

# ggplot2 Graph Gallery

Categories and distributions: *amounts*

Martin Frigaard

2022-05-22

# Load data packages

```
library(palmerpenguins)  
library(fivethirtyeight)  
library(ggplot2movies)
```

# palmerpenguins



package website

```
penguins <- palmerpenguins::penguins
penguins
```

species	island	bill_length_mm	bill_depth_mm	flipper_length_m m	body_mass_g	sex										year
Adelie	Torgersen	39.1	18.7	181	3750	male										2007
Adelie	Torgersen	39.5	17.4	186	3800	female										2007
Adelie	Torgersen	40.3	18	195	3250	female										2007
Adelie	Torgersen															2007
Adelie	Torgersen	36.7	19.3	193	3450	female										2007
1-5 of 344 rows					Previous	1	2	3	4	5	...	69	Next			

package website

*All datasets are listed below with descriptions*

		Search
dataset	title	
US_births_1994_2003	Some People Are Too Superstitious To Have A Baby On Friday The 13th	
US_births_2000_2014	Some People Are Too Superstitious To Have A Baby On Friday The 13th	
ahca_polls	American Health Care Act Polls	
airline_safety	Should Travelers Avoid Flying Airlines That Have Had Crashes in the Past?	
antiquities_act	Trump Might Be The First President To Scrap A National Monument	
1-5 of 129 rows		Previous 1 2 3 4 5 ... 26 Next

# ggplot2movies



package website

We're using `movies_data` (derived version of the `ggplot2movies::movies`)

movies\_data

title	year	length	budget	rating	mpaa	genres
100 Mile Rule	2002	98	1100000	5.6	R	comedy
13 Going On 30	2004	98	37000000	6.4	PG-13	drama, comedy, romance
15 Minutes	2001	120	42000000	6.1	R	drama
2 Fast 2 Furious	2003	107	76000000	5.1	PG-13	action
2046	2004	129	12000000	7.6	R	drama, romance
1-5 of 751 rows				Previous	1	2
					3	4
					5	...
					151	Next

# Comparing Categories and Distributions



**Amounts**

# Amounts: Bars



*The bar chart (or graph) is typically used to display counts. Bar charts can be arranged vertically or horizontally, stacked, diverging, or dodged. In `ggplot2`, bar charts can be built using `geom_bar()` or `geom_col()`*

# Amounts: Bars



movies\_data

title	year	length	budget	rating	mpaa	genres									
100 Mile Rule	2002	98	1100000	5.6	R	comedy									
13 Going On 30	2004	98	37000000	6.4	PG-13	drama, comedy, romance									
15 Minutes	2001	120	42000000	6.1	R	drama									
2 Fast 2 Furious	2003	107	76000000	5.1	PG-13	action									
2046	2004	129	12000000	7.6	R	drama, romance									
1-5 of 751 rows				Previous	1	2	3	4	5	...	151	Next			



# Amounts: Bars



Map `mpaa` to the `x` axis and to the `fill` aesthetic inside the `aes()` of `geom_bar()`, and add the labels

R Code

Plot

```
labs_geom_bar <- labs(  
  x = "MPAA rating",  
  title = "IMDB movie information/user ratings")  
  
ggplot(data = movies_data,  
       aes(x = mpaa)) +  
  geom_bar(aes(fill = mpaa)) +  
  labs_geom_bar
```

# Amounts: Grouped Bars



*To create grouped bar charts (compare the values of a numerical variable across the levels of a categorical variable) we can use the `geom_col()` function.*

# Amounts: Grouped Bars



Map `mpaa` to the `x` axis, `rating` to the `y` axis, and `mpaa` to `fill` inside the `aes()` of `geom_col()`, and add the labels

R Code

Plot

```
labs_geom_col <- labs(  
  x = "MPAA rating",  
  y = "Average IMDB user rating",  
  title = "IMDB movie information/user ratings")
```

```
ggplot(data = movies_data,  
       aes(x = mpaa,  
           y = rating)) +  
  geom_col(aes(fill = mpaa)) +  
  labs_geom_col
```

# Amounts: Stacked Bars



*We can also use bars to look at numeric and categorical variables using `geom_bar()` by setting `fill` argument.*

# Amounts: Stacked Bars



Map `flipper_length_mm` to the `x` axis, `sex` to `fill`, the `geom_bar()` layer, and add the labels

R Code

Plot

```
labs_geom_bar_stacked <- labs(  
  x = "Flipper length (millimeters)",  
  title = "Adult foraging penguins")
```

```
# remove missing sex  
penguins_stacked <- filter(penguins,  
                           !is.na(sex))  
ggplot(data = penguins_stacked,  
       aes(x = flipper_length_mm,  
           fill = sex)) +  
  geom_bar() +  
  labs_geom_bar_stacked
```

# Amounts: More Stacked Bars



Map `island` to the `x` axis, `flipper_length_mm` to the `y` axis, `sex` to `fill`, the `geom_bar()` layer (with `position` and `stat`), and add the labels

# Amounts: More Stacked Bars



R Code    Plot

```
geom_bar_stacked_2 <- labs(  
  x = "Island in Palmer Archipelago",  
  y = "Flipper length (millimeters)",  
  title = "Adult foraging penguins")  
ggplot(data = penguins,  
  aes(x = island,  
    y = flipper_length_mm,  
    fill = sex)) +  
# use this to determine how many  
# sex values are NA (and in what  
# categories)  
  geom_bar(position = "stack",  
    stat = "identity") +  
  geom_bar_stacked_2
```

# Amounts: Diverging Bars



*If you have a numeric variable with positive and negative values, consider using diverging bars with `geom_bar()`*

R Code

Data

```
unisex_names <- fivethirtyeight::unisex_names
unisex_names_diff <- mutate(unisex_names,
  male_female_diff = male_share - female_share,
  diff_cat = if_else(
    male_female_diff > 0,
    true = "More common male name",
    false = "More common female name"))
sample_names <- slice_sample(unisex_names_diff, n = 10)
```



# Amounts: Diverging Bars



Here we use the `reorder()` function to arrange the values of `male_female_diff` by `name`, and map the `diff_cat` to `label`.

R Code

Plot

```
labs_geom_bar_diverg <- labs(  
  x = "Name",  
  y = "Male share - female share",  
  title = "Most Common Unisex Names In America",  
  fill = "Difference category")  
ggplot(data = sample_names,  
  aes(x = reorder(x = name,  
    male_female_diff),  
    # reorder this by x  
    y = male_female_diff,  
    label = diff_cat)) +  
  geom_bar(  
    aes(fill = diff_cat),  
    stat = "identity",  
    width = .5) +  
  labs_geom_bar_diverg
```

# Amounts: Diverging Bars (vertical)



Diverging bar-charts can be arranged vertically, too. For vertically arranged bars, we switch the `x` and `y` axis variables (and the `reorder()` function).

R Code

Plot

```
labs_geom_bar_diverg_vert <- labs(  
  x = "Name",  
  y = "Male share - female share",  
  title = "Most Common Unisex Names In America",  
  fill = "Difference category")  
ggplot(data = sample_names,  
  aes(x = male_female_diff,  
    # reorder this by x  
    y = reorder(x = name,  
      male_female_diff),  
    label = diff_cat)) +  
  geom_bar(  
    aes(fill = diff_cat),  
    stat = "identity",  
    width = .5) +  
  labs_geom_bar_diverg_vert
```