DSBA 5122: Visual Analytics

Class 3: Visual Representations Basics II

Ryan Wesslen

February 4, 2019

Basic Principles of Visualization: Cairo, Chapter 5

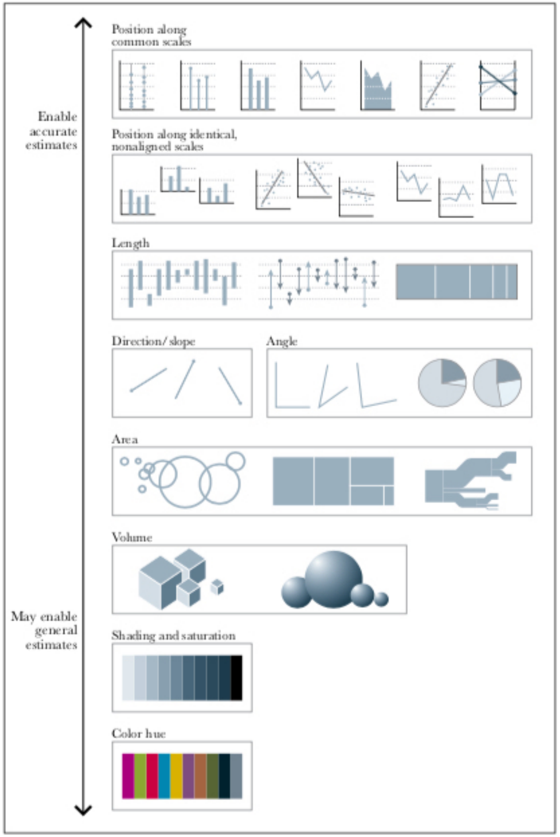

Position along common scales
Enable accurate estimates
Position along identical, nonaligned scales
Length
Direction/ slope
Angle
Area
Volume
May enable general estimates
Shading and saturation
Color hue

Figure 5.5 Scale of elementary perceptual tasks, inspired by William Cleveland and Robert McGill.

What if we want to show "high" and "low" levels?

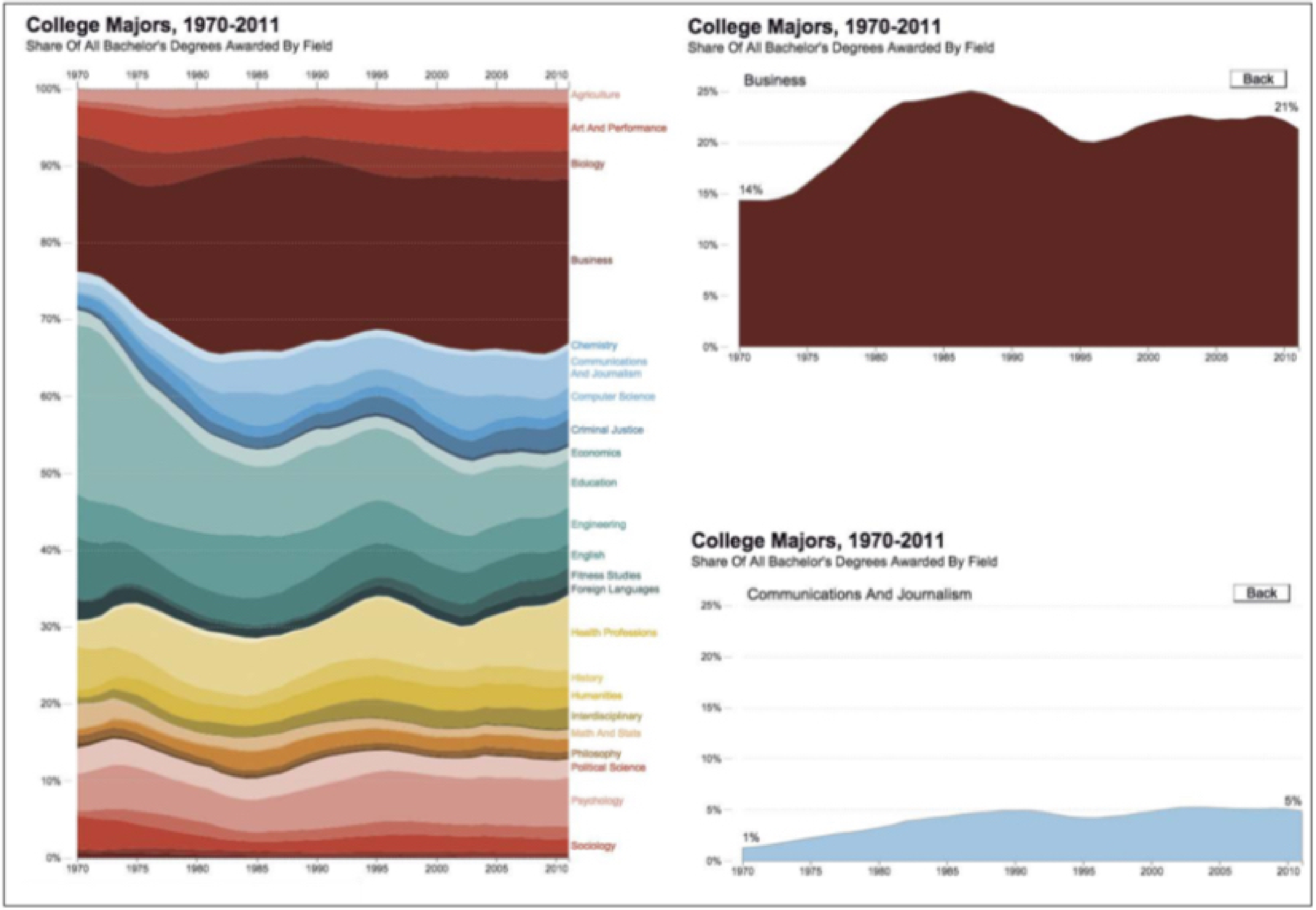


Figure 5.9 Visualization by NPR, http://www.npr.org/sections/money/2014/05/09/310114739/whats-your-major-four-decades-of-college-degrees-in-1-graph.

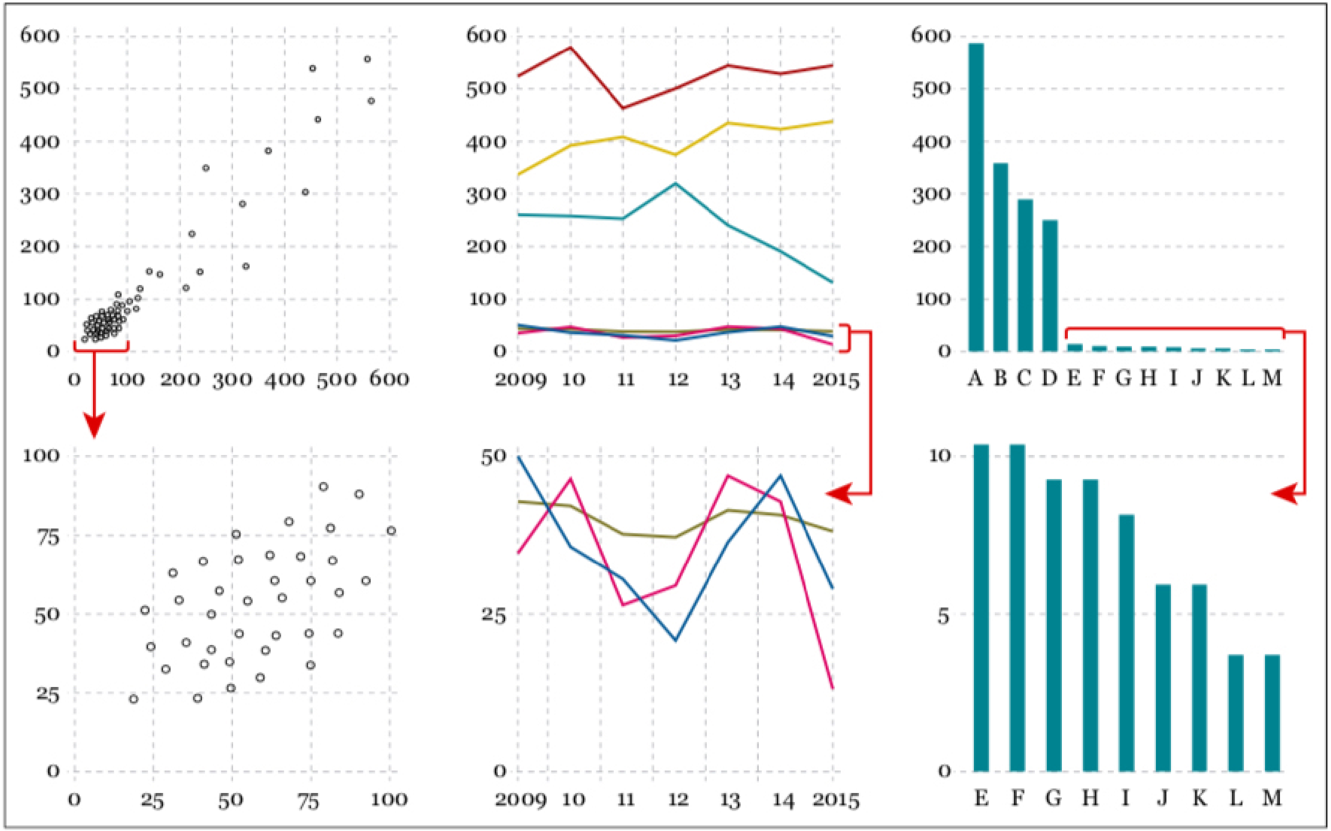


Figure 5.14 Two different scales for subsets of the same data.

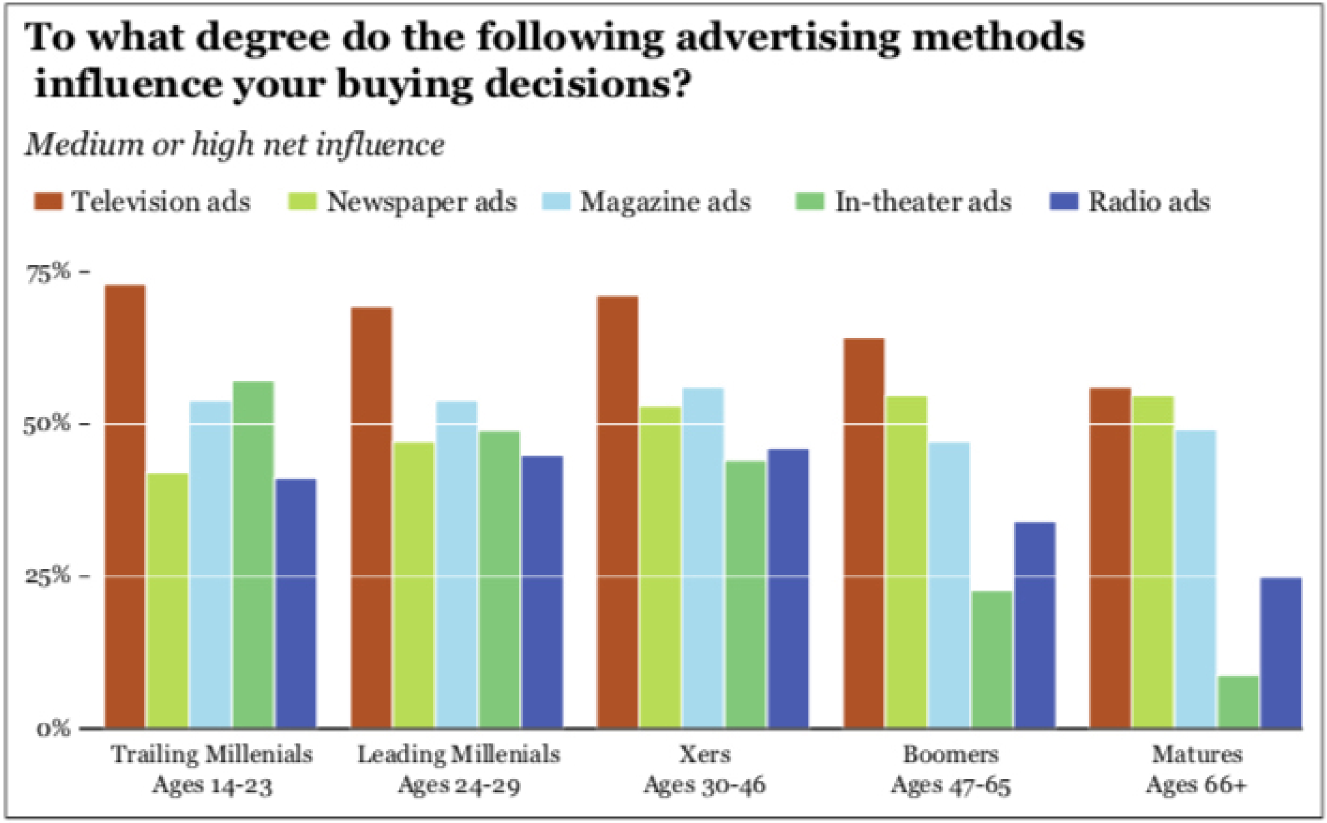


Figure 5.15 Data source: Deloitte's Digital Democracy Survey.

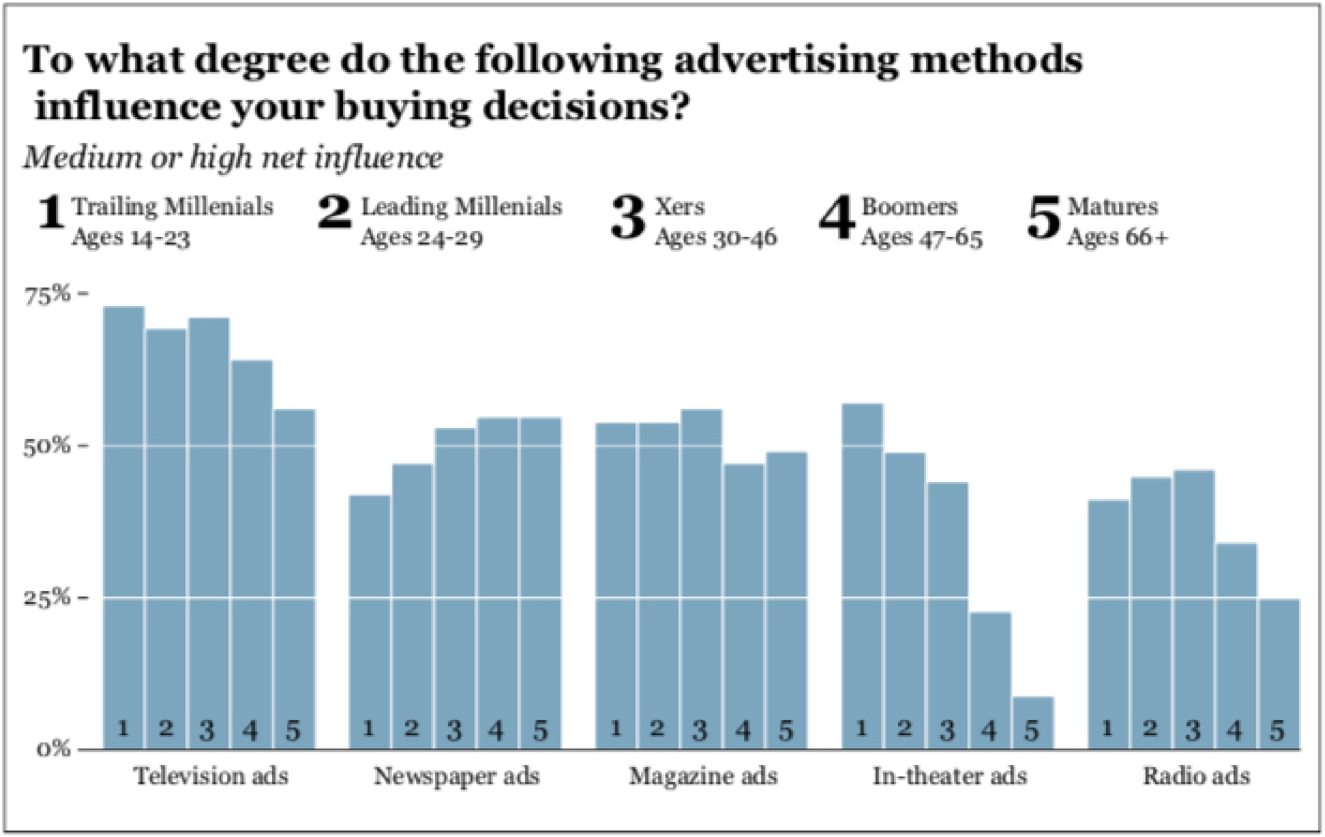


Figure 5.16 Reorganizing the data from Figure 5.15.

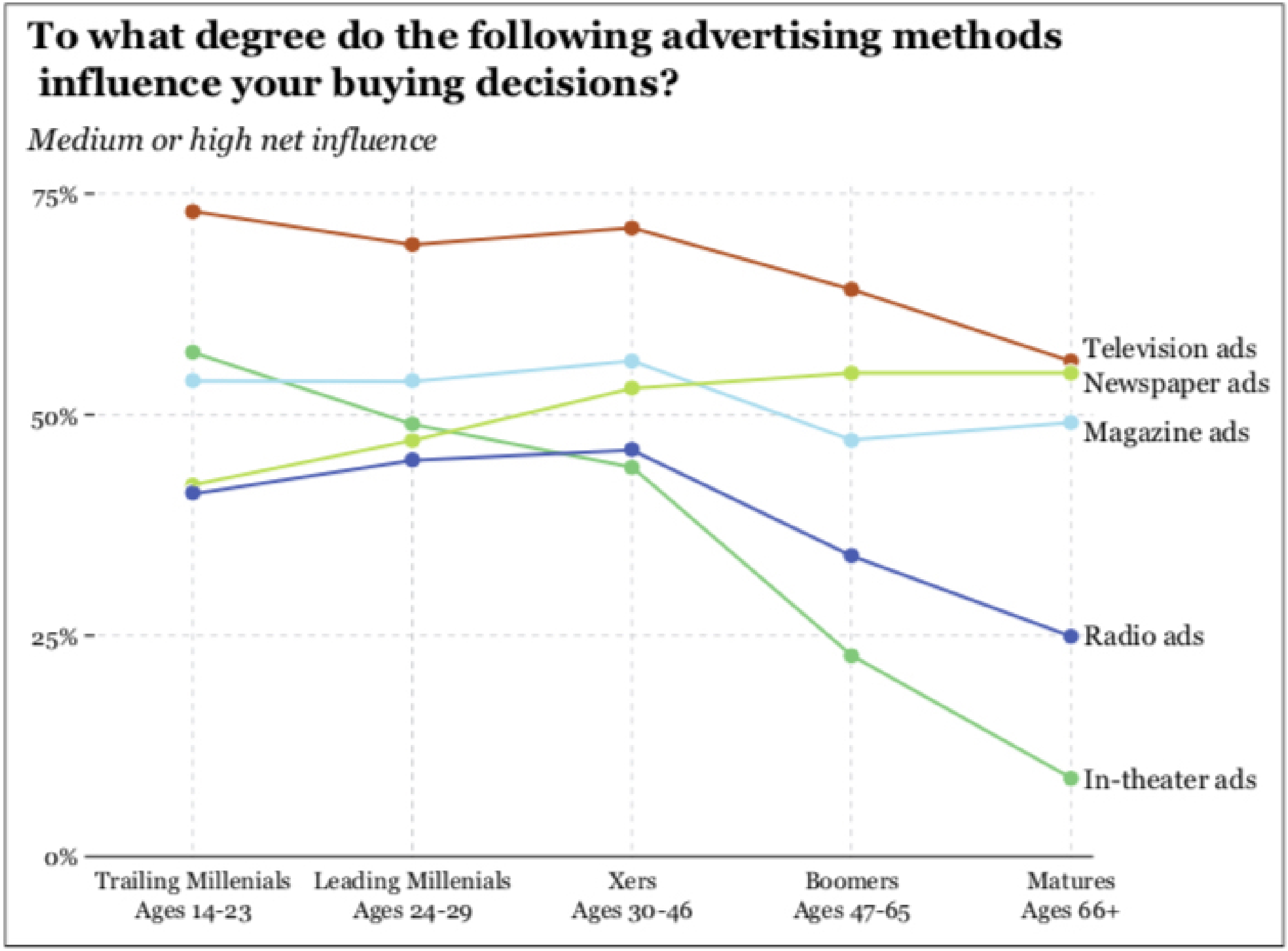


Figure 5.17 Line chart with the same data used in Figure 5.15.

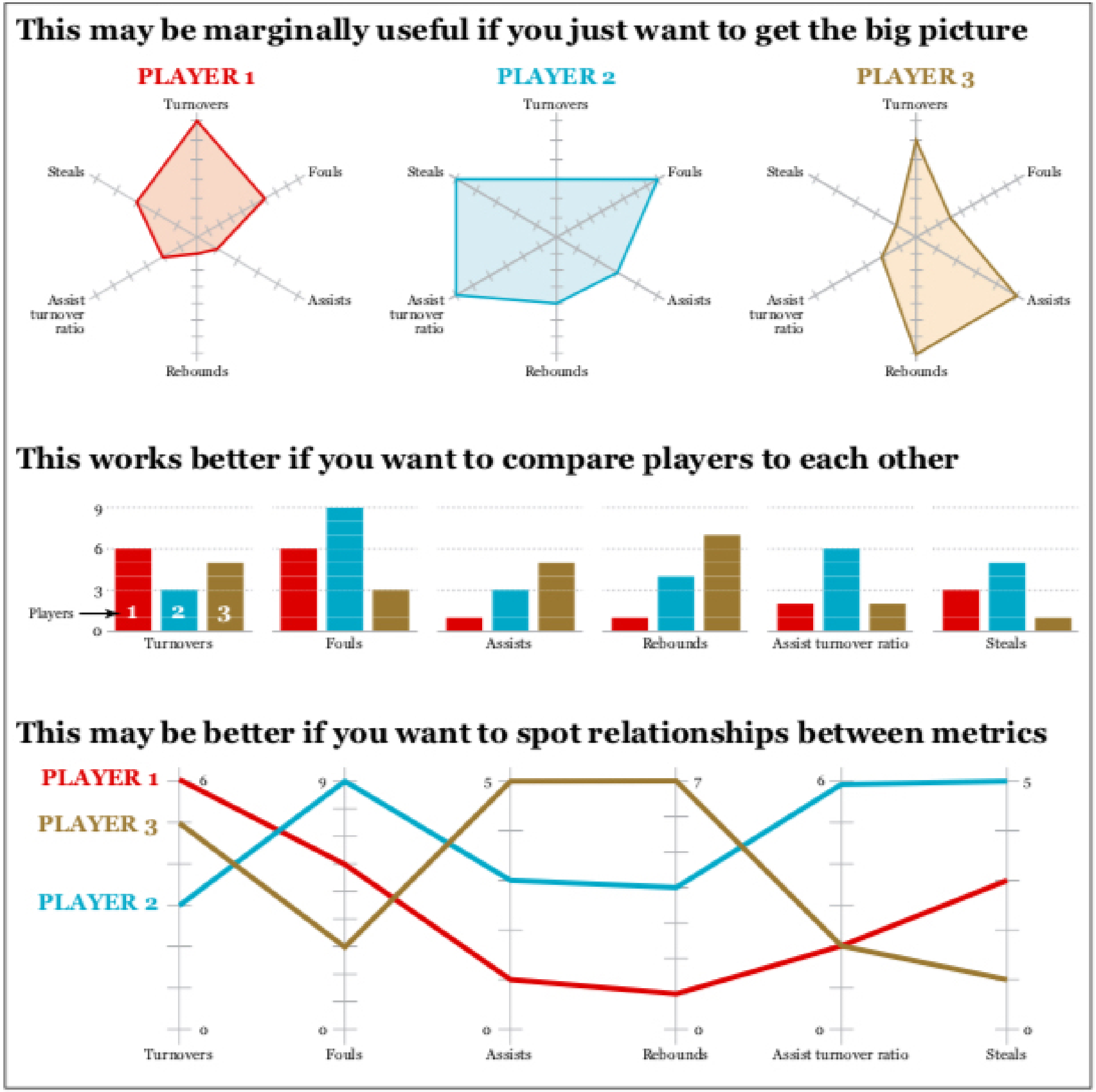


Figure 5.19 Radar charts aren't usually very effective. These are all fake charts, by the way.

Directory of Visualizations: Wilke, Chapter 5 (with tidyverse)

```
library(tidyverse)
```

```
## ── Attaching packages ──────────────────────────────────────────────── tidyverse 1.2.1 ──
```

```
## ✔ ggplot2 3.1.0     ✔ purrr   0.3.0
## ✔ tibble  2.0.1     ✔ dplyr   0.7.8
## ✔ tidyr   0.8.2     ✔ stringr 1.3.1
## ✔ readr   1.3.1     ✔ forcats 0.3.0
```

```
## Warning: package 'tibble' was built under R version 3.5.2
```

```
## Warning: package 'purrr' was built under R version 3.5.2
```

```
## ── Conflicts ─────────────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
```

For this section, I'm going to use the **mpg** dataset.

```
head(mpg,n=5)
```

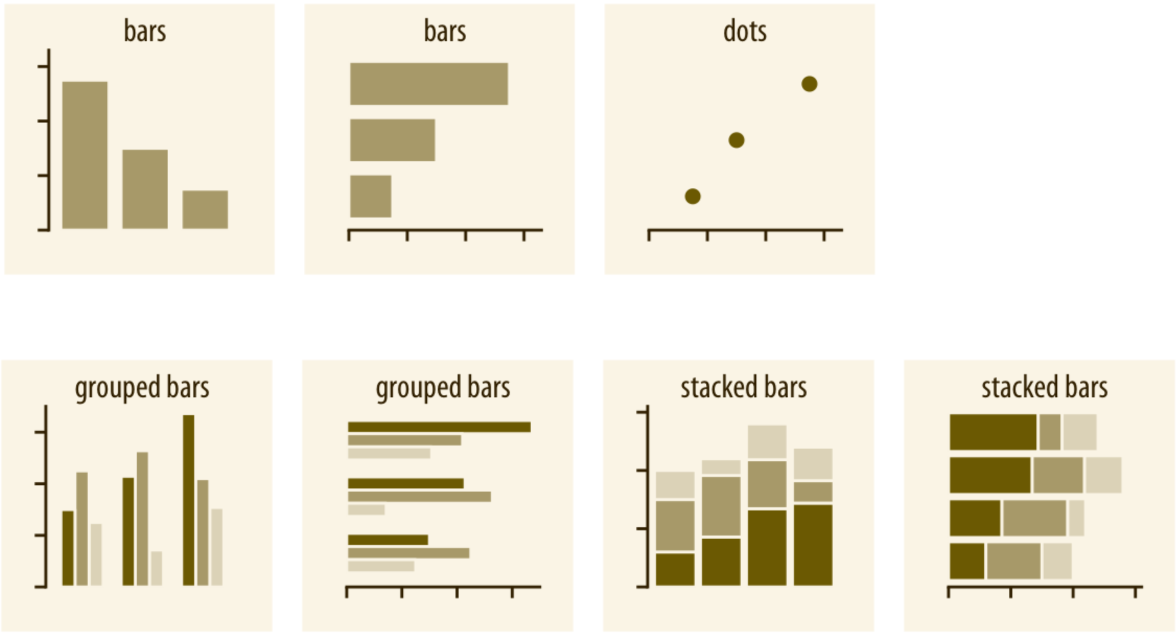
```
## # A tibble: 5 x 11
##   manufacturer model displ  year   cyl trans  drv     cty   hwy fl    class
##   <chr>        <chr> <dbl> <int> <int> <chr>  <chr> <int> <int> <chr> <chr>
## 1 audi         a4      1.8  1999     4 auto(… f        18    29 p     comp…
## 2 audi         a4      1.8  1999     4 manua… f        21    29 p     comp…
## 3 audi         a4      2    2008     4 manua… f        20    31 p     comp…
## 4 audi         a4      2    2008     4 auto(… f        21    30 p     comp…
## 5 audi         a4      2.8  1999     6 auto(… f        16    26 p     comp…
```

```
# glimpse is from dplyr
glimpse(mpg)
```

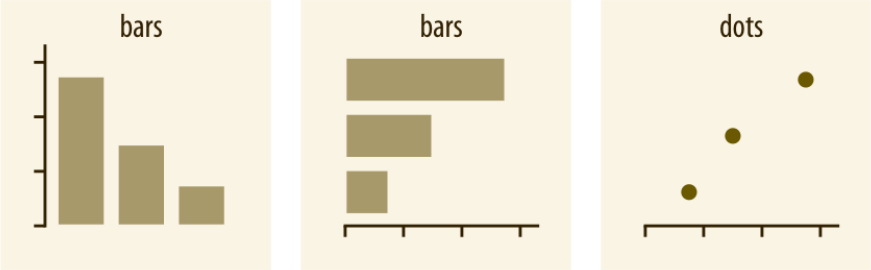
```
## Observations: 234
## Variables: 11
## $ manufacturer <chr> "audi", "audi", "audi", "audi", "audi", "audi", "au…
## $ model        <chr> "a4", "a4", "a4", "a4", "a4", "a4", "a4", "a4 quatt…
## $ displ        <dbl> 1.8, 1.8, 2.0, 2.0, 2.8, 2.8, 3.1, 1.8, 1.8, 2.0, 2…
## $ year         <int> 1999, 1999, 2008, 2008, 1999, 1999, 2008, 1999, 199…
## $ cyl          <int> 4, 4, 4, 4, 6, 6, 6, 4, 4, 4, 4, 6, 6, 6, 6, 6, 6, …
## $ trans        <chr> "auto(l5)", "manual(m5)", "manual(m6)", "auto(av)",…
## $ drv          <chr> "f", "f", "f", "f", "f", "f", "f", "4", "4", "4", "…
## $ cty          <int> 18, 21, 20, 21, 16, 18, 18, 18, 16, 20, 19, 15, 17,…
## $ hwy          <int> 29, 29, 31, 30, 26, 26, 27, 26, 25, 28, 27, 25, 25,…
## $ fl           <chr> "p", "p", "p", "p", "p", "p", "p", "p", "p", "p", "…
## $ class        <chr> "compact", "compact", "compact", "compact", "compac…
```

Amounts

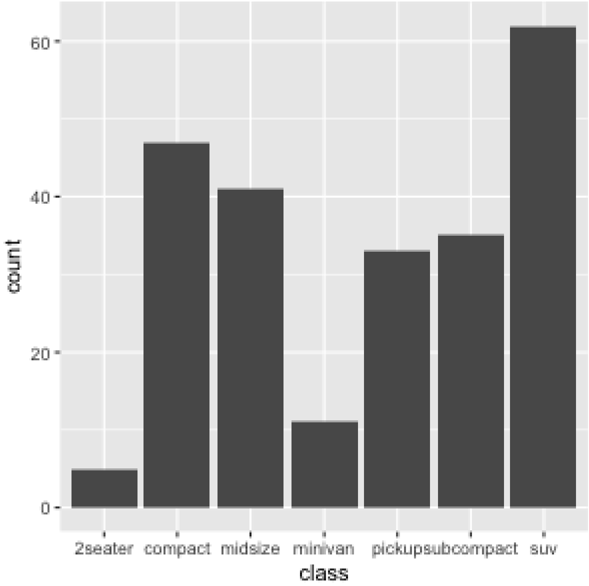
Descriptive statistics like averages and counts by one or two categorical groups (covariates or features). These use **absolute values**, rather than values, therefore **scale matters**.



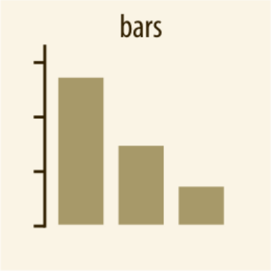
Amounts

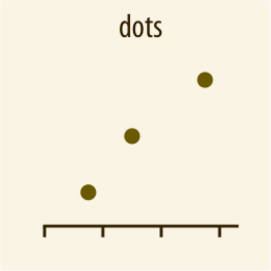


```r
ggplot(mpg, aes(x = class)) +
  geom_bar()
```

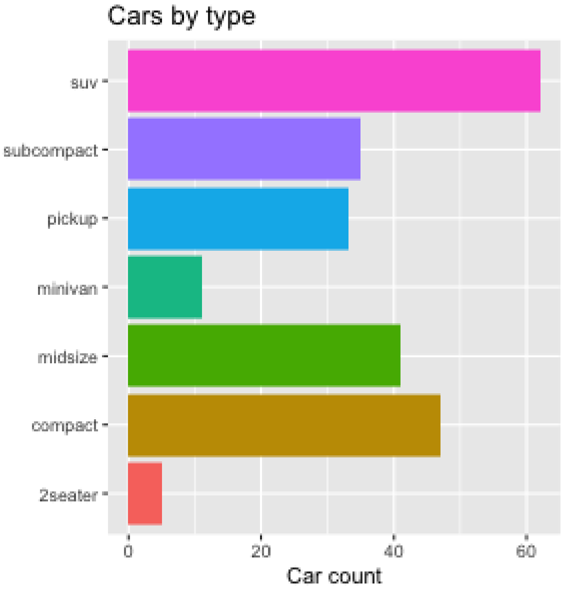


Amounts

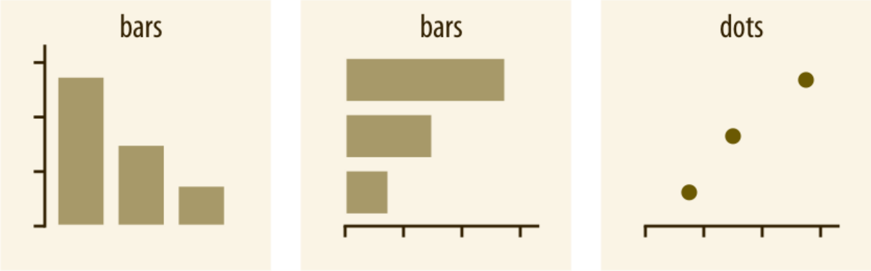




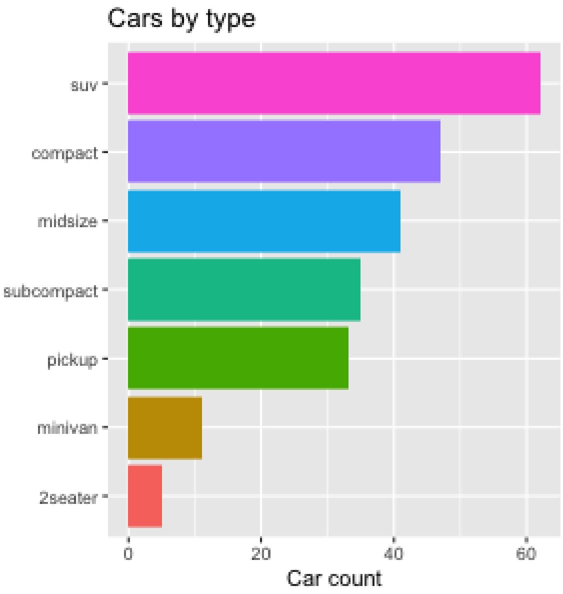
```
ggplot(mpg, aes(x = class, fill = class)) +
  geom_bar() +
  coord_flip() +
  labs(x = " ", y = "Car count",
       title = "Cars by type") +
  theme(legend.position = "none")
```



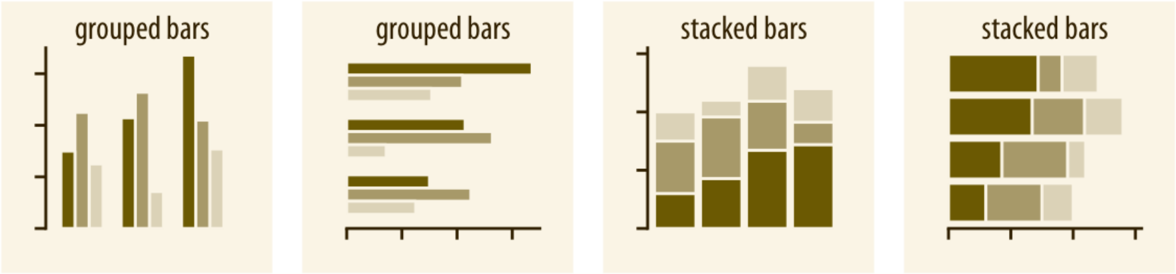
Amounts



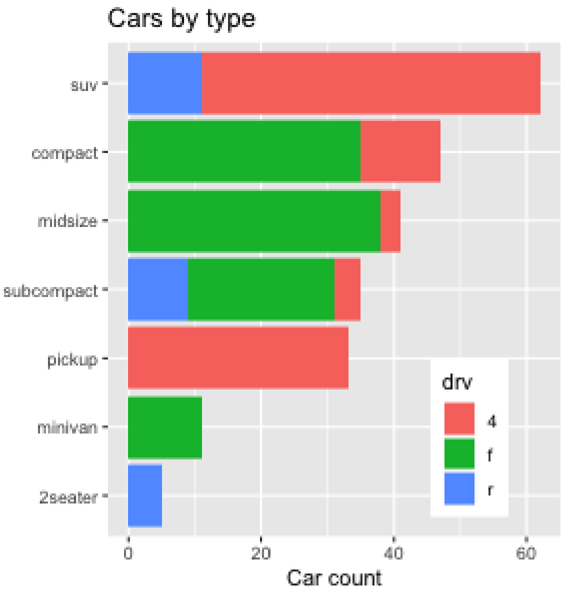
```
l <- c("2seater","minivan","pickup",
       "subcompact","midsize","compact","suv")

mpg %>%
  mutate(class = factor(class, levels = l)) %>%
  ggplot(aes(x = class, fill = class)) +
  geom_bar() +
  coord_flip() +
  labs(x = " ", y = "Car count",
       title = "Cars by type") +
  theme(legend.position = "none")
```



Amounts

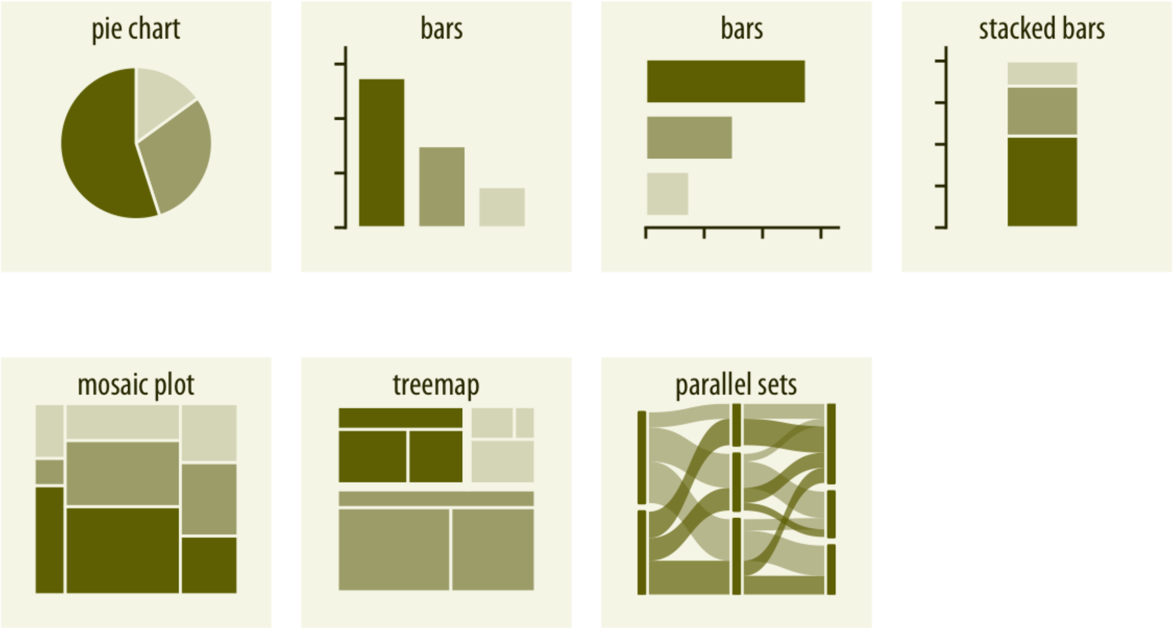


```r
l <- c("2seater","minivan","pickup",
       "subcompact","midsize","compact","suv")

mpg %>%
  mutate(class = factor(class, levels = l)) %>%
  ggplot(aes(x = class, fill = drv)) +
  geom_bar() +
  coord_flip() +
  labs(x = " ", y = "Car count",
       title = "Cars by type") +
  theme(legend.position = c(0.8,0.2))
```

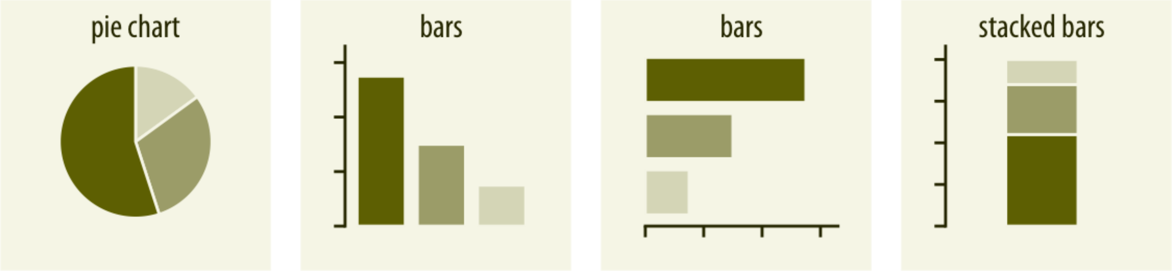


Proportions

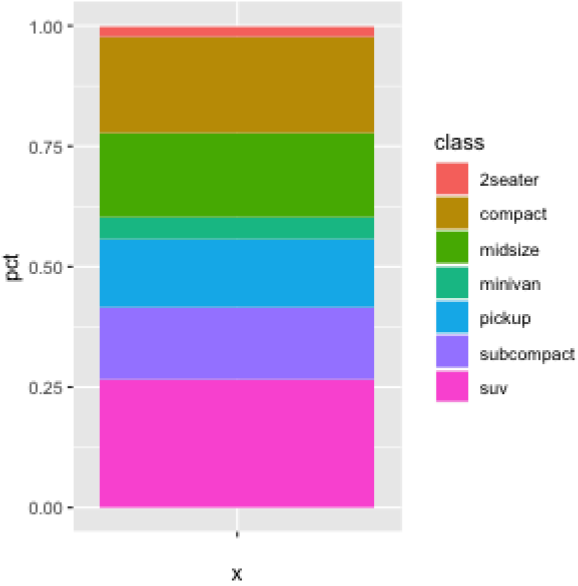
Relative values to compare sizes of categories.



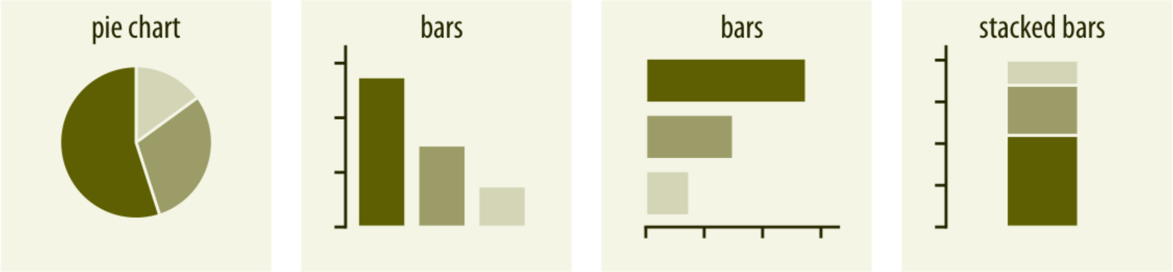
Proportions



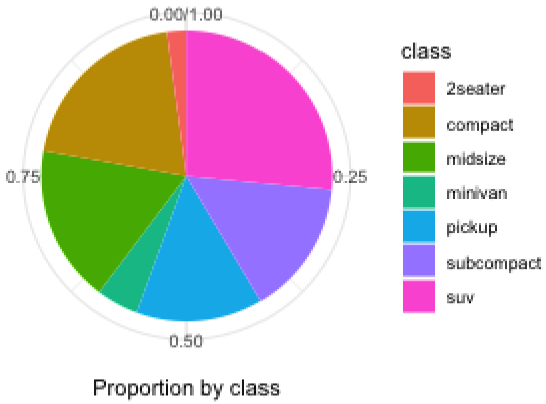
```r
p <- mpg %>%
  count(class) %>%
  mutate(pct = n / sum(n)) %>%
  ggplot(aes(x = "", y = pct, fill = class)) +
  geom_bar(width = 1, stat = "identity")

p
```



Proportions



```
p <- mpg %>%
  count(class) %>%
  mutate(pct = n / sum(n)) %>%
  ggplot(aes(x = "", y = pct, fill = class)) +
  geom_bar(width = 1, stat = "identity")

p + coord_polar("y", start=0) +
  theme_minimal() +
  labs(x = " ", y = "Proportion by class")
```



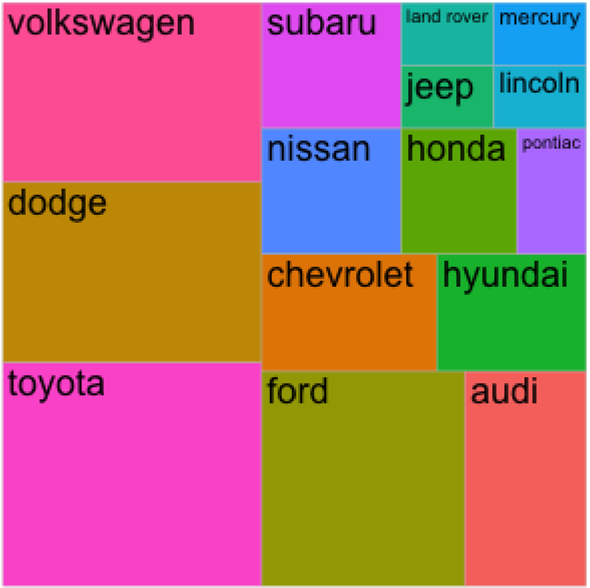
Proportions

mosaic plot

treemap

parallel sets

```
library(treemapify)
mpg %>%
  filter(year == 1999) %>%
  count(manufacturer) %>%
  ggplot(aes(area = n,
             fill = manufacturer,
             label = manufacturer)) +
  geom_treemap() +`
  geom_treemap_text() +
  theme(legend.position = "none")
```

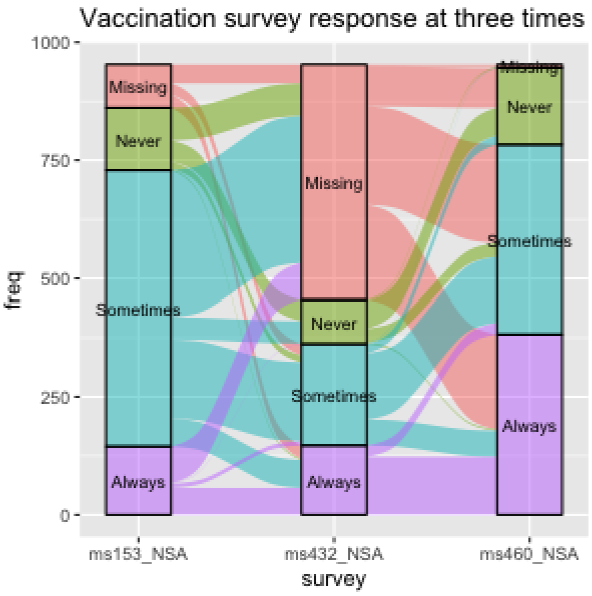


Proportions

mosaic plot

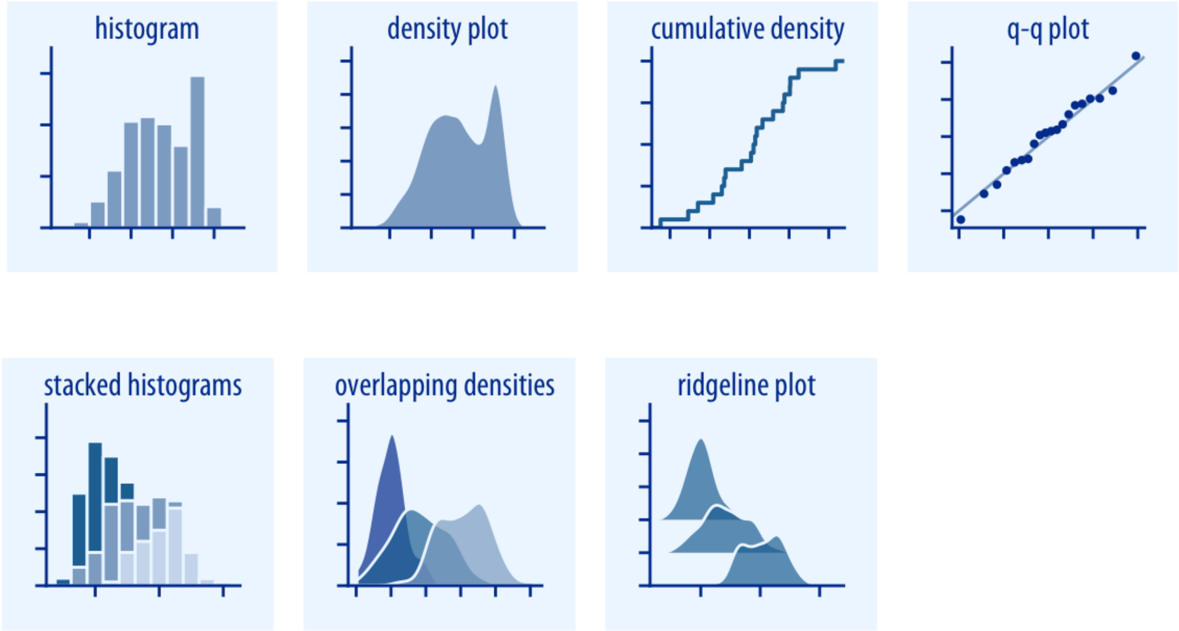
parallel sets

```
library(ggalluvial)

data(vaccinations)

ggplot(vaccinations,
       aes(x = survey, y = freq,
           alluvium = subject, stratum = respons
           fill = response, label = response)) +
  scale_x_discrete(expand = c(.1, .1)) +
  geom_flow() +
  geom_stratum(alpha = .5) +
  geom_text(stat = "stratum", size = 3) +
  theme(legend.position = "none") +
  labs(title = "Vaccination survey response at t
```

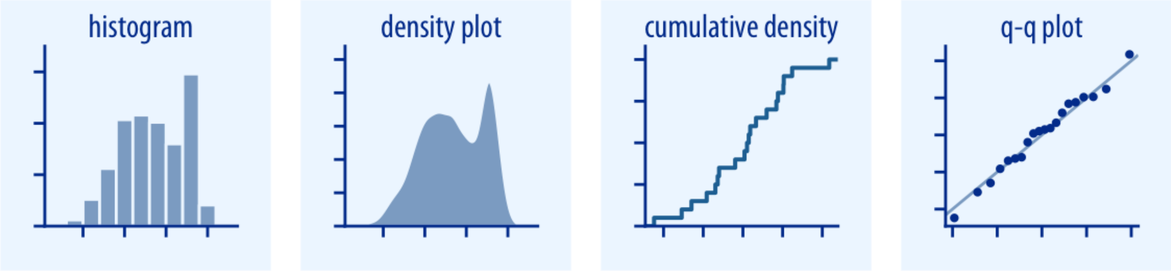


Distributions

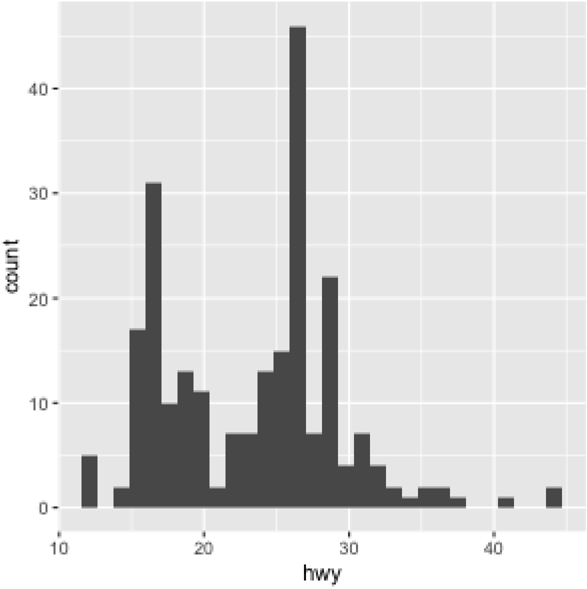
What is the variance? How evenly spread are the values?



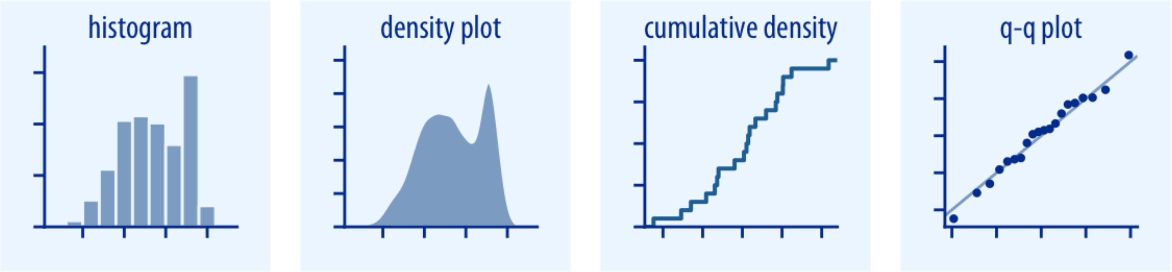
Distributions



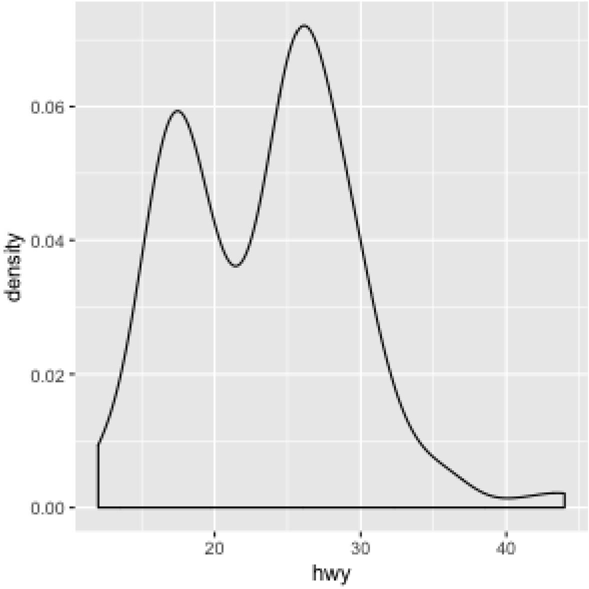
```
ggplot(mpg, aes(x = hwy)) +
  geom_histogram()
```



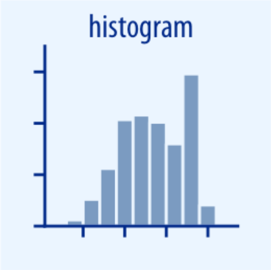
Distributions

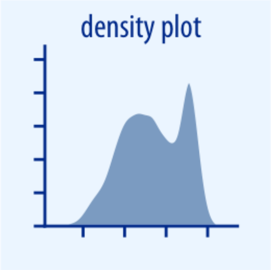


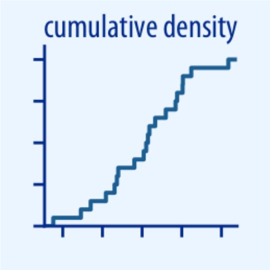
```
ggplot(mpg, aes(x = hwy)) +
  geom_density()
```

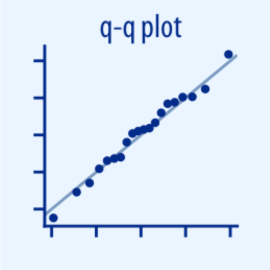


Distributions

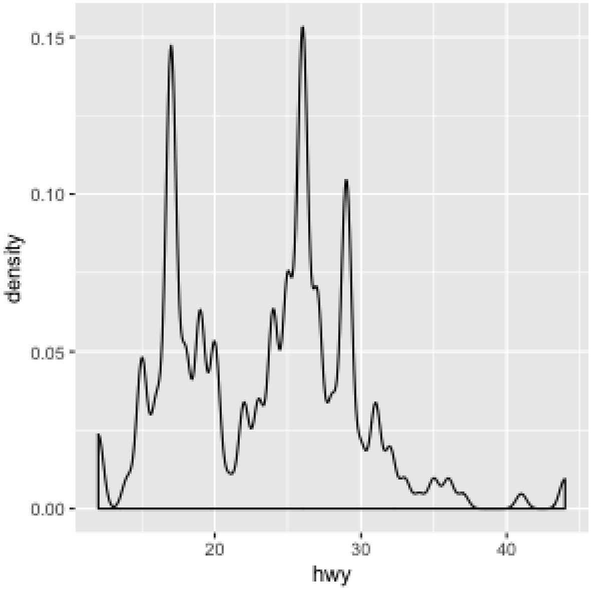






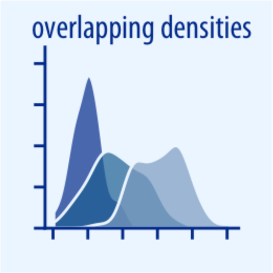


```r
ggplot(mpg, aes(x = hwy)) +
  geom_density(adjust = 0.2) # adjust kernel
```



Distributions

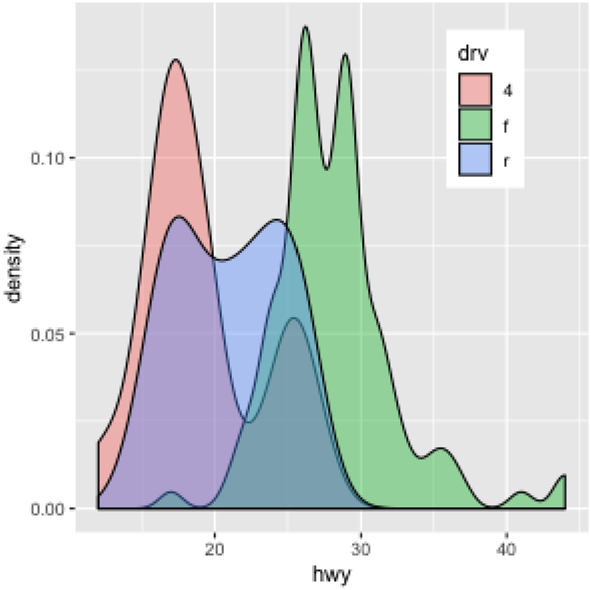
stacked histograms



overlapping densities

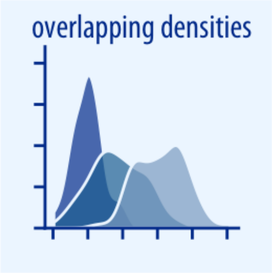
ridgeline plot

```
ggplot(mpg, aes(x = hwy, fill = drv)) +
  geom_density(alpha = 0.4) +
  theme(legend.position = c(0.8,0.8))
```



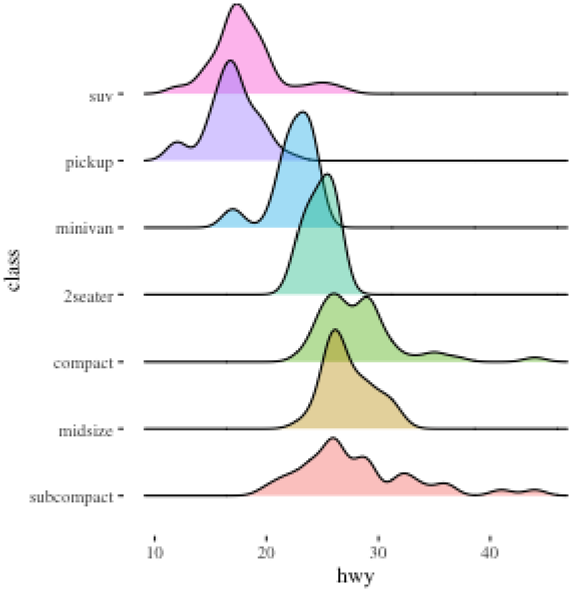
Distributions

stacked histograms



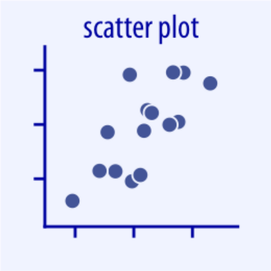
ridgeline plot

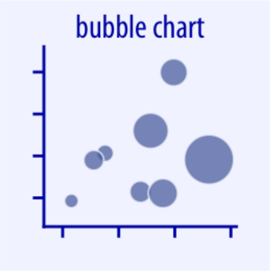
```r
library(ggridges)
library(ggthemes)

l2 <- c("subcompact","midsize","compact",
       "2seater","minivan","pickup","suv")

mpg %>%
  mutate(class = factor(class, levels = l2)) %>%
  ggplot(aes(x = hwy, y = class, fill = class))
  geom_density_ridges(alpha = 0.4) +
  theme_tufte() +
  theme(legend.position = "none")
```

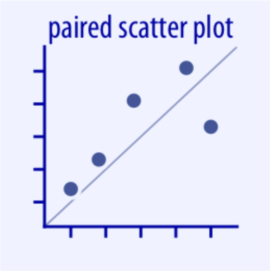


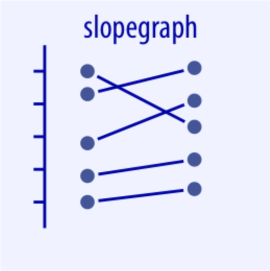
x-y relationships

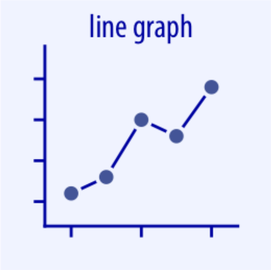
What is the relationship between two or more variables?

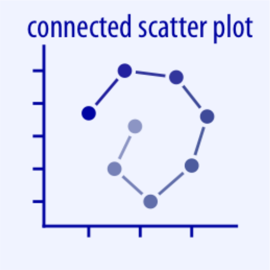


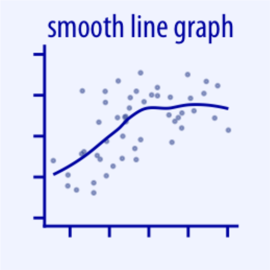




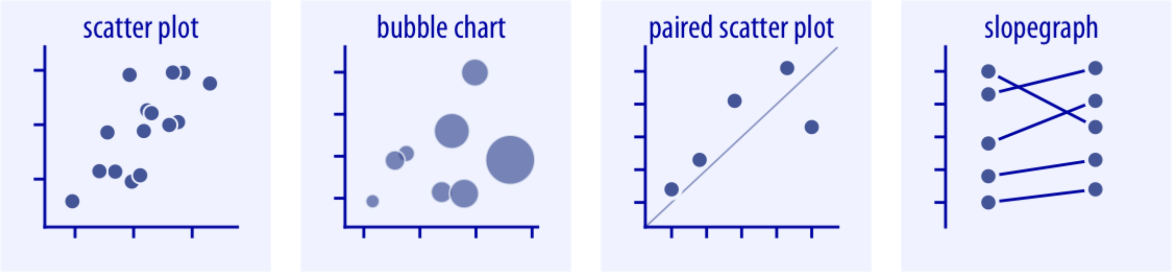




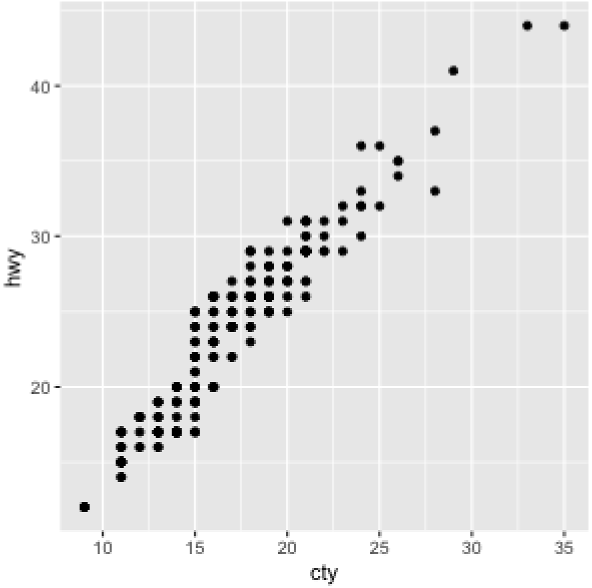




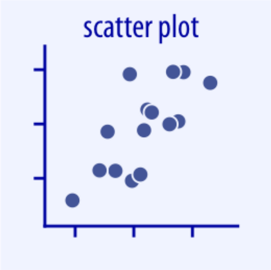
x-y relationships

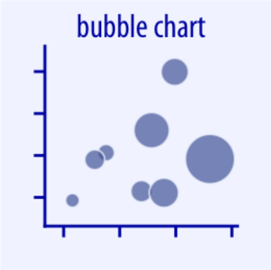


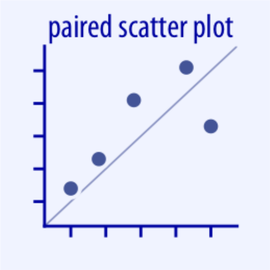
```
ggplot(mpg, aes(x = cty, y = hwy)) +
  geom_point()
```

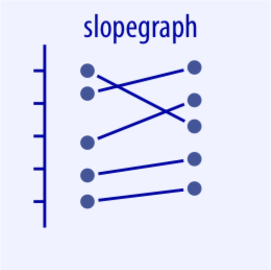


x-y relationships

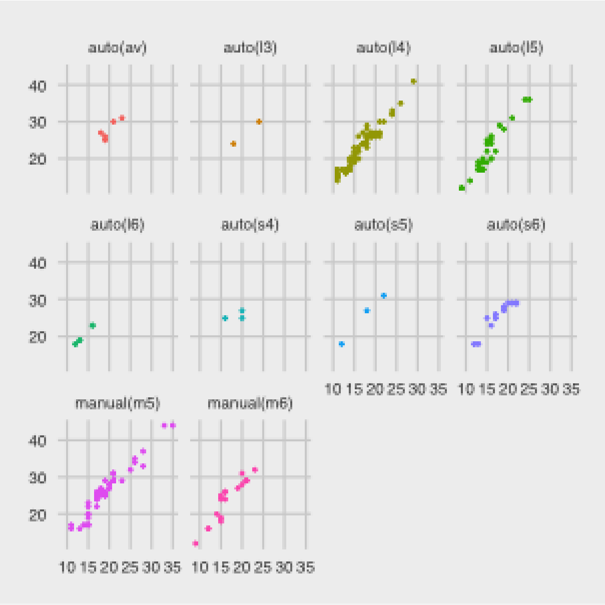




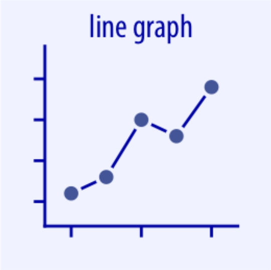


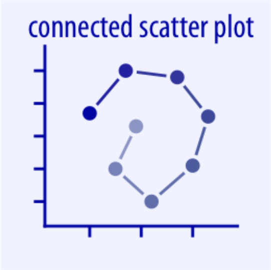


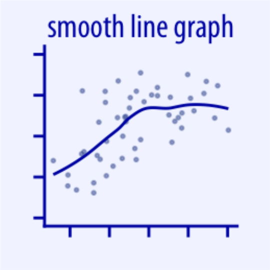
```
library(ggthemes)

ggplot(mpg, aes(x = cty, y = hwy)) +
  geom_point(aes(color = trans), size = 0.5) +
  facet_wrap(~trans) +
  theme_fivethirtyeight() +
  theme(legend.position = "none",
        text = element_text(size=10))
```



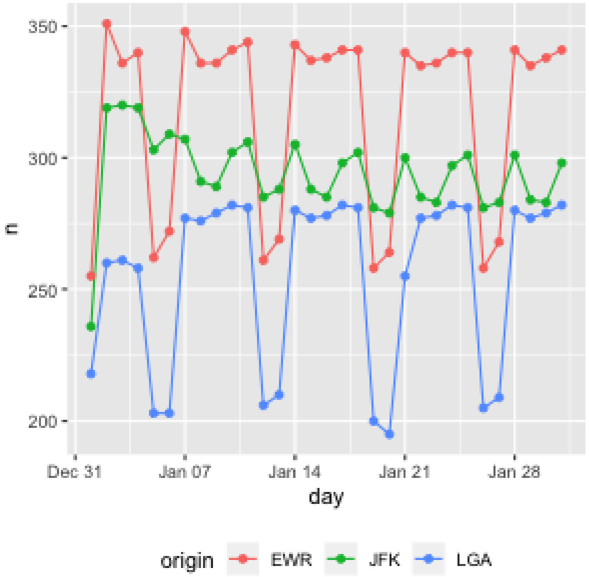
x-y relationships



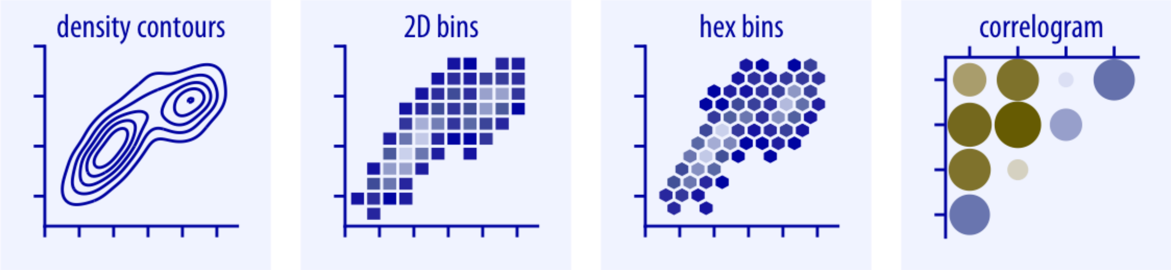




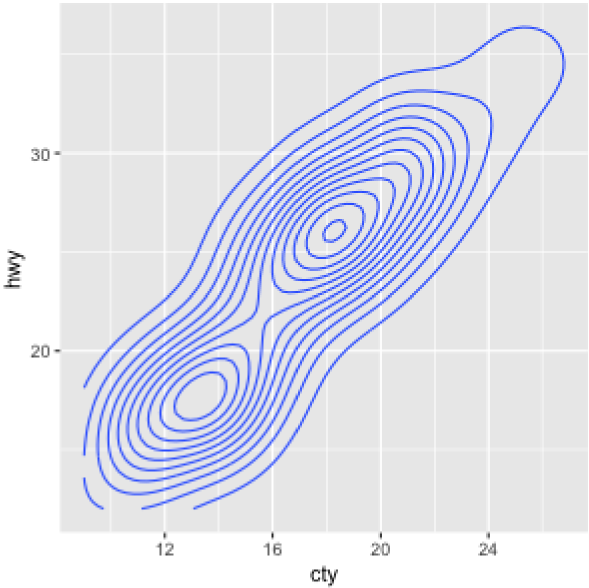
```
library(nycflights13)

# break up by data manipulation
df <- flights %>%
  mutate(day=as.Date(time_hour)) %>%
  filter(day < "2013-02-01") %>%
  count(day,origin)

# and ggplot
ggplot(df, aes(x=day, y=n, color=origin)) +
  geom_line(aes(group=origin)) +
  geom_point() +
  theme(legend.position="bottom")
```



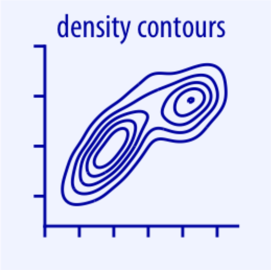
x-y relationships

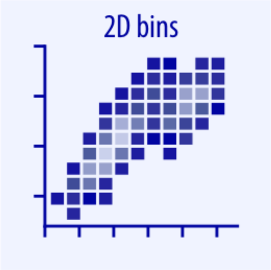


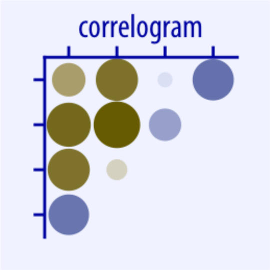
```
filter(mpg, class != "2seater") %>%
  ggplot(aes(x = cty, y = hwy)) +
  geom_density_2d()
```



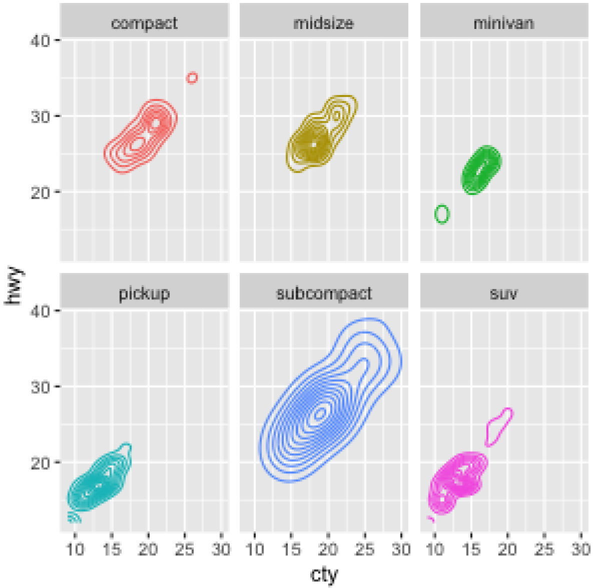
x-y relationships



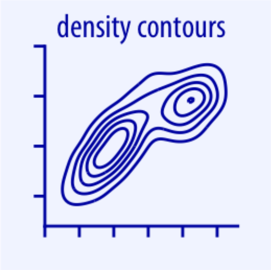


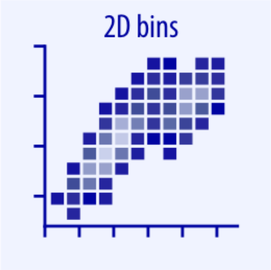


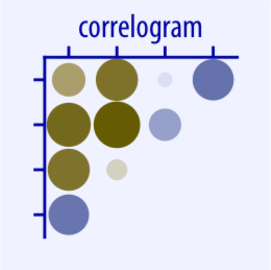
```
filter(mpg, class != "2seater") %>%
  ggplot(aes(x = cty, y = hwy)) +
  geom_density_2d(aes(color = class)) +
  facet_wrap(~class) +
  theme(legend.position = "none")
```



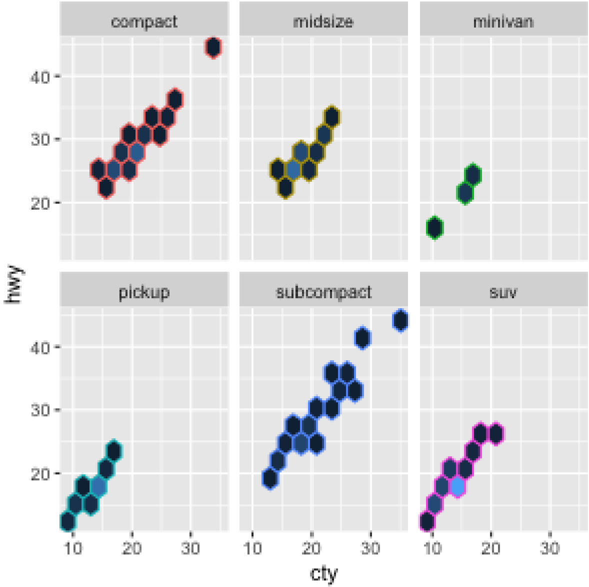
x-y relationships



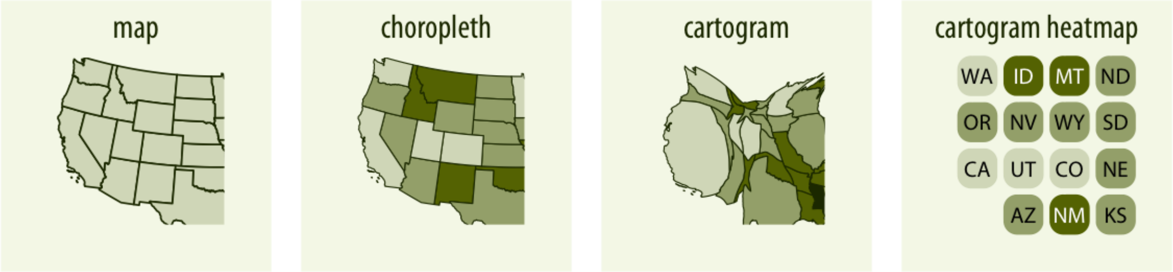




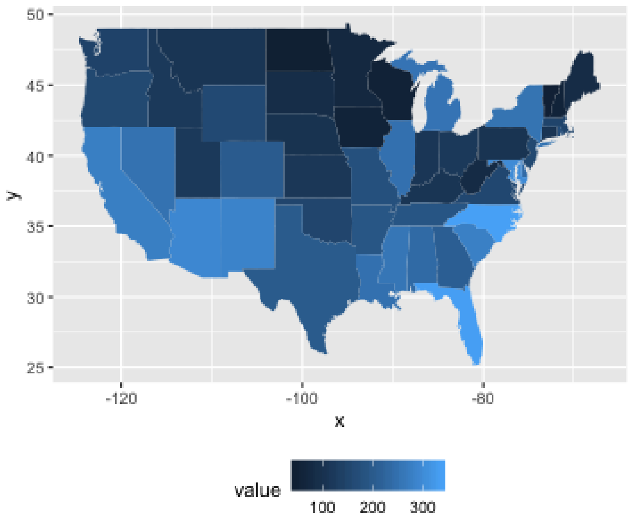
```
filter(mpg, class != "2seater") %>%
  ggplot(aes(x = cty, y = hwy)) +
  geom_hex(aes(color = class), bins = 10) +
  facet_wrap(~class) +
  theme(legend.position = "none")
```



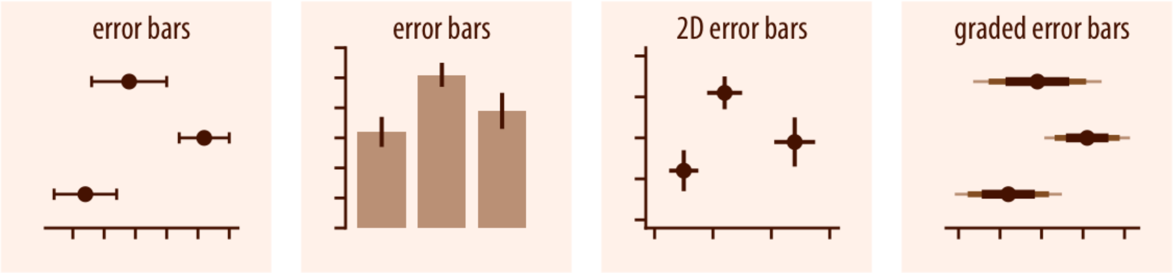
Geospatial



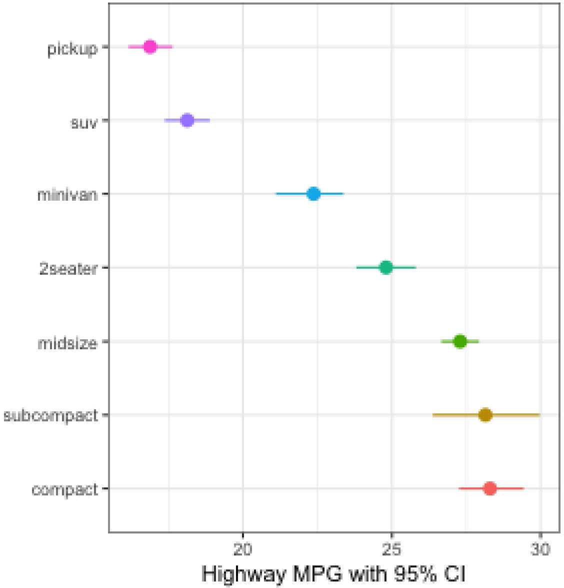
```r
library(maps)

crimes <- USArrests %>%
  rownames_to_column(var = "state") %>%
  mutate(state = tolower(state)) %>%
  gather("variable","value",-state)

states_map <- map_data("state")

crimes %>%
  filter(variable == "Assault") %>%
  ggplot(aes(map_id = state)) +
  geom_map(aes(fill = value), map = states_map)
  expand_limits(x = states_map$long,
                y = states_map$lat) +
  theme(legend.position = "bottom")
```



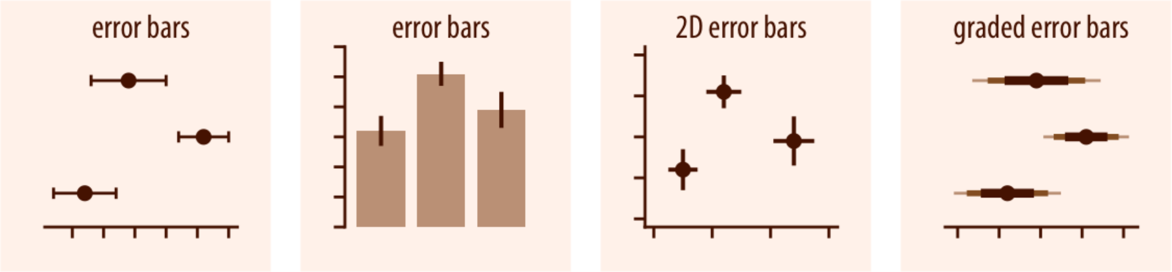
Uncertainty



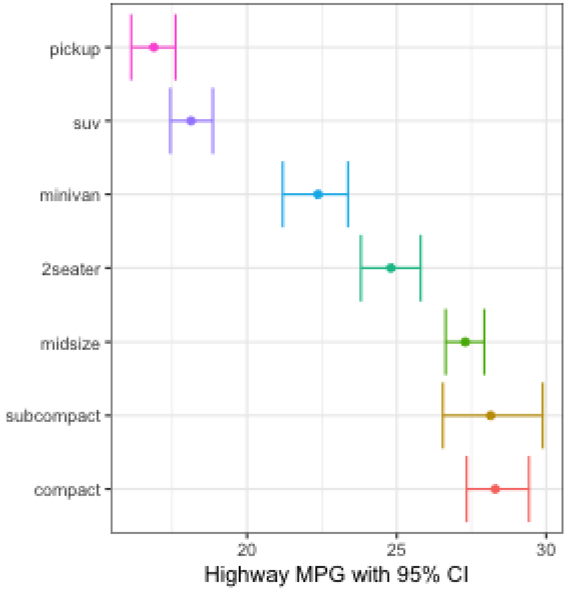
```
l3 <- c("compact","subcompact","midsize",
       "2seater","minivan","suv","pickup")

# avg highway mpg with boostrapped 95% CI
mpg %>%
  mutate(class = factor(class, levels = l3)) %>%
  ggplot(aes(x = class, y = hwy, color = class))
  stat_summary(fun.y = mean, geom = "point") +
  `stat_summary(fun.data = mean_cl_boot,
               geom = "pointrange")` +
  theme_bw() +
  coord_flip() +
  theme(legend.position = "none") +
  labs(x = " ", y = "Highway MPG with 95% CI")
```



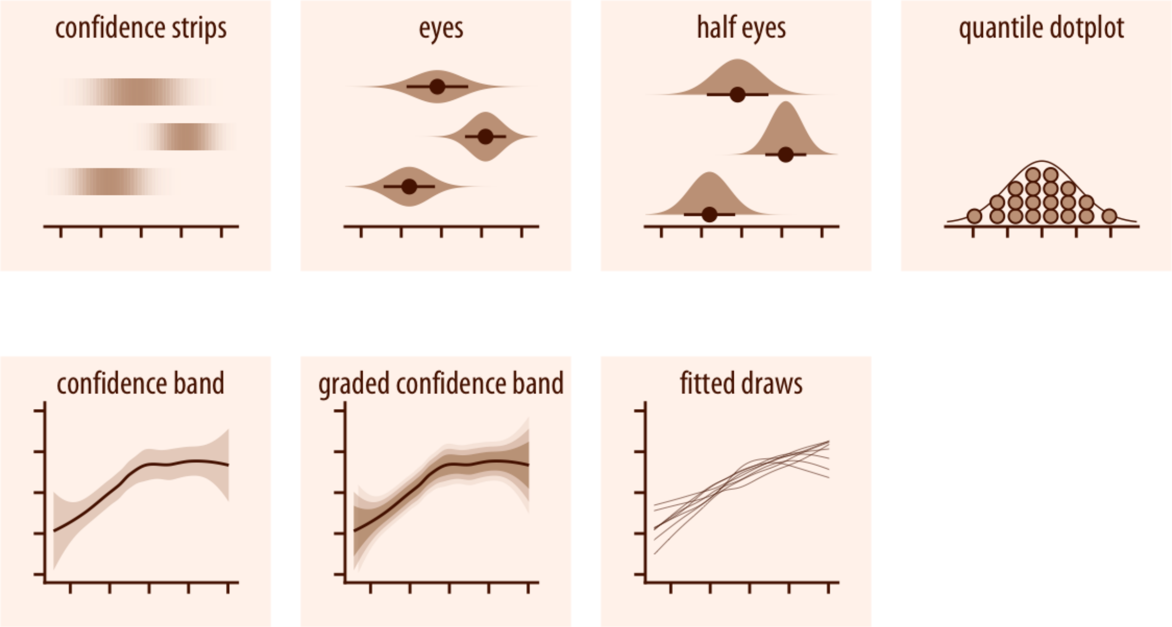
Uncertainty



```
l3 <- c("compact","subcompact","midsize",
       "2seater","minivan","suv","pickup")

# avg highway mpg with boostrapped 95% CI
mpg %>%
  mutate(class = factor(class, levels = l3)) %>%
  ggplot(aes(x = class, y = hwy, color = class))
  stat_summary(fun.y = mean, geom = "point") +
  stat_summary(fun.data = mean_cl_boot,
               geom = "errorbar") +
  theme_bw() +
  coord_flip() +
  theme(legend.position = "none") +
  labs(x = " ", y = "Highway MPG with 95% CI")
```



Uncertainty


confidence strips
eyes
half eyes
quantile dotplot
confidence band
graded confidence band
fitted draws