Session 14
Data Visualization 2

R for Social Data Science

Jeffrey Ziegler, PhD

Assistant Professor in Political Science & Data Science
Trinity College Dublin

Fall 2022

Road map for today

Last time:

- Plotting in base R

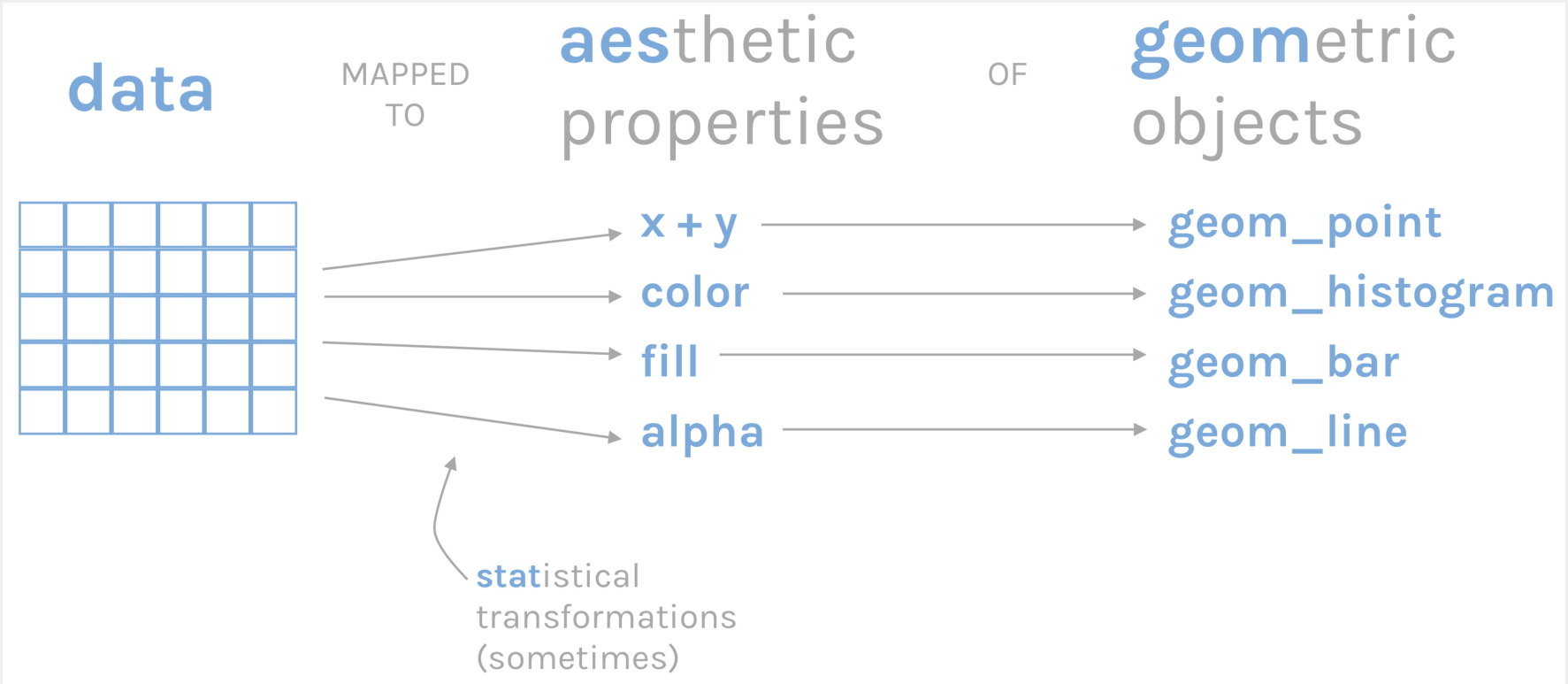
This time:

- Plotting in `ggplot`

GGPLOT BASICS

- Similar to base plot, we open our plot using `ggplot()`, in which the arguments are:
  - Data frame containing data to be plotted
  - Mapping of variables to visual properties of graph
  - Mappings are placed within ‘aes‘ function (where ‘aes‘ stands for aesthetics)

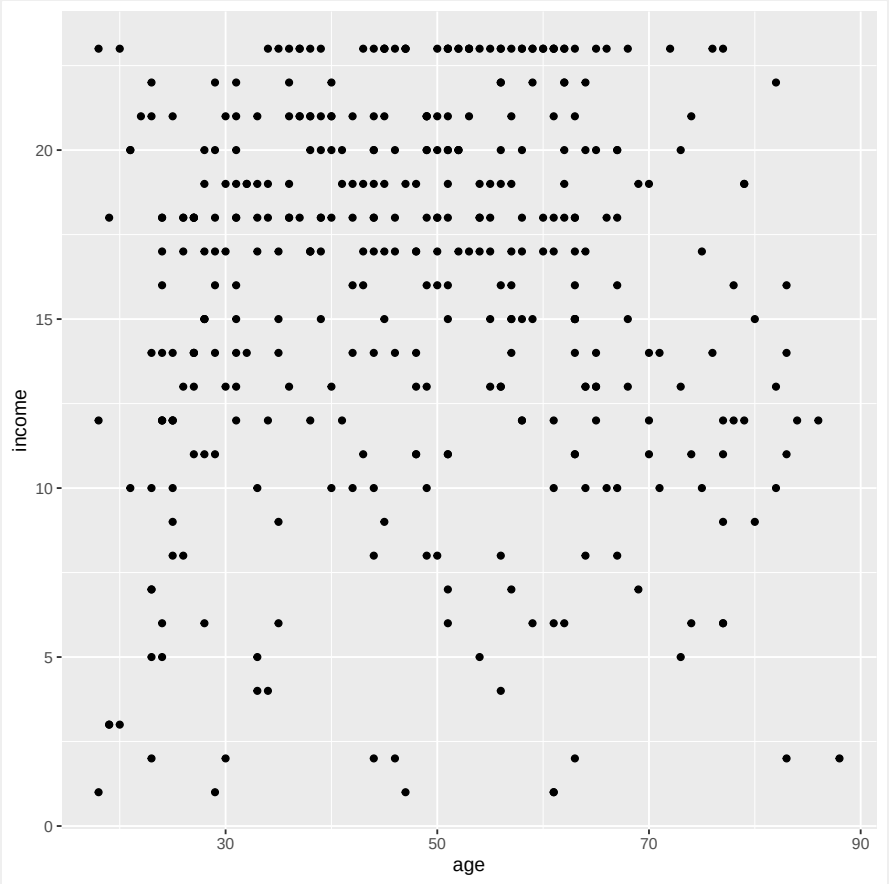
Grammar of Ggplot graphics



Scatterplots

- Scatter plot can be created using `geom_point()`

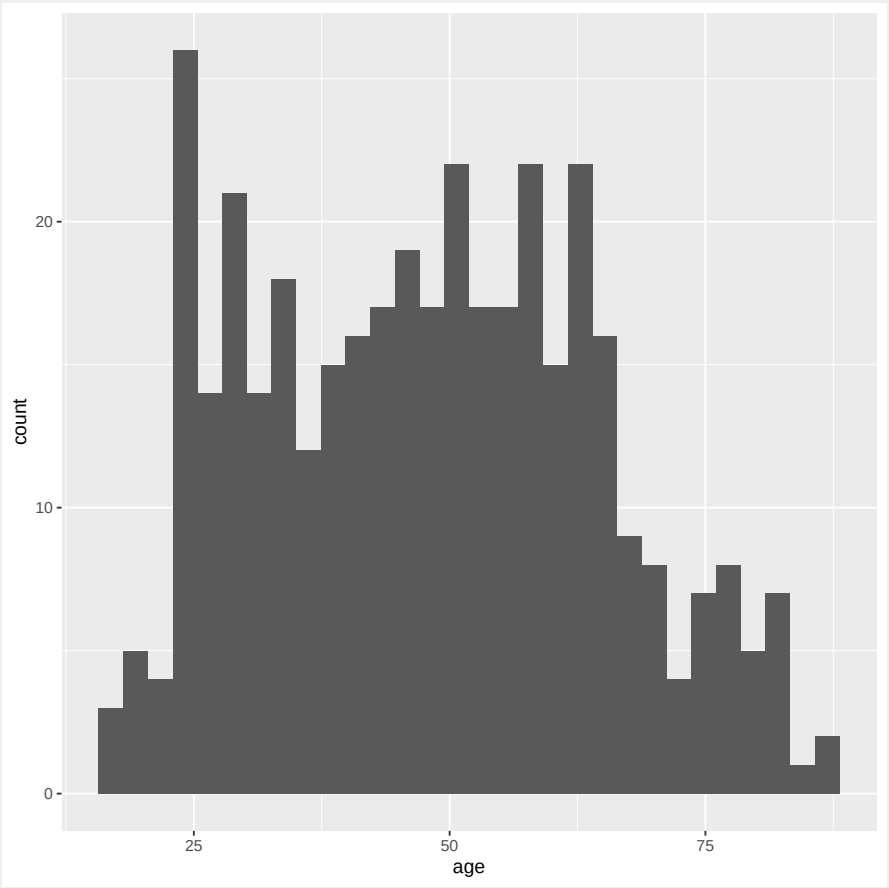
```
1 ggplot(data = anes, mapping = aes(x = age, y = income)) + geom_
      point()
```



Histograms

Histogram can be created using `geom_histogram()`

```
1  ggplot(data = anes, mapping = aes(x = age)) + geom_histogram()
```

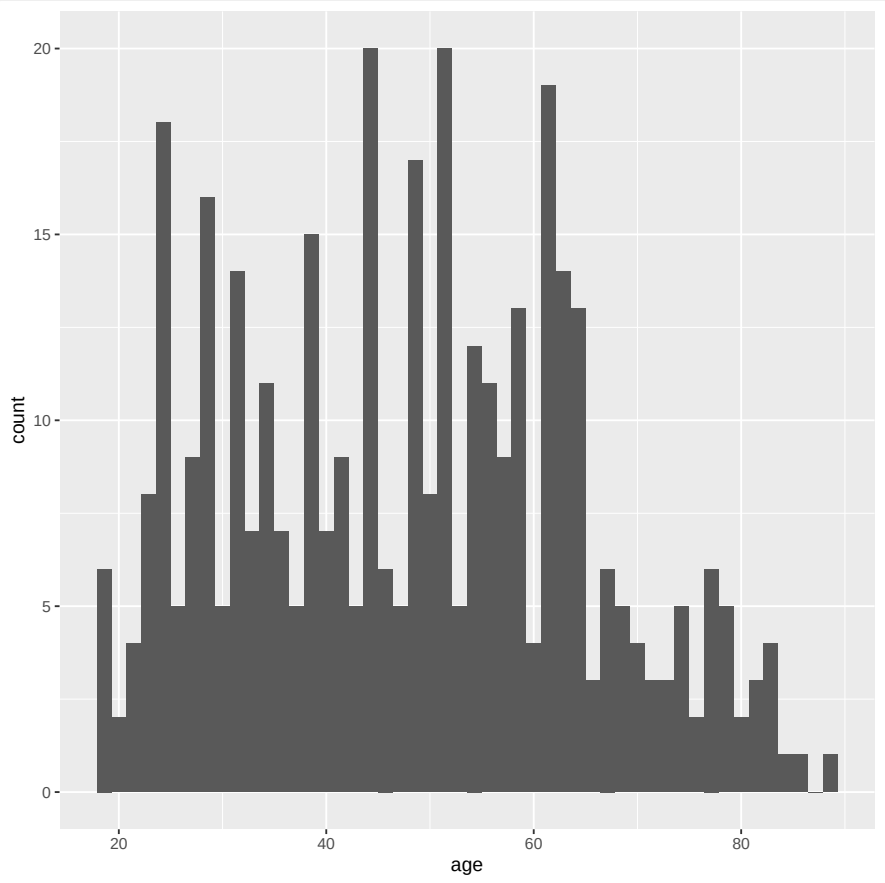


Histograms

How does ggplot know what to plot on y axis?

- It's using default statistical transformation is 'stat = "bin"
- We can adjust the number of bins using the 'bins' argument

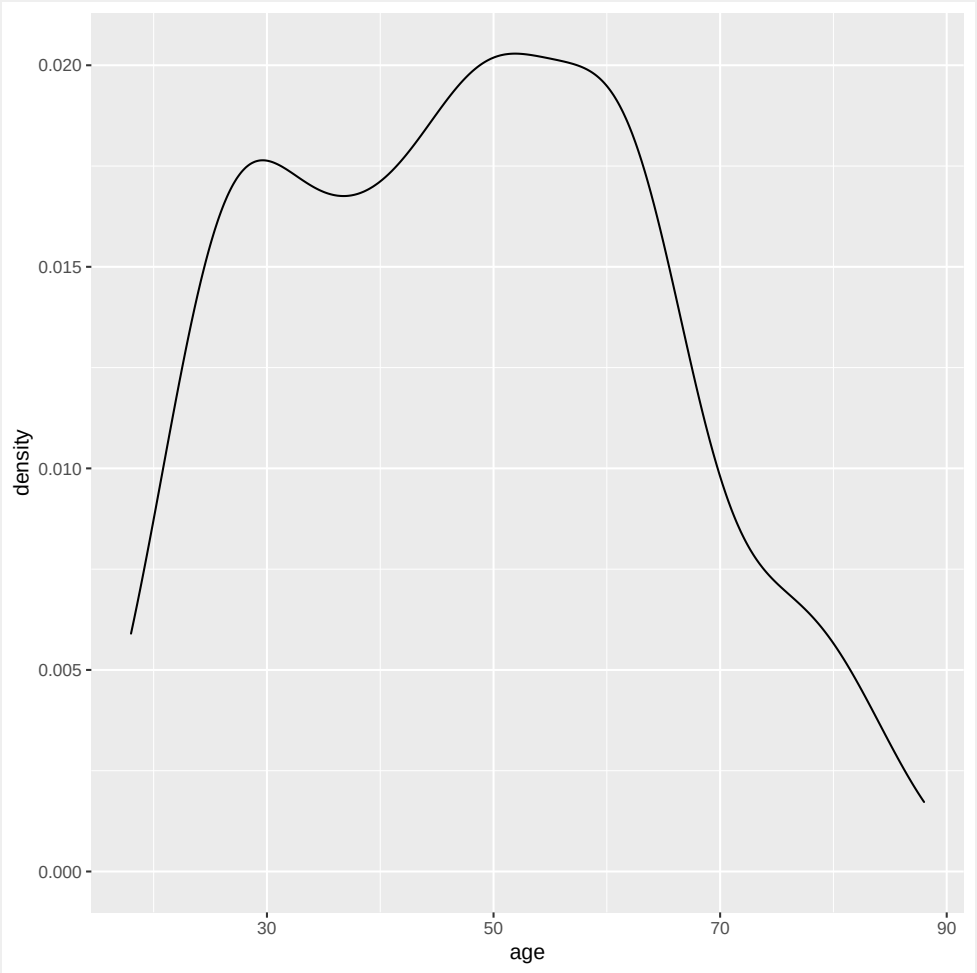
```
1  ggplot(data = anes, mapping = aes(x = age)) + geom_histogram(stat="bin", bins=50)
```



Density plot

Density plot can be created using `geom_density()`

```
1 ggplot(data = anes, mapping = aes(x = age)) + geom_density()
```



Bar Chart

There are two basic approaches to making bar charts, both of which use `geom_bar`

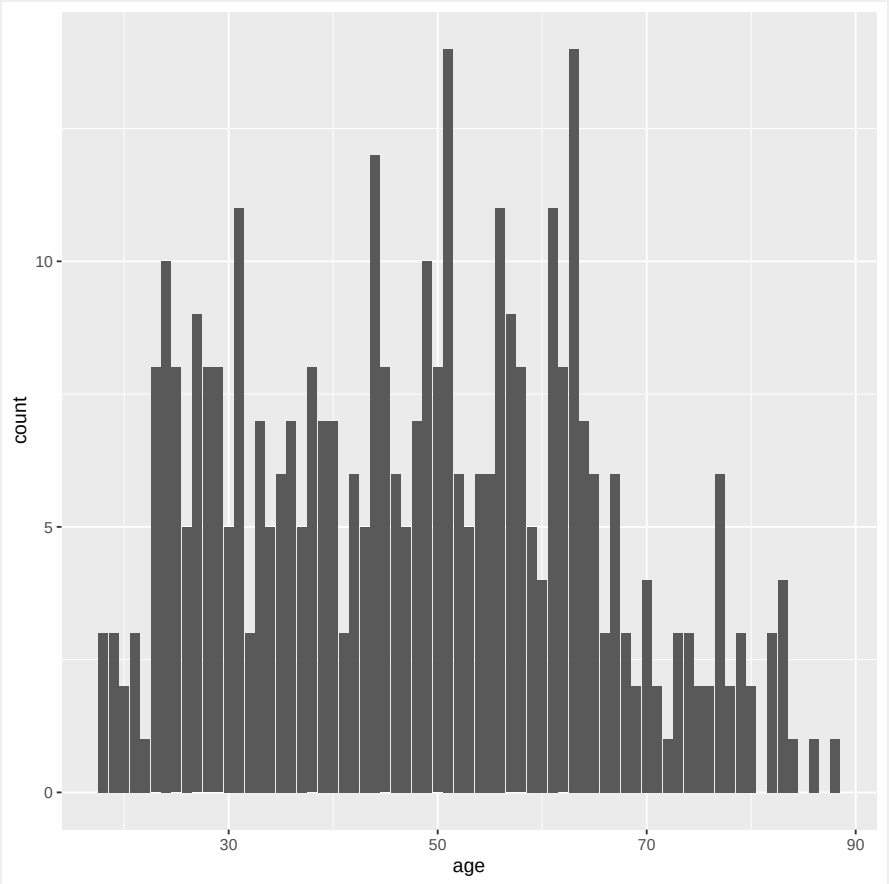
- Approach #1 - Use your full dataset
  - Only assign a variable to x axis
  - Let ggplot use default ‘stat‘ transformation (‘stat = "count"‘) to generate counts that it then plots on y axis
- Approach #2 - Wrangle your data frame before plotting
  - Possibly creating a new data frame in the process
  - Assign variables to x and y axes
  - Use ‘stat = "identity"‘ to tell ggplot to use data exactly as it is

Bar chart - Approach #1

Default statistical transformation for `geom_bar` is 'count'

- Will give us same result as our previous plot

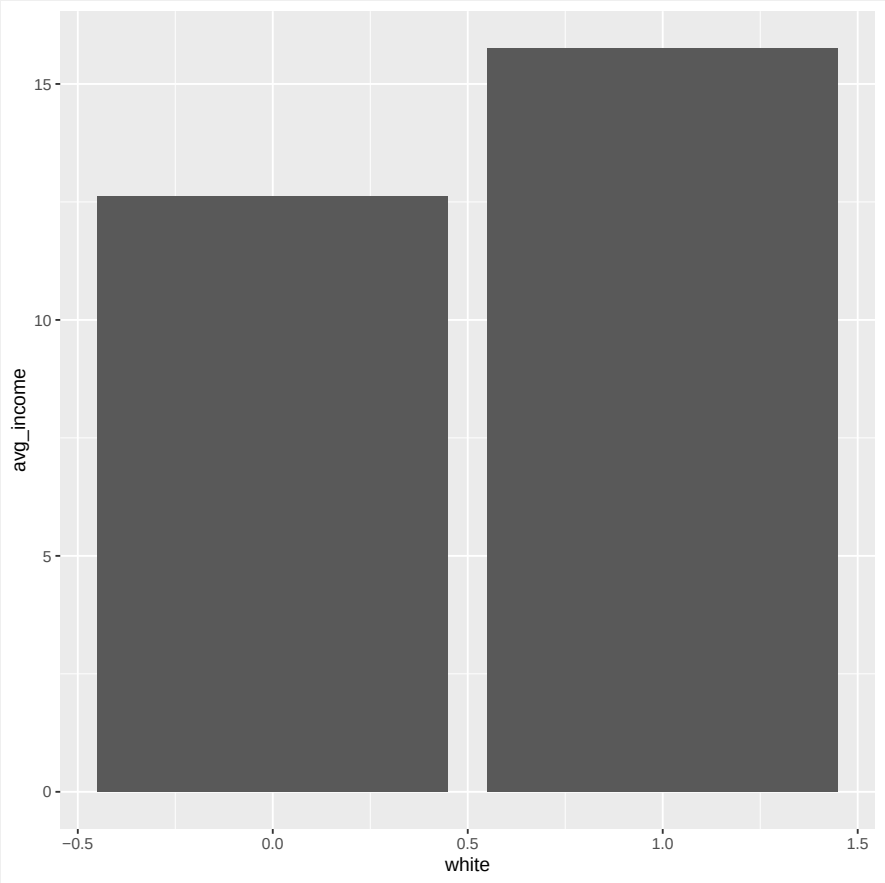
```
1 ggplot(data = anes, mapping = aes(x = age)) + geom_bar()
```



Bar chart - Approach #2

- It's often easier to do our analysis work, save a data frame, and then use this to plot
- 'stat = "identity"' here tells ggplot to use exact data points without any 'stat' transformations

```
1 anes %>%
2   filter(female == 1) %>%
3   group_by(white) %>%
4   summarize(avg_income = mean(
      income)) %>%
5   ggplot(aes(x = white, y = avg_
      income)) +
6   geom_bar(stat = "identity")
```

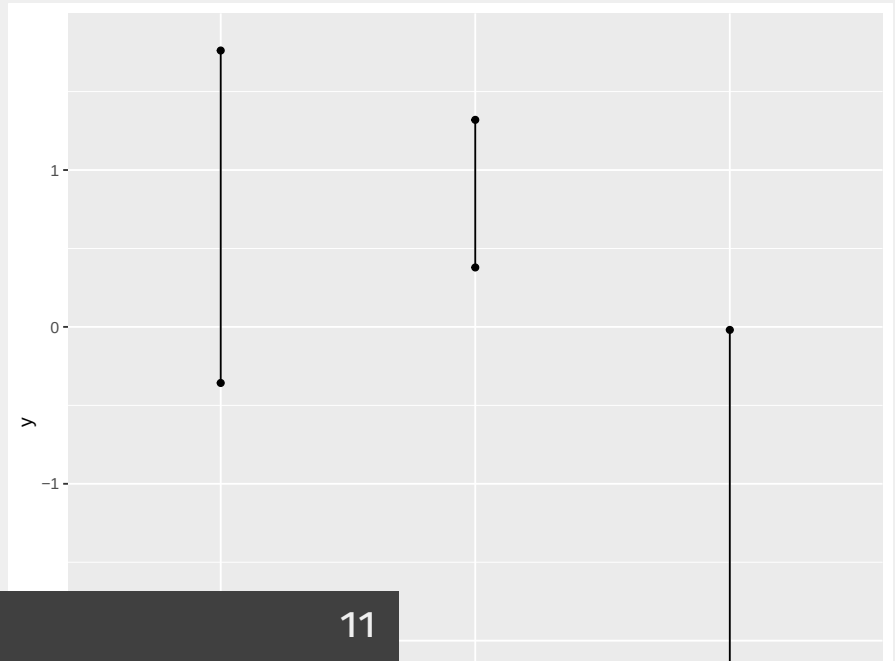


Line chart

Use geom_line()

```
data.frame(x=rep(c("a", "b", "c"),2),
           y=rnorm(6)) %>%
  ggplot(aes(x = x, y=y)) + geom_line() + geom_point()
dev.off()

pdf("../graphics/facet_gg.pdf")
ggplot(data = anes, aes(x = age, y = income, color = female)) +
    geom_point() + facet_wrap(~white)
dev.off()
```

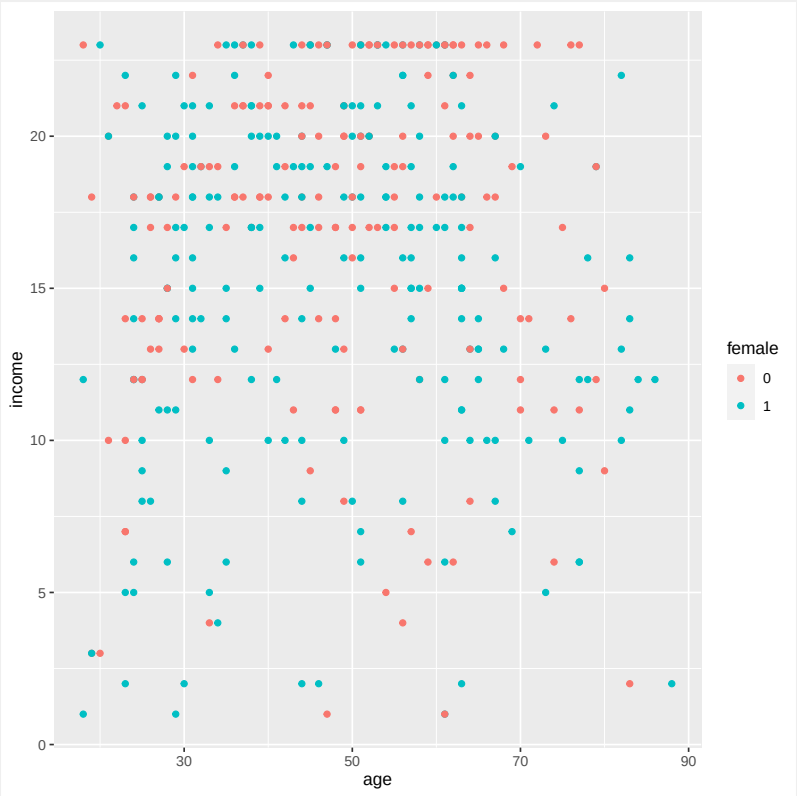


Color and fill

We add color argument **within** 'aes' so that data in that variable is mapped to those aesthetic properties

- Note that each option in the gender variable (male and female) is mapped to a color (women = teal, men = orange)

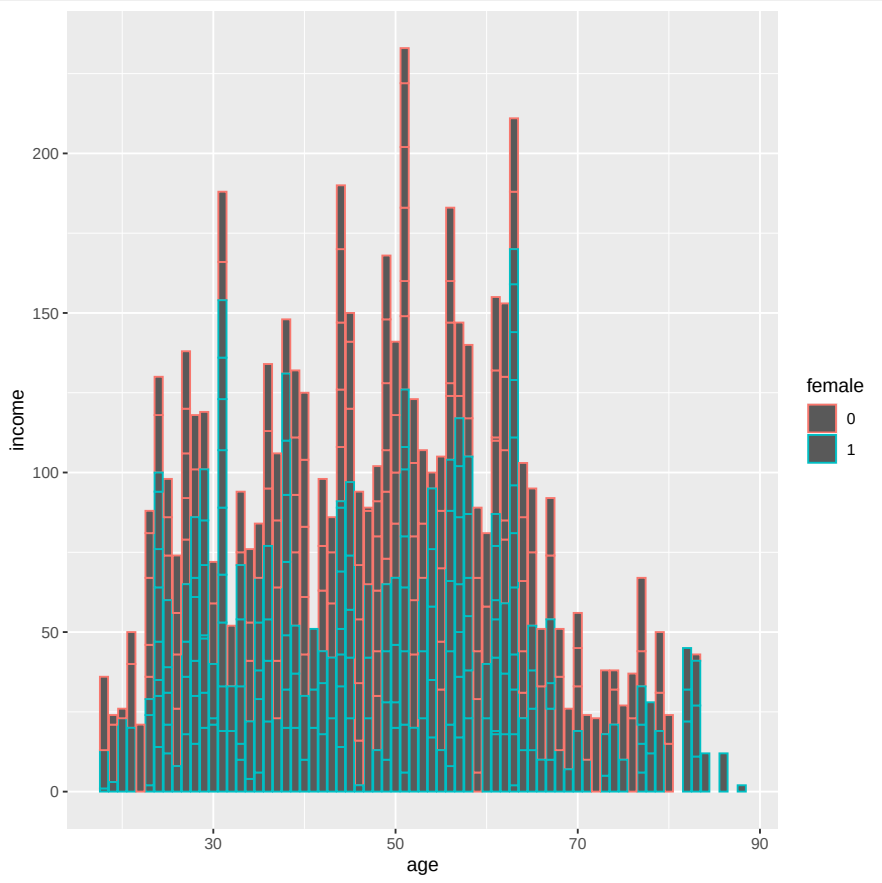
```
1  ggplot(data = anes, aes(x = age, y = income, color = female)) + geom_point()
```



Color and fill

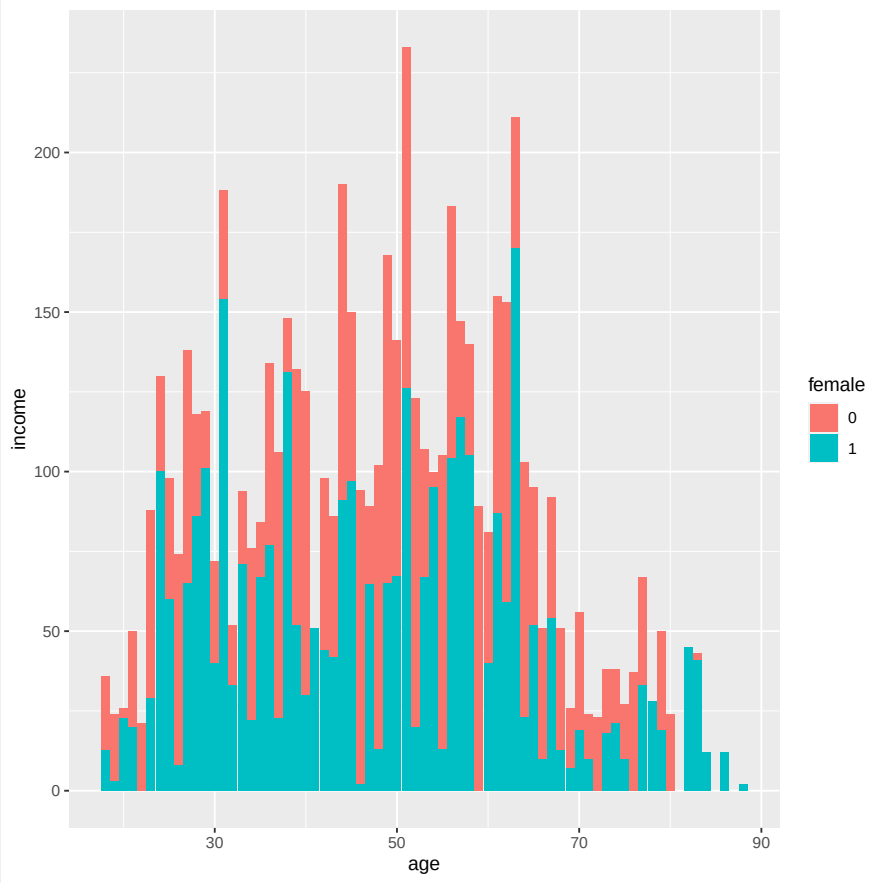
Let's try the same thing with a bar chart

```
ggplot(data = anes, aes(x = age,
    y = income, color = female))
    + geom_bar(stat = "identity")
```



That didn't work! Let's try 'fill' instead

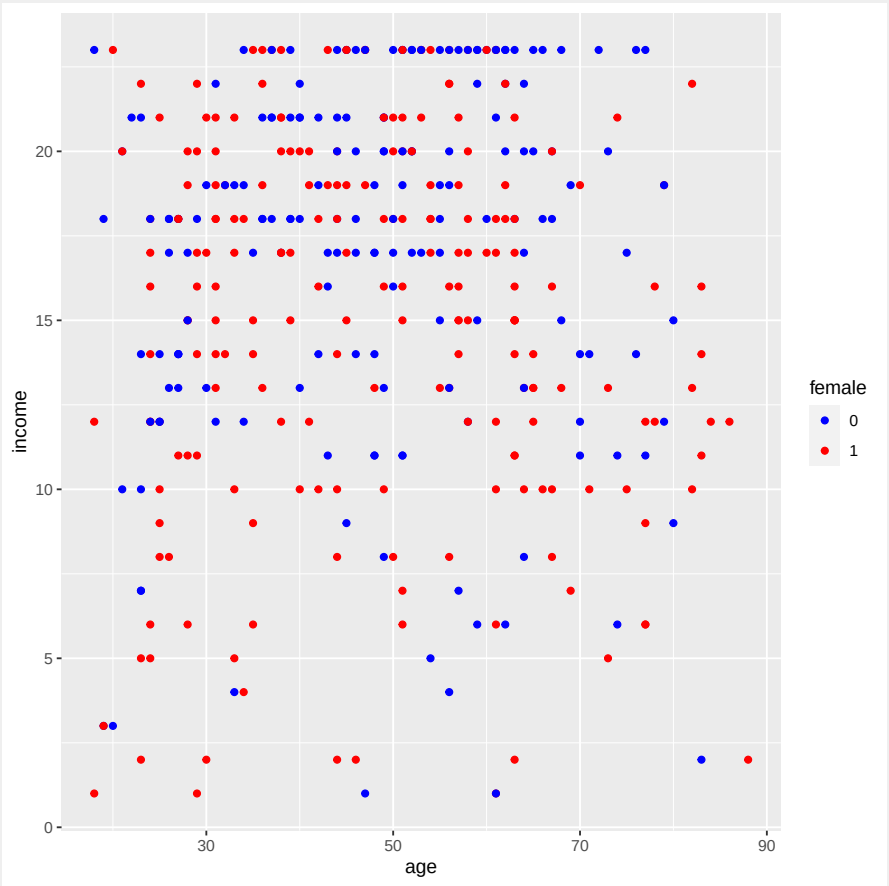
```
ggplot(data = anes, aes(x = age,
    y = income, fill = female)) +
    geom_bar(stat = "identity")
```



Scales

We can change which colors data is mapped to by using a ‘scale‘ function

```
1  ggplot(data = anes, aes(x = age, y = income, color = female)) + geom_point() + scale_
       color_manual(values = c("blue", "red"))
```

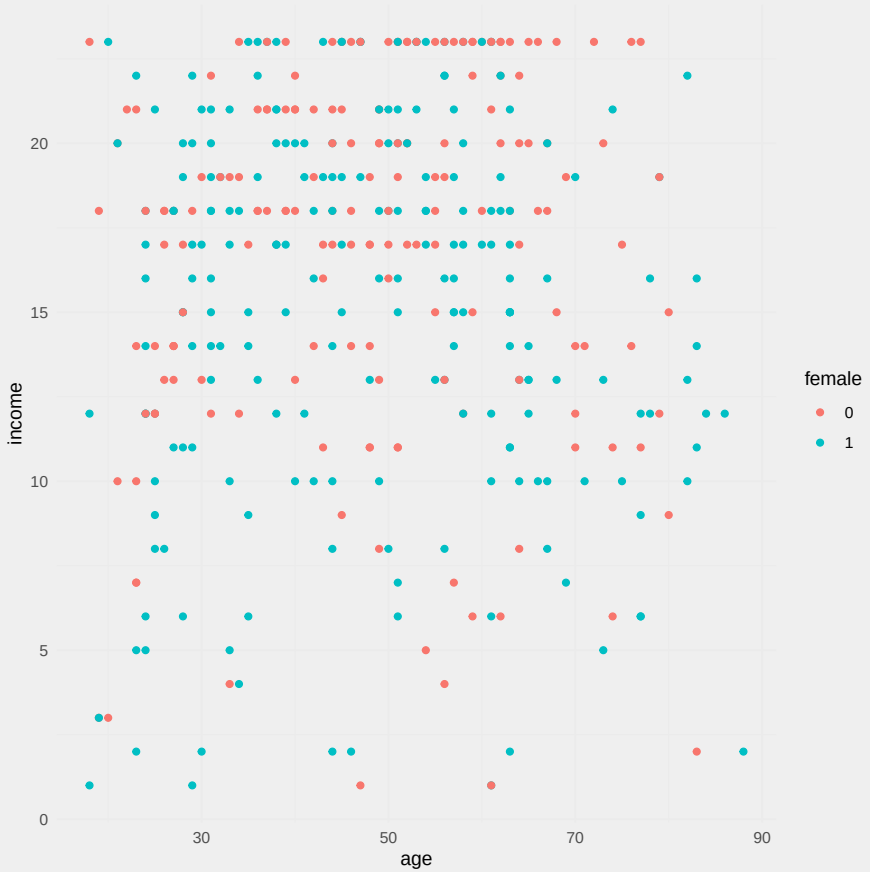


Themes

- To add a theme to a plot, we use the ‘theme_‘ set of functions
- There are several built-in themes
  - ‘theme_minimal‘
  - ‘theme_light‘
- There are also packages that give you themes you can apply to your plots (‘ggthemes‘)
- We can then use a theme from this package (‘theme_economist‘) to make our plots look like those in the Economist

Themes

```
ggplot(data = anes, aes(x = age,
    y = income, color = female))
    + geom_bar(stat = "identity")
```



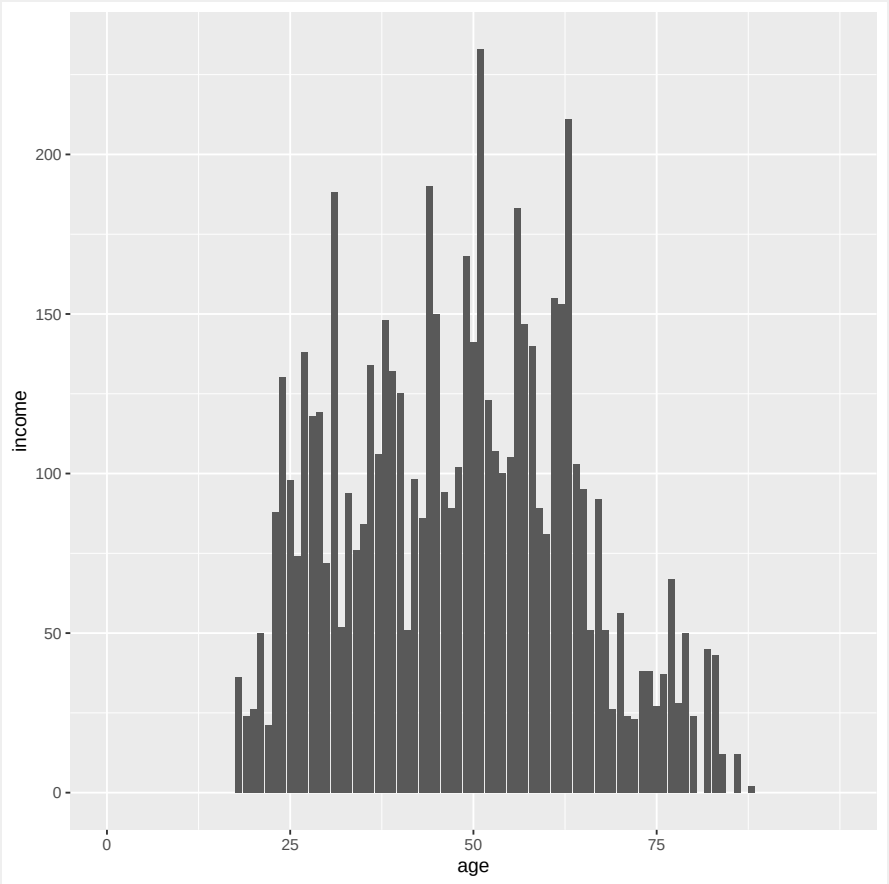

```
ggplot(data = anes, aes(x = age,
    y = income, fill = female)) +
     geom_bar(stat = "identity")
```

X AND Y AXES

- Adjusting our x and y axes is similar to base R
- Remember that x and y axes are considered an aesthetic properties in same way color is in ggplot
- We adjust our x and y axes using the ‘scale_‘ set of functions
- Which exact function you use depends on your data
  - Ex: Use ‘scale_y_continuous‘ if you have continuous data on y axis
  - ‘limits‘ argument sets minimum and maximum values that display
  - ‘breaks‘ argument determines which axis labels show up

Scale_continuous

```
ggplot(data = anes, mapping = aes(x = age, y = income)) + geom_col
    () + scale_x_continuous(limits = c(0, 100), breaks = c(0, 25,
    50, 75))
```



Text and Labels

- Text is just another geom, we use ‘geom_text‘ to add labels to our figures
- We can use ‘hjust‘ and ‘vjust‘ arguments to horizontally and vertically adjust text
  - ‘vjust = 0‘ puts the labels on outer edge of bars
  - ‘vjust = 1‘ puts the labels at inner edge of bars

Plot Labels

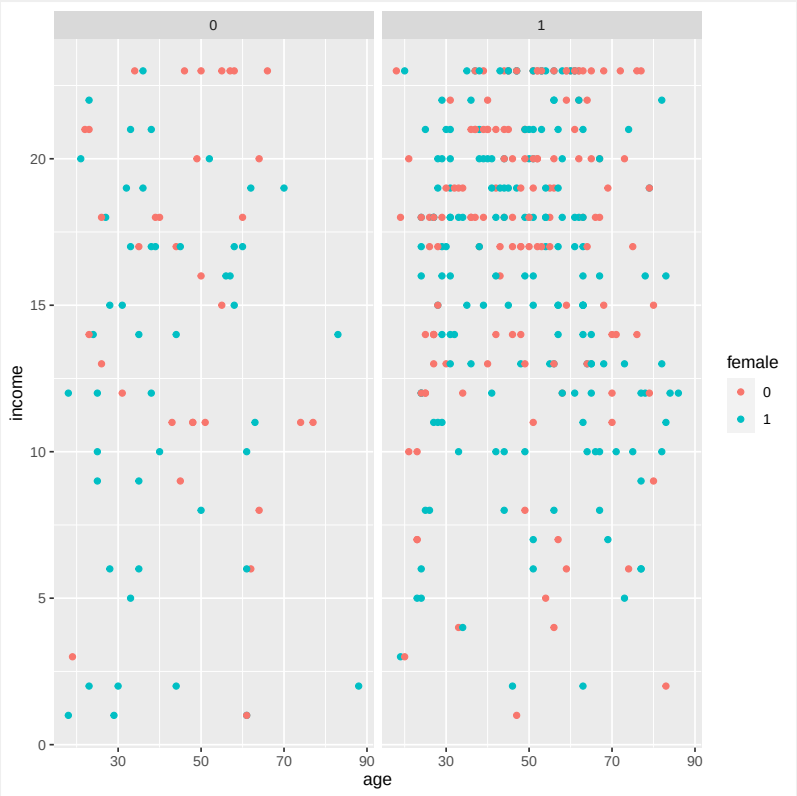
- To add labels to our plot, we use ‘labs‘
- We can a title to the plot with ‘title‘ argument
- We can add a ‘subtitle‘ as well
- We can change the x and y axis labels using the ‘x‘ and ‘y‘ arguments
- To change the title above the legend, we use the name of aesthetic that is being shown

Facets

One of the most powerful features of ggplot is facetting

- You can make small multiples by adding just a line of code using ‘facet_wrap‘ function

```
ggplot(data = anes, aes(x = age, y = income, color = female)) +
    geom_point() + facet_wrap(~white)
```



Tutorial - Plotting trends over time

- We want to report and plot the average income by age group
  1. Create a variable that breaks respondents into 10 groups
  2. Calculate the average income for each group by gender and ethnicity
  3. Plot the trend for the average income by group over time

Overview

This week:

- Plotting in base R
- Plotting in `ggplot`

Next week:

- Gathering electronic data