JAK RADZIĆ SOBIE Z BRAKAMI DANYCH

Prezentacja na podstawie artykułu Nicholasa Tierney i Dianny Cook

Tymoteusz Kwieciński i Marta Szuwarska

Expanding Tidy Data Principles to Facilitate Missing Data Exploration, Visualization and Assessment of Imputations

Dianne Cook, źródło: https://magazine.amstat.org/blog/2020/03/01/dianne-cook-2/

Nicholas Tireney, źródło: https://github.com/njtierney

Plan prezentacji

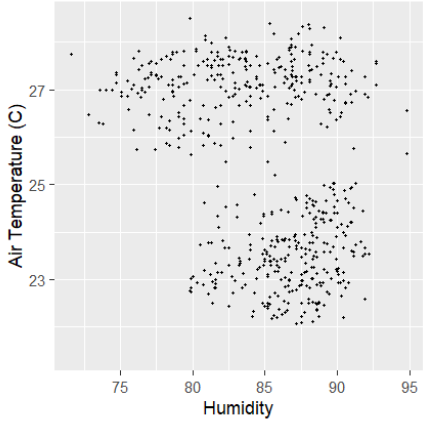
- Wstęp
- Background
- Existing software
- *Tidy framework*
- Funkcje pakietu *naniar*
- Wizualizacje i sposoby podsumowania braków danych
- Bardzo życiowe przykłady

Główna idea – ułatwienie pracy z brakami danych

- Rozszerzenie zasad dotyczących ***tidy data*** na braki danych
- Ułatwienie rozpoznawania i **eksploracji** braków danych
- **Wizualizacja** missing values
- Ułatwienie oceny **imputacji** danych

Dlaczego powstał pakiet *naniar*?

- Brak odpowiednich narzędzi pozwalających obsługiwać braki danych
- Potrzeba jednego środowiska do ułatwienia operacji oraz oceny imputacji braków danych
- Zgodność z zasadami *tidy data*



Warning message:
removed 171 rows containing missing values (geom_point).

Jak pakiet ggplot2 domyślnie radzi sobie z brakami danych

BACKGROUND

Czyli teoretyczne wejście w temat

Tidy data

- Każda zmienna ma swoją kolumnę
- Każda obserwacja ma swój wiersz
- Każda wartość ma swoją komórkę

Tidy tools

Manipulacja danych

Wizualizacje

Modelowanie

TIDY DATA

+

TIDY TOOLS

=

TIDYVERSE

Tidy tools do pracy z brakami danych?

Shadow matrix

Jest to macierz B reprezentująca braki w danych $Y = (Y_{miss}, Y_{obs})$ taka, że:

$$b_{ij} = \begin{cases} 1, & \text{dla brakującego } y_{ij} \\ 0, & \text{dla niebrakującego } y_{ij} \end{cases}$$

Model brakujących danych

Modelem brakujących danych będziemy nazywać $\mathbb{P}(b_{ij} \mid Y_{obs}, Y_{miss}, \psi)$, gdzie ψ jest parametrem prawdopodobieństwa braku.

Rodzaje braków

MCAR
missing completely at random

$$\mathbb{P}(B = 1 \mid Y_{obs}, Y_{miss}, \psi) = \mathbb{P}(B = 1 \mid \psi)$$

MAR
missing at random

$$\mathbb{P}(B = 1 \mid Y_{obs}, Y_{miss}, \psi) = \mathbb{P}(B = 1 \mid Y_{obs}, \psi)$$

MNAR
missing not at random

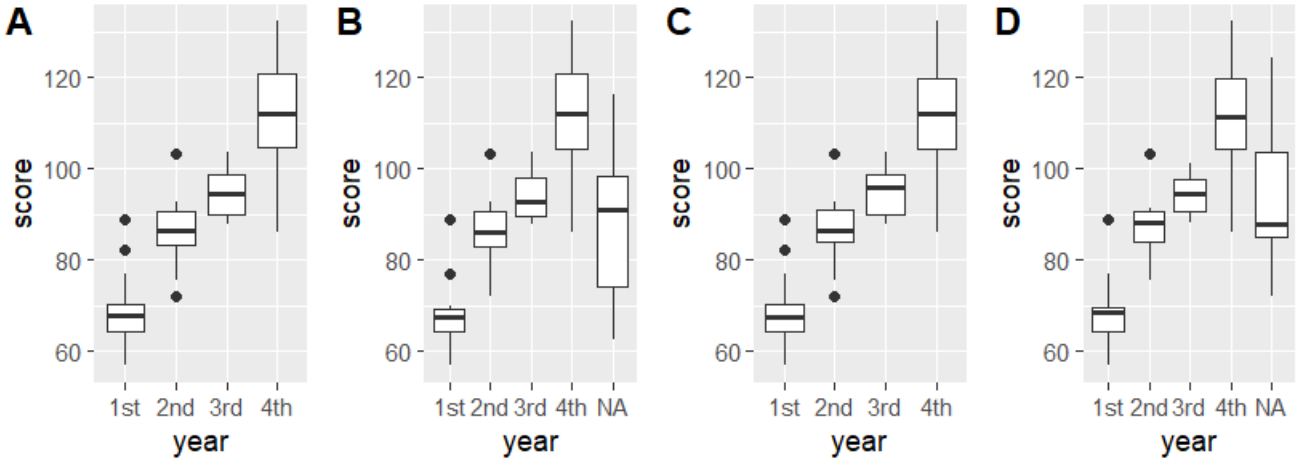
$$\mathbb{P}(B = 1 \mid Y_{obs}, Y_{miss}, \psi) \neq \mathbb{P}(B = 1 \mid Y_{obs}, \psi)$$

EXISTING SOFTWARE

Czyli jakie narzędzia radzące sobie z brakami danych już istnieją?

Istniejące pakiety

- Ogólnie są niewystarczające
- Wiele pakietów pełniących różne funkcje: imputacje, eksploracje braków, wizualizacje, podsumowania braków danych
- Brak jednoznacznych, *tidy* outputów

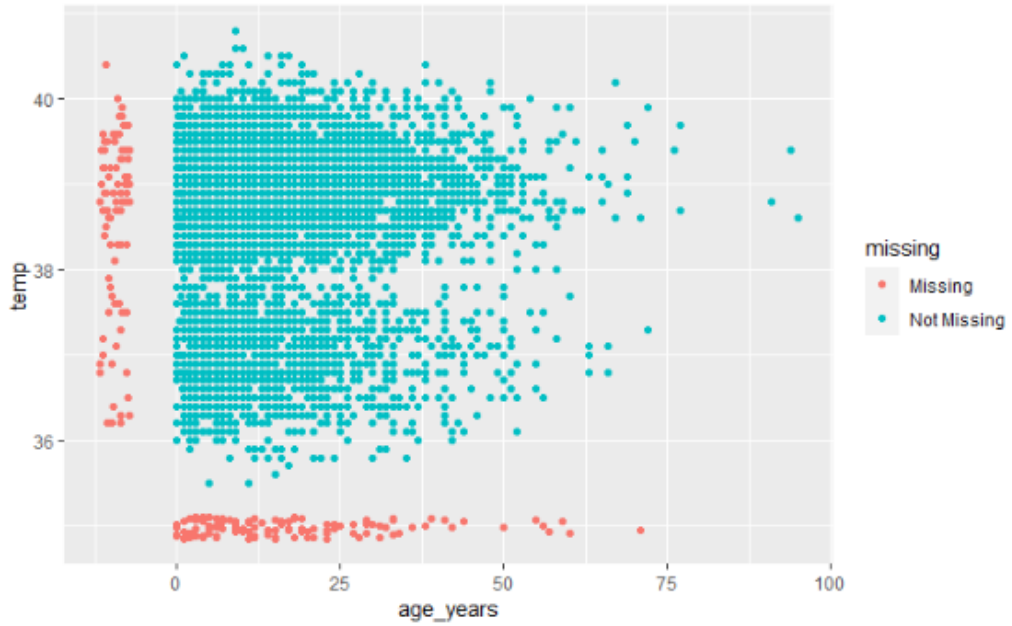


Pakiet *simputation*

- Interfejs do obsługi imputacji braków danych
- Jednoznaczny schemat działania - zawsze zwraca ramkę danych z zaimputowanymi wartościami
- Nie dostarcza informacji które dane zostały zaimputowane
- **Może współpracować z pakietem *naniar***

Co chcemy zmienić i czego potrzebujemy

- Idea shadow matrix – ale poprawiona, z wartościami NA i !NA
- Jednolitość outputu
- Zgodność z *tidy data principles*
- Lepsza obsługa imputacji
- Wizualizacja braków danych

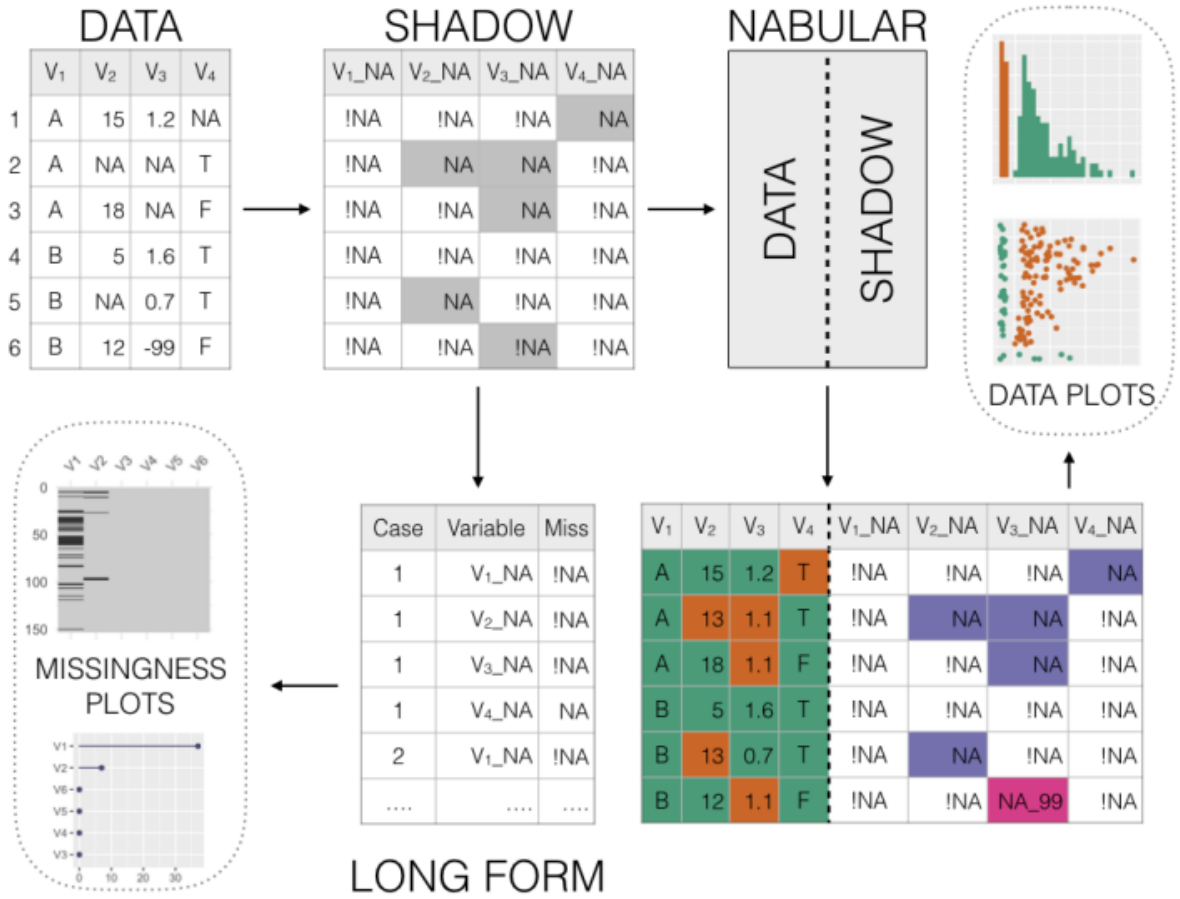


TIDY FRAMEWORK

Czyli jak zastosować tidy principles do struktur danych i operacji?

Nowa shadow matrix i nabular data

- Etykiety brakujących danych (NA, !NA)
- Specjalne wartości brakujących danych
- Skoordynowane nazwy zmiennych
- Łączność z oryginalnymi danymi



OPERACJE NA BRAKACH DANYCH

Czyli jakie funkcje ma pakiet *naniar*?

Operacje na brakujących danych

- **Sprawdzanie** czy istnieją braki w powszechnym typie
- **Oznaczenie** braków danych
- **Dodanie podsumowania**
- Dodanie **shadow matrix**
- **Oflagowanie** różnych rodzajów, np. nietypowych braków
- **Imputacja** braków danych
- **Śledzenie** procesu imputacji i eksploracji braków

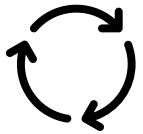
Funkcje biblioteki

- **Sprawdzanie** czy istnieją braki w powszechnym typie - ***scan***
- **Oznaczenie** braków danych - ***replace***
- **Dodanie podsumowania - *add***
- Dodanie **shadow matrix - *shadow***
- **Oflagowanie** różnych rodzajów, np. nietypowych braków – ***flag***
- **Imputacja** braków danych – ***impute***
- **Śledzenie** procesu imputacji i eksploracji braków - ***track***

scan

- Istnieje wiele różnych formatów zapisu braków danych, np. *N/A, MISSING, -99, NA, nan,* itp.
- Funkcja *miss_scan_count* zlicza występowanie braków
- Common_na_numbers
- common_na_strings

replace



- **Ujednolicenie** wyglądu braków danych
- Zamiana braków danych zapisanych w różny sposób na jednoznaczne oznaczenie braków danych
- Kilka opcji zamiany: w zależności od spełnionych warunków, w odpowiednich kolumnach, jeżeli wiersz spełnia pewien warunek

add

- Add_count() – funkcja w dplyr która zainspirowała twórców
- Funkcje z tej kategorii dodają dodatkową kolumnę podsumowującą

Function	Adds column which
`add_n_miss(data)`	contains the number missing values in a row
`add_any_miss(data)`	contains whether there are any missing values in a row
`add_prop_miss(data)`	contains the proportion of missing values in a row
`add_miss_cluster(data)`	contains the missing value cluster

Table 1: Overview of the `add` functions in **naniar**

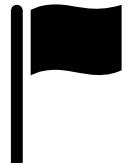
shadow

- Tworzy *nabular data*
- W istocie do istniejącej ramki danych dołącza *shadow matrix*

```
R> nabular(dat_ms)

# A tibble: 5 x 6
      x y         z x_NA  y_NA  z_NA
  <dbl> <chr> <dbl> <fct> <fct> <fct>
1     1 A      -100 !NA   !NA   !NA
2     3 N/A     -99 !NA   !NA   !NA
3    NA <NA>    -98 NA    NA    !NA
4   -99 E      -101 !NA   !NA   !NA
5   -98 F        -1 !NA   !NA   !NA
```

flag



- Istnieją różne typy braków danych
- Dzięki funkcji *recode_shadow* możemy oznaczać różne typy braków danych

```
R> nabular(dat_ms) %>%
+    recode_shadow(x = .where(x == -99 ~ "broken_sensor"))

# A tibble: 5 x 6
      x y          z x_NA              y_NA  z_NA
* <dbl> <chr> <dbl> <fct>             <fct> <fct>
1     1 A      -100 !NA               !NA   !NA
2     3 N/A     -99 !NA               !NA   !NA
3    NA <NA>    -98 NA                NA    !NA
4   -99 E      -101 NA_broken_sensor  !NA   !NA
5   -98 F        -1 !NA               !NA   !NA
```

impute

- Pakiet zawiera kilka funkcji przeprowadzającą podstawowe imputacje
- Bazowo działają na wektorach
- Funkcja *impute_below* pozwala imputowane wartości przedstawiać na wykresach
- Pakiet *simputation* oferuje znacznie bardziej rozbudowane funkcje imputacji

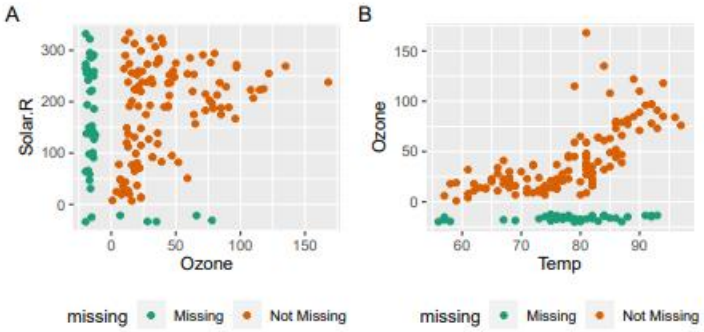
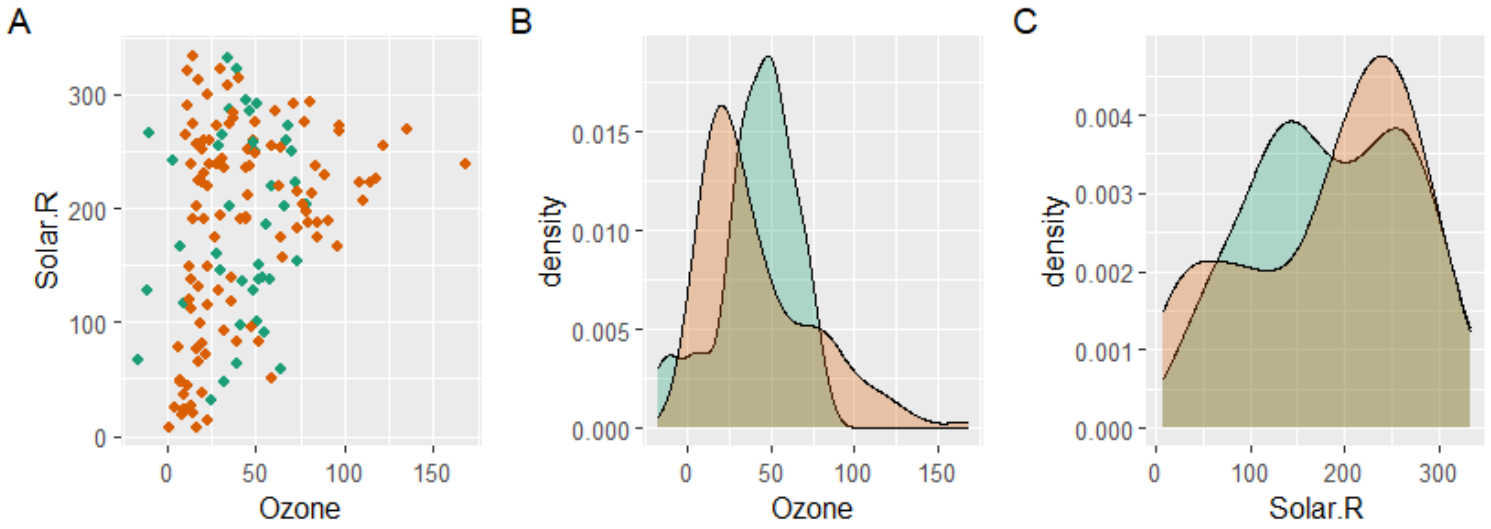


Figure 8: Scatterplots with missings displayed at 10% below for the airquality dataset. Scatterplots of ozone and solar radiation (A), and ozone and temperature (B). There are missings in ozone and solar radiation, but not temperature.

track

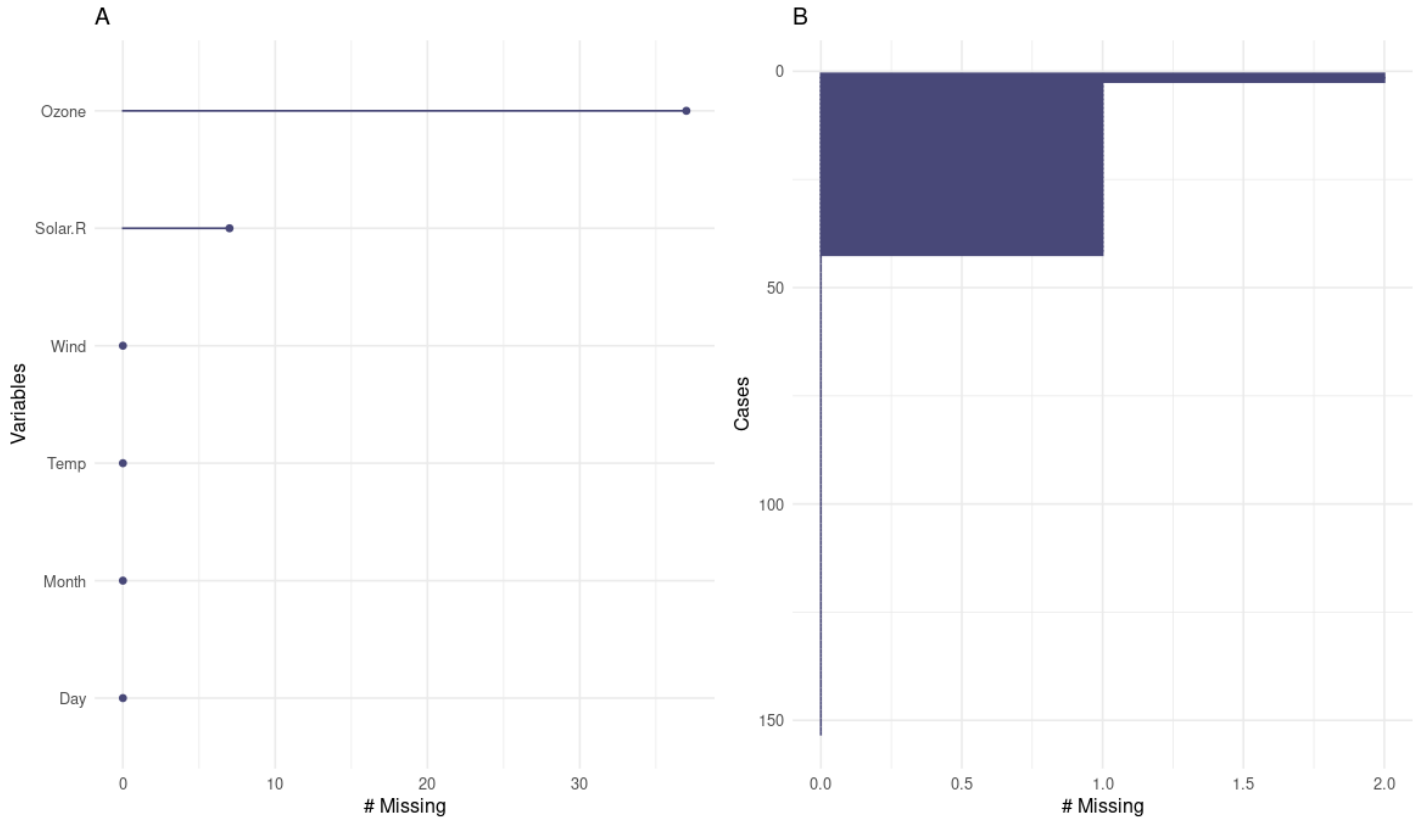
- Aby ocenić jakość imputacji, pakiet *naniar* pozwala śledzić ten proces
- Pakiet współpracuje z bardziej zaawansowanymi sposobami imputacji z *simputation*



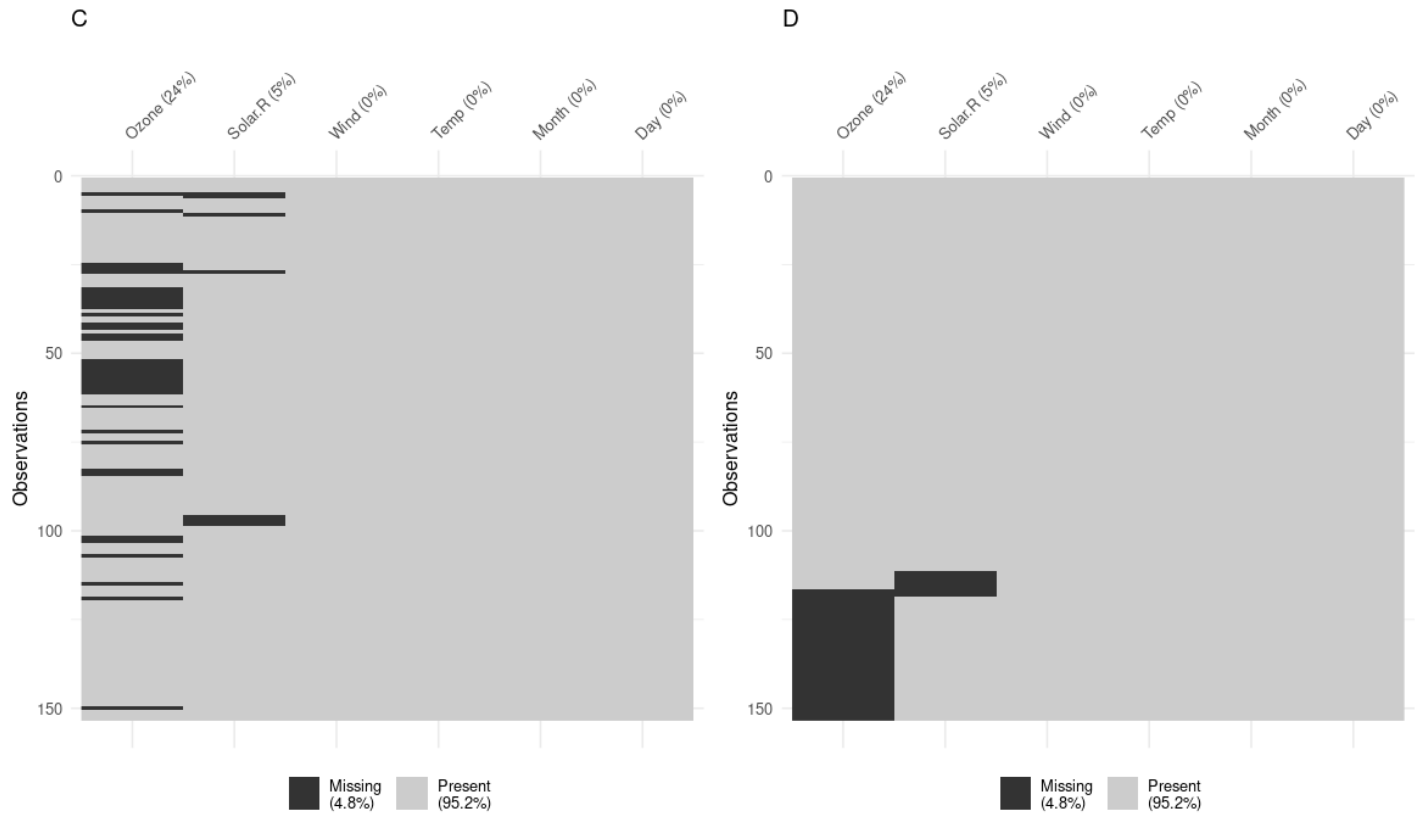
GRAFIKA

Czyli jak przedstawić braki danych na wykresach?

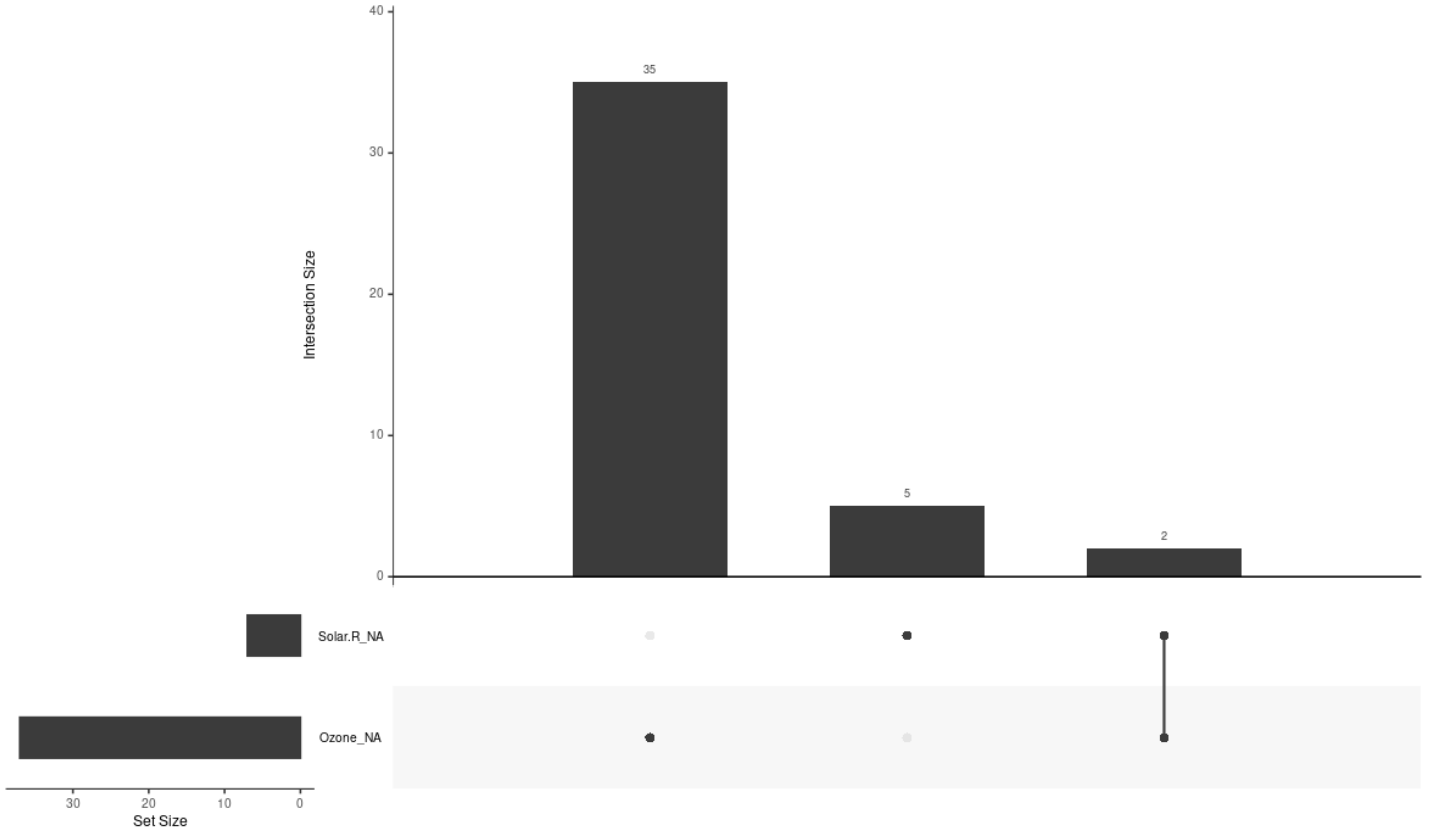
gg_miss_var() i gg_miss_case()



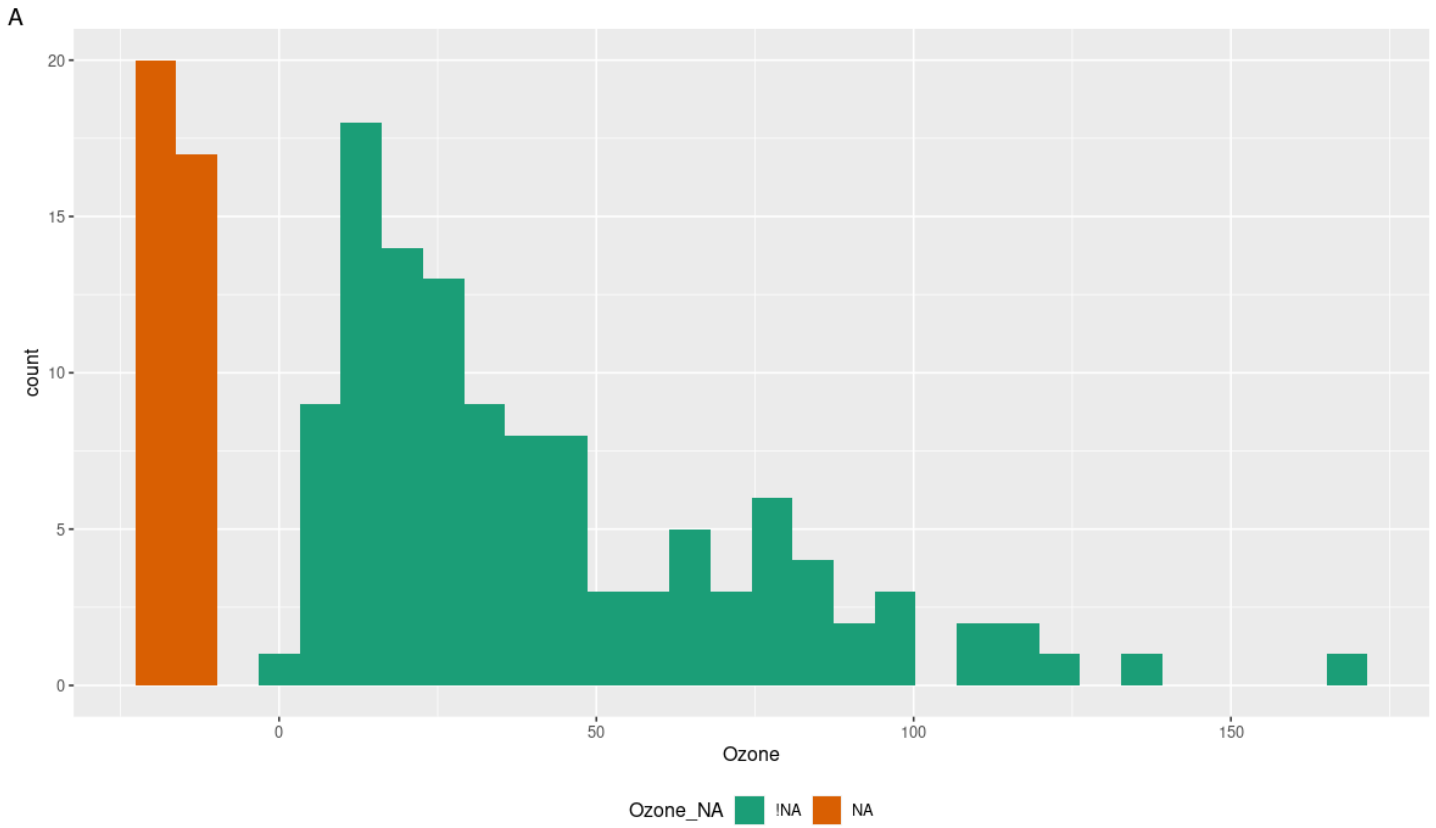
vis_miss()

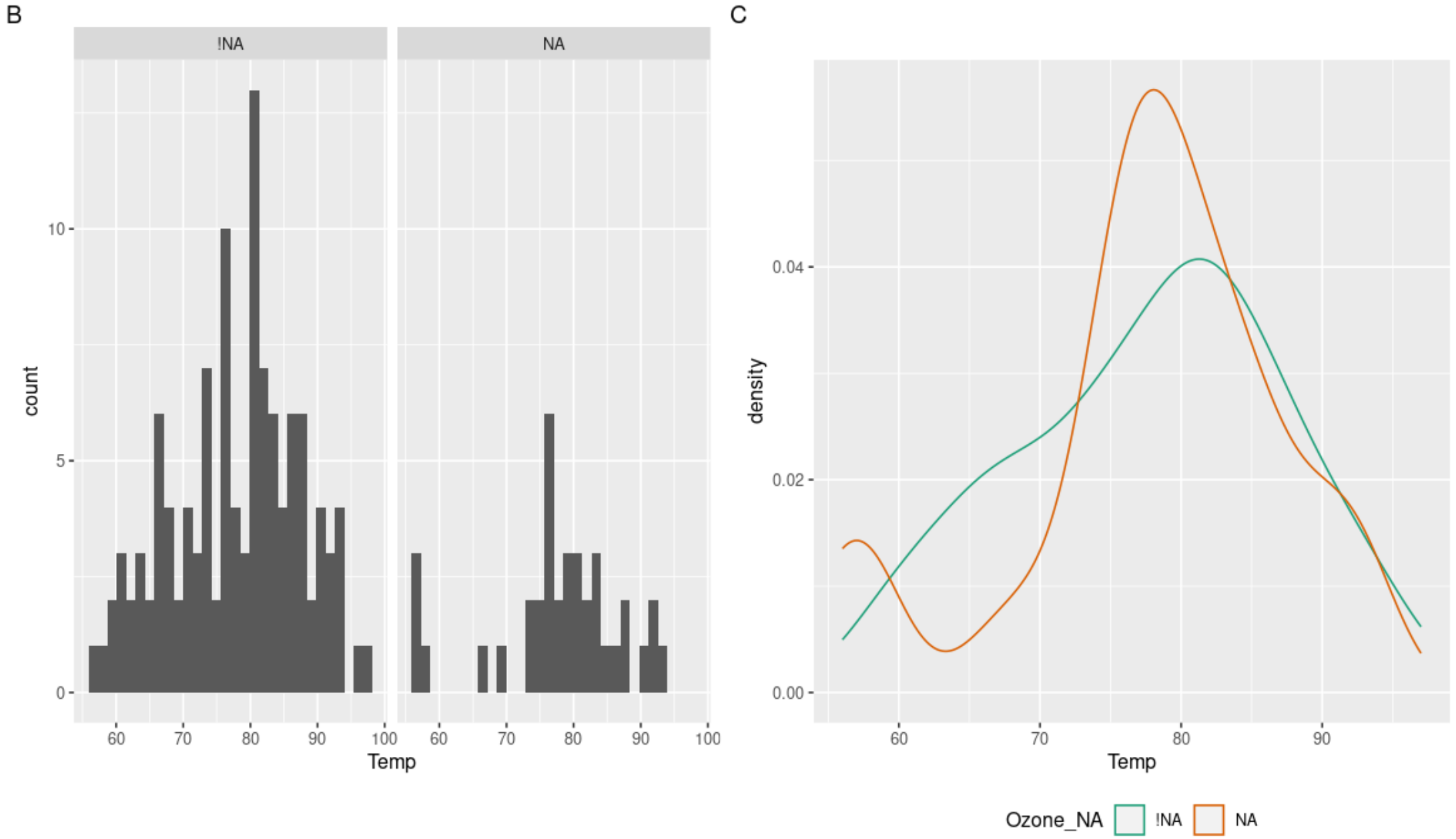

C
D
Ozone (24%)
Solar.R (5%)
Wind (0%)
Temp (0%)
Month (0%)
Day (0%)
Observations
0
50
100
150
Missing (4.8%)
Present (95.2%)

gg_miss_upset()

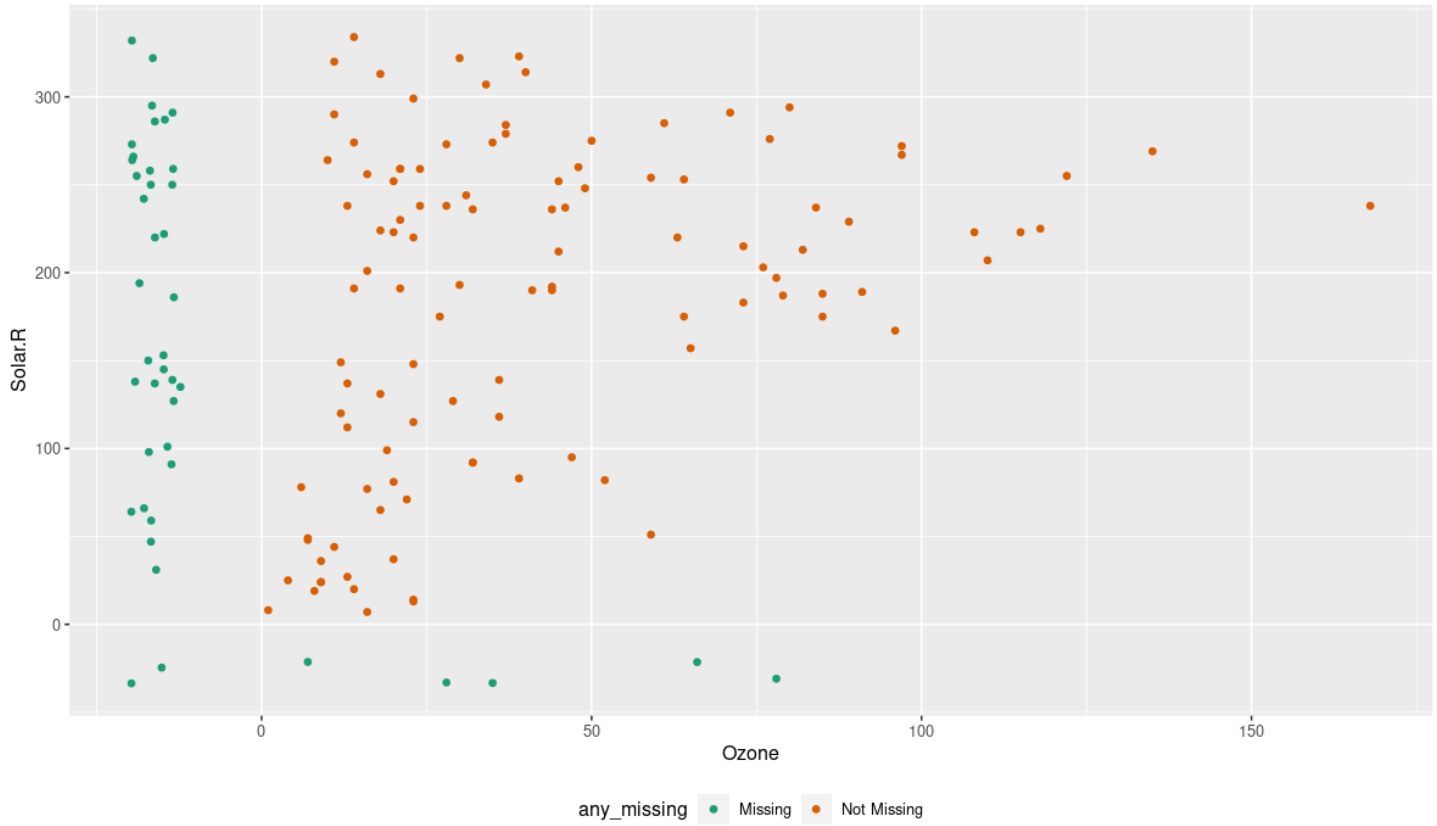


Wykresy jednowymiarowe

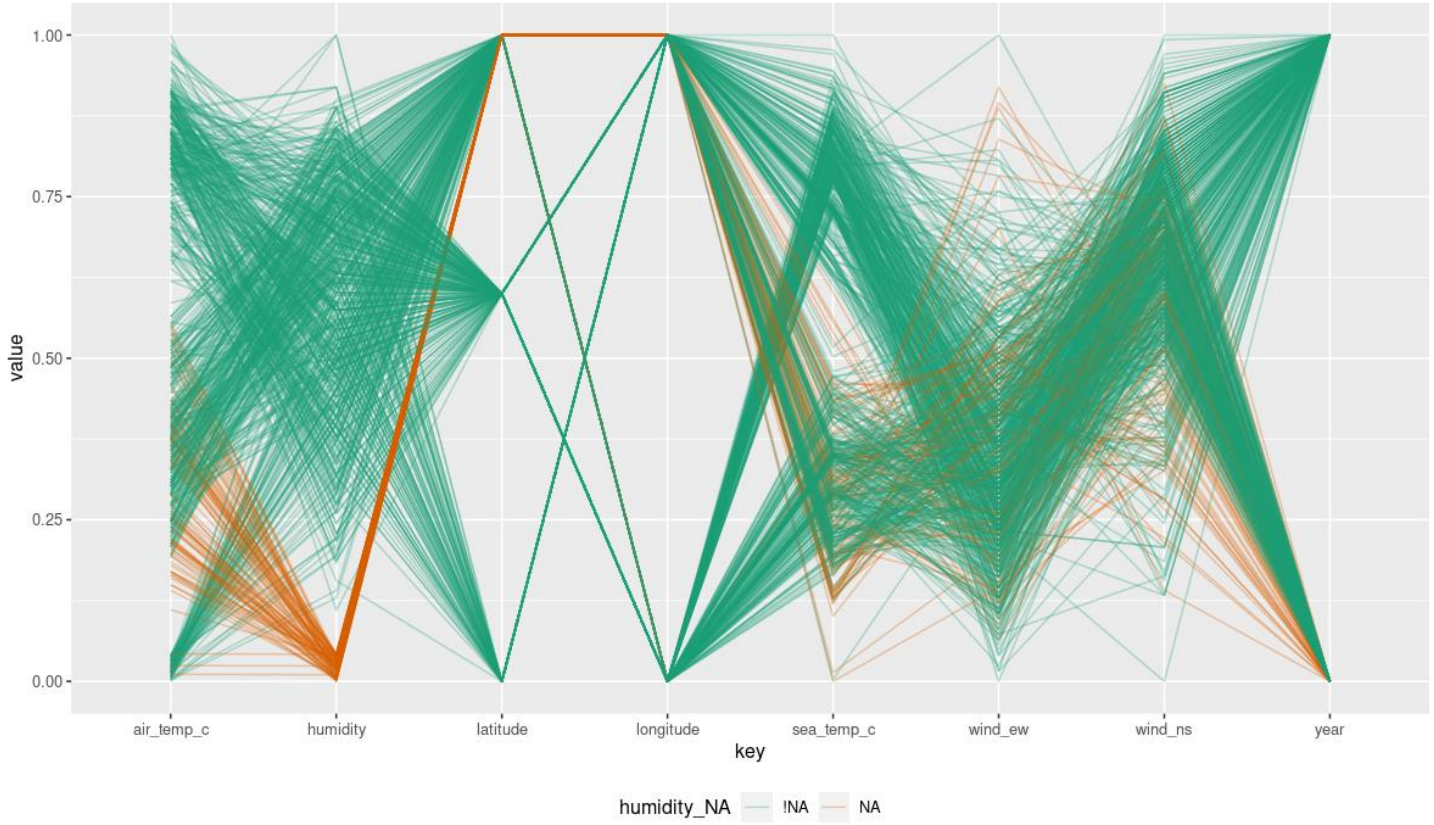



B
!NA
NA
count
10
5
0
60
70
80
90
100
Temp
C
density
0.04
0.02
0.00
60
70
80
90
Temp
Ozone_NA
!NA
NA

Wykresy dwuwymiarowe



Wykresy wielowymiarowe



SUMMARIES BRAKÓW DANYCH

Czyli jak pakiet *naniar* może podsumować braki danych w sposób tekstowy?

Po co nam podsumowania braków?

- Sprawdzenie które kolumny mają wiele braków
- Eksploracja zależności między brakami

Missing function	Missing value	Complete function	Complete value
`n_miss`	44.00	`n_complete`	874.00
`prop_miss`	0.05	`prop_complete`	0.95
`pct_miss`	4.79	`pct_complete`	95.21
`pct_miss_case`	27.45	`prop_complete_case`	72.55
`pct_miss_var`	33.33	`pct_complete_var`	66.67

Table 3: Single number summaries of missingness and completeness of the airquality dataset. The functions follow consistent naming, making them easy to remember, and their use clear.

Grupowane podsumowanie braków

- *Naniar* umożliwia podsumowanie braków ze względu na grupy
- Współpracuje z *dplyr*
- pozwala to odkrywać zależności między brakami

Month	Variable	n_miss	pct_miss
5	Ozone	5	16.1
5	Solar.R	4	12.9
5	Wind	0	0.0
5	Temp	0	0.0
5	Day	0	0.0
6	Ozone	21	70.0
6	Solar.R	0	0.0
6	Wind	0	0.0
6	Temp	0	0.0
6	Day	0	0.0

Table 6: Output of `airquality %>% group_by(Month) %>% miss_var_summary()` provides a grouped summary of the missingness in each variable, for each month of the airquality dataset. Only the first 10 rows are shown. There are more ozone missings in June than May.

DZIĘKUJEMY ZA UWAGĘ