Week 2: tidyverse, normal probabilities, ggplot

Monica Alexander

16/01/2022

By the end of this lab you should know

- How to read in data from a csv file
- How to view a dataset in R
- What the pipe `%>%` is
- The functions `select`, `arrange`, `filter`, `mutate` and `summarize`
- How to calculate probabilities for the Normal distribution
- Know what `ggplot` is a do a first plot

Load in the `tidyverse` package

The first thing we need to do (and the first thing you will usually need to do) is to load in the `tidyverse` R package

```
library(tidyverse)
```

Reading in data

We now want to read in the GSS data:

```
# make sure the file name points to where you've saved the gss file
# for example, I have it saved in a "data" folder
gss <- read_csv(file = "../data/gss.csv")
```

You can look at the gss file by going to the "Environment" pane and clicking on the table icon next to the gss object, or by typing `View(gss)` into the console.

Important functions

This section illustrates some important functions that make manipulating datasets like the gss dataset much easier.

select

We can select a column from a dataset. For example the code below selects the column with the respondents age:

```
select(gss, age)
```

```
## # A tibble: 20,602 x 1
##      age
##    <dbl>
##  1  52.7
##  2  51.1
##  3  63.6
##  4  80
##  5  28
##  6  63
##  7  58.8
##  8  80
##  9  63.8
## 10  25.2
## # ... with 20,592 more rows
```

```
select(gss, age, education)
```

```
## # A tibble: 20,602 x 2
##      age education
##    <dbl> <chr>
##  1  52.7 High school diploma or a high school equivalency certificate
##  2  51.1 Trade certificate or diploma
##  3  63.6 Bachelor's degree (e.g. B.A., B.Sc., LL.B.)
##  4  80   High school diploma or a high school equivalency certificate
##  5  28   College, CEGEP or other non-university certificate or di...
##  6  63   High school diploma or a high school equivalency certificate
##  7  58.8 Less than high school diploma or its equivalent
##  8  80   Less than high school diploma or its equivalent
##  9  63.8 High school diploma or a high school equivalency certificate
## 10  25.2 Less than high school diploma or its equivalent
## # ... with 20,592 more rows
```

The pipe

Instead of selecting the age column like above, we can make use of the pipe function. This is the %>% notation. It looks funny but it may help to read it as like saying "and then". On a more technical note, it takes the first part of code and *pipes* it into the first argument of the second part and so on. So the code below takes the gss dataset AND THEN selects the age column:

```
gss %>%
  select(age)
```

```
## # A tibble: 20,602 x 1
##      age
```

```
##    <dbl>
##  1  52.7
##  2  51.1
##  3  63.6
##  4  80
##  5  28
##  6  63
##  7  58.8
##  8  80
##  9  63.8
## 10  25.2
## # ... with 20,592 more rows
```

Notice that the commands above don't save anything. Assign the age column to a new object called `gss_age`

```
gss_age <- gss %>% select(age)
gss_age
```

```
## # A tibble: 20,602 x 1
##      age
##    <dbl>
##  1  52.7
##  2  51.1
##  3  63.6
##  4  80
##  5  28
##  6  63
##  7  58.8
##  8  80
##  9  63.8
## 10  25.2
## # ... with 20,592 more rows
```

arrange

The `arrange` function sorts columns from lowest to highest value. So for example we can select the age column then arrange it from smallest to largest number. Note that this involves using the pipe twice (so taking gss AND THEN selecting age AND then arranging age).

Side note: you need not press enter after each pipe but it helps with readability of the code.

filter

To filter rows based on some criteria we use the `filter` function. e.g. filter to only include those aged 30 or less:

Filter takes any logical arguments. If we want to filter by participants who identified as *Female*, we use `==` operator.

mutate

We can add columns using the mutate function. For example we may want to add a new column called `age_plus_1` that adds one year to everyone's age:

`summarize`

The `summarize` function is used to give summaries of one or more columns of a dataset. For example, we can calculate the mean age of all respondents in the gss:

Review questions

1. Create a new R Markdown file for these review questions
2. Find the mean age at first birth (age_at_first_birth) of respondents in the GSS.
3. Create a new dataset that just contains GSS respondents who are less than 20 years old.
4. How many rows does the dataset in step 4 have?
5. What is the largest case id in the dataset in step 4?

Calculating Normal probabilities

We can calculate the cumulative probability that a value coming from a Normal distribution with a certain mean and standard deviation is below a certain value using the `pnorm` function. E.g. what the probability that a value is less than -1.25 from a standard Normal distribution (mean 0 and standard deviation 1) is

```
pnorm(-1.25, mean = 0, sd = 1)
```

```
## [1] 0.1056498
```

Questions

- What's the probability that a value is above -1.25?
- What's the probability that a value is between -1.25 and 1.25?
- Pretend that population heights are Normally distributed with mean 160 and standard deviation 10 (in cm). Find the probability that a randomly selected person has a height less than yours?

My first `ggplot`

`ggplot` is a powerful visualization package. It provides many options to make beautiful graphs, maps, plots of all sort.

We are going to make a histogram of ages at first marriage in the gss dataset.