

```

library(ggplot2)

# 1. Visualizing a single variable ----
## 1.1 Binary variables ----
### 1.1a Barplot (count) ----
ggplot(mtcars) + geom_bar(aes(x=vs))

### 1.1b Barplot (proportion) ----
ggplot(mtcars) + geom_bar(aes(x=vs, y=..count../sum(..count..)))

## 1.2 Non-binary categorical ----
### 1.2a Waffle plot ----
# install.packages("waffle", repos = "https://cinc.rud.is")
library(waffle)
library(dplyr)

iris %>%
  count(Species) %>%
  ggplot(aes(fill = Species, values = n)) +
  geom_waffle(size = 1,
              colour = "white",
              na.rm=TRUE,
              flip = TRUE,
              make_proportional = TRUE) +
  theme_void() +
  coord_equal()

## 1.3 Continuous ----
### 1.3a Histogram ----
ggplot(iris) + geom_histogram(aes(x=Sepal.Length))

### 1.3b Density ----
ggplot(iris) + geom_density(aes(x=Sepal.Length))

# 2. Visualizing multiple variables ----
## 2.1 Multiple categorical variables ----
### 2.1a Barplot (proportion) ----
mtcars %>%
  mutate(vs = factor(vs), am = factor(am)) %>%
  ggplot(.) +
  geom_bar(aes(x=vs,
              y=..count../sum(..count..),
              group=am,
              fill=am),
          position = 'dodge')

### 2.1b Mosaic plot ----
# library(ggmosaic)
# library(tidyr)

ggplot(ggmosaic::fly) +
  geom_mosaic(aes(x = product(do_you_recline, rude_to_recline), fill=do_you_recline)) +
  theme(panel.grid = element_blank(),
        legend.position = 'none')

### 2.1c Categorical heatmap ----
ggmosaic::fly %>%

```

```

count(rude_to_recline, do_you_recline, .drop = FALSE) %>%
ggplot(.) +
  geom_tile(aes(x = rude_to_recline,
                y = do_you_recline,
                fill = n)) +
  scale_fill_gradient(low = 'navy', high = "yellow")

## 2.2 Multiple continuous variables ----
### 2.2a Scatterplot ----
ggplot(iris) +
  geom_point(aes(x=Petal.Length,
                y=Petal.Width,
                color=Species))

### 2.2b Two-dimensional density ----
ggplot(iris, aes(x=Petal.Length, y=Petal.Width)) +
  stat_density2d(aes(fill = ..density..),
                geom = "raster", contour = FALSE) +
  scale_fill_gradient(low = "black", high = "white")

## 2.3 Hybrid ----
### 2.3a Jitterplot ----
ggplot(mtcars) +
  geom_jitter(aes(x=cyl, y=am))

# You can also specify 'width' and 'height'
# as additional arguments to refine the
# jitter. Remember, jitter is random:
# jitter plots will change whenever you
# re-run the code.

### 2.3b Barplot with +/- indicator ----
library(gcookbook)
climate_sub <- climate %>%
  filter(Source == "Berkeley" & Year >= 1900) %>%
  mutate(pos = Anomaly10y >= 0)

ggplot(climate_sub, aes(x = Year, y = Anomaly10y, fill = pos)) +
  geom_col(position = "identity", colour = "black", size = 0.25) +
  scale_fill_manual(values = c("#CCEEFF", "#FFDDDD"), guide = 'none')

### 2.3c Cleveland dot plot ----
library(gcookbook)
tophit <- tophitters2001[1:15, ]

ggplot(tophit) +
  geom_point(aes(x = avg, y = reorder(name, avg)))

# The `reorder` function is helpful
# when making plots of a non-binary categorical
# variable and a corresponding continuous variable.

```