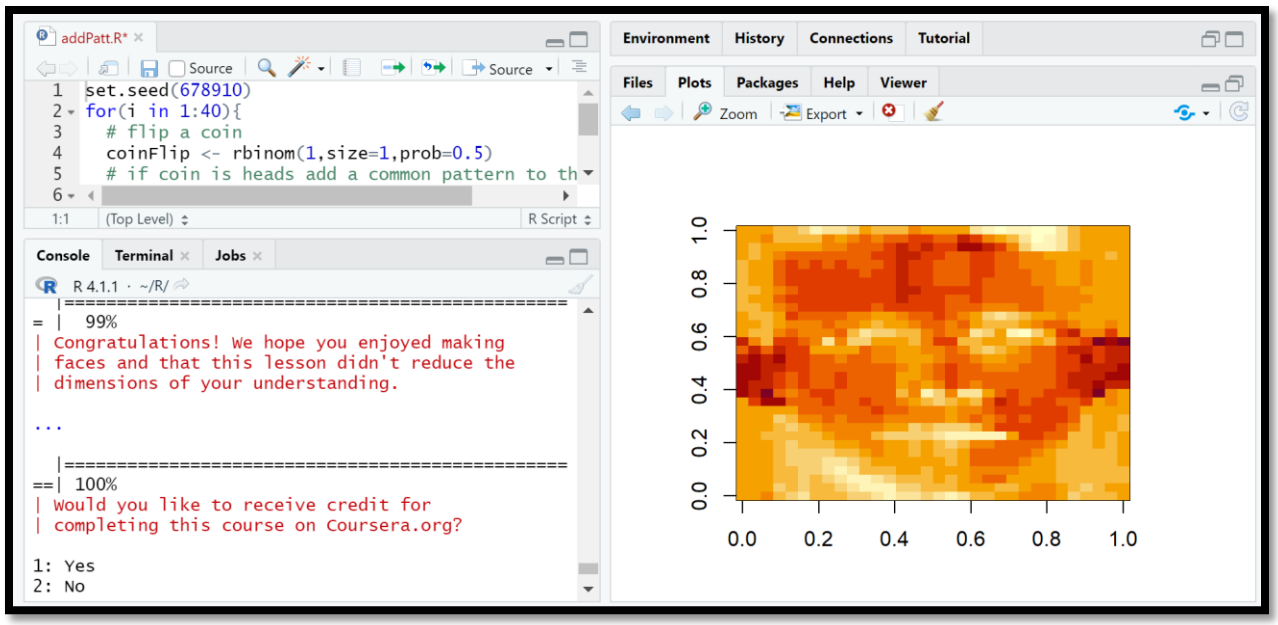
```
...

  |================================
  |  66%
| According to the plot, how much of the variance
| does the second element account for?

1: 11%
2: 18%
3: .1%
4: 53%

Selection:
```

Environment History Connections Tutorial

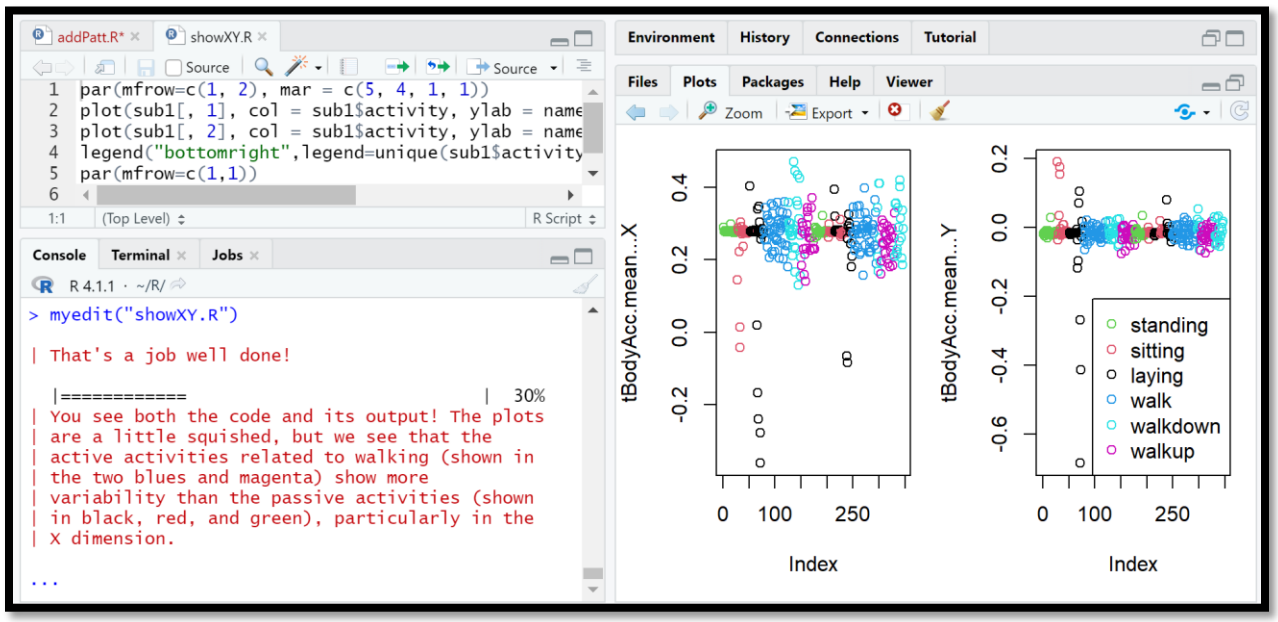
Files Plots Packages Help Viewer

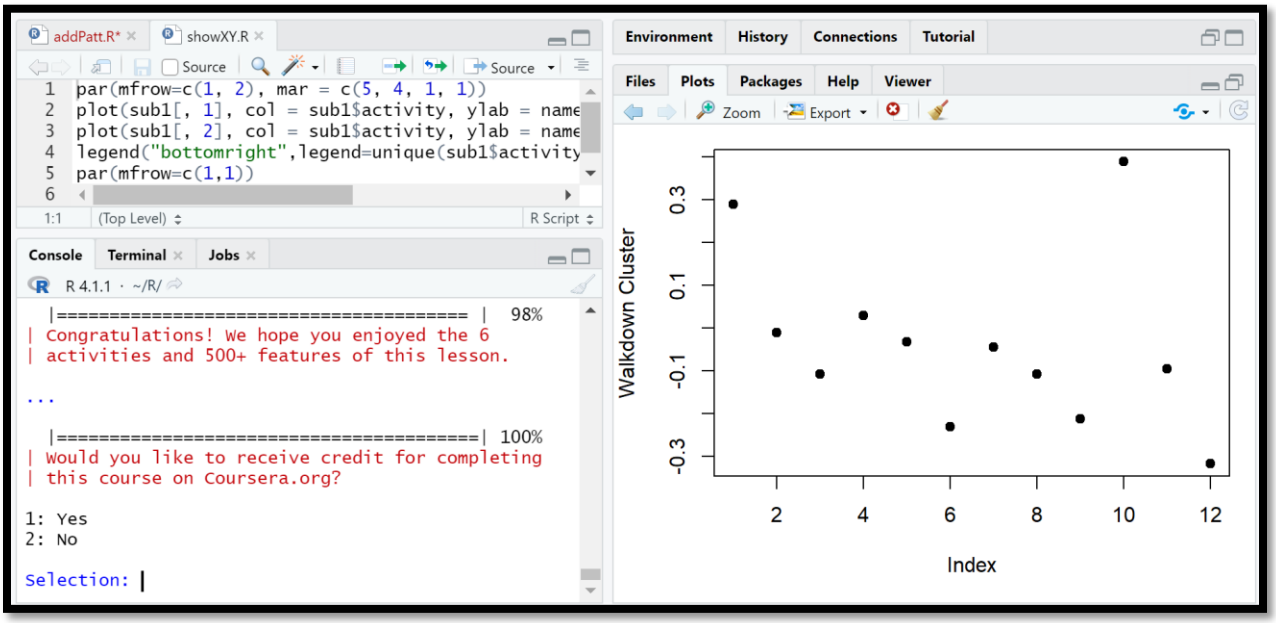
Zoom Export



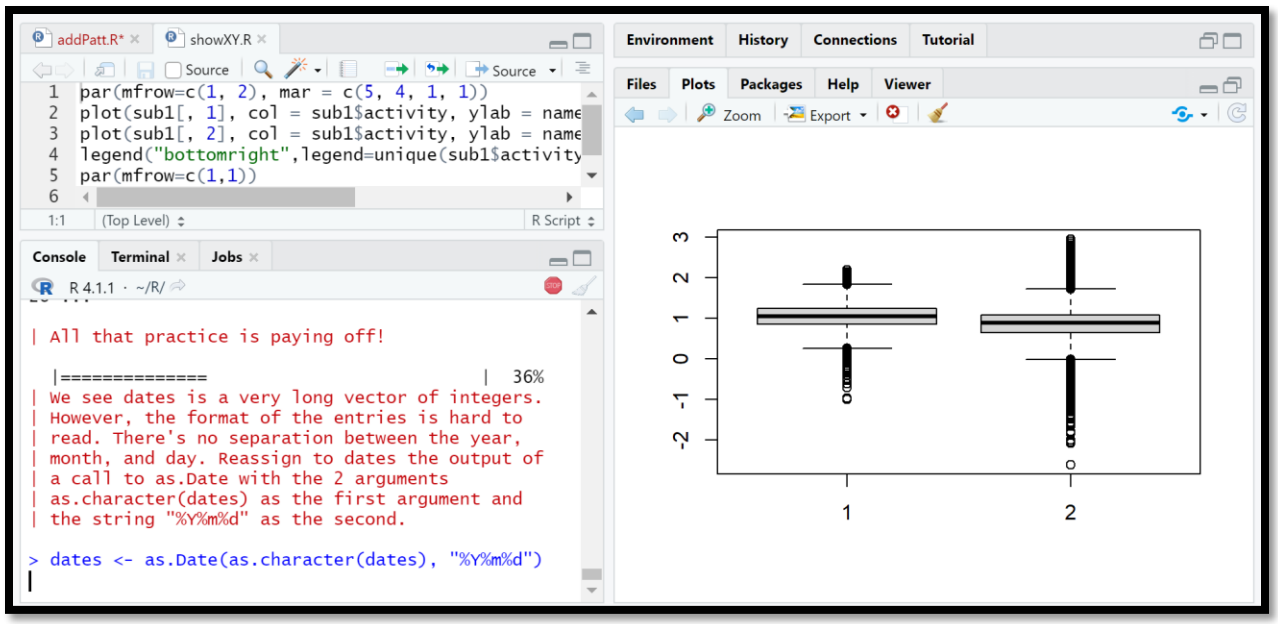


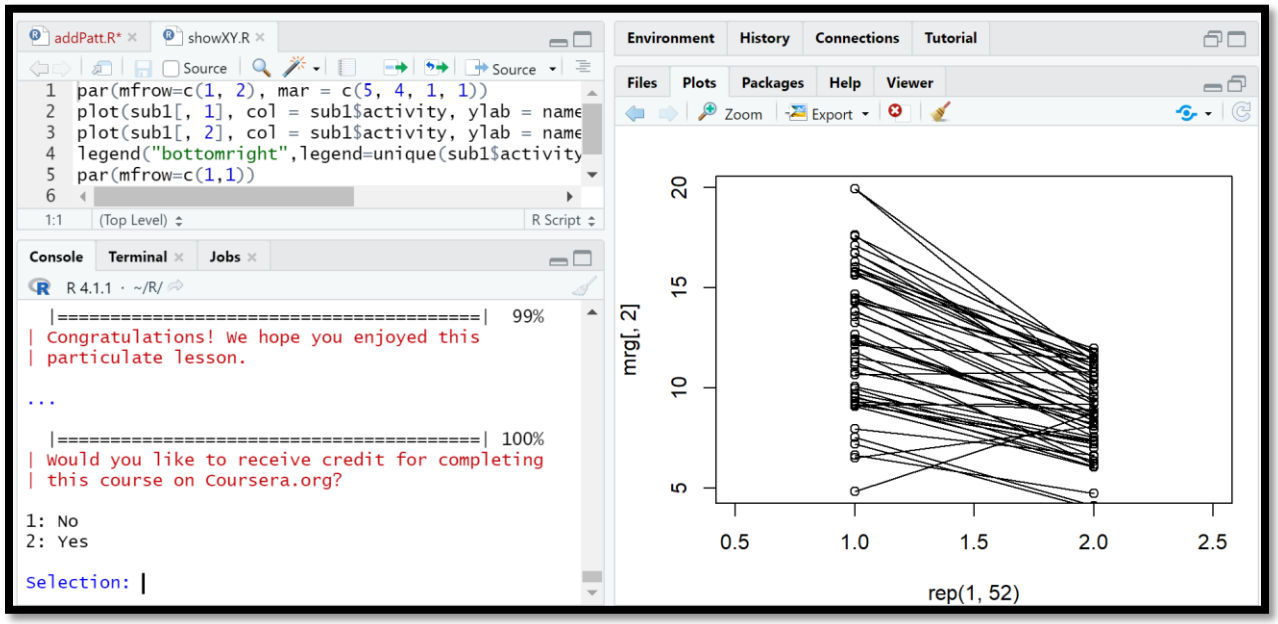
14: Clustering Example





15: Case Study


addPatt.R*
showXY.R
Source
par(mfrow=c(1, 2), mar = c(5, 4, 1, 1))
plot(sub1[, 1], col = sub1$activity, ylab = name
plot(sub1[, 2], col = sub1$activity, ylab = name
legend("bottomright",legend=unique(sub1$activity
par(mfrow=c(1,1))
1:1
(Top Level)
R Script
Console
Terminal
Jobs
R 4.1.1 · ~/R/
| All that practice is paying off!
|==============| 36%
| We see dates is a very long vector of integers.
| However, the format of the entries is hard to
| read. There's no separation between the year,
| month, and day. Reassign to dates the output of
| a call to as.Date with the 2 arguments
| as.character(dates) as the first argument and
| the string "%Y%m%d" as the second.
> dates <- as.Date(as.character(dates), "%Y%m%d")
Environment
History
Connections
Tutorial
Files
Plots
Packages
Help
Viewer
Zoom
Export
-2
-1
0
1
2
3
1
2


addPatt.R*
showXY.R
Source
par(mfrow=c(1, 2), mar = c(5, 4, 1, 1))
plot(sub1[, 1], col = sub1$activity, ylab = name
plot(sub1[, 2], col = sub1$activity, ylab = name
legend("bottomright",legend=unique(sub1$activity
par(mfrow=c(1,1))
1:1
(Top Level)
R Script
Console
Terminal
Jobs
R 4.1.1 · ~/R/
|==========| 99%
| Congratulations! We hope you enjoyed this
| particulate lesson.
...
|==========| 100%
| Would you like to receive credit for completing
| this course on Coursera.org?
1: No
2: Yes
Selection:
Environment
History
Connections
Tutorial
Files
Plots
Packages
Help
Viewer
Zoom
Export
mrg[, 2]
5
10
15
20
0.5
1.0
1.5
2.0
2.5
rep(1, 52)