Welcome!

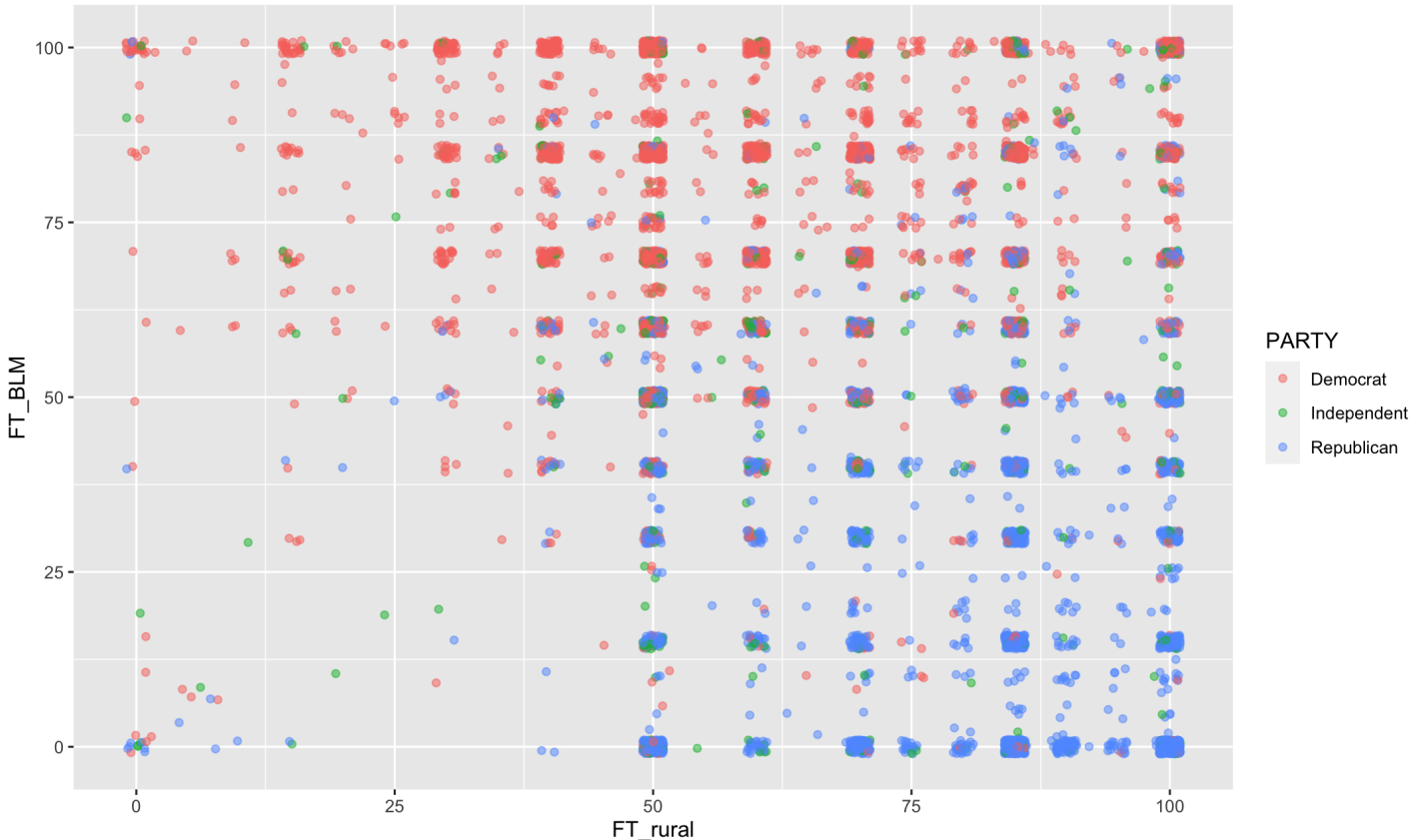
Week 3.2: Colors and Labels

Jennifer Lin

PS 490: R Workshop

2021-11-15

Last Week


100
75
50
25
0
FT_BLM
0
25
50
75
100
FT_rural
PARTY
Democrat
Independent
Republican

This Week -- we add color!

So your graph from last week was basic. It shows the general trends of your data but might do so in inefficient ways

- The labels on the axes do not communicate something meaningful
- The legend is poorly labeled
- R's default colors might not be the most beautiful, especially, if you are coloring/filling by a particular variable
- The default scales might not be as informative

So we will fix those problems today.

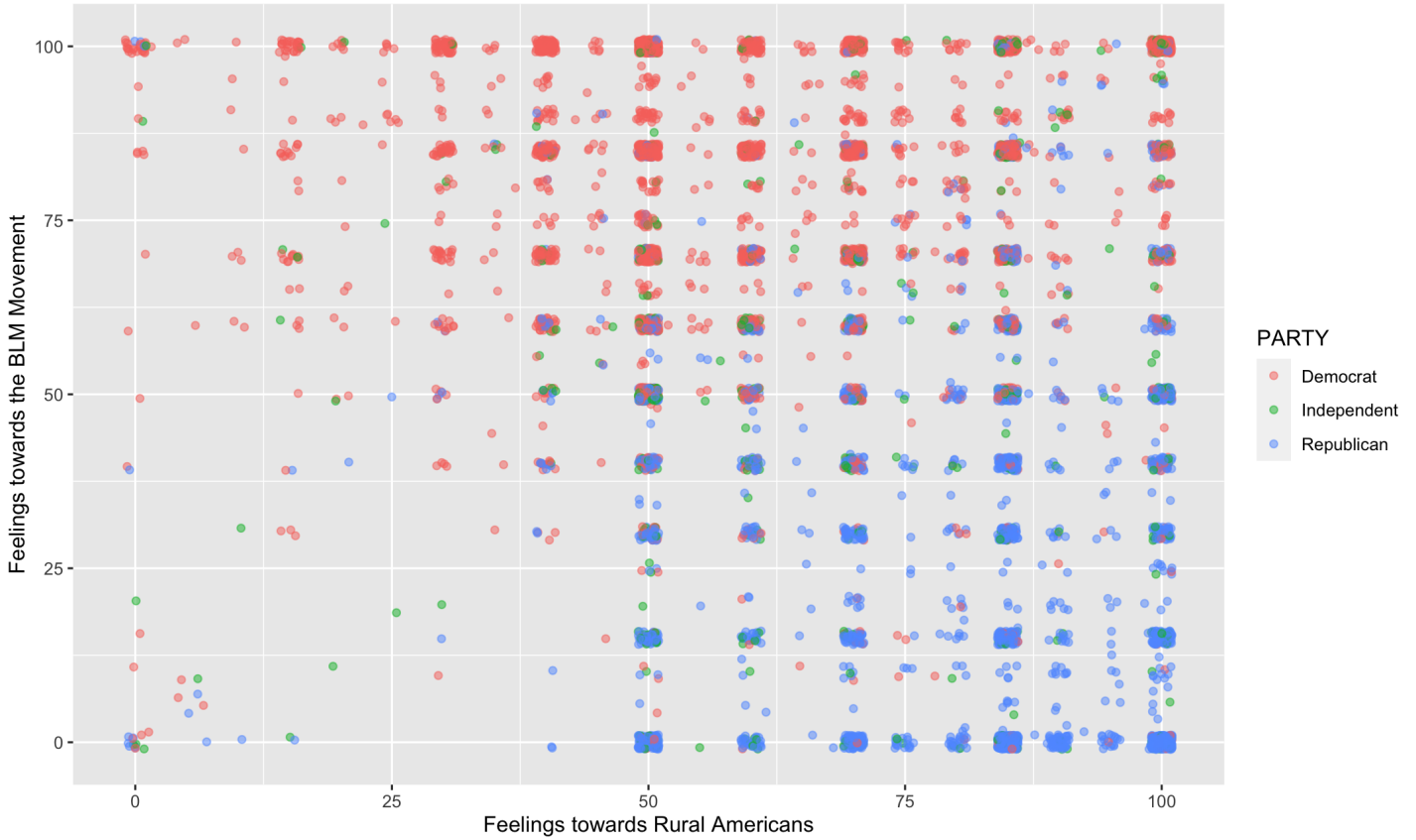
The Label Layer

Changing X and Y axis labels

To change the x and y axis labels, we use `xlab()` and `ylab()` respectively.

Within the parentheses, we add the new labels that we want in quotes ("") as such:

```
ANES %>%
  filter(!is.na(PARTY)) %>%
  ggplot(aes(x = FT_rural, y = FT_BLM, color = PARTY))+
  geom_point(position = position_jitter(1, 1), alpha = .5)+
  xlab("Feelings towards Rural Americans")+
  ylab("Feelings towards the BLM Movement")
```


Feelings towards the BLM Movement
100
75
50
25
0
0
25
50
75
100
Feelings towards Rural Americans
PARTY
Democrat
Independent
Republican

Adding Title, Legend Names and Subtitles

To add a title, subtitle, caption, or legend label to our graph, we can use the options provided by `labs()`

```
labs(
  color    = "",
  fill     = "",
  title    = "",
  subtitle = "",
  caption  = ""
)
```

- The `color`/`fill` option labels the legend and it is dependent on whether you used COLOR or FILL in the data layer
- `title` adds a main title to the top of the plot
- `subtitle` adds a subtitle to the top of the plot
- `caption` adds a caption to the bottom of the plot

```
ANES %>%
  filter(!is.na(PARTY)) %>%
  ggplot(aes(x = FT_rural, y = FT_BLM, color = PARTY))+
  geom_point(position = position_jitter(1, 1), alpha = .5)+
  xlab("Feelings towards Rural Americans")+
  ylab("Feelings towards the BLM Movement")+
  labs(
    color = "Political Party",
    title = "Feelings towards Rural Americans and
    the BLM Movement",
    subtitle = "Analysis from the American National Elections Stud
    caption = "Data: ANES 2020
    Author: Jennifer Lin"
  )
```

A few things to note:

1. Since it is a scatterplot, and I used color in the data layer, my label option for my legend is `color`
2. When you press `ENTER` to break a line in the quotes, it inserts a line break in the outcome.


Feelings towards Rural Americans and
the BLM Movement
Analysis from the American National Elections Studies
Feelings towards the BLM Movement
100
75
50
25
0
0
25
50
75
100
Feelings towards Rural Americans
Political Party
Democrat
Independent
Republican
Data: ANES 2020
Author: Jennifer Lin

Exercise: Adding labels to your graph

1. Revisit your code from last class. Copy the results that you have from the data and graph layers onto the code for this week.
2. Using `xlab()`, `ylab()` and `labs()` to add a x axis label, y axis label, title, subtitle, legend label and optional caption.

The Scale Layer

Changing the Colors in Our Plots in R

`ggplot` comes with a lot of `scale_*_*()` functions. These help us override defaults.

Most `ggplot` scales come in the format of `scale_[SOMETHING]_[SOMEHOW]()`

- `[SOMETHING]` -- What do you want to scale? Color, the x-axis (x), the fill, y-axis (y)?
- `[SOMEHOW]` -- How do you want it to start the rescale process? Transformed, gradients, manual?

Arguments

Most scales take the following arguments

- `name` = Name the thing you are scaling
- `breaks` = Locate where you want to break it
- `values` = Assign each break point a value (for colors or fills)
- `limits` = Set upper and lower bounds (if applicable)

Common Scale Uses

- `scale_x_continuous()`
  - `[SOMETHING]` = x-axis
  - `[SOMEHOW]` = continuously
- `scale_fill_manual()`
  - `[SOMETHING]` = shape fill
  - `[SOMEHOW]` = manually
- `scale_colour_brewer(palette = "[PALETTE NAME]")`
  - `[SOMETHING]` = color
  - `[SOMEHOW]` = Using the R Color Brewer palette

scale_color_manual()

If you are using a `fill`, this situation is the same thing, just change `color` to `fill`

So I want to introduce my own colors to the mix since the R default colors are completely trash.

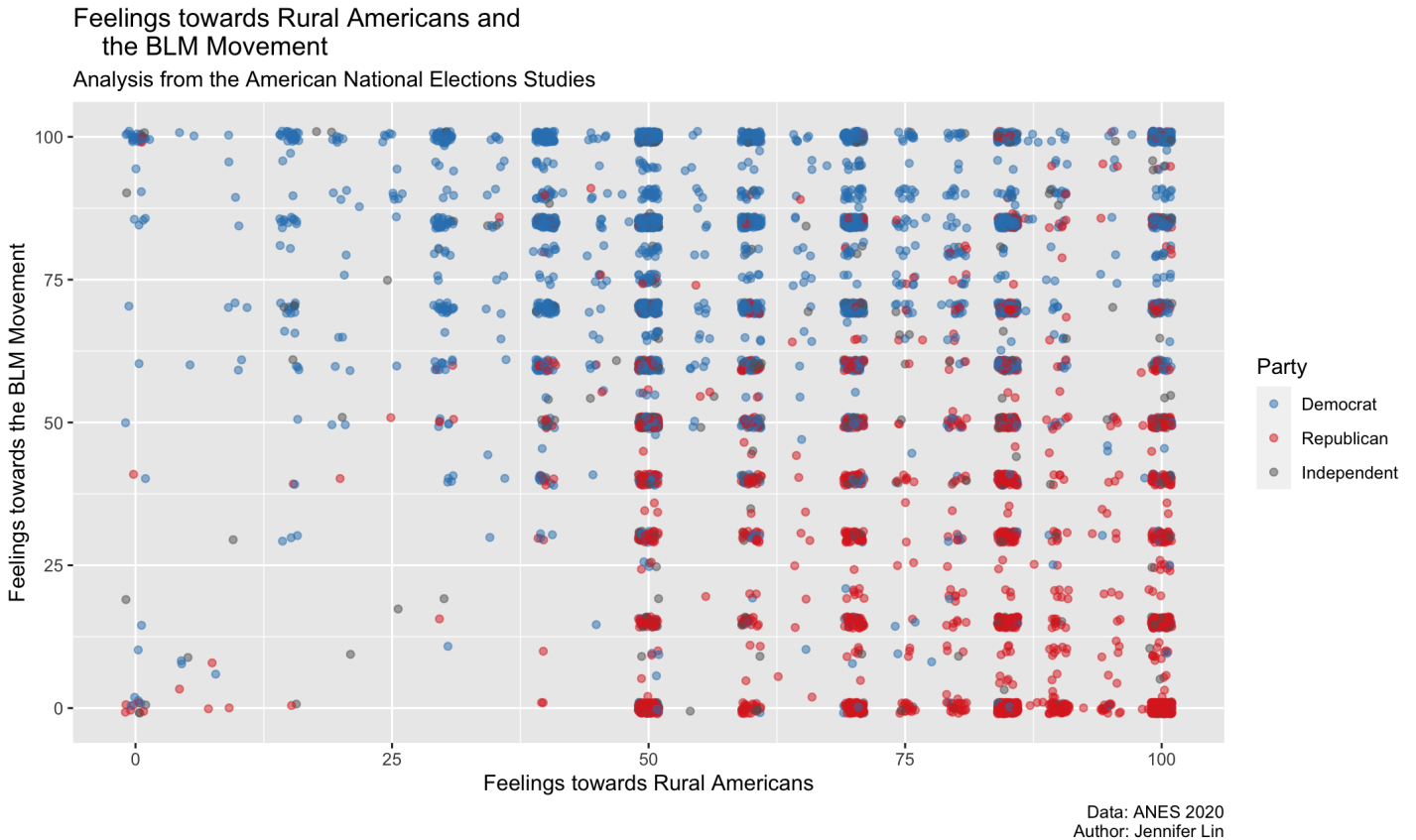
```
  scale_color_manual(
    name = "Party",
    breaks = c("Democrat",  "Republican", "Independent"),
    values = c("Democrat" = "#3182bd", "Republican" = "#de2d26", '
  )
```

```
  scale_color_manual(
    name = "Party",
    breaks = c("Democrat",  "Republican", "Independent"),
    values = c("Democrat" = "#3182bd", "Republican" = "#de2d26", '
  )
```

Here, I am using the `name`, `breaks` and `values` arguments

- `name` changes my legend label name from the `labs()` argument earlier (or from defaults)
- `breaks` set the categories for the legend. NOTICE that your legend will appear in this order
- `values` allows you to set colors manually. You can list the color ("red", "blue" etc) or use HEX codes (but keep them in quotes!)

```
ANES %>%
  filter(!is.na(PARTY)) %>%
  ggplot(aes(x = FT_rural, y = FT_BLM, color = PARTY))+
  geom_point(position = position_jitter(1, 1), alpha = .5)+
  xlab("Feelings towards Rural Americans")+
  ylab("Feelings towards the BLM Movement")+
  labs(
    color = "Political Party",
    title = "Feelings towards Rural Americans and
    the BLM Movement",
    subtitle = "Analysis from the American National Elections Stud
    caption = "Data: ANES 2020
    Author: Jennifer Lin"
  )+
  scale_color_manual(
    name = "Party",
    breaks = c("Democrat",  "Republican", "Independent"),
    values = c("Democrat" = "#3182bd", "Republican" = "#de2d26", "
  )
```


Feelings towards Rural Americans and
the BLM Movement
Analysis from the American National Elections Studies
Feelings towards the BLM Movement
100
75
50
25
0
0
25
50
75
100
Feelings towards Rural Americans
Party
Democrat
Republican
Independent
Data: ANES 2020
Author: Jennifer Lin

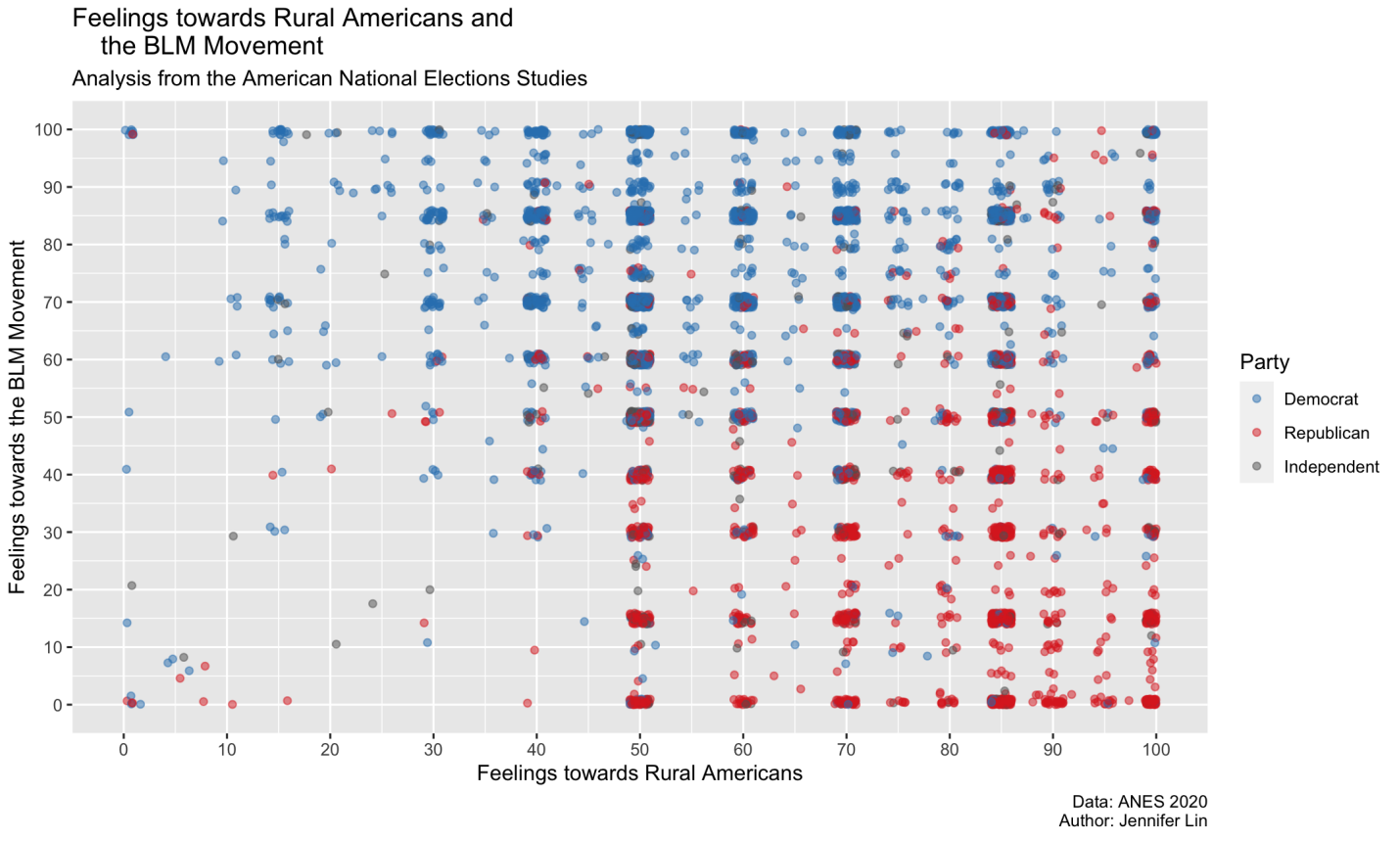
scale_x_continuous()

Now suppose I do not like the fact that the axis labels of my graph are too far apart, and I want to manipulate the continuous scale so that it shows more breaks.

```
scale_x_continuous(
  breaks = seq(0, 100, 10),
  limits = c(0, 100)
)+
```

Here, I use the `breaks` and `limits` options

- `breaks` tell R the numbers to include on the axis and how frequently I want to scale to add a new tick for a number. Since I am lazy, I am not writing all 10 numbers. Rather, I use `seq(BEGINNING, END, BY)` to count.
- `limits` tell R to cut the scale off at a certain point. In this case, I am telling it to go for the full scale (0-100)


Feelings towards Rural Americans and
the BLM Movement
Analysis from the American National Elections Studies
Feelings towards the BLM Movement
100
90
80
70
60
50
40
30
20
10
0
0
10
20
30
40
50
60
70
80
90
100
Feelings towards Rural Americans
Party
Democrat
Republican
Independent
Data: ANES 2020
Author: Jennifer Lin

Exercise: Changing your scales

1. Using the graph that you created in the last exercise, change the scales on your continuous variable
2. Change the color of your fills or lines, either by variable, or revisit your graph layer and change it globally
3. CHALLENGE: Go online and find a HEX code generator and add a color that is not an R default color (i.e. your color should include a #).

Your Submission to the Lab Assignment for this week

1. Export the graph as a PDF using the "Export" button on the upper left hand corner of the plot window.
2. Upload your PDF AND the code, with your answers to the questions
3. Don't worry about the background or the font sizes. We will work on that next week.