The full_join verb

JOINING DATA WITH DPLYR

Chris Cardillo
Data Scientist

Left and right joins

```
batwing %>%
  left_join(batwing, by = c("part_num", "color_id"), suffix = c("_batmobile", "_batwing"))
```

```
# A tibble: 309 x 4
   part_num color_id quantity_batmobile quantity_batwing
   <chr>        <dbl>              <dbl>            <dbl>
 1 3023            0                 22               22
 2 3024            0                 22               22
 3 3623            0                 20               20
 4 11477           0                 18               18
 5 99207          71                 18               18
 6 2780            0                 17               17
 7 3666            0                 16               16
 8 22385           0                 14               14
 9 3710            0                 14               14
10 99563           0                 13               13
# … with 299 more rows
```

The full join

first table   second table

first table	second table
a	a
b	c
c	d
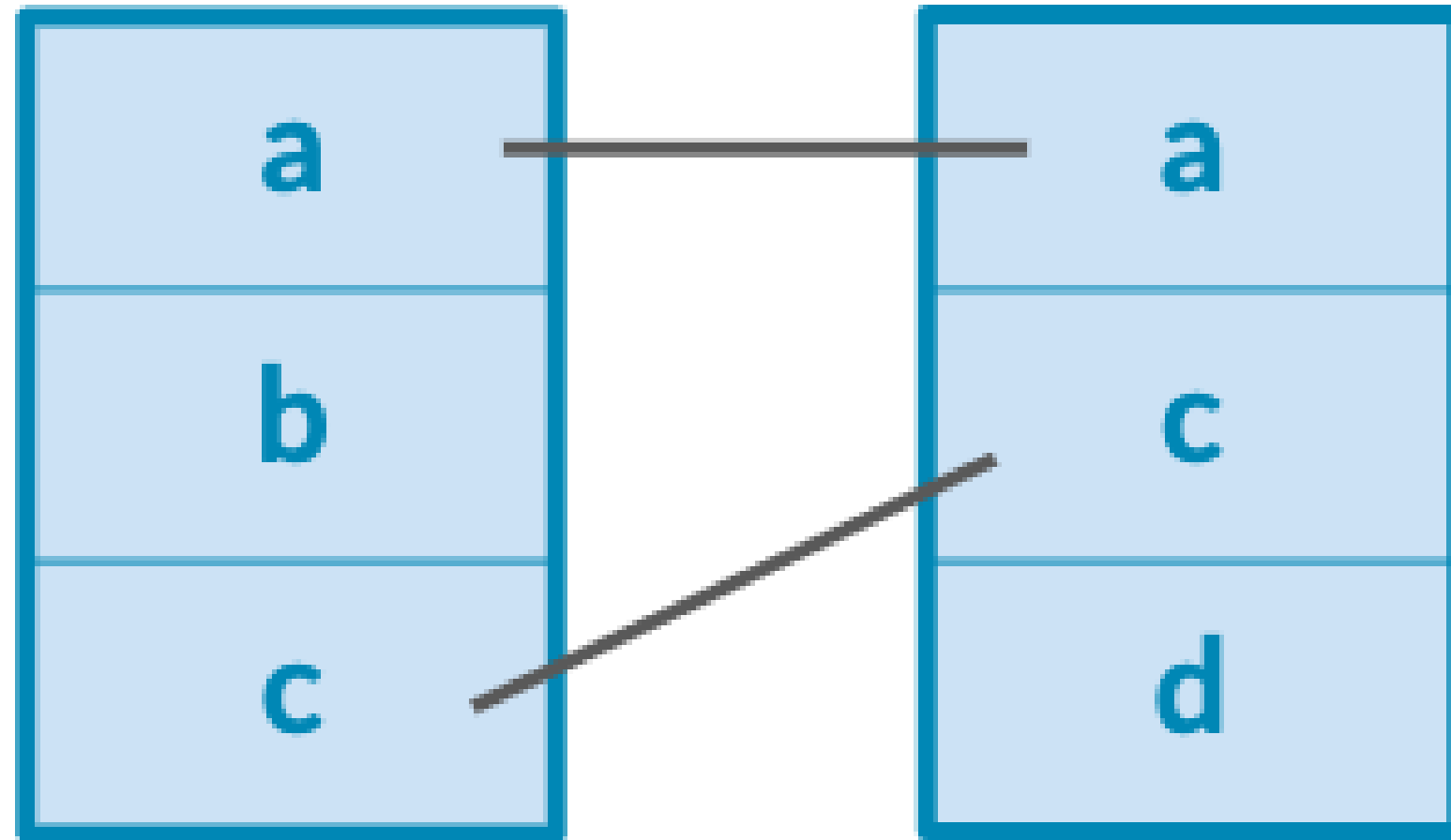
Joining and filtering

```r
inventory_parts_joined <- inventories %>%
  inner_join(inventory_parts, by = c("id" = "inventory_id")) %>%
  arrange(desc(quantity)) %>%
  select(-id, -version)


batmobile <- inventory_parts_joined %>%
  filter(set_num == "7784-1") %>%
  select(-set_num)


batwing <- inventory_parts_joined %>%
  filter(set_num == "70916-1") %>%
  select(-set_num)
```

Batmobile vs. Batwing

```
# A tibble: 173 x 3
   part_num color_id quantity
   <chr>       <dbl>    <dbl>
 1 3023           72       62
 2 2780            0       28
 3 50950           0       28
 4 3004           71       26
 5 43093           1       25
 6 3004            0       23
 7 3010            0       21
 8 30363           0       21
 9 32123b         14       19
10 3622            0       18
# … with 163 more rows
```

```
# A tibble: 309 x 3
   part_num color_id quantity
   <chr>       <dbl>    <dbl>
 1 3023            0       22
 2 3024            0       22
 3 3623            0       20
 4 11477           0       18
 5 99207          71       18
 6 2780            0       17
 7 3666            0       16
 8 22385           0       14
 9 3710            0       14
10 99563           0       13
# … with 299 more rows
```

Joining it all together

Left join: keep all batmobile

```
batmobile %>%
  left_join(batwing, by = c("part_num", "color_id"), suffix = c("_batmobile", "_batwing"))
```

Right join: keep all batwing

```
batmobile %>%
  right_join(batwing, by = c("part_num", "color_id"), suffix = c("_batmobile", "_batwing"))
```

Full join: keep all both

```
batmobile %>%
  full_join(batwing, by = c("part_num", "color_id"), suffix = c("_batmobile", "_batwing"))
```

Full join result

```
batmobile %>%
  full_join(batwing, by = c("part_num", "color_id"), suffix = c("_batmobile", "_batwing"))
```

```
# A tibble: 440 x 4
   part_num color_id quantity_batmobile quantity_batwing
   <chr>       <dbl>              <dbl>            <dbl>
 1 3023           72                 62               NA
 2 2780            0                 28               17
 3 50950           0                 28                2
 4 3004           71                 26                2
 5 43093           1                 25                6
 6 3004            0                 23                4
 7 3010            0                 21               NA
 8 30363           0                 21               NA
 9 32123b         14                 19               NA
10 3622            0                 18                2
# … with 430 more rows
```

Replace NA: multiple variables

```
library(tidyr)

batmobile %>%
  full_join(batwing, by = c("part_num", "color_id"), suffix = c("_batmobile", "_batwing")) %>%
  replace_na(list(quantity_batmobile = 0,
                  quantity_batwing = 0))
```

Let's practice!

JOINING DATA WITH DPLYR

The semi- and anti-join verbs

JOINING DATA WITH DPLYR

Chris Cardillo
Data Scientist

Mutating verbs

- `inner_join`

- `left_join`

- `right_join`

- `full_join`

Review: left join

```
batmobile %>%
  left_join(batwing, by = c("part_num", "color_id"), suffix = c("_batmobile", "_batwing"))
```

```
# A tibble: 173 x 4
   part_num color_id quantity_batmobile quantity_batwing
   <chr>       <dbl>              <dbl>            <dbl>
 1 3023           72                 62               NA
 2 2780            0                 28               17
 3 50950           0                 28                2
 4 3004           71                 26                2
 5 43093           1                 25                6
 6 3004            0                 23                4
 7 3010            0                 21               NA
 8 30363           0                 21               NA
 9 32123b         14                 19               NA
10 3622            0                 18                2
# … with 163 more rows
```
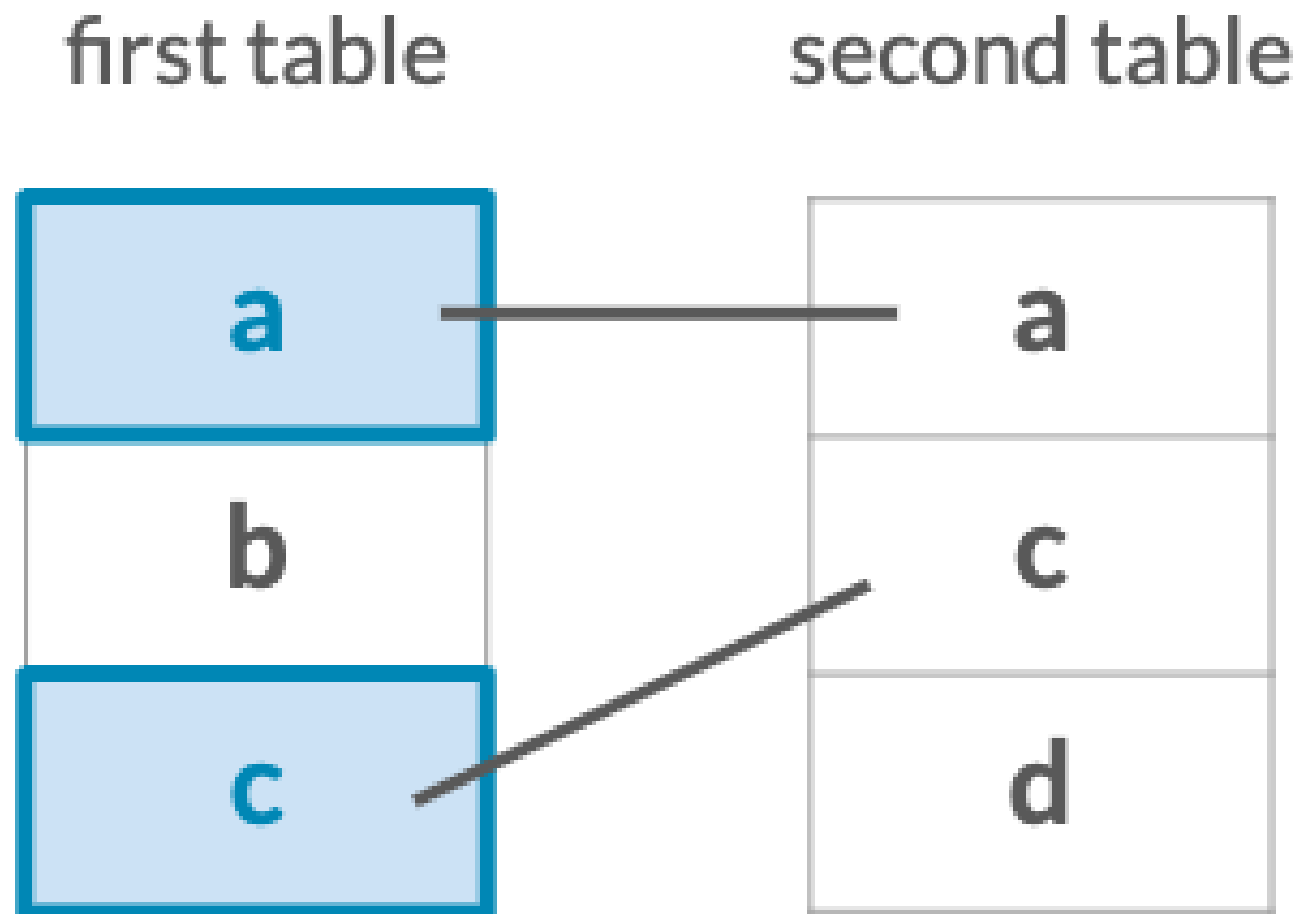
Filtering joins

- Keeps or removes observations from the first table

- Doesn't add new variables

- `semi_join()`
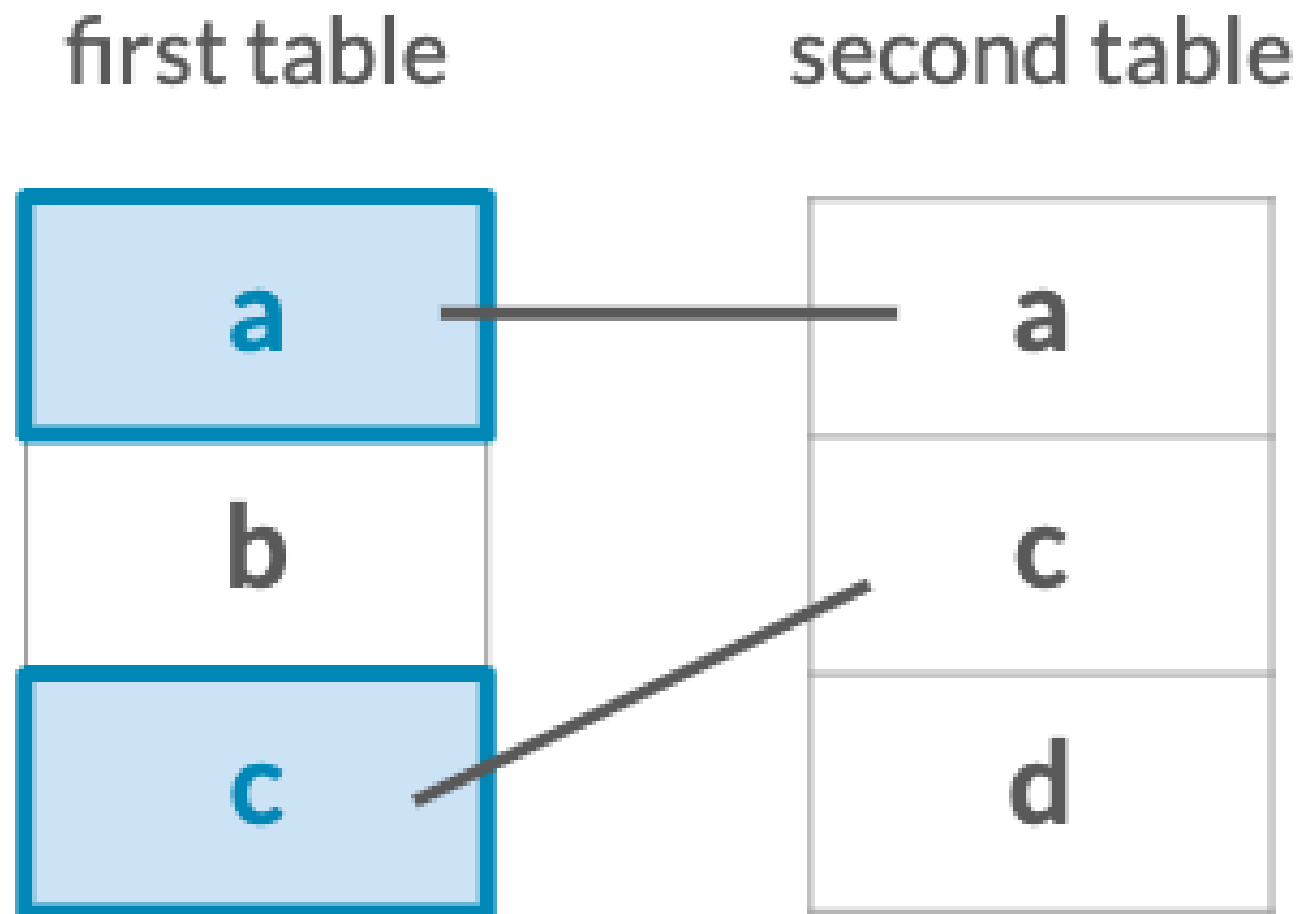
- `anti_join()`

Filtering joins

Semi join

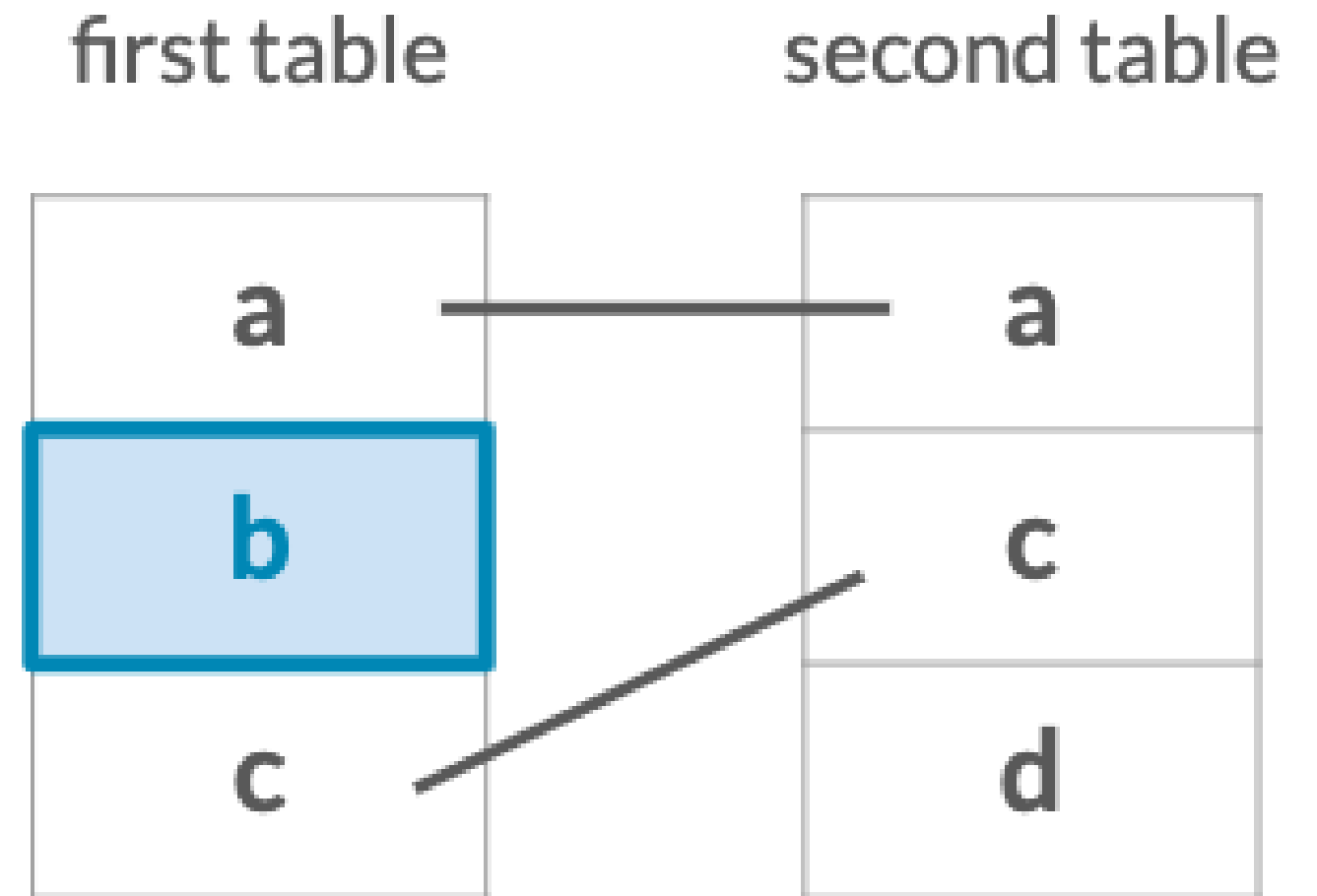- What observations in X are **also** in Y?

Filtering joins

Semi join

- What observations in X are **also** in Y?

first table   second table

first table	second table
a	**a**
b	c
c	d

Anti join

- What observations in X are **not** in Y?

first table   second table

first table	second table
a	a
b	c
c	d

The semi join

```
batmobile %>%
  semi_join(batwing, by = c("color_id", "part_num"))
```

```
# A tibble: 45 x 3
   part_num color_id quantity
   <chr>       <dbl>    <dbl>
 1 2780            0       28
 2 50950           0       28
 3 3004           71       26
 4 43093           1       25
 5 3004            0       23
 6 3622            0       18
 7 4286            0       16
 8 3039            0       12
 9 4274           71       12
10 3001            0       11
# … with 35 more rows
```

The anti join

```
batmobile %>%
  anti_join(batwing, by = c("color_id", "part_num"))
```

```
# A tibble: 128 x 3
   part_num color_id quantity
   <chr>       <dbl>    <dbl>
 1 3023           72       62
 2 3010            0       21
 3 30363           0       21
 4 32123b         14       19
 5 50950         320       18
 6 6541            0       18
 7 3040b           0       14
 8 3298            0       14
 9 3660            0       14
10 42022           0       14
# … with 118 more rows
```

Filtering with semi join

```
themes %>%
  semi_join(sets, by = c("id" = "theme_id"))
```

```
# A tibble: 569 x 3
     id name          parent_id
  <dbl> <chr>             <dbl>
 1    1 Technic              NA
 2    2 Arctic Technic        1
 3    3 Competition           1
 4    4 Expert Builder        1
 5    5 Model                 1
 6    6 Airport               5
 7    7 Construction          5
 8    9 Fire                  5
 9   10 Harbor                5
10   11 Off-Road              5
# … with 559 more rows
```

Filtering with anti join

```
themes %>%
  anti_join(sets, by = c("id" = "theme_id"))
```

```
# A tibble: 96 x 3
      id name        parent_id
   <dbl> <chr>           <dbl>
 1     8 Farm                5
 2    24 Airport            23
 3    25 Castle             23
 4    26 Construction       23
 5    27 Race               23
 6    28 Harbor             23
 7    29 Train              23
 8    32 Robot              23
 9    34 Building           23
10    35 Cargo              23
# … with 86 more rows
```

The joining verbs

- `inner_join`

- `left_join`

- `right_join`

- `full_join`

- `semi_join`

- `anti_join`

Let's practice!

JOINING DATA WITH DPLYR

Visualizing set differences

JOINING DATA WITH DPLYR

Chris Cardillo
Data Scientist

Aggregating sets into colors

```
batmobile_colors <- batmobile %>%
  group_by(color_id) %>%
  summarize(total = sum(quantity))


batmobile_colors
```

```
batwing_colors <- batwing %>%
  group_by(color_id) %>%
  summarize(total = sum(quantity))


batwing_colors
```

```
# A tibble: 12 x 2
   color_id total
      <dbl> <dbl>
 1        0   543
 2        1    33
 3        4    16
 4       14    20
 5       15    16
 6       36    15
 7       57     8
 8       71   202
 9       72   160
10      182     8
# … with 2 more rows
```

```
# A tibble: 20 x 2
   color_id total
      <dbl> <dbl>
 1        0   418
 2        1    45
 3        4    81
 4       14    22
 5       15    22
 6       19    10
 7       25     1
 8       34     3
 9       36     9
10       46    21
# … with 10 more rows
```

Comparing color schemes of sets

```
batmobile_colors %>%
  full_join(batwing_colors, by = "color_id", suffix = c("_batmobile", "_batwing")) %>%
  replace_na(list(total_batmobile = 0, total_batwing = 0))
```

```
# A tibble: 22 x 3
   color_id total_batmobile total_batwing
      <dbl>           <dbl>         <dbl>
 1        0             543           418
 2        1              33            45
 3        4              16            81
 4       14              20            22
 5       15              16            22
 6       36              15             9
 7       57               8             3
 8       71             202           158
 9       72             160           213
10      182               8            14
# … with 12 more rows
```

Adding the color names

```
batmobile_colors %>%
  full_join(batwing_colors, by = "color_id", suffix = c("_batmobile", "_batwing")) %>%
  replace_na(list(total_batmobile = 0, total_batwing = 0)) %>%
  inner_join(colors, by = c("color_id" = "id"))
```

```
# A tibble: 22 x 5
   color_id total_batmobile total_batwing name               rgb
      <dbl>           <dbl>         <dbl> <chr>              <chr>
 1        0             543           418 Black              #05131D
 2        1              33            45 Blue               #0055BF
 3        4              16            81 Red                #C91A09
 4       14              20            22 Yellow             #F2CD37
 5       15              16            22 White              #FFFFFF
 6       36              15             9 Trans-Red          #C91A09
 7       57               8             3 Trans-Neon Orange  #FF800D
 8       71             202           158 Light Bluish Gray  #A0A5A9
 9       72             160           213 Dark Bluish Gray   #6C6E68
10      182               8            14 Trans-Orange       #F08F1C
# … with 12 more rows
```

Adding percentages

```r
batmobile_colors %>%
  full_join(batwing_colors, by = "color_id", suffix = c("_batmobile", "_batwing")) %>%
  replace_na(list(total_batmobile = 0, total_batwing = 0)) %>%
  inner_join(colors, by = c("color_id" = "id")) %>%
  mutate(total_batmobile = total_batmobile / sum(total_batmobile),
         total_batwing = total_batwing / sum(total_batwing))
```

```
# A tibble: 22 x 5
   color_id total_batmobile total_batwing name              rgb
      <dbl>           <dbl>         <dbl> <chr>             <chr>
 1        0           0.516         0.397 Black             #05131D
 2        1          0.0314        0.0428 Blue              #0055BF
 3        4          0.0152        0.0770 Red               #C91A09
 4       14          0.0190        0.0209 Yellow            #F2CD37
 5       15          0.0152        0.0209 White             #FFFFFF
 6       36          0.0143       0.00856 Trans-Red         #C91A09
 7       57         0.00760       0.00285 Trans-Neon Orange #FF800D
 8       71           0.192         0.150 Light Bluish Gray #A0A5A9
 9       72           0.152         0.202 Dark Bluish Gray  #6C6E68
10      182         0.00760        0.0133 Trans-Orange      #F08F1C
# … with 12 more rows
```

The difference between fractions

```
colors_joined <- batmobile_colors %>%
  full_join(batwing_colors, by = "color_id", suffix = c("_batmobile", "_batwing")) %>%
  replace_na(list(total_batmobile = 0, total_batwing = 0)) %>%
  inner_join(colors, by = c("color_id" = "id")) %>%
  mutate(total_batmobile = total_batmobile / sum(total_batmobile),
         total_batwing = total_batwing / sum(total_batwing),
         difference = total_batmobile - total_batwing)

colors_joined
```

```
# A tibble: 22 x 6
   color_id total_batmobile total_batwing name              rgb     difference
      <dbl>           <dbl>         <dbl> <chr>             <chr>        <dbl>
 1        0           0.516         0.397 Black             #05131D      0.119
 2        1          0.0314        0.0428 Blue              #0055BF     -0.0114
 3        4          0.0152        0.0770 Red               #C91A09     -0.0618
 4       14          0.0190        0.0209 Yellow            #F2CD37     -0.00190
 5       15          0.0152        0.0209 White             #FFFFFF     -0.00570
 6       36          0.0143       0.00856 Trans-Red         #C91A09      0.00570
 7       57         0.00760       0.00285 Trans-Neon Orange #FF800D      0.00475
 8       71           0.192         0.150 Light Bluish Gray #A0A5A9      0.0418
 9       72           0.152         0.202 Dark Bluish Gray  #6C6E68     -0.0504
10      182         0.00760        0.0133 Trans-Orange      #F08F1C     -0.00570
# … with 12 more rows
```

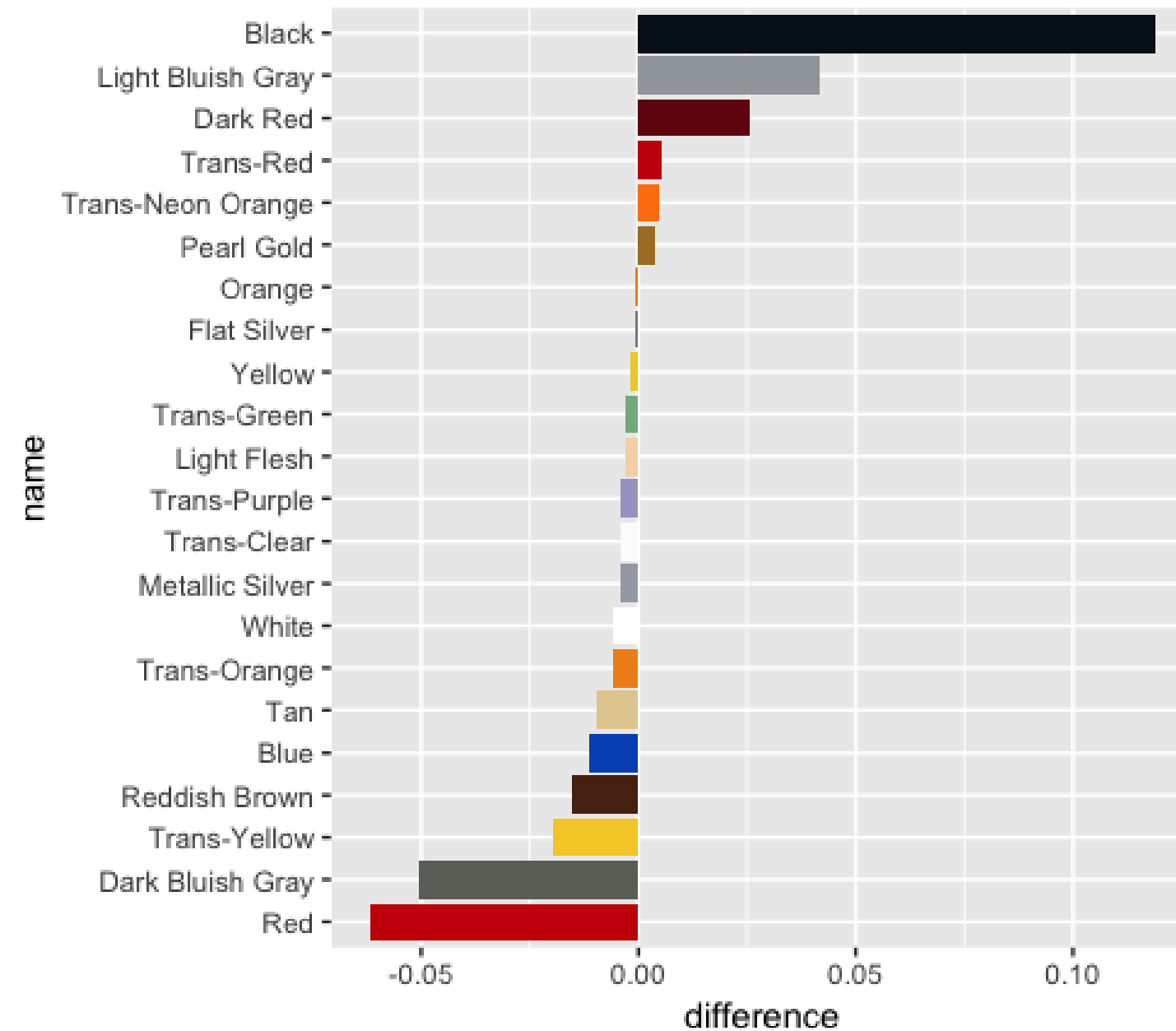# Visualizing the data

```r
library(ggplot2)
library(forcats)

color_palette <- setNames(colors_joined$rgb, colors_joined$name)

colors_joined %>%
  mutate(name = fct_reorder(name, difference)) %>%
  ggplot(aes(name, difference, fill = name)) +
  geom_col() +
  coord_flip() +
  scale_fill_manual(values = color_palette, guide = FALSE)
```

Visualizing the data

Comparing Batman and Star Wars themes

Let's practice!

JOINING DATA WITH DPLYR