

Basic Syntax

`ggplot(data, aes(x = variable1, y = variable2)) + geom_type() + [other functions]`

ggplot

- **ggplot()**: Initializes the plot with data and aesthetics.
 - **data**: Dataset used for plotting.
 - **aes()**: Aesthetic mappings (e.g., x and y axes).
- **functions**: Add geometries to the plot.
 - **geom_point()**: Scatter plot.
 - **geom_line()**: Line plot.
 - **geom_bar()**: Bar plot.
 - **geom_histogram()**: Histogram.
 - **geom_boxplot()**: Boxplot.
- **labs()**: Add labels and titles.
 - **labs(title = "Title", x = "X Axis", y = "Y Axis")**
- **theme()**: Customize plot appearance.
 - **theme_minimal()** # Clean minimal
 - **theme_bw()** # Black and white theme

Aesthetic Mappings (aes)

- **Position**: x, y
- **Color**: color (point or line color)
- **Shape**: shape (for points)
- **Size**: size (point or line size)
- **Fill**: fill (used in bar charts, histograms)

`ggplot(data, aes(x = var1, y = var2, color = group)) + geom_point()`

Facets

- Use **facet_wrap()** or **facet_grid()** for splitting the plot by categories.

`ggplot(data, aes(x = var1, y = var2)) + geom_point() + facet_wrap(~ category)`

Customizing Legends

Control legend position: `theme(legend.position = "top")`

Title: `scale_color_manual(name = "Legend Title", values = c("red", "blue"))`

Statistical Transformations

- **Smooth line**: Add a regression line with `geom_smooth()`
- `ggplot(data, aes(x = var1, y = var2)) + geom_point() + geom_smooth(method = "lm")`

Adding Text or Annotations

- Use **geom_text()** to add labels directly to points.
- `ggplot(data, aes(x = var1, y = var2, label = group)) + geom_point() + geom_text()`

Summary of When to Use Each Plot:

Variables	Plot Type
1 Quantitative	Histogram, Boxplot
2 Quantitative	Scatter Plot, Line Plot
1 Categorical	Bar Plot, Pie Chart
1 Categorical + 1 Quantitative	Boxplot, Violin Plot, Bar Plot
2 Categorical	Stacked Bar Plot, Mosaic Plot
2 Quantitative + 1 Categorical	Scatter Plot (colored), Facets
Multivariate	Heatmap, Pair Plot

```
starbucks_us_by_state <- starbucks |>
  filter(Country == "US") |>
  count(State.Province) |>
  mutate(state_name = str_to_lower(abbr2state(State.Province)))

census_pop_est_2018 <- read_csv("https://mac-stat.github.io/data/us-census-2018-state-pop-est.csv") |>
  separate(state, into = c("dot", "state"), extra = "merge") |>
  select(-dot) |>
  mutate(state = str_to_lower(state))

starbucks_with_2018_pop_est <-
  starbucks_us_by_state |>
  left_join(census_pop_est_2018,
    by = c("state_name" = "state")
  ) |>
  mutate(starbucks_per_10000 = (n / est_pop_2018) * 10000)

head(starbucks_with_2018_pop_est)
...

library(ggthemes)

starbucks_contiguous_us <- starbucks |>
  filter(Country == "US", State.Province != "AK", State.Province != "HI")

ggplot(starbucks_with_2018_pop_est, aes(map_id = state_name, fill = starbucks_per_10000)) +
  geom_map(map = states_map) +
  geom_point(
    data = starbucks_contiguous_us,
    aes(x = Longitude, y = Latitude),
    size = 0.05,
    alpha = 0.2,
    inherit.aes = FALSE
  ) +
  expand_limits(x = states_map$long, y = states_map$lat) +
  theme_map() +
  scale_fill_gradientn(name = "Starbucks per 10,000 citizens", colors = c("darkgreen", "green", "lightgreen"), values = scales::rescale(seq(0, 100, by = 5))) +
  labs(title = "A Choroplethic Map of all Starbucks Locations in the Contiguous United States") +
  labs(caption = "Created by Izzy Flood")

#| fig-cap
#| fig-alt
```