Intro to Data Visualization in R

Exercise files

Download all exercise files at
http://www.github.com/summerofgeorge/rtraining/.

Introducing ggplot2 for graphics

In this unit, we will be using the package ggplot2, part of the tidyverse, to create plots for exploring our data.

We will again use the hsbraw.csv from the UCLE IDRE group. Let's load the tidyverse and get to it.

This lesson is based on IDRE's training module at https://stats.idre.ucla.edu/stat/data/intro_r/intro_r_flat.html.

```r
library(tidyverse)
d<-read_csv("C:/RFiles/hsbraw.csv" )
```

Basic syntax of a ggplot2 plot

The basic specification for a ggplot2 plot is to specify which variables are mapped to which aspects of the graph (called aesthetics) and then to choose a shape (called a geom) to display on the graph.

For example, we can choose to map one variable to the x-axis, another variable to the y-axis, and to use geom_point() as the shape to plot, which produces a scatter plot.

what `ggplot2` wants

Within the `ggplot()` function we specify (Note that the package is named `ggplot2` while this function is called `ggplot()`):

- the dataset

what ggplot2 wants

Within the `ggplot()` function we specify (Note that the package is named `ggplot2` while this function is called `ggplot()`):

- the dataset
- inside an `aes()` function, we then specify which variables are mapped to which aesthetics, which can include:

what ggplot2 wants

Within the `ggplot()` function we specify (Note that the package is named `ggplot2` while this function is called `ggplot()`):

- the dataset
- inside an `aes()` function, we then specify which variables are mapped to which aesthetics, which can include:
- x-axis and y-axis

what `ggplot2` wants

Within the `ggplot()` function we specify (Note that the package is named `ggplot2` while this function is called `ggplot()`):

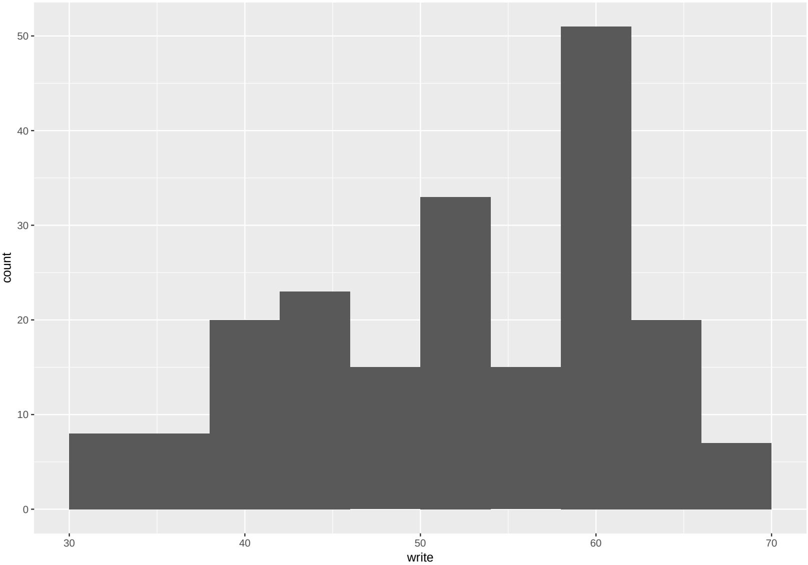
- the dataset
- inside an `aes()` function, we then specify which variables are mapped to which aesthetics, which can include:
- x-axis and y-axis
- color, size, and shape of objects

Exploring continuous variables: Histograms

Histograms bin continuous variables into intervals and count the frequency of observations in each interval.

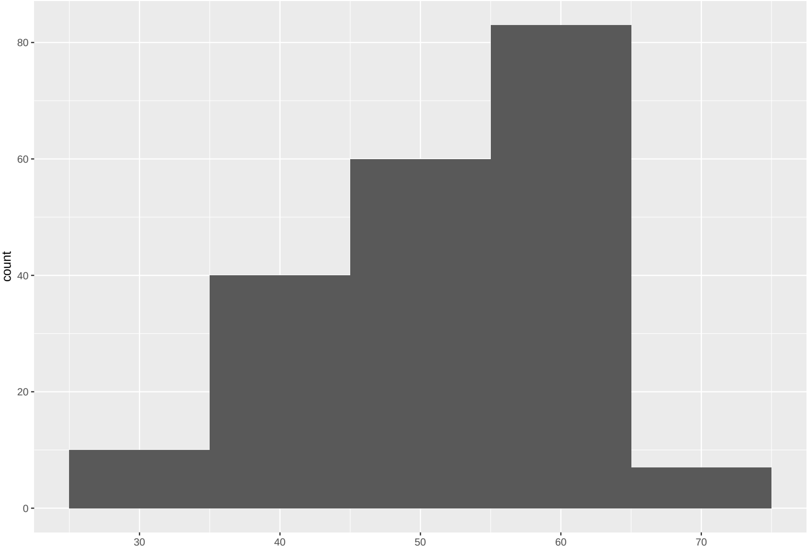
For histograms and density plots, we map the variable of interest to `x`.

```r
#use the bins= argument to control the # of intervals
ggplot(d,aes(x=write))+geom_histogram(bins=10)
```



To change the width of the bin in the histogram we can use `bidwith` in geom_histogram:

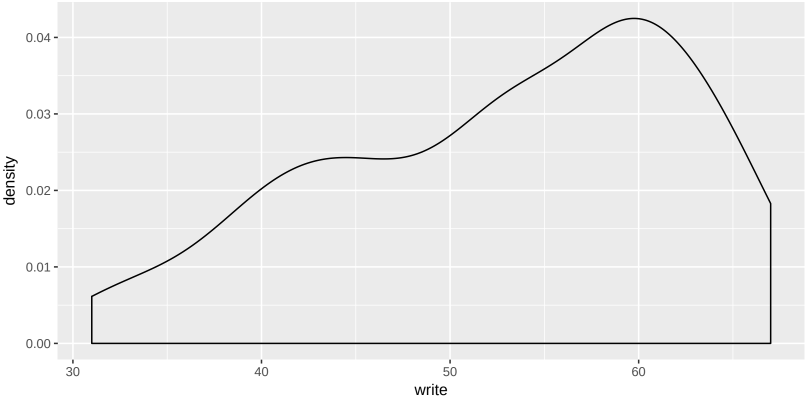
```
ggplot(d, aes(x=write))+geom_histogram(binwidth=10)
```



Exploring continuous variables: Density plots

Density plots smooth out the shape of histograms.

```
ggplot(d, aes(x=write))+geom_density()
```

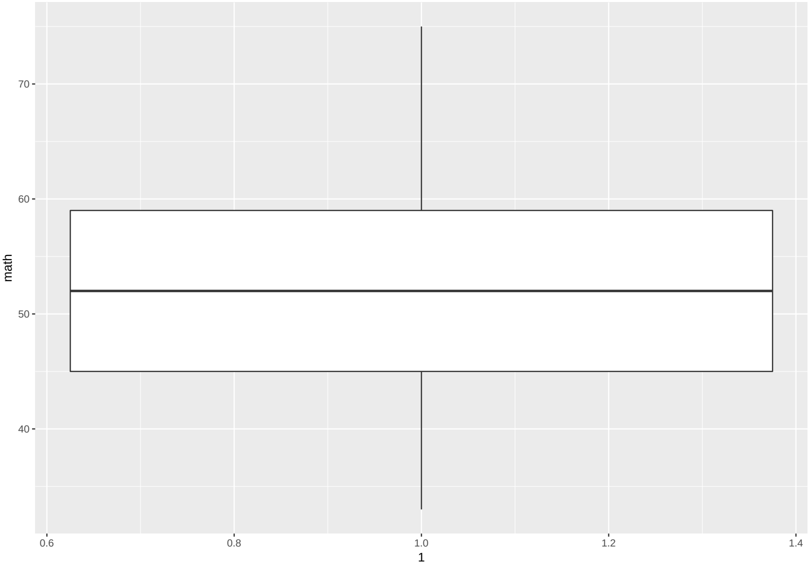


Exploring continuous variables: boxplots

Boxplots show the median, lower and upper quartiles and outliers.

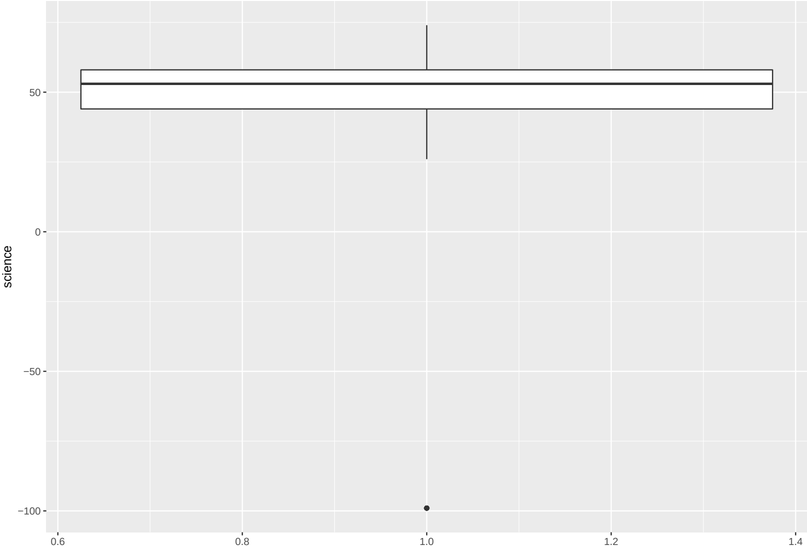
Unlike histograms and desntiy plots, we want to map the variable of interest to `y` instead of `x`. If we are making a single boxplot, we need an arbitrary value for `x`, just as a place holder.

```r
#for the overall distribution of one variable, specify x=1
ggplot(d, aes(x = 1, y = math)) + geom_boxplot()
```



Data exploration can help us identify suspicious looking values.

```
#for the overall distribution of one variable, specify x=1
ggplot(d, aes(x = 1, y = science)) + geom_boxplot()
```



Exploring categorical variables.

For categorical variables, summary statistics such as mean, median and variance cannot be calculated meaningfully.

Instead, we will use frequency tables to summarize the distribution of each category using the `table` function.

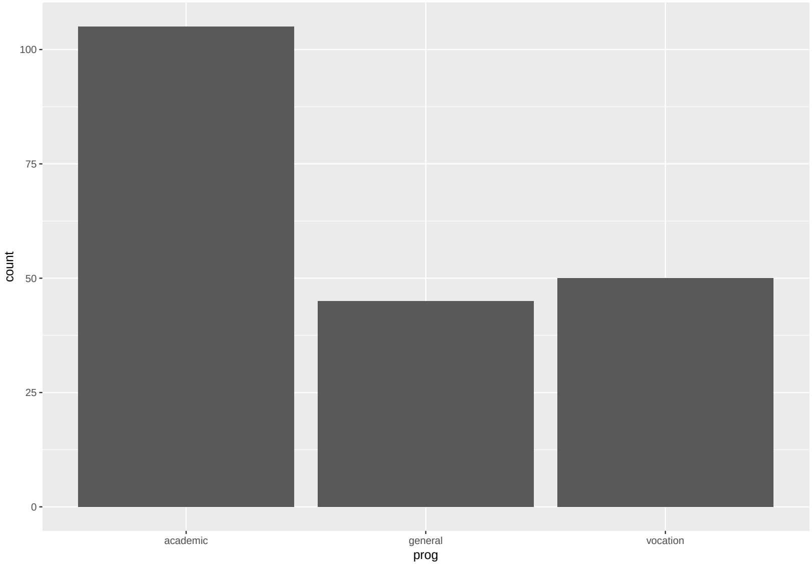
Use `prop.table` on the table provided by `table` to see frequencies stated in proportions.

```
#table() produces counts

table(d$female)
##
## female   male
##    109     91

#for proportions, use prop.table(table())

prop.table(table(d$female))
##
## female   male
##  0.545  0.455
```

Exploring categorial vars: bar graphs

Distributions of categorical variables are often depicted by bar graphs, which are easily made in `ggplot2`. By default, geom_bar() counts the number of observations for each value of the variable mapped to `x`.

```r
ggplot(d, aes(x=prog))+geom_bar()
```



Exploring relationships between two variables

After inspecing distributions of variables individually, we proceed to explore relationships between variables.

In particular, we want to examine whether the values of one variable might be associated with another.

We will use different numerical and graphical methods for exploration depending on whether the variables are both continuous, both categorical, or one of each.

Exploring continuous by continuous numerically

Correlations provide quick assesssments of whether two continuous variables are linearly related to one another.

The `cor()` function estimates correlations. If supplied with two vectors, `cor` will estimate a single correlation. If supplied a data frame with several variables, `cor` will estimate a correlation matrix.

```
cor(d$write, d$read)
## [1] 0.5967765

scores<-d[,c("read","write","math","science","socst")]
cor(scores)
##              read     write      math   science      socs
## read    1.0000000 0.5967765 0.6622801 0.1709428 0.181492
## write   0.5967765 1.0000000 0.6174493 0.1289845 0.150458
## math    0.6622801 0.6174493 1.0000000 0.2051668 0.189864
## science 0.1709428 0.1289845 0.2051668 1.0000000 0.936167
## socst   0.1814928 0.1504587 0.1898648 0.9361672 1.000000
```

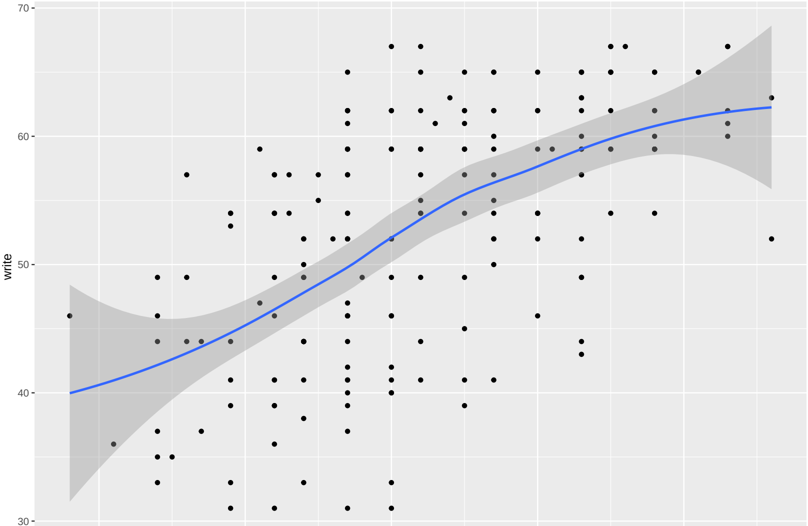
Exploring continuous by continuous graphically

Scatter plots are an obvious choice to depict the relationship between two varialbes. We can also add a loess smoothing plot (geom_smooth()) that provides a best-fit curve to the data.

Note that the further layers are added with a +.

Here we examine the relationship between reading and writing test scores.

```
ggplot(d,aes(x=read, y=write))+
  geom_point()+
  geom_smooth()
## `geom_smooth()` using method = 'loess'
```



Exploring continuous by categorical: grouping data frames

When exploring the relationship between a continuous and a categorical variable, we are often interested in whether the distribution of the continuous variable is the same between classes.

For example, we might wnt to know whether the means and variances of math test scores are the same between males and females.

For this, we will group them like in `dplyr`

```r
by_female<-group_by(d,female)
```

Then we will use summarize to get the means and variances of math by gender:

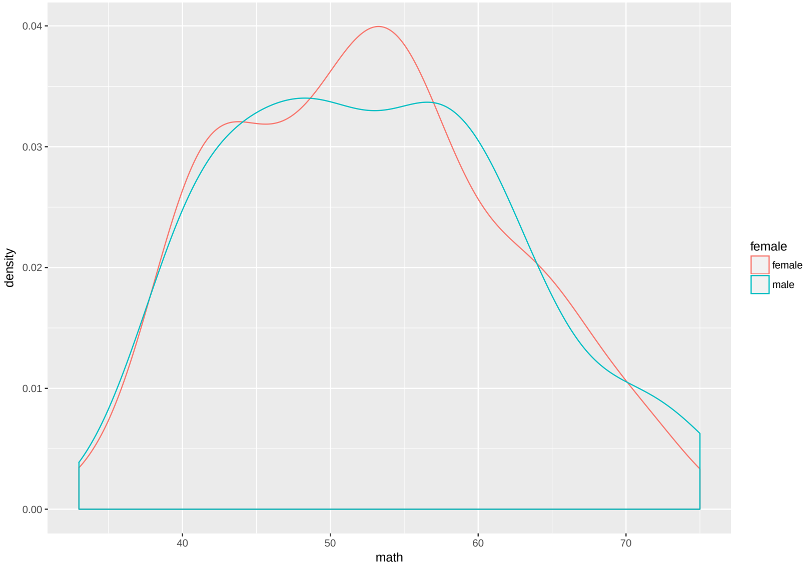
```
summarize(by_female,mean(math),var(math))
## # A tibble: 2 x 3
##   female `mean(math)` `var(math)`
##   <chr>         <dbl>       <dbl>
## 1 female         52.4        83.7
## 2 male           52.9        93.4
```

Exploring continuous by categorical graphically

To plot distribtuions of the continuous varialbes by gorups defined by the categorical variables, we will plot separate density plots of the continuous varialbes for each group of the categorical variable.

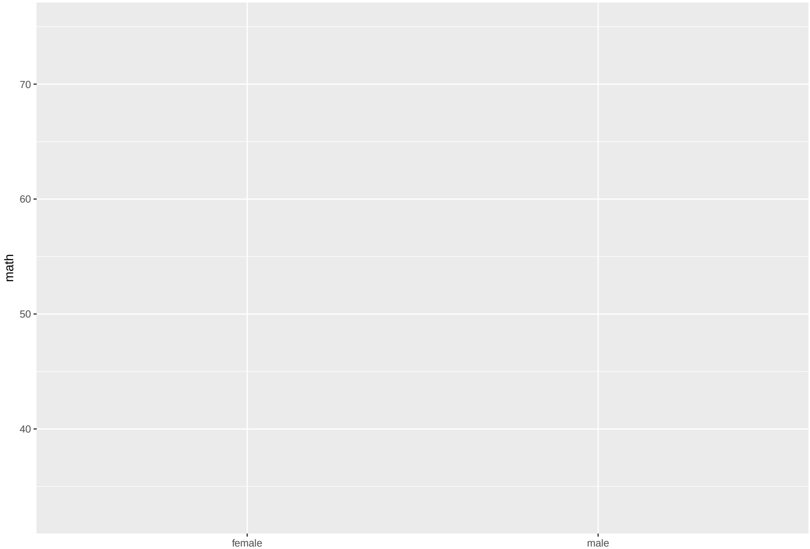
The grouping variable is commonly mapped to aesthetics that take on categories themselves, such as `color` or `shape` but can be mapped to `x` as well if it is numeric.

```
ggplot(d,aes(x=math,color=female))+
  geom_density()
```



Boxplots of math by female show the same similar-looking distributions.

```r
ggplot(d,aes(x=female,y=math))
```



Exploring categorical by categorical numerically.

Two-way and multi-way frequency tables are used to explore the relationships between categorical variables.

We can use `table` and `prop.table` again. With `prob.table` use `margin=1` for row proportions and `margin=2` for column proportions. Omitting `margin=` will give proportions of the total.

Here, we check whether the proportions of observations that fall into each education program are about the same across socioeconomic statuses.

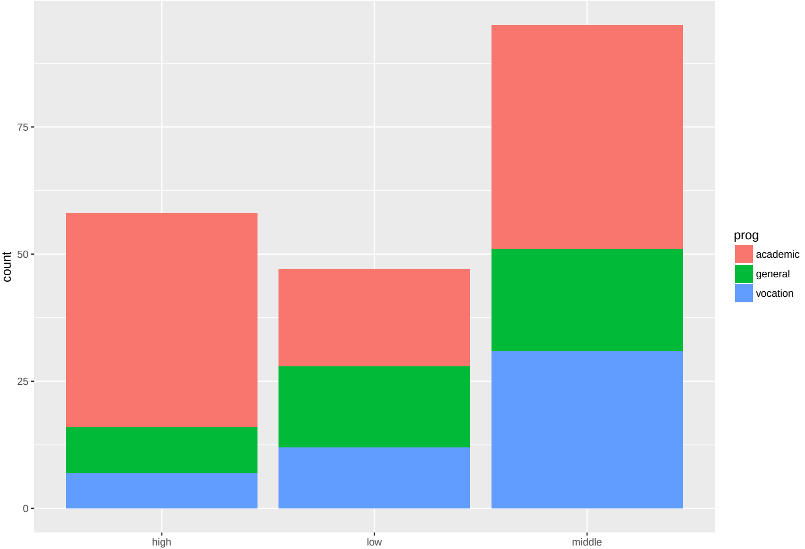
```
my2way<-table(d$prog,d$ses)

#counts in each crossing of prog and ses
my2way
##
##            high low middle
##   academic   42  19     44
##   general     9  16     20
##   vocation    7  12     31
```

Exploring categorical by categorical graphically

We can add a categorical variable to the bar graph of the other categorical variable to depict their relationship.

Here we map prog to `fill`, the color used to fill the bars of the bar graph.

```
ggplot(d,aes(x=ses,fill=prog)) +geom_bar()
```



Questions?