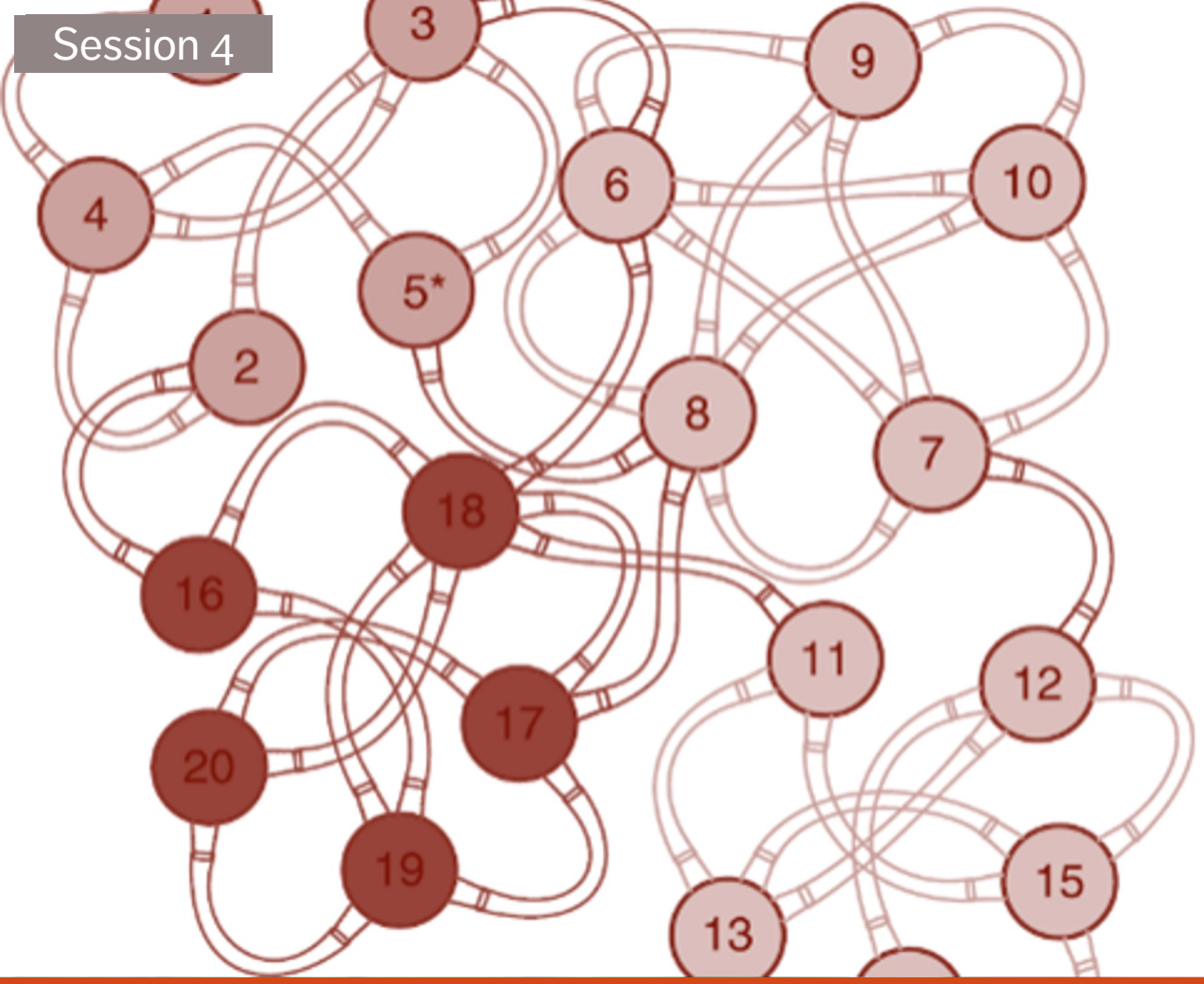
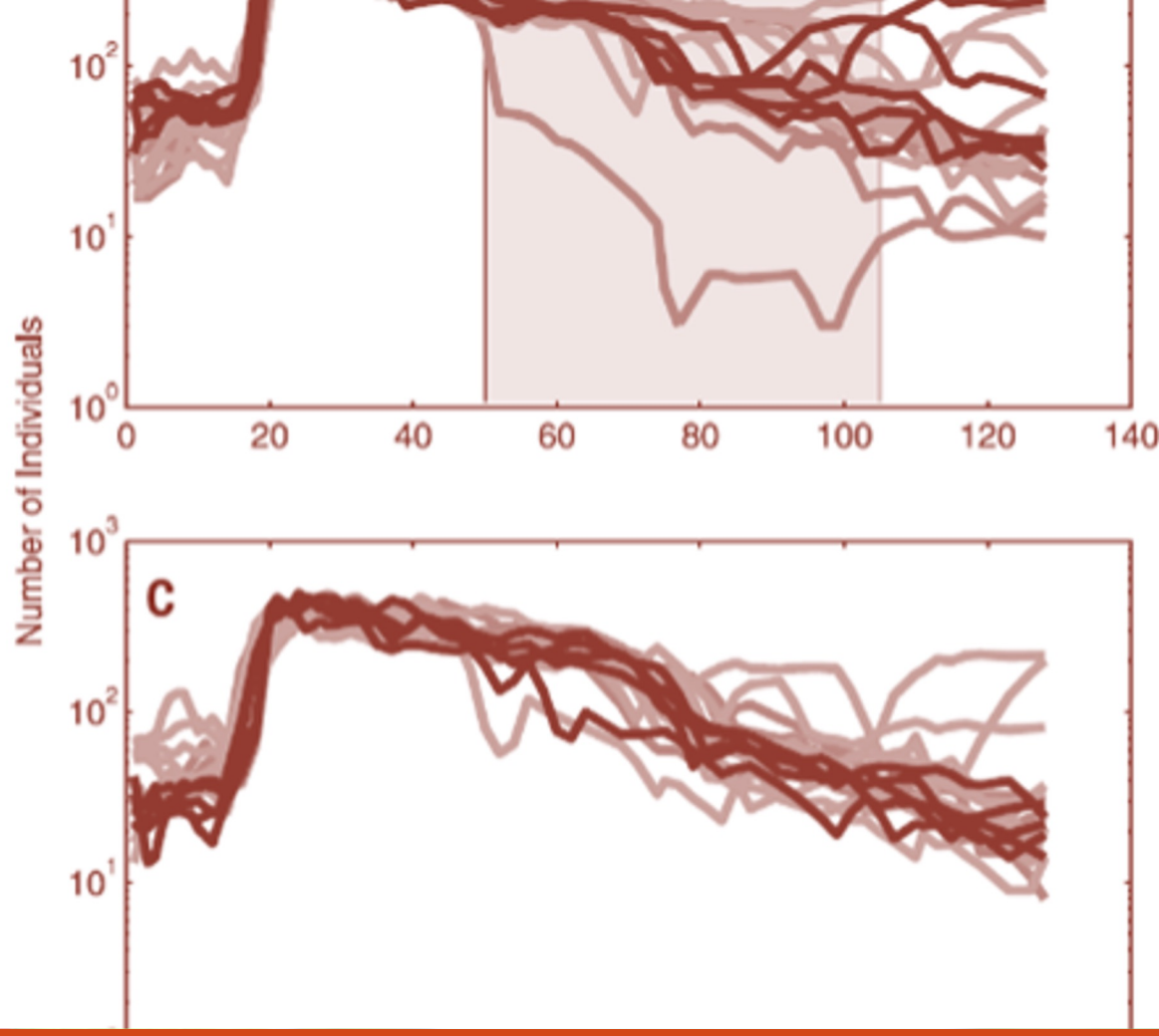




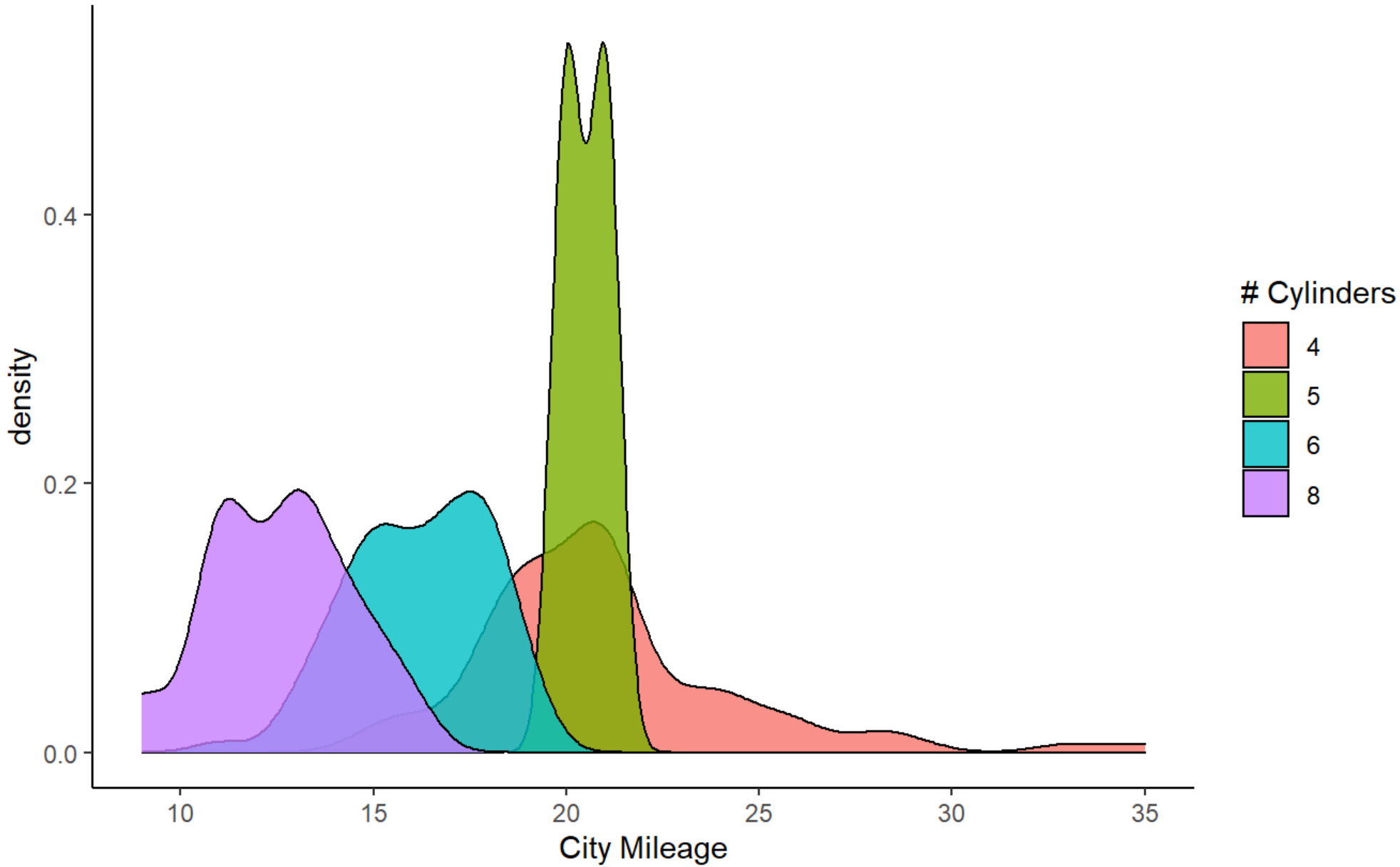
12. ggplot2 Examples and Miscellanea

Example 1

```
1  library(ggplot2)
2  theme_set(theme_classic())
3
4  # Plot
5  g <- ggplot(mpg, aes(cty))
6  g + geom_density(aes(fill=factor(cyl)), alpha=0.8) +
7      labs(title="Density Plot",
8           subtitle="City Mileage Grouped by Number of cylinders",
9           caption="Source: mpg",
10          x="City Mileage",
11          fill="# Cylinders")
```

Density Plot

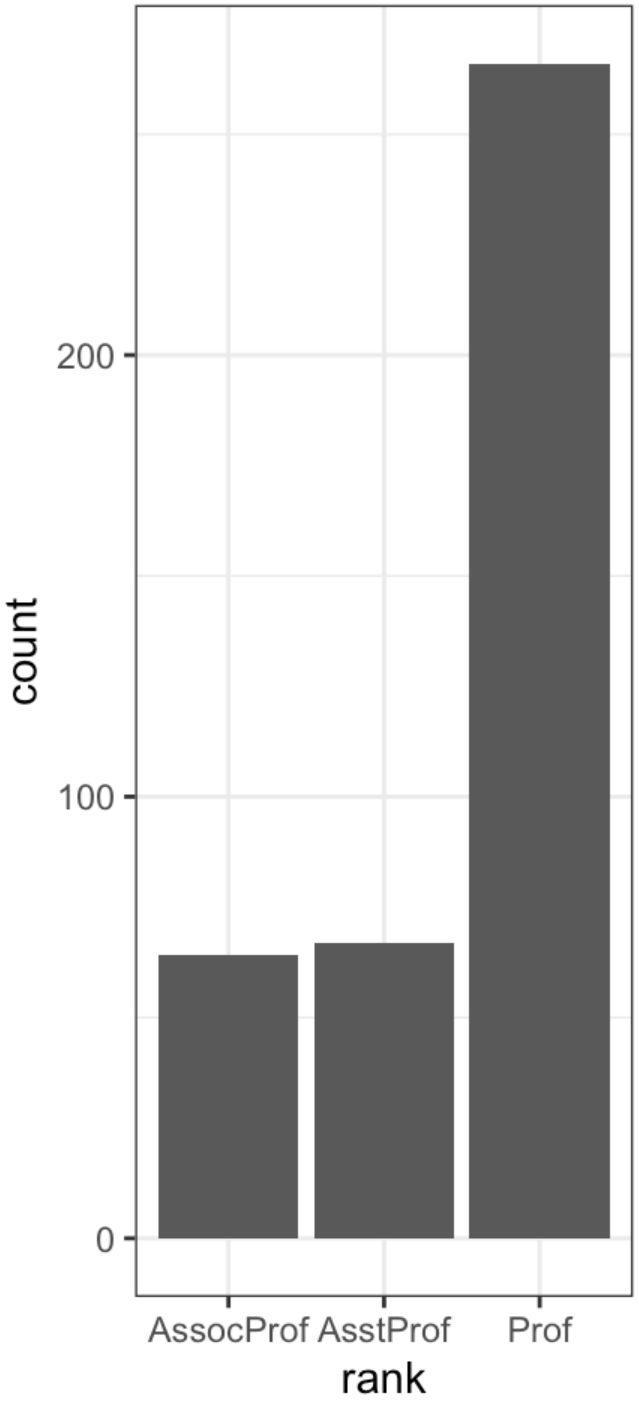
City Mileage Grouped by Number of cylinders

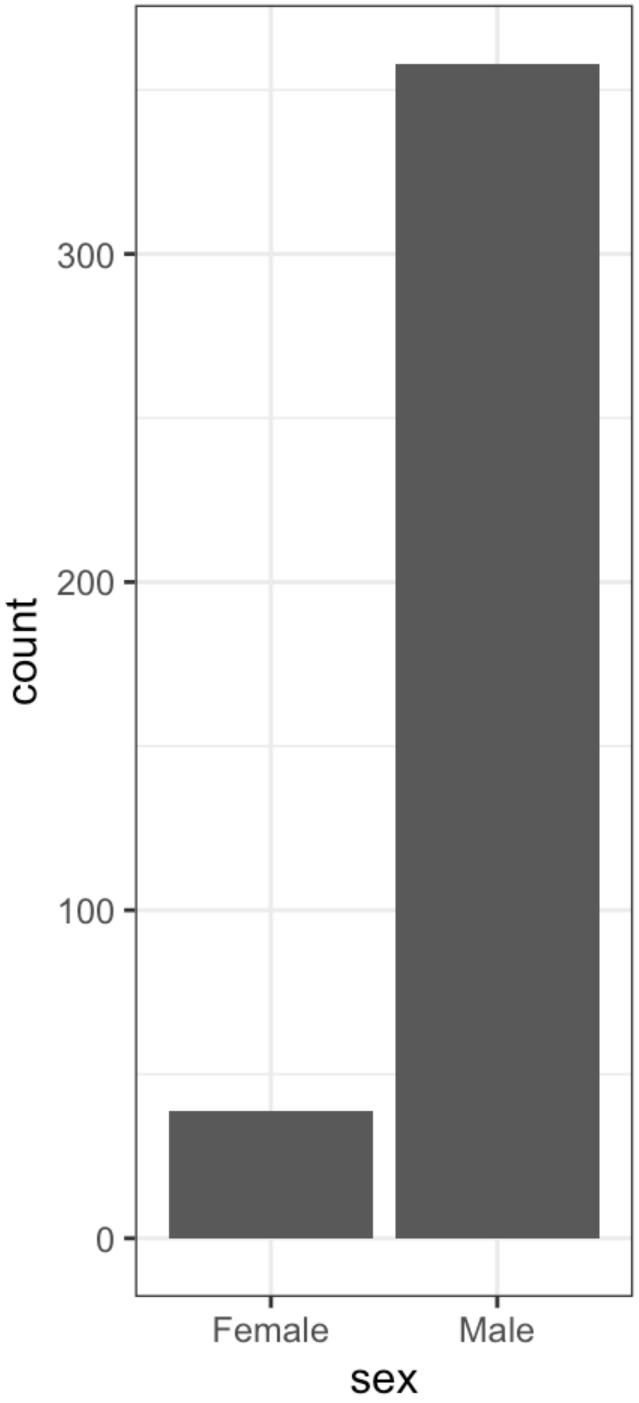


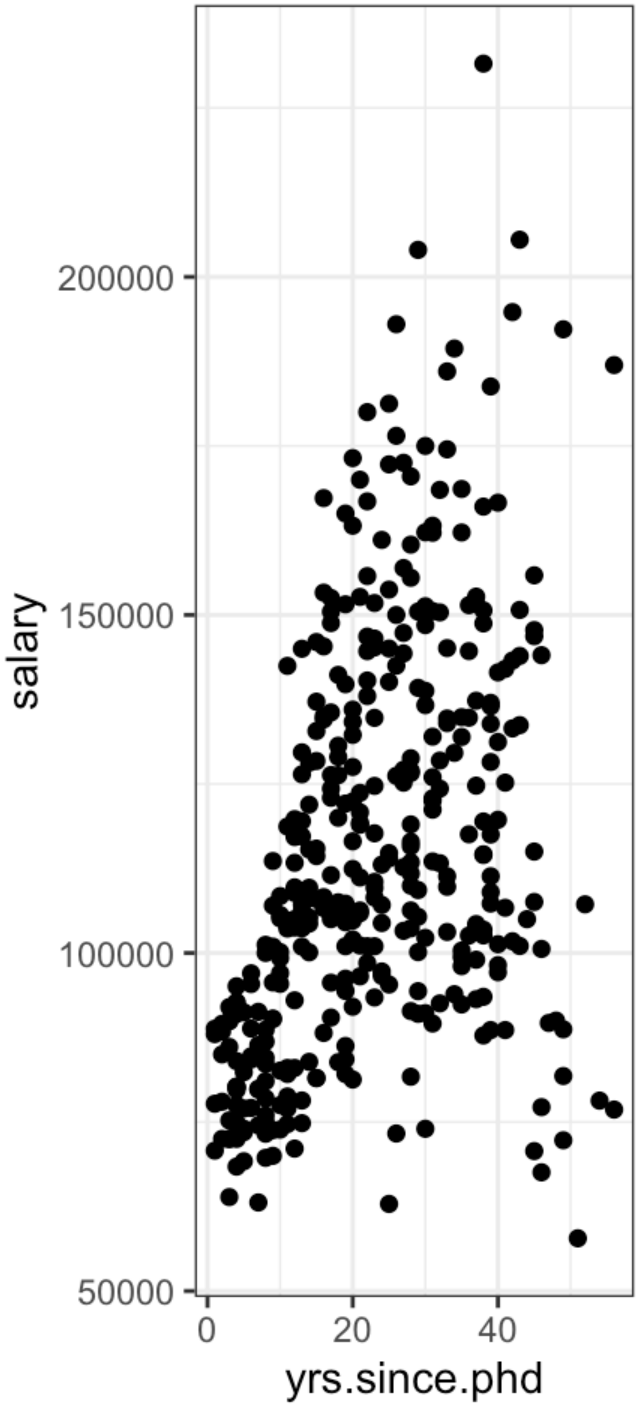
Source: mpg

Example 2

```
1 library(ggplot2)
2 p1 <- ggplot(data=Salaries, aes(x=rank)) + geom_bar()
3 p2 <- ggplot(data=Salaries, aes(x=sex)) + geom_bar()
4 p3 <- ggplot(data=Salaries, aes(x=yrs.since.phd, y=salary)) + geom_point()
5
6 library(gridExtra)
7 grid.arrange(p1, p2, p3, ncol=3)
```

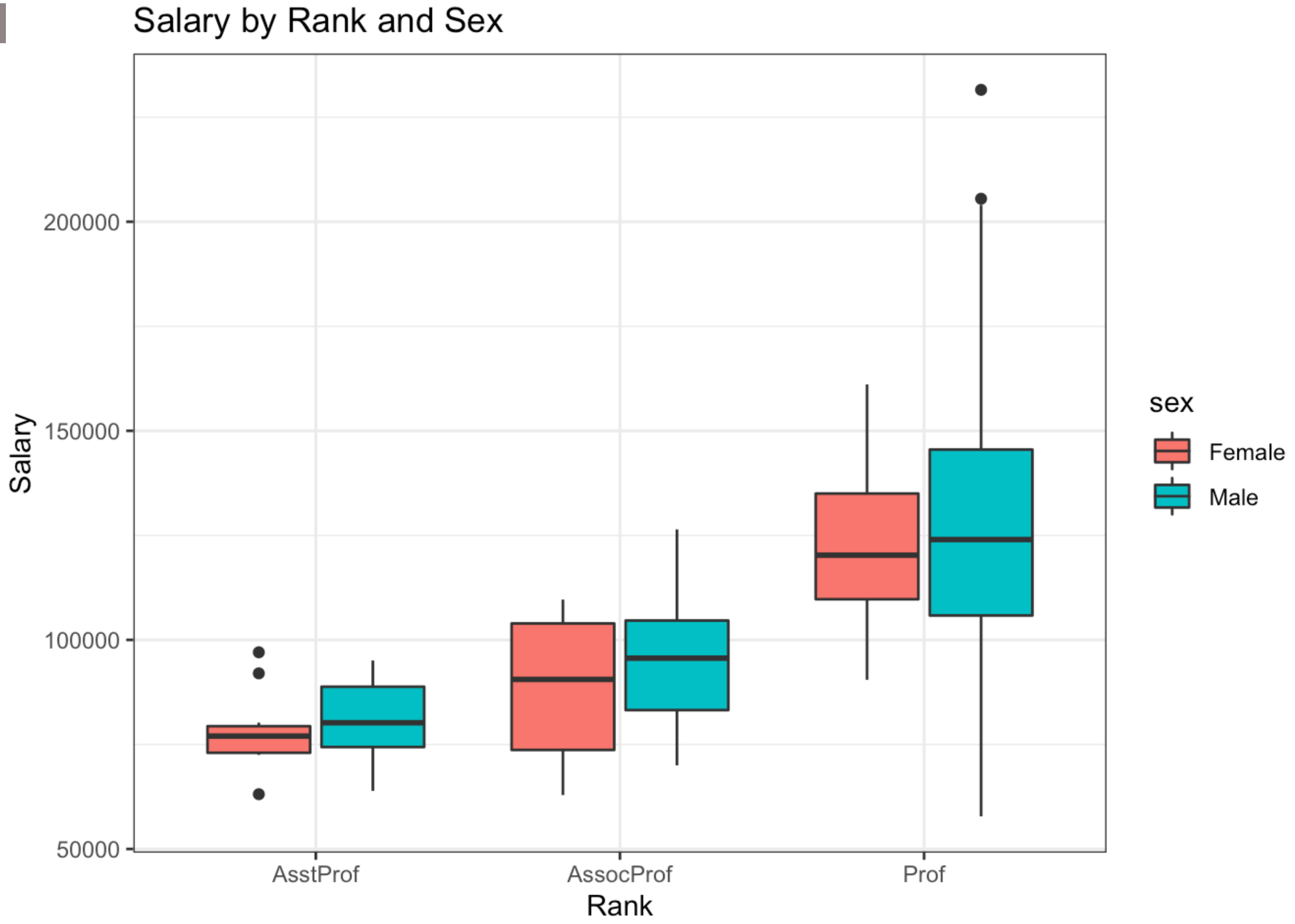

count
200
100
0
AssocProf
AsstProf
Prof
rank


count
300
200
100
0
Female
Male
sex


salary
200000
150000
100000
50000
0
20
40
yrs.since.phd

Example 3

```
1  library(ggplot2)
2  mytheme <- theme(plot.title=element_text(face="bold", size=14, color="brown"),
3                   axis.title=element_text(size=10, color="brown"),
4                   axis.text=element_text(size=9, color="black"),
5                   panel.background=element_rect(fill="white",color="black"),
6                   panel.grid.major.y=element_line(color="grey", linetype=1),
7                   panel.grid.minor.y=element_line(color="grey", linetype=2),
8                   panel.grid.minor.x=element_blank(),
9                   legend.position="top")
10
11 ggplot(Salaries, aes(x=reorder(rank,salary), y=salary, fill=sex)) +
12        geom_boxplot() +
13        labs(title="Salary by Rank and Sex", x="Rank", y="Salary")
```

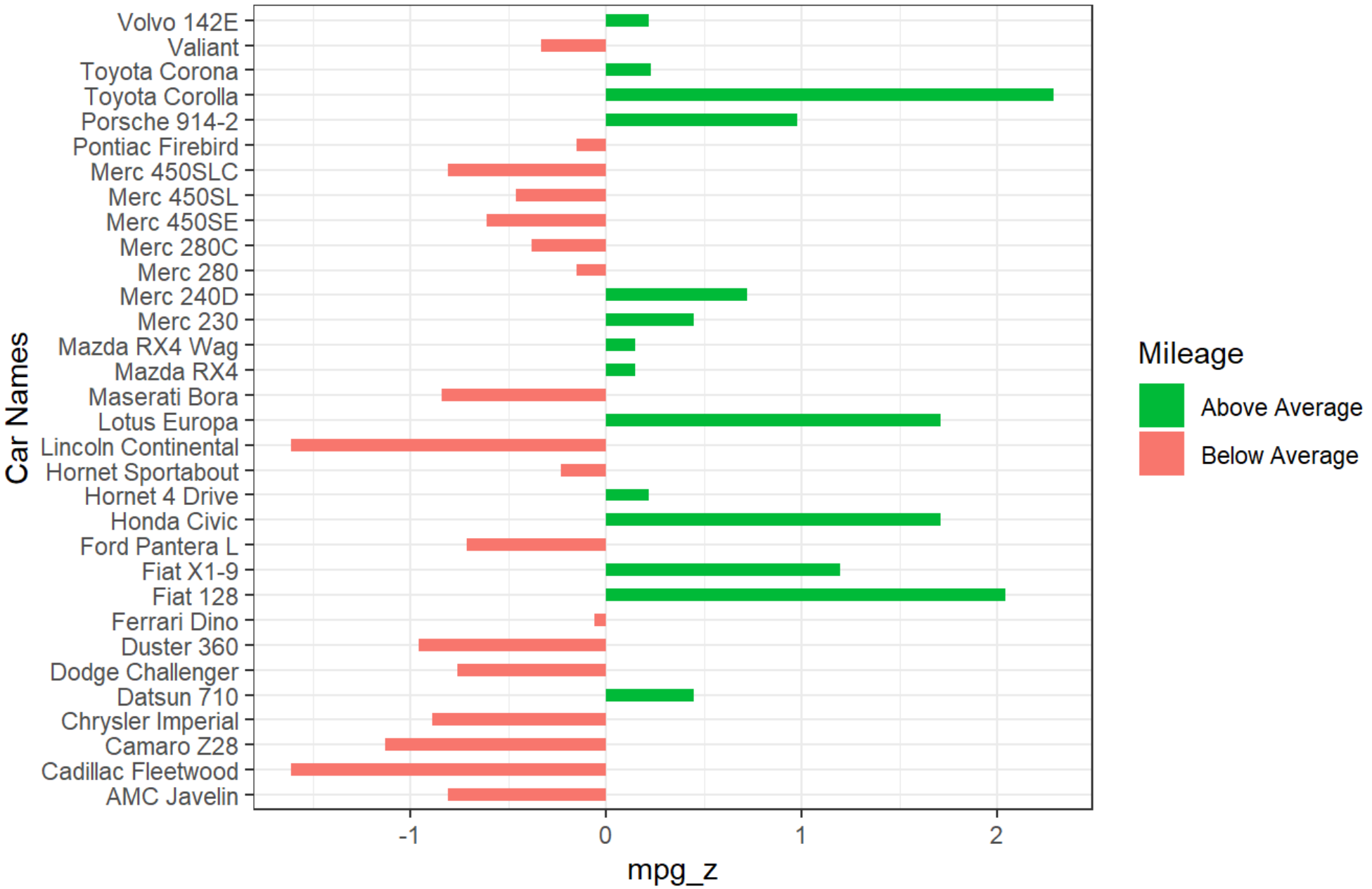

Salary by Rank and Sex
Salary
200000
150000
100000
50000
AsstProf
AssocProf
Prof
Rank
sex
Female
Male

Example 4

```r
library(ggplot2)
theme_set(theme_bw())

mtcars_new <- mtcars |>
  tibble::rownames_to_column(var = "car_name") |> # convert row names to a new column
  dplyr::mutate(car_name = as.factor(car_name),  # convert to factor to retain sorted order
  mpg_z = round(scale(mpg), 2),  # compute normalized mpg
  mpg_type = ifelse(mpg_z < 0, "below", "above")  # above / below avg flag
  ) |>
  dplyr::arrange(mpg_z) # sort

# Diverging Barcharts
ggplot(mtcars_new, aes(x= car_name, y=mpg_z, label=mpg_z)) +
  geom_bar(stat='identity', aes(fill=mpg_type), width=.5)  +
  scale_fill_manual(name="Mileage",
                    labels = c("Above Average", "Below Average"),
                    values = c("above"="#00ba38", "below"="#f8766d")) +
  labs(subtitle="Normalised mileage from 'mtcars'",
       title= "Diverging Bars",
       x = "Car Names") +
  coord_flip()
```

Diverging Bars

Normalised mileage from 'mtcars'

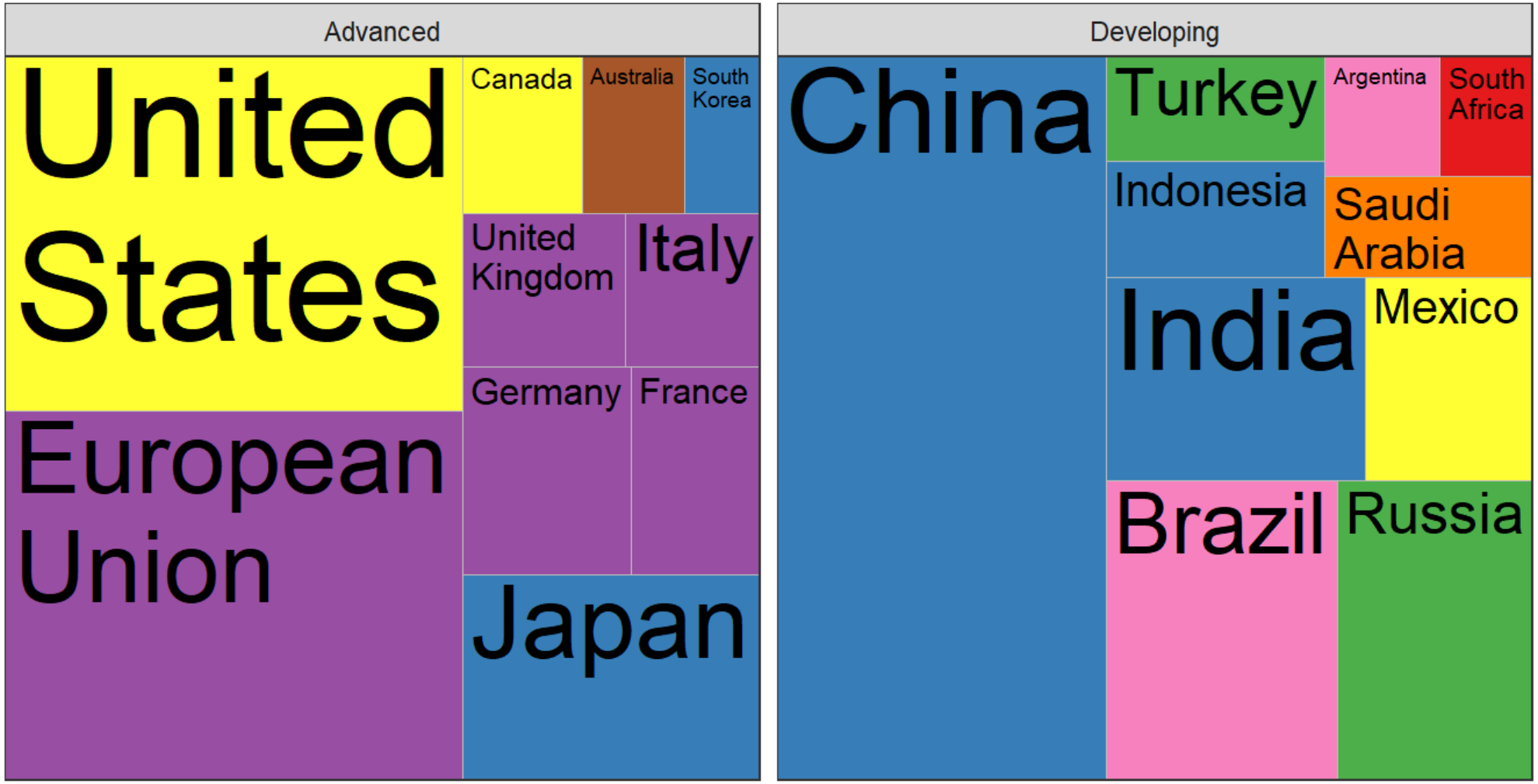


Example 5

```
library(devtools)
#devtools::install_github("wilkox/treemapify")
library(treemapify)
library(ggplot2)
data(G20)
head(G20)

ggplot(G20, aes(area = gdp_mil_usd, fill = region, label = country)) +
  geom_treemap() +
  geom_treemap_text(grow = T, reflow = T, colour = "black") +
  facet_wrap( ~ econ_classification) +
  scale_fill_brewer(palette = "Set1") +
  theme(legend.position = "bottom") +
  labs(
    title = "The G-20 major economies",
    caption = "The area of each country is proportional to its relative GDP
    within the economic group (advanced or developing)",
    fill = "Region"
  )
```

Session 4

The G-20 major economies

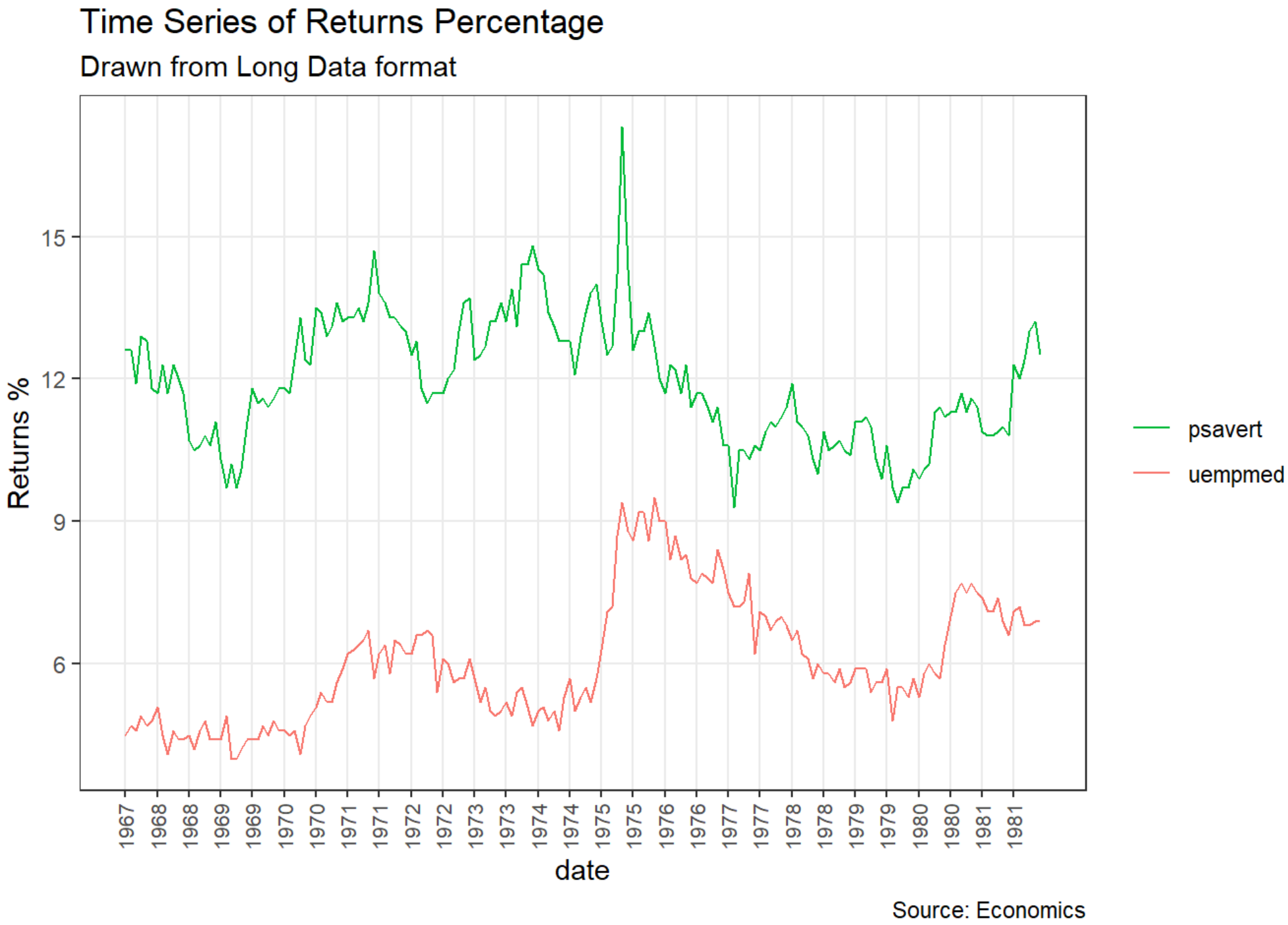


The area of each country is proportional to its relative GDP within the economic group (advanced or developing)

Example 6

```r
data(economics_long, package = "ggplot2")
head(economics_long)

library(ggplot2)
library(lubridate)
theme_set(theme_bw())
df <- economics_long[economics_long$variable %in% c("psavert", "uempmed"), ]
df <- df[lubridate::year(df$date) %in% c(1967:1981), ]

# labels and breaks for X axis text
brks <- df$date[seq(1, length(df$date), 12)]
lbls <- lubridate::year(brks)
```

```r
ggplot(df, aes(x=date)) +
  geom_line(aes(y=value, col=variable)) +
  labs(title="Time Series of Returns Percentage",
       subtitle="Drawn from Long Data format",
       caption="Source: Economics",
       y="Returns %",
       color=NULL) +  # title and caption
  scale_x_date(labels = lbls, breaks = brks) +  # change to monthly ticks and
  scale_color_manual(labels = c("psavert", "uempmed"),
                     values = c("psavert"="#00ba38", "uempmed"="#f8766d")) +
  theme(axis.text.x = element_text(angle = 90, vjust=0.5, size = 8),  # rotate
        panel.grid.minor = element_blank())  # turn off minor grid
```

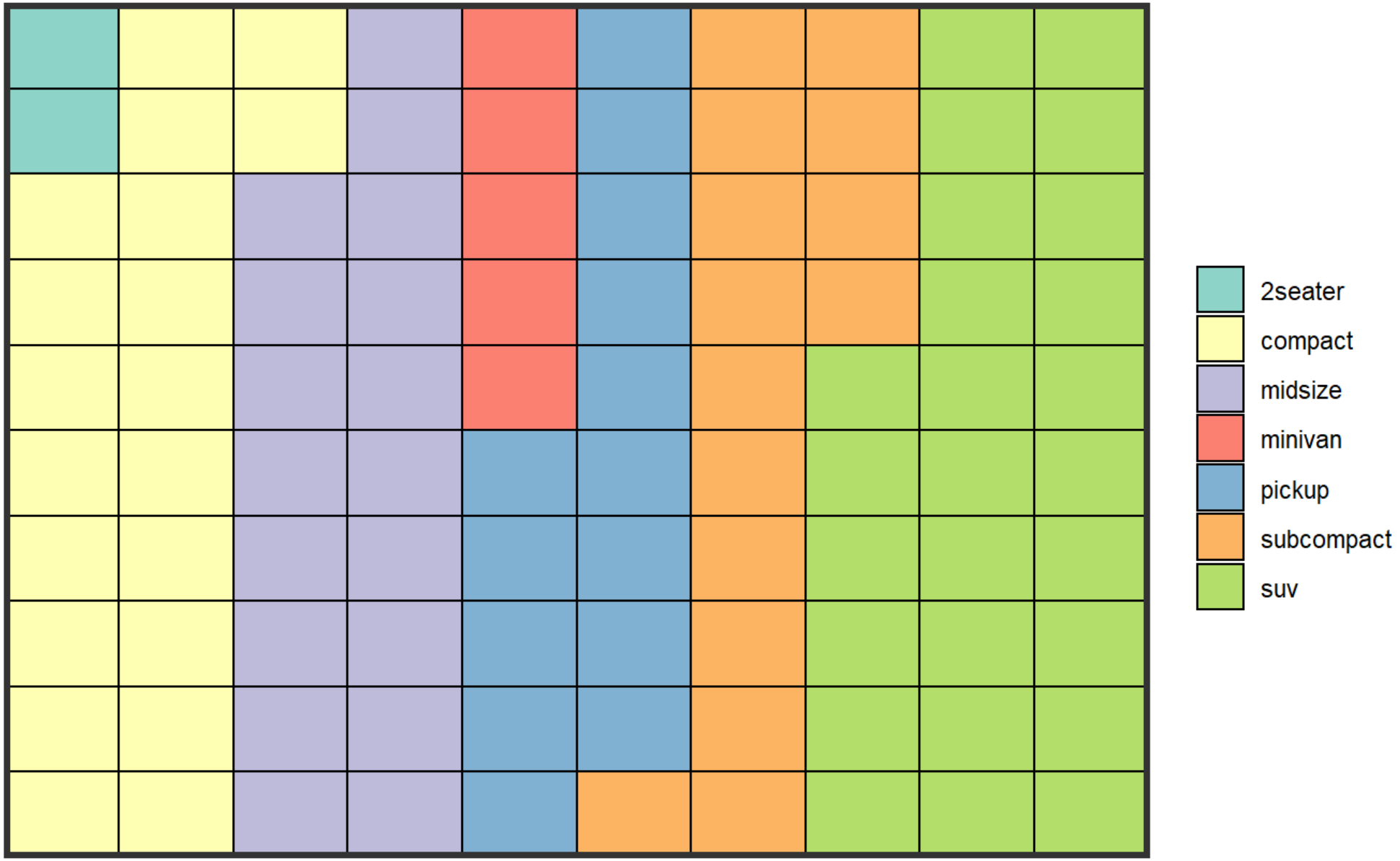

Time Series of Returns Percentage
Drawn from Long Data format
Returns %
15
12
9
6
1967
1968
1968
1969
1969
1970
1970
1971
1971
1972
1972
1973
1973
1974
1974
1975
1975
1976
1976
1977
1977
1978
1978
1979
1979
1980
1980
1981
1981
date
psavert
uempmed
Source: Economics

Example 7

```
library(ggplot2)
var <- mpg$class  # the categorical data
## Prep data (nothing to change here)
nrows <- 10
df <- expand.grid(y = 1:nrows, x = 1:nrows)
categ_table <- round(table(var) * ((nrows*nrows)/(length(var))))
categ_table

df$category <- factor(rep(names(categ_table), categ_table))
ggplot(df, aes(x = x, y = y, fill = category)) +
        geom_tile(color = "black", size = 0.5) +
        scale_x_continuous(expand = c(0, 0)) +
        scale_y_continuous(expand = c(0, 0), trans = 'reverse') +
        scale_fill_brewer(palette = "Set3") +
        labs(title="Waffle Chart", subtitle="'Class' of vehicles",
             caption="Source: mpg") +
        theme(panel.border = element_rect(size = 2),
              plot.title = element_text(size = rel(1.2)),
              axis.text = element_blank(),
              axis.title = element_blank(),
              axis.ticks = element_blank(),
              legend.title = element_blank(),
              legend.position = "right")
```

Waffle Chart

'Class' of vehicles

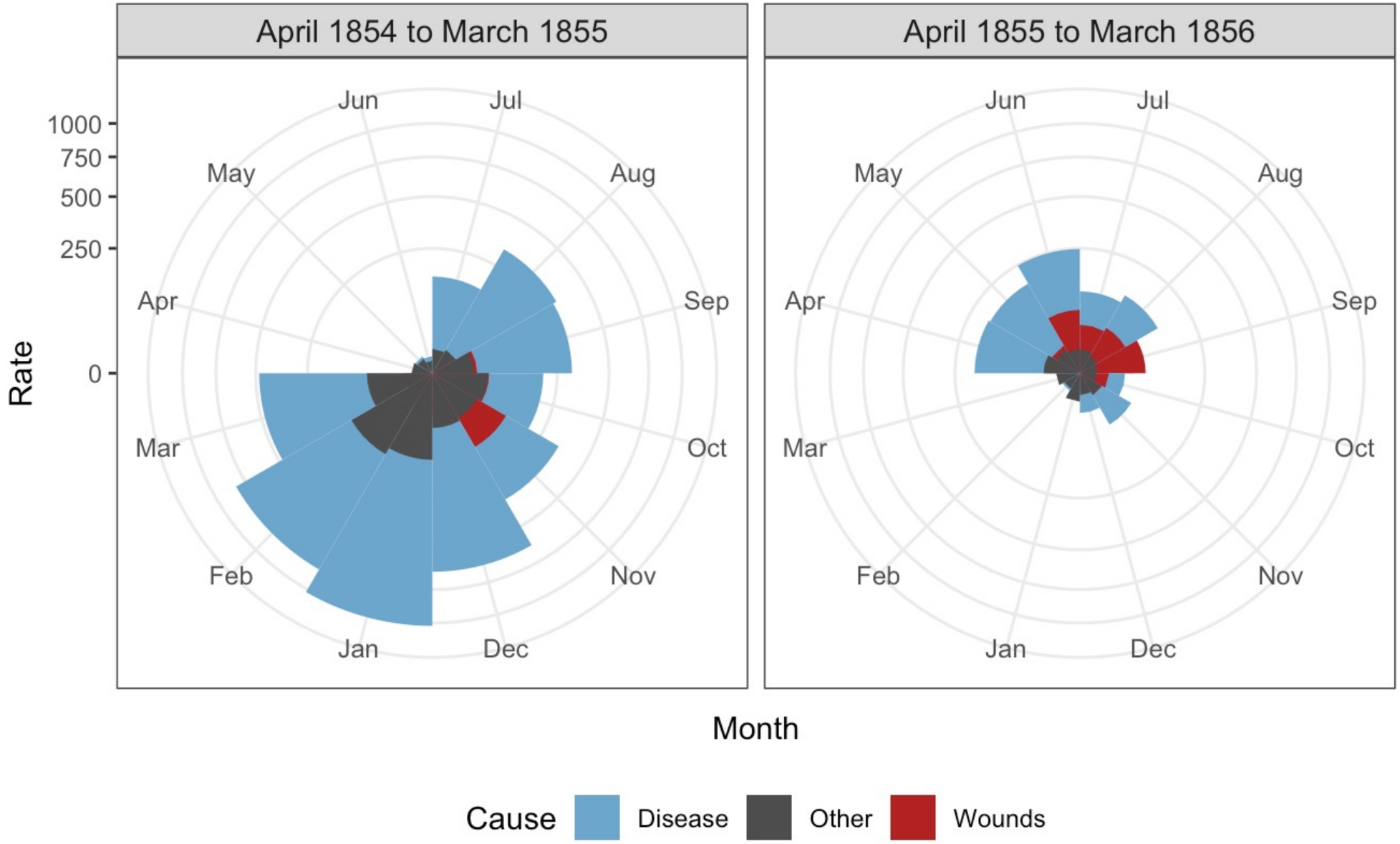


Source: mpg

Example 8

```
library(tidyverse)
library(HistData)

Nightingale %>%
  select(Date, Month, Year, contains("rate")) %>%
  pivot_longer(cols = 4:6, names_to = "Cause", values_to = "Rate") %>%
  mutate(Cause = gsub(".rate", "", Cause),
         period = ifelse(Date <= as.Date("1855-03-01"), "April 1854 to March 1855", "April 1855 to March 1856"),
         Month = fct_relevel(Month, "Jul", "Aug", "Sep", "Oct", "Nov", "Dec", "Jan", "Feb", "Mar", "Apr", "May", "Jun")) %>%
  ggplot(aes(Month, Rate)) +
  geom_col(aes(fill = Cause), width = 1, position = "identity") +
  coord_polar() +
  facet_wrap(~period) +
  scale_fill_manual(values = c("skyblue3", "grey30", "firebrick")) +
  scale_y_sqrt() +
  theme_bw() +
  theme(axis.text.x = element_text(size = 9),
        strip.text = element_text(size = 11),
        legend.position = "bottom",
        plot.margin = unit(c(10, 10, 10, 10), "pt"),
        plot.title = element_text(vjust = 5)) +
  ggtitle("Diagram of the Causes of Mortality in the Army in the East")
```

Diagram of the Causes of Mortality in the Army in the East



Example 9

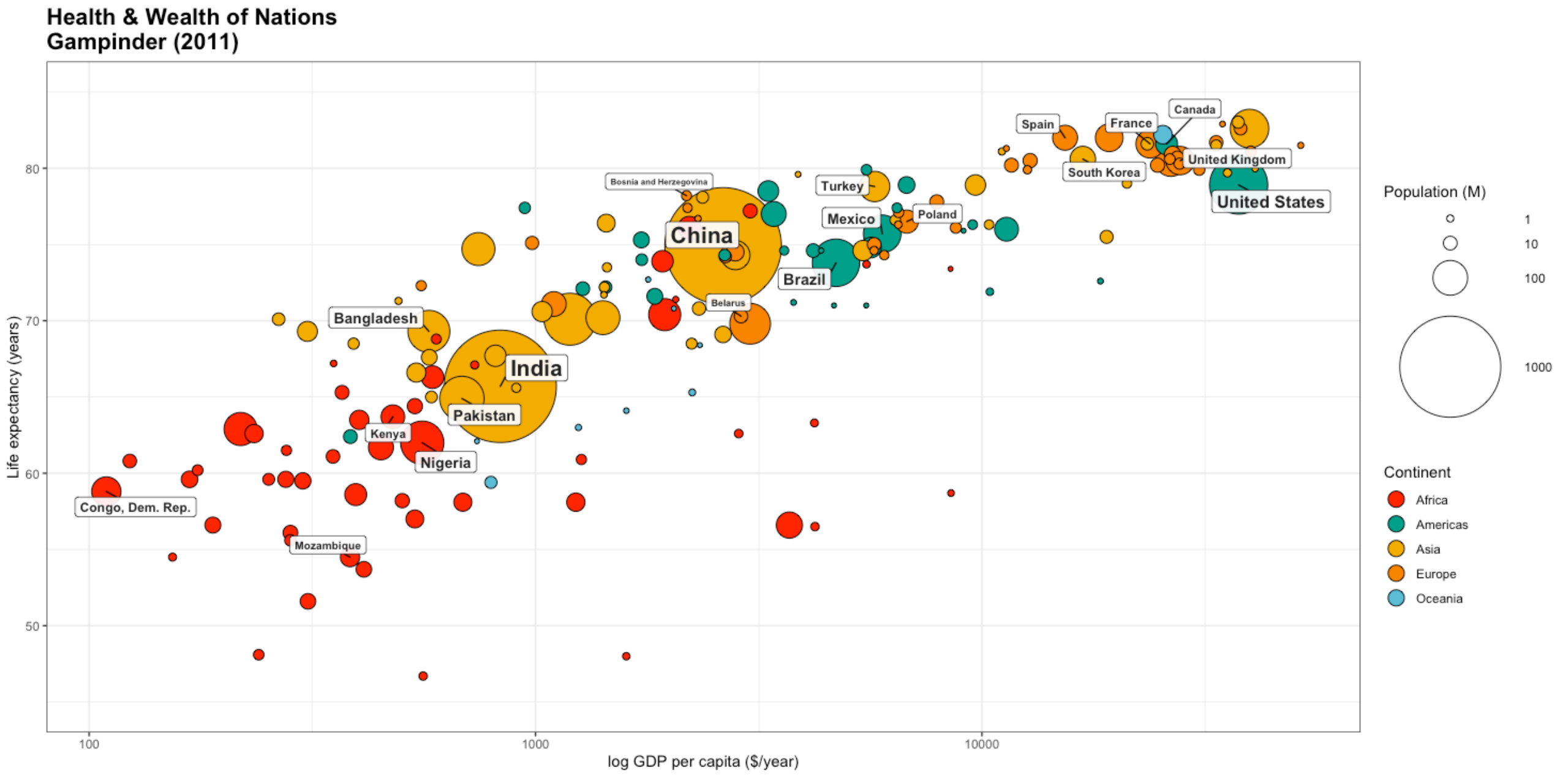
```
library(tidyverse)
gapminder_ds <- dslabs::gapminder # for the dataset
library(wesanderson)  # for the colour palette
library(ggrepel)      # for country names on chart
```

```
yr <- 2009
chart_title <- paste("Health & Wealth of Nations \nGampinder (",yr,")",sep="")
```

```
# sort the countries by inverse population
gapminder_ds <- gapminder_ds |>
  dplyr::arrange(year, dplyr::desc(population))

# select which countries will have their names labelled
num_countries <- 20

# tidyverse + ggplot2
filtered_gapminder <- gapminder_ds %>%
  filter(year==yr) %>%
  mutate(pop_m = population/1e6, gdppc=gdp/population)

weights <- filtered_gapminder$pop_m/sum(filtered_gapminder$pop_m)
p <- sample(nrow(filtered_gapminder), num_countries, prob = weights)

filtered_gapminder$country_display <- ifelse(ifelse(1:185 %in% p, TRUE,FALSE),as.character(filtered_gapminder$country),"")
```

```
filtered_gapminder %>%
  ggplot(aes(x = gdppc,
             y=life_expectancy,
             size=pop_m)) +
  geom_point(aes(fill=continent), pch=21) +
  scale_fill_manual(values=wes_palette(n=5, name="Darjeeling1")) +
  scale_x_log10() +
  geom_label_repel(aes(label=country_display, size=sqrt(pop_m/pi)),
                   alpha=0.9,
                   fontface="bold",
                   min.segment.length = unit(0, 'lines'),
                   show.legend=FALSE) +
  ggtitle(chart_title) +
  theme(plot.title = element_text(size=14, face="bold")) +
  xlab('log GDP per capita ($/year)') +
  ylab('Life expectancy (years)') +
  ylim(45,85) +
  scale_size_continuous(range=c(1,40),
                        breaks = c(1,10,100,1000),
                        limits = c(0, 1500),
                        labels = c("1","10","100","1000")) +
  guides(fill = guide_legend(override.aes = list(size = 5))) +
  labs(fill="Continent", size="Population (M)") +
  theme_bw() +
  theme(plot.title = element_text(size=16, face="bold"))
```

Session 4


Health & Wealth of Nations
Gampinder (2011)
Life expectancy (years)
80
70
60
50
100
1000
10000
log GDP per capita ($/year)
Bosnia and Herzegovina
Turkey
Spain
France
Canada
United Kingdom
South Korea
United States
Mexico
Poland
China
Brazil
Belarus
Bangladesh
India
Pakistan
Kenya
Nigeria
Congo, Dem. Rep.
Mozambique
Population (M)
1
10
100
1000
Continent
Africa
Americas
Asia
Europe
Oceania

Suggested Reading

ggplot2 Examples and Miscellanea

Data Understanding, Data Analysis, Data Science
Data Visualization and Data Exploration

ggplot2 Visualizations in R

- Examples (except the U.S. Census PUMS Data example)

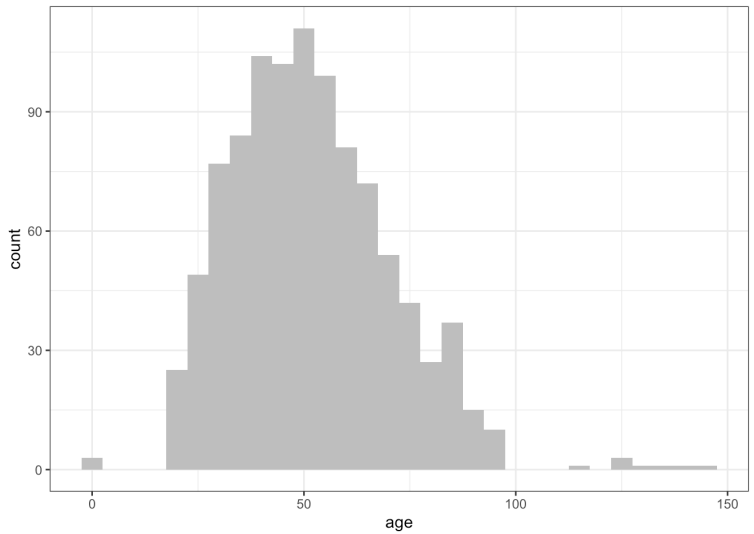
Exercises

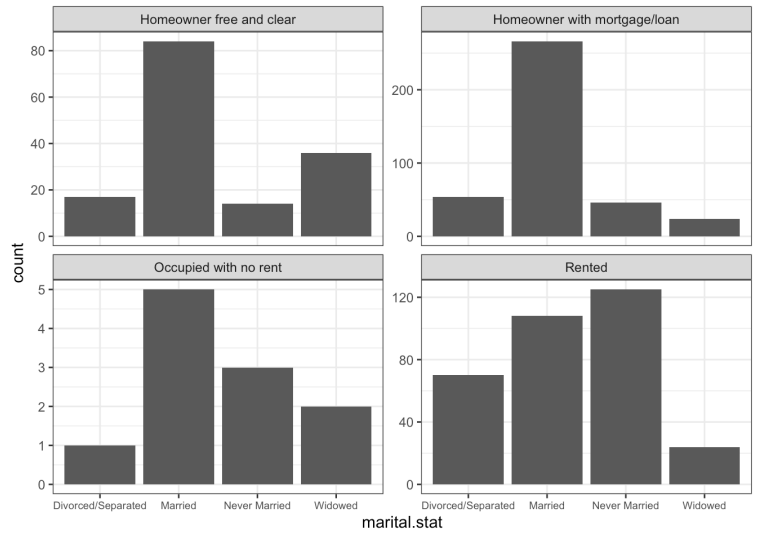
ggplot2 Examples and Miscellanea

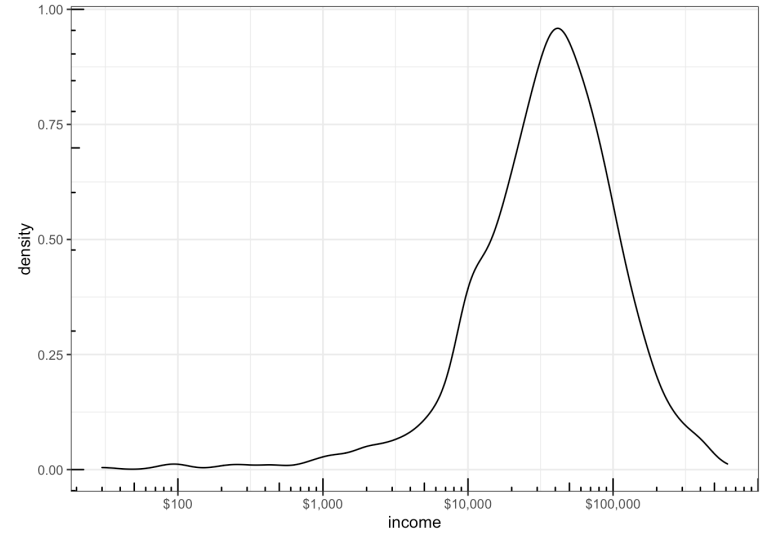
(Try this exercise without first consulting the U.S. Census PUMS Data example in DUDADS).

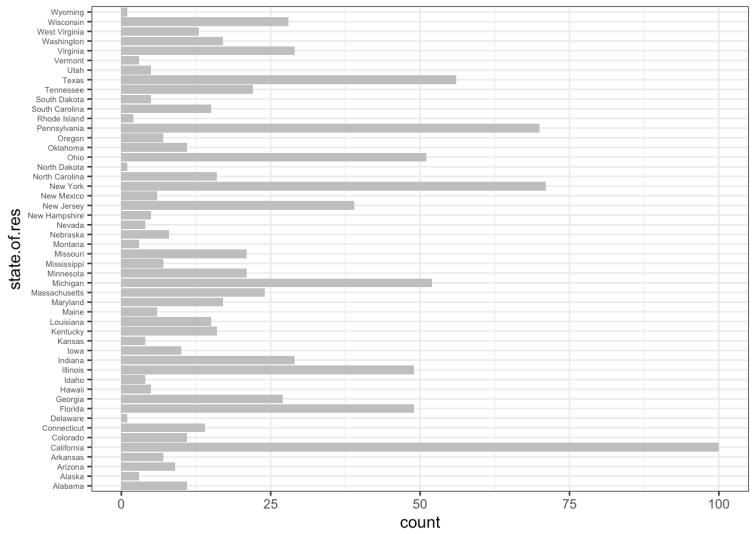
The custdata.tsv file is derived from U.S. Census PUMS data. This synthetic dataset contains customer information for individuals whose health insurance status is known.

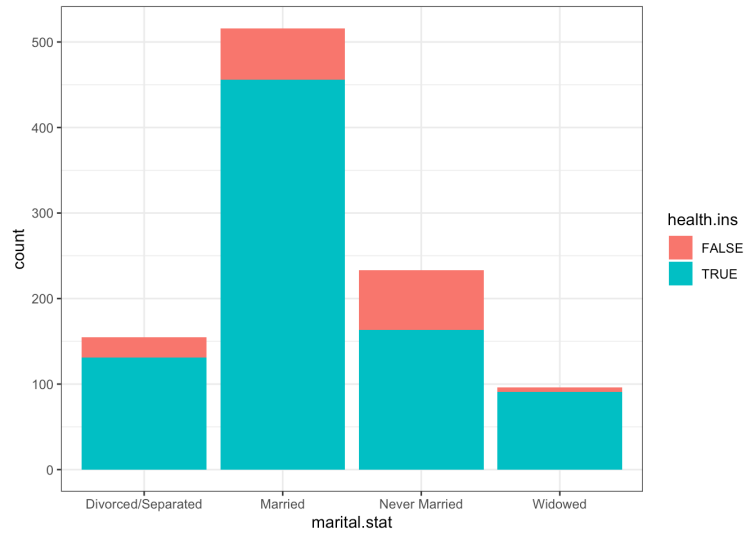
Recreate the following ggplot2 charts (ignore missing values for the time being).

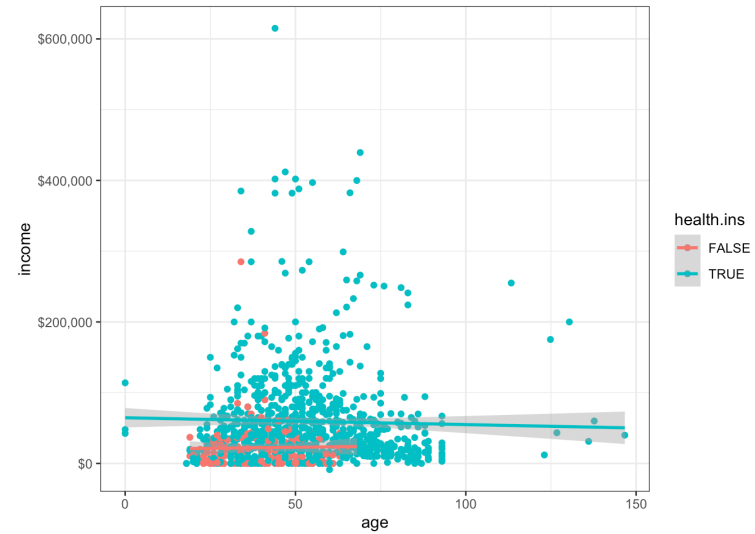

count
age
0
30
60
90
0
50
100
150


Homeowner free and clear
Homeowner with mortgage/loan
Occupied with no rent
Rented
count
marital.stat
Divorced/Separated
Married
Never Married
Widowed


density
income
0.00
0.25
0.50
0.75
1.00
$100
$1,000
$10,000
$100,000


state.of.res
count
0
25
50
75
100


count
marital.stat
health.ins
FALSE
TRUE
Divorced/Separated
Married
Never Married
Widowed


income
age
health.ins
FALSE
TRUE
$0
$200,000
$400,000
$600,000