Lab 1 - Introduction to R

# Introduction

This week’s lab is an introduction to R and how to import data. To run this week’s lab, you will need to download the **bfro\_reports\_geocoded.csv** file available on D2L. For credit, answer the questions and upload them to the associated D2L quiz. If you would like, you may take the D2L quiz twice and use it to check some of your answers - just make sure you save your written answers somewhere else as D2L will not save those for you if you retake the quiz. Your highest score will be kept.

Please note that if you do not understand something in this lab, you should talk to your section leader or your professor. This week’s lab covers essential pieces of how to use R, and you want to make sure you use this week to master these so that the labs in the future are not harder than they need to be.

# Learning Outcomes

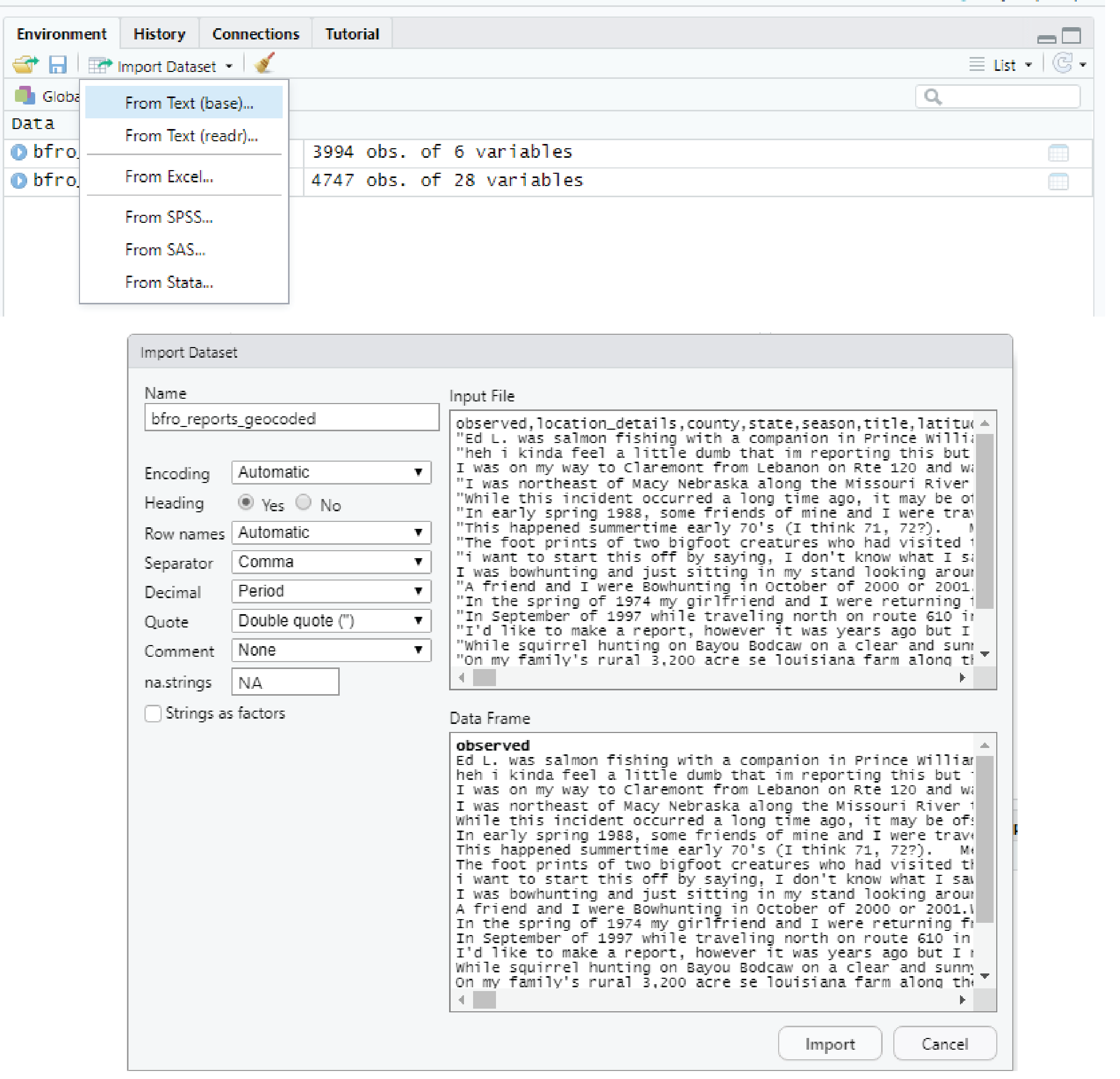
By the end of this lab you should be able to:

* Import data into R using the import dataset button
* Name new dataframes and vectors using = or <-
* Assign columns as vectors in R
* Identify your filepath using the import dataset button
* Use the hist() function to make a basic graphic of your data
* Identify and fix basic errors in your code

# Part 1 - Load and Rename Data

## Part 1.1 Load Data

In the upper right hand corner there is a button called **Import Dataset** under the Environment tab. Click the drop down menu, and click **From Text (base)**. That will pop up a window that gives you a lot of options (encoding, heading, etc).

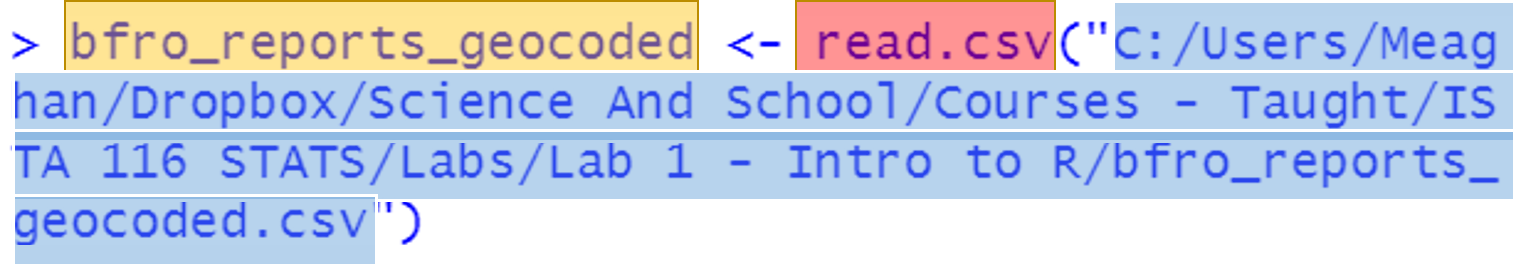


The most important option here to note is the **Heading** section. Make sure that is always on Yes! But otherwise, don’t change any of the other options, just hit **Import**.

You now should have an object in your environment window called *bfro\_reports\_geocoded*. This is a list of Bigfoot sightings across North America. We’re going to use it to answer some questions about Bigfoot’s preferred climate. If you look in the console, you’ll see that R told you some information about what it ran in order to import this data. This information is important, so we’ll take a look at it!

## Part 1.2 Understanding Data Import

Here’s an example of what R outputs for your instructor when she runs this code with some extra color coding:



There are three parts here that are very important:

* Yellow: the name of our R Object
* Red: the code used to help import our data and then view it, also called functions
* Blue: the full **filepath**, or location of the file

These are important because in future labs, you will not be using the Import Dataset Button. Instead, you will be importing your data using code, because in the long run it is a thousand times faster and prone to fewer mistakes. In your professor’s experience, the part that students struggle with the most is recognizing what their filepath is. Filepaths in R:

* Must have forward slashes, not backwards (so / not \ )
* Are enclosed in quotation marks (")
* Have the name of the file (brfo\_reports\_geocoded)
* Have the file extension (.csv)

Some of you may be working on Macs and the filepath that R will spit out may have a tilde (~) as an abbreviation. That’s fine!

### QUESTION 1: Copy and paste your full filepath, with quotations, to demonstrate that you recognize where your file is on your computer.

## Part 1.3 Rename Data

The name of our data is really long, so we’re going to quickly rename it here. There are plenty of ways to rename data in R. We’re going to actually copy the file and rename it in the process, using what are called **assignment operators** in R. Assignment operators just tell R “take this, copy it, and rename it as this.” There are two assignment operators in R: = and <- . To copy and rename, you put the new name on the left, use the operator of your choice, and the thing you’re renaming on the right. Like so:

Bigfoot <- bfro\_reports\_geocoded  
Bigfoot = bfro\_reports\_geocoded

You may use the assignment operator of your choice. Your professor prefers to use the arrow, because it reminds you that the new thing is on the left, but if the equals sign makes more sense to you that is fine! Some rules for naming objects in R:

* They can be any number of characters, starting at one (so x, xxxxxxxxxxx, and xyasksdfsdf are all fine names)
* They can’t start with a number (i.e. 2X can’t be a name)
* You can use \_ or . in your names, just not at the start (i.e. B.B is fine, but .BB is not)
* Capitalization is important (i.e. X is not the same thing as x)
* If you name it the same as something that is already in your environment, it will overwrite your data
* Rude names are fine, and often cathartic - half of your instructor’s scripts are filled with things called poop and various curse words

### QUESTION 2: Which chunk of code would you use if you wanted to name your dataframe WowCoolFun:

A. WowCoolFun == bfro\_reports\_geocoded  
B. WowCoolFun <- bfro\_reports\_geocoded  
C. bfro\_reports\_geocoded = WowCoolFun  
D. WowCoolFun = "bfro\_reports\_geocoded"

# Part 2 - Explore Data

Spreadsheets in R are typically imported into a type of R data called a **dataframe**. These have column names, row names, and data organized just like a typical spreadsheet. You can click on the data in the Environment window and see all of this data there, but there are also ways of calling out this data in R.

## Part 2.1 Column Names

First, let’s figure out what the column names are. To do this, you will be using a **function** in R. Functions are like verbs, or action words - they tell R to do something. A function is a word with parentheses at the end - you put the thing you’re interested in inside the parentheses. This makes a full sentence in R!

The function for figuring out column names is colnames(). You run it like so:

colnames(Bigfoot)  
colnames(bfro\_reports\_geocoded)

This spits out a list of column names in your console. You’ll notice some of them are really long, and have specific abbreviations. Because R is VERY literal, if you don’t spell things exactly or have the exact right capitalization you won’t be able to use the column. So using colnames() makes it so you can copy the column names and paste them.

Columns are not actually called columns in R, however. If you look in the environment window, you’ll see that the dimensions of this dataframe are listed.

### QUESTION 3: This data set has 28 columns. Look at the environment window - what is the word for “columns” in R?

1. obs.
2. variables
3. chr.
4. num.

### QUESTION 4: 4747 people reported sightings of sasquatch. In excel, that means that this data set has 4747 rows. What does R call rows?

1. obs.
2. variables
3. chr.
4. num.

## Part 2.2 Use Dollar Sign to Call Out Columns

In this dataset there is a column that contains the state of the siting. But to look at it, you can’t just type “state” into R and have it understand that’s the data you’re interested in. This is because often you are dealing with many different dataframes in R, so you need to tell it two pieces of information: the column name, but also what dataframe it’s pulling it from. To do that, you use **$**, with the dataframe on the left and the column name on the right. Like so:

Bigfoot$state  
bfro\_reports\_geocoded$location\_details

The helpful thing about RStudio is that it will give you auto-fill options as well. So if you type Bigfoot$ in the console or the script window, it will give you a list of the column names that you can click on. Convenient!

### QUESTION 5: Which of the following would call out the pressure column?

A. Bigfoot$Pressure  
B. pressure  
C. pressure$Bigfoot  
D. Bigfoot$pressure

## Part 2.3 Save Column as Variable

But it is still a lot to type some of these things over, and over, and over again. So sometimes it can be easier to store this information as a **vector** in R. That means you just name the column as an object in R, using assignment operators just like we did with the name of the dataframe itself. You can name these according to the same rules, as well. For example:

x <- Bigfoot$state  
state <- Bigfoot$state  
pooooooopturd <- Bigfoot$state

Then, if you want to call them out for some reason, you just type the name of your vector into R:

x  
state  
pooooooopturd

NOTE: If you did indeed use the extremely mature third option name, you may notice that the number of O’s is obnoxious to type. This means it’s probably not a very good object name, because it’s easy to misspell, even if R allows it and it is very funny to your professor’s 3rd-grader level humor.

### QUESTION 6: Which of the following would save the column dew\_point as a vector called DP?

A. DP <- dew\_point  
B. DP <- Bigfoot$DP  
C. DP <- Bigfoot$dew\_point  
D. Bigfoot$dew\_point -> DP

# Part 3 - Plot Data

For the last part of this lab we’re going to show you how to make a few basic plots to explore this Bigfoot sighting data. Specifically, we’re going to look and see if there is a specific temperature or precipitation that is most common for Bigfoot sightings, with the thought that perhaps these sightings are made by people who are either overheating or perhaps can’t tell a wet bear from a Sasquatch.

## Part 3.1 Name Variables

To start with, we’re going to rename our variables so they are easier to call out. We are using temperature\_high and precip\_intensity as our variables.

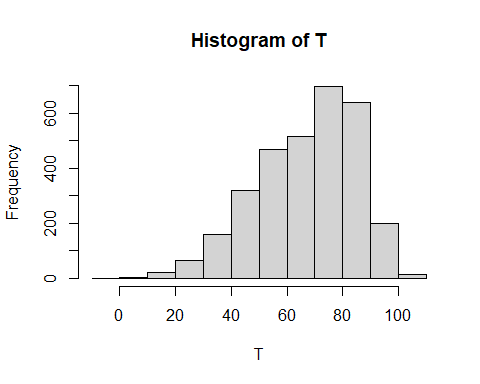
T <- Bigfoot$temperature\_high  
R <- Bigfoot$precip\_intensity

NOTE: You’ll notice that many of the values in this vector are “NA”. That is how R interprets missing data. They won’t interfere with what we are doing, but when we get a little more into our data sometimes NA values can be a problem. It’s always good practice to check out your data for things like NA values.

## Part 3.2 Make a Histogram

A histogram is a display of one variable of data that shows how common each possibility is. The y axis on your histogram is how much of your data falls into each bar, and the x axis is what that data represents. To run a histogram in R, use the hist() function, like so:

hist(T)



You can see that the most common temperatures that people are seeing Bigfoot at is about 70-80 degrees. It isn’t specified but since this is an American dataset, we are assuming this is in Fahrenheit - and while it isn’t impossible to get heatstroke and hallucinate Bigfoot at these temperatures, it’s not super likely. So what about precipitation? Use the hist() function again to look at precipitation intensity:

hist(R)

Weird looking graph, right? Intensity of precipitation is a measurement of how much rain is falling over time - a number near 0 means that even if it is raining, it’s more of a sprinkle than anything. So our theory that people can’t see Black Bears well when it is raining isn’t a good explanation for Sasquatch sightings.

Which means that Bigfoot is real, right??? Well, no, not necessarily. But for the purposes of this lab, we certainly can’t disprove it.

### QUESTION 7: If you named a vector of dew point as DP, which code would make a histogram of dew point?

A. hist(DP)  
B. hist(Bigfoot$DP)  
C. hist("DP")  
D. hist DP;

While renaming variables is convenient, you can run histograms on data frame columns without doing that. For example:

hist(Bigfoot$wind\_speed)

# Part 4. Quiz Time

Here are a few more questions to make sure that you understand the content of this lab, since this lab is fundamental to your success going forward in R.

### QUESTION 8: Bigfoot$Cloud\_Cover returns an error that just says “NULL.” What’s wrong with this code?

### QUESTION 9: "C:/Users/Meaghan/Dropbox/Science And School/Courses - Taught/ISTA 116/2020 Labs/Lab 1 - Intro to R/bfro\_reports\_geocoded.csv" is your instructor’s filepath. What is the name of the folder on this computer that contains the csv file for this lab?

### QUESTION 10: "C:\Users\Meaghan\Dropbox\Science And School\bfro\_reports\_geocoded.csv" is not a file path that R would recognize. What is wrong with it?

### QUESTION 11: Which of these is an appropriate name for a data frame?

1. 1Bigfoot
2. Bigfoot1\_.
3. \_.1Bigfoot
4. .\_1\_

That’s it! Please upload your answers to the associated D2L quiz by the deadline, and reach out to your section leaders or professor if you have any issues or questions. Also, feel free to read through some of the answers in the ‘observed’ column of this dataset… some are pretty funny.