2.exploration

Joao Lopes

2024-10-07

Contents

1. VARIATION

[from https://r4ds.hadley.nz/eda#variation]

```
library("nycflights13") #collection of datasets
library("skimr")        #function skim() for descriptive statistics
library("tidyverse")    #collection of packages for data analysis
```

```
#metadata
?diamonds
```

```
#data inspection
glimpse(diamonds)

#descriptive statistics
skim(diamonds)
```

1.1. TYPICAL VALUES

```
#visualize distribution
ggplot(diamonds, aes(x = carat)) +
  geom_histogram(binwidth = 0.5)

#visualize distribution in detail
diamonds |>
  filter(carat < 3) |>                #filter in smaller diamonds
  ggplot(aes(x = carat)) +
  geom_histogram(binwidth = 0.01) +
  geom_vline(xintercept=c(seq(0,3,0.5)),linetype = "dashed",color="red")

#visualize distribution
ggplot(diamonds, aes(x = y)) +
  geom_histogram(binwidth = 0.5)

#visualize distribution in detail
ggplot(diamonds, aes(x = y)) +
  geom_histogram(binwidth = 0.5) +
  coord_cartesian(ylim = c(0, 50))
```

1.2. UNUSUAL VALUES

```
#look at the data
diamonds |>
  filter(between(y, 3, 20)) |>         #filter out unusual values
  select(price, x, y, z) |>
  arrange(y)

#drop entire row of data
diamonds2 <- diamonds |>
  filter(between(y, 3, 20))

#replacing with missing values
diamonds2 <- diamonds |>
  mutate(y = if_else(y < 3 | y > 20, NA, y))
```

```r
ggplot(diamonds2, aes(x = x, y = y)) +
  geom_point()

ggplot(diamonds2, aes(x = x, y = y)) +
  geom_point(na.rm = TRUE)

#meaningless missing values
diamonds |>
  filter(!between(y, 3, 20))

#meaningful missing values
flights |>
  mutate(
    cancelled = is.na(dep_time),
    sched_hour = sched_dep_time %/% 100,
    sched_min = sched_dep_time %% 100,
    sched_dep_time = sched_hour + (sched_min / 60)
  ) |>
  ggplot(aes(x = sched_dep_time)) +
  geom_density(aes(color = cancelled), bw = 1/4)
```

---

2. MISSING VALUES

[from https://r4ds.hadley.nz/missing-values]

```
library("nycflights13") #collection of datasets
library("tidyverse")    #collection of packages for data analysis
```

2.1. EXPLICIT MISSING VALUES

```
treatment <- tribble(
  ~person,           ~treatment, ~response,
  "Derrick Whitmore", 1,         7,
  NA,                 2,         10,
  NA,                 3,         NA,
  "Katherine Burke",  1,         4
)
```

Last observation carried forward

```
treatment |>
  fill(everything())
```

Fixed values

```
treatment |>
  mutate(response = coalesce(response, 0)) |>
  fill(everything())
```

Numeric value represents NA

```
csv <- "
person,          treatment, response
Derrick Whitmore,        1,        7
Derrick Whitmore,        2,       10
Derrick Whitmore,        3,       99
Katherine Burke,         1,        4"
read_csv(csv, na = "99")

read_csv(csv) |>
  mutate(response = na_if(response, 99))
```

NaN

```
v <- c(NA, NaN)
v * 10
v == 1
is.na(v)
is.nan(v)

0 / 0
0 * Inf
```

```r
Inf - Inf
sqrt(-1)
```

2.2. IMPLICIT MISSING VALUES

```r
stocks <- tibble(
  year  = c(2020, 2020, 2020, 2020, 2021, 2021, 2021),
  qtr   = c(   1,    2,    3,    4,    2,    3,    4),
  price = c(1.88, 0.59, 0.35,   NA, 0.92, 0.17, 2.66)
)
#1. Price in 2020 Q4 is explicitly missing
#2. Price in 2021 Q1 is implicitly missing
```

Pivoting

```r
stocks |>
  pivot_wider(
    names_from = qtr,
    values_from = price
  ) |>
  pivot_longer(
    cols = -year,
    names_to = "qtr",
    values_to = "price"
  )
```

Complete

```r
stocks |>
  complete(year, qtr)

stocks |>
  complete(year = 2019:2021, qtr)
```

Joins

```r
flights |>
  distinct(faa = dest) |>              #calculate the distinct "dest"
  anti_join(airports)                  #obtain absent "dest"

flights |>
  distinct(tailnum) |>                 #calculate the distinct "dest"
  anti_join(planes)                    #obtain absent "dest"
```

2.3. FACTORS AND EMPTY GROUPS

```r
health <- tibble(
  name   = c("Ikaia", "Oletta", "Leriah", "Dashay", "Tresaun"),
  smoker = factor(c("no", "no", "no", "no", "no"), levels = c("yes", "no")),
  age    = c(34, 88, 75, 47, 56),
)
```

```r
#empty groups with count
health |> count(smoker)
health |> count(smoker, .drop = FALSE)

#empty groups with ggplot
ggplot(health, aes(x = smoker)) +
  geom_bar() +
  scale_x_discrete()
ggplot(health, aes(x = smoker)) +
  geom_bar() +
  scale_x_discrete(drop = FALSE)

#empty groups with group_by()
health |>
  group_by(smoker, .drop = FALSE) |>
  summarize(
    n = n(),
    mean_age = mean(age),
    min_age = min(age),
    max_age = max(age),
    sd_age = sd(age)
  )
health |>
  group_by(smoker) |>
  summarize(
    n = n(),
    mean_age = mean(age),
    min_age = min(age),
    max_age = max(age),
    sd_age = sd(age)
  ) |>
  complete(smoker)
```

---

3. COVARIATION

[from https://r4ds.hadley.nz/eda#covariation]

```
library("hexbin")       #function geom_hex() for data visualization
library("tidyverse")    #collection of packages for data analysis
```

3.1. CATEGORICAL AND NUMERICAL VARIABLES

```
#use geom_freqpoly()
ggplot(diamonds, aes(x = price, color = cut)) +
  geom_freqpoly(binwidth = 500, linewidth = 0.75)

#use geom_freqpoly() with densities
ggplot(diamonds, aes(x = price, y = after_stat(density), color = cut)) +
  geom_freqpoly(binwidth = 500, linewidth = 0.75)

#use geom_density()
ggplot(diamonds, aes(x = price, color = cut)) +
  geom_density(bw = 500, linewidth = 0.75)

#use geom_boxplot()
ggplot(diamonds, aes(x = cut, y = price)) +
  geom_boxplot()
```

3.2. TWO CATEGORICAL VARIABLES

```
#table
diamonds |> count(color, cut) |>
  pivot_wider(
    names_from = "cut",
    values_from = "n"
  )

#geom_count()
ggplot(diamonds, aes(x = cut, y = color)) +
  geom_count()

#geom_tile()
diamonds |>
  count(color, cut) |>
  ggplot(aes(x = cut, y = color)) +
  geom_tile(aes(fill = n))

#geom_bar()
ggplot(diamonds, aes(x = cut, fill = color)) +
  geom_bar()
```

3.3. TWO NUMERICAL VARIABLES

```
smaller <- diamonds |>
  filter(carat < 3)                    #filter in smaller diamonds
```

```r
#geom_point()
ggplot(smaller, aes(x = carat, y = price)) +
  geom_point()

#geom_point() with alpha
ggplot(smaller, aes(x = carat, y = price)) +
  geom_point(alpha = 0.01)

#geom_bin2d()
ggplot(smaller, aes(x = carat, y = price)) +
  geom_bin2d()

#geom_hex()
ggplot(smaller, aes(x = carat, y = price)) +
  geom_hex()

#geom_boxplot()
ggplot(smaller, aes(x = carat, y = price)) +
  geom_boxplot(aes(group = cut_width(carat, 0.1)))
```

---

4. PATTERNS AND MODELS

[from https://r4ds.hadley.nz/eda#patterns-and-models]

```
library("tidymodels") #collection of packages for data modelling
library("tidyverse")  #collection of packages for data analysis
```

```
set.seed(123)
diamonds2 <- diamonds |>
  mutate(
    log_price = log(price),
    log_carat = log(carat)
  ) |>
  slice_sample(n = 1000)

diamonds2 |>
  ggplot(aes(x = log_carat, y = log_price)) +
  geom_point()

#functions linear_reg() and fit()
diamonds_fit <- linear_reg() |>
  fit(log_price ~ log_carat, data = diamonds2)

#function tidy()
tidy(diamonds_fit)
summary(diamonds_fit$fit)$coef

#function augment()
diamonds_aug <- augment(diamonds_fit, new_data = diamonds2) |>
  mutate(.resid = exp(.resid))

#unexplained variation in price using carat
ggplot(diamonds_aug, aes(x = carat, y = .resid)) +
  geom_point()

#modelling unexplained variation in price using cut
ggplot(diamonds_aug, aes(x = cut, y = .resid)) +
  geom_boxplot()
```