Data Visualization

First Steps: Getting Started

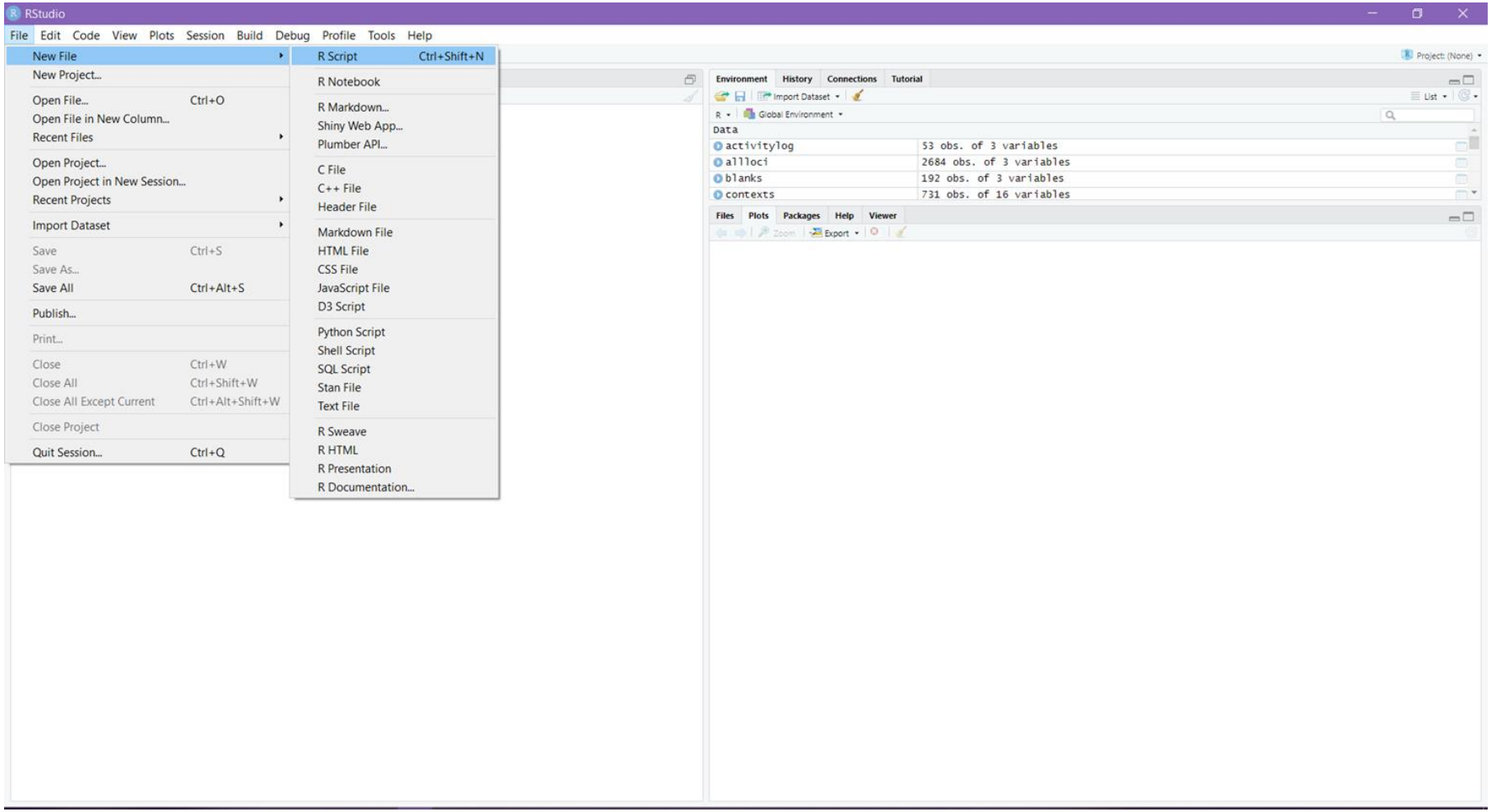
Ciara Zogheib

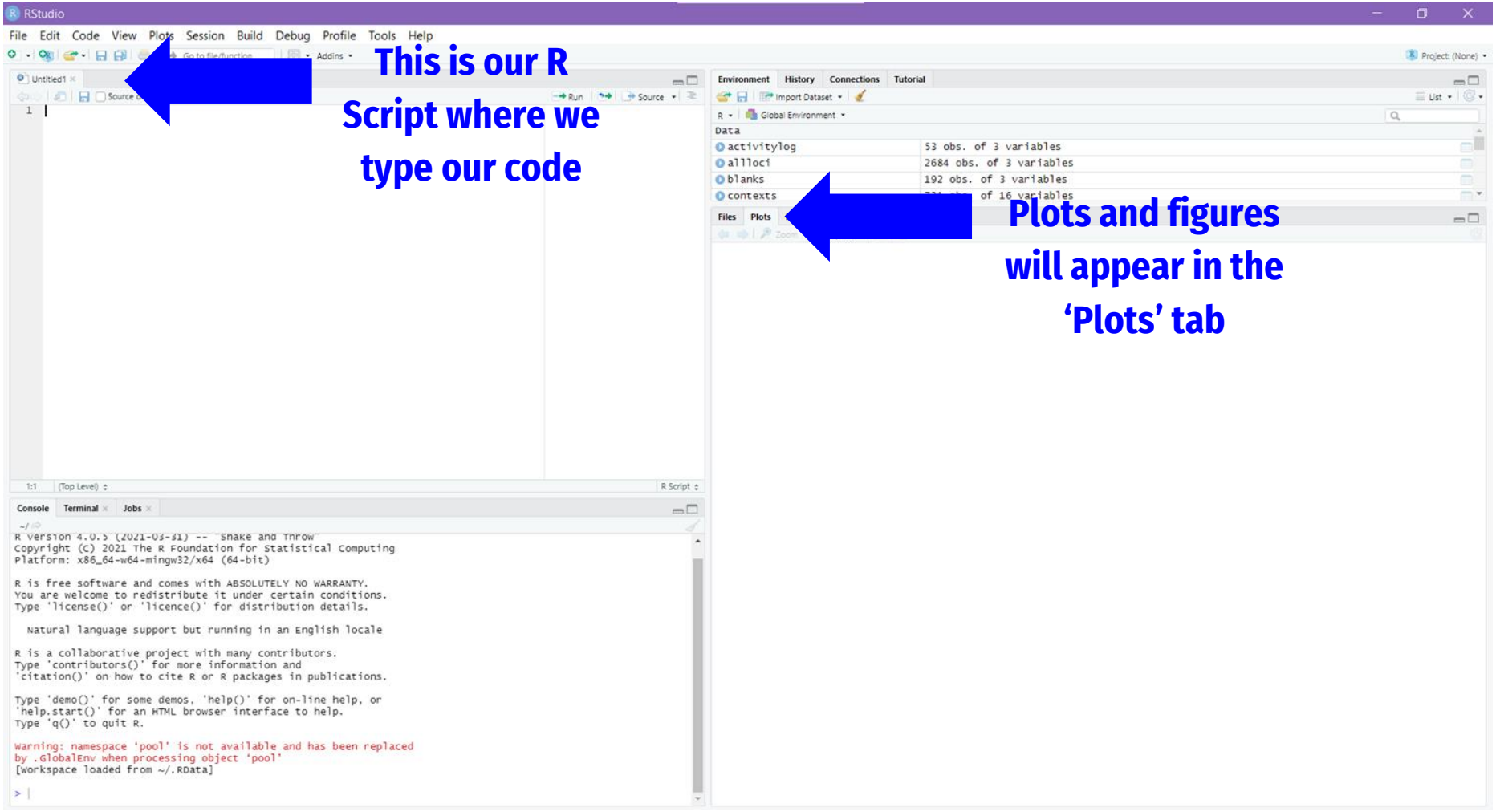
Data Sciences Institute, University of Toronto

In this lesson, we will...

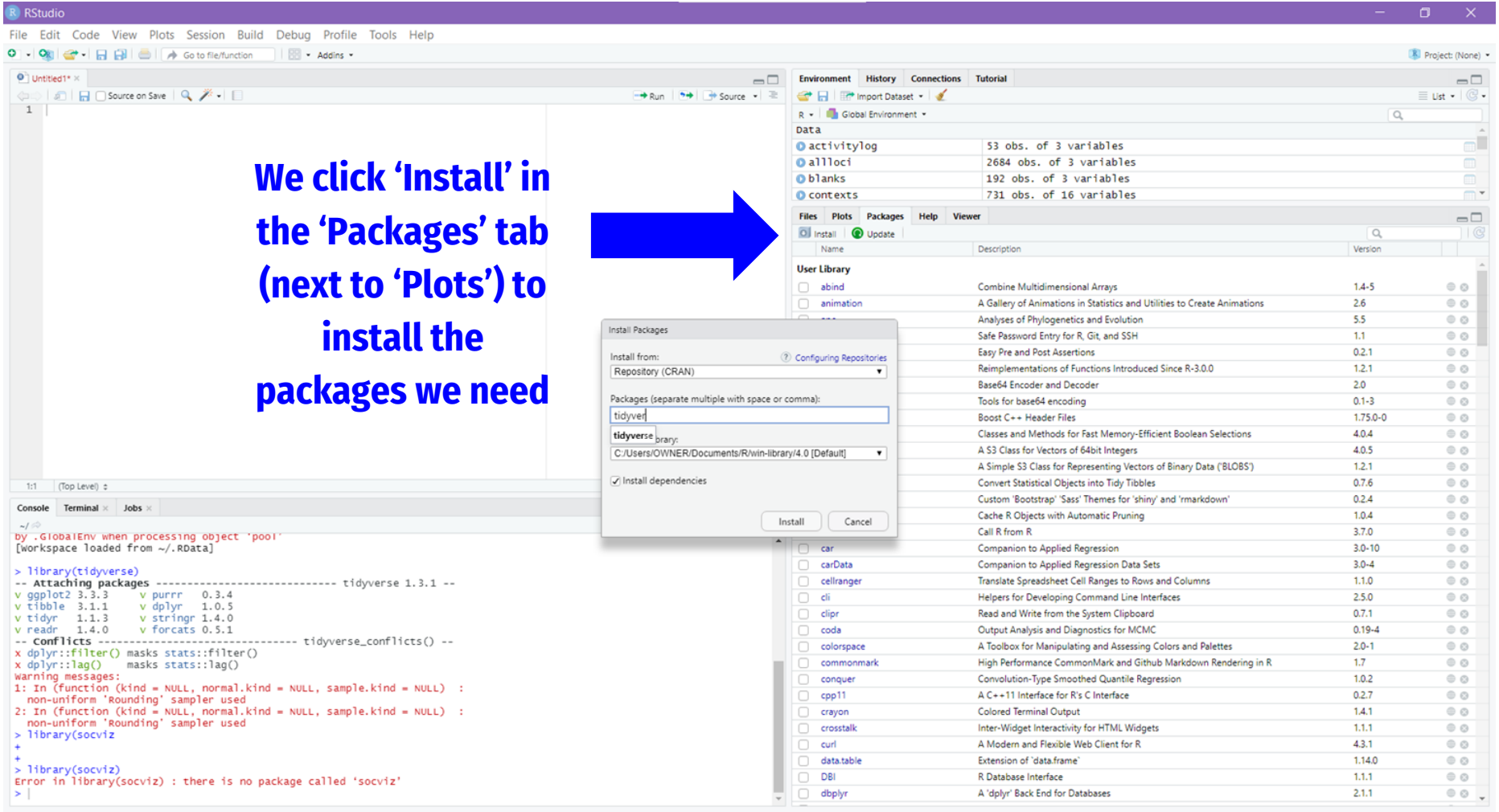
- Work through practice code from Chapter 2 (*Getting Started*) of Healy, K. (2018). Data Visualization: A Practical Introduction. Princeton University Press. This code will let us:
  - Set up our workspace in RStudio
  - Get the data we want to visualize into RStudio
  - Make our first data viz with RStudio!

Setting up R for data visualization


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
New File
New Project...
Open File... Ctrl+O
Open File in New Column...
Recent Files
Open Project...
Open Project in New Session...
Recent Projects
Import Dataset
Save Ctrl+S
Save As...
Save All Ctrl+Alt+S
Publish...
Print...
Close Ctrl+W
Close All Ctrl+Shift+W
Close All Except Current Ctrl+Alt+Shift+W
Close Project
Quit Session... Ctrl+Q
R Script Ctrl+Shift+N
R Notebook
R Markdown...
Shiny Web App...
Plumber API...
C File
C++ File
Header File
Markdown File
HTML File
CSS File
JavaScript File
D3 Script
Python Script
Shell Script
SQL Script
Stan File
Text File
R Sweave
R HTML
R Presentation
R Documentation...
Project: (None)
Environment History Connections Tutorial
Import Dataset
List
R
Global Environment
Data
activitylog 53 obs. of 3 variables
allloci 2684 obs. of 3 variables
blanks 192 obs. of 3 variables
contexts 731 obs. of 16 variables
Files Plots Packages Help Viewer
Zoom
Export


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Untitled1
Source
Run
Source
1:1
(Top Level)
R Script
Console Terminal Jobs
R version 4.0.5 (2021-03-31) -- "Shake and Throw"
Copyright (C) 2021 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)
R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.
Natural language support but running in an English locale
R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.
Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.
Warning: namespace 'pool' is not available and has been replaced
by .GlobalEnv when processing object 'pool'
[Workspace loaded from ~/.RData]
Environment History Connections Tutorial
Import Dataset
Global Environment
Data
activitylog 53 obs. of 3 variables
allloci 2684 obs. of 3 variables
blanks 192 obs. of 3 variables
contexts
Files Plots
This is our R Script where we type our code
Plots and figures will appear in the 'Plots' tab

We click 'Install' in the 'Packages' tab (next to 'Plots') to install the packages we need



Load required packages

- Once our packages have been installed, type the following in your script:

```
library(tidyverse)
library(socviz)
library(ggplot2)
```

- ctrl+enter (Windows) or cmd+enter (Mac) over highlighted code to run it and load the libraries

(Healy, 2018)

Load required packages

- We will (later!) be using the ggplot library to make our visualizations

```
library(ggplot2)
```

Getting our data into R

For our purposes...

- In this class, we will be using already-prepared datasets from the socviz library rather than separate data files
- BUT when visualizing data for your own purposes, you will most likely need to read in data from flat files
- We do this using the `read_csv()` function from the readr package to assign our dataset to an object from our computer or from a URL:

```
ourdata <- read_csv(/location/of/file.csv)

ourdata <- read_csv(file="fakewebsite.com/file.csv")
```

(Healy, 2018)

Preparing our data for visualizing

- For graphing with ggplot (AKA what we want to do), our data should be in long format as opposed to wide format
- That is: **in long formatted data, every observation should be a row, and every variable should be a column**

(Healy, 2018)

Make a basic figure

Load our sample dataset

- We will use a preloaded sample dataset called ‘gapminder’, which can be loaded as follows:

```
install.packages("gapminder")

library(gapminder)

gapminder
```

- We can see that our dataset contains data about countries over several years

(Healy, 2018)

Make a scatterplot

- To make some sense of what is happening in our dataset, we can make a simple scatterplot as follows:

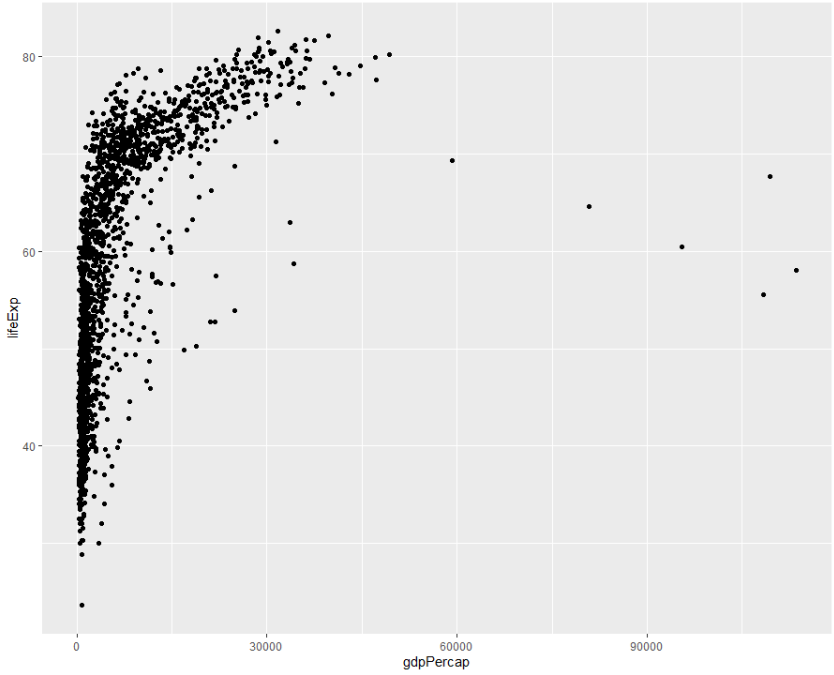
```
p <- ggplot(data = gapminder,
            mapping = aes(x = gdpPercap, y = lifeExp))
p + geom_point()
```

- (Don’t worry about the code yet, we’ll break it down later)

(Healy, 2018)

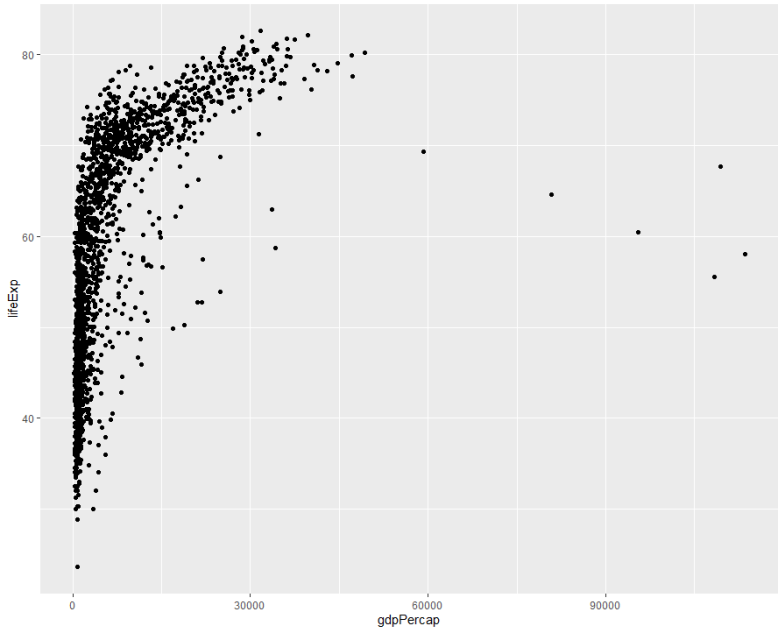
Make a scatterplot

- If we run the code, we should see an output in our ‘plots’ pane:



(Healy, 2018)

Assess our first plot



The Good	The Not-So-Good
• Mostly legible • Has axis labels • Shows some kind of relationship between variables	• Font size is tiny • Not visually interesting or pleasing • What are we trying to show?

Next...

- Making a plot
- Understanding the different components of the ggplot we made in this class