Statystyczna Analiza Danych – laboratorium

Wprowadzenie

Dorota Celińska-Kopczyńska

Uniwersytet Warszawski

Zajęcia 1
2/3 marca 2023

Prowadząca

- dr Dorota Celińska-Kopczyńska
  - mail: dot@mimuw.edu.pl
  - strona: `mimuw.edu.pl/~dot`
  - Moodle: `https://moodle.mimuw.edu.pl/course/view.php?id=1733`
  - dyżur: czwartek 19:00-20:00, po umówieniu e-mailem

Forma zajęć

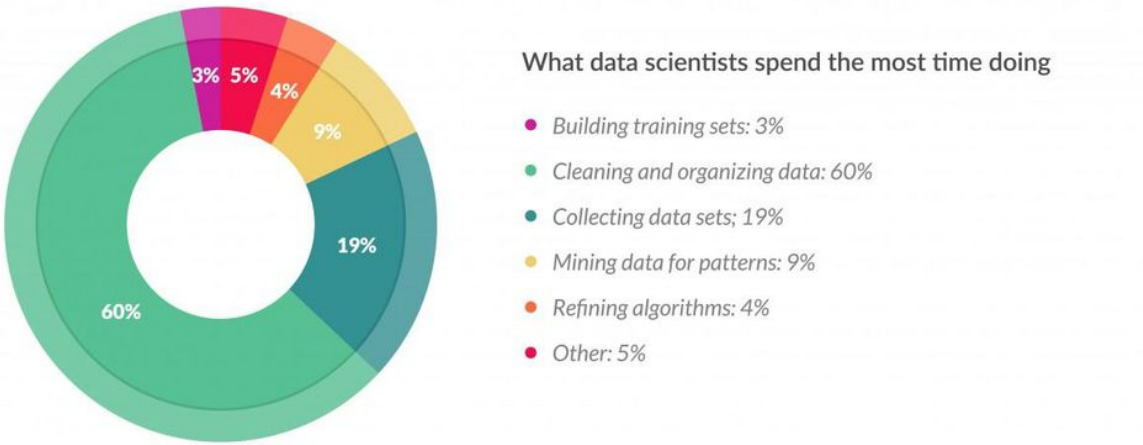
- Zajęcia w formie stacjonarnej, 14 spotkań w semestrze letnim
- Możliwe uczestnictwo o innej porze niż jest się zapisanym
- W razie przejścia na tryb zdalny spotkania w Zoom, asynchroniczna komunikacja w wydziałowym Moodle

Elementy zaliczenia w ramach laboratorium

1. Dwa projekty zaliczeniowe po 15 pkt
   - pierwszy projekt z podstaw statystyki i regresji liniowej
   - drugi projekt z zastosowaniem zbioru o wysokiej liczebności
2. Punkty za aktywność do 10 pkt (pula dodatkowa)
   - zadania do wykonania w trakcie laboratorium (od zajęć 2)
   - pytania, problemy, ciekawe pomysły
   - max 3 razy zadanie domowe (suma punktów mniejsza niż 10)

Przetwarzanie danych

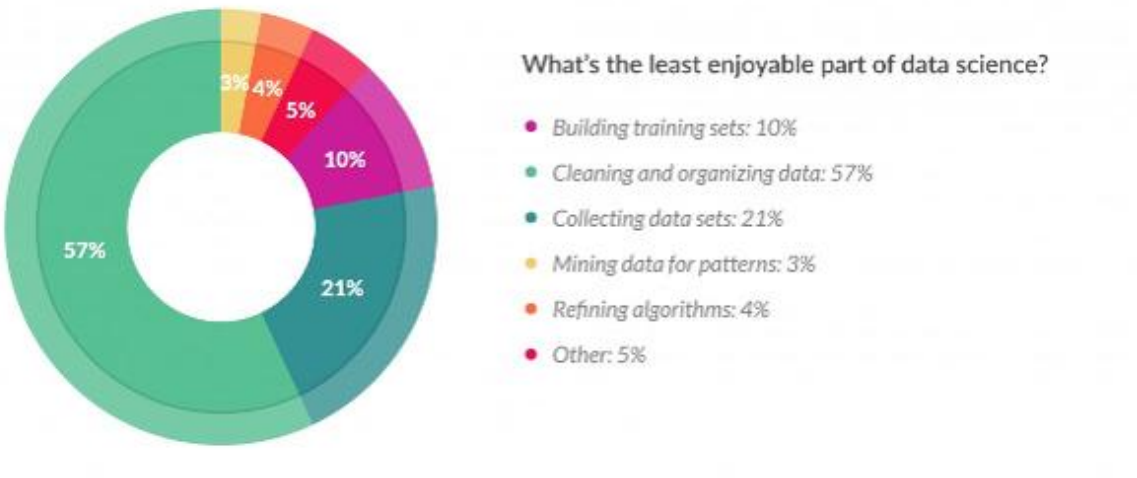
- Data scientist spędza 60% czasu na czyszczeniu i organizacji danych. Pozyskiwanie danych jest na drugim miejscu z wynikiem 19% czasu, co oznacza, że 80% czasu jest przeznaczane na przygotowaniu i opracowywaniu danych dla analizy.



https://www.forbes.com/sites/gilpress/2016/03/23/data-preparation-most-time-consuming-least-enjoyable-data-science-task-survey-says/19d366d06f63

Przetwarzanie danych

- Jednocześnie 57% data scientistów uznaje przygotowanie danych za najmniej przyjemną część ich pracy



https://www.forbes.com/sites/gilpress/2016/03/23/data-preparation-most-time-consuming-least-enjoyable-data-science-task-survey-says/19d366d06f63

Obróbka danych

- Smutne, ale prawdziwe – **spędza się** wiele **godzin** na przyszykowaniu zbioru danych
- Najczęściej nie jest to wyzwaniem, nie daje satysfakcji ani nie jest zabawne. Po prostu żmudne.
- Jest jednak dobra strona – później to **Państwo** decydują, z jakiego narzędzia będą chcieli korzystać
- Warto, żeby to było coś, co Państwo polubią

Obróbka danych – możliwości

- Języki programowania (szczególnego zastosowania: R, Julia, Matlab
- Języki programowania (ogólnego przeznaczenia): Python, C++
- Pakiety statystyczne: Stata, SPSS, Statistica
- Arkusze kalkulacyjne
- alternatywne narzędzia: awk, sed, grep, narzędzia powłoki
- ... inne możliwości, których tu nie wymieniono

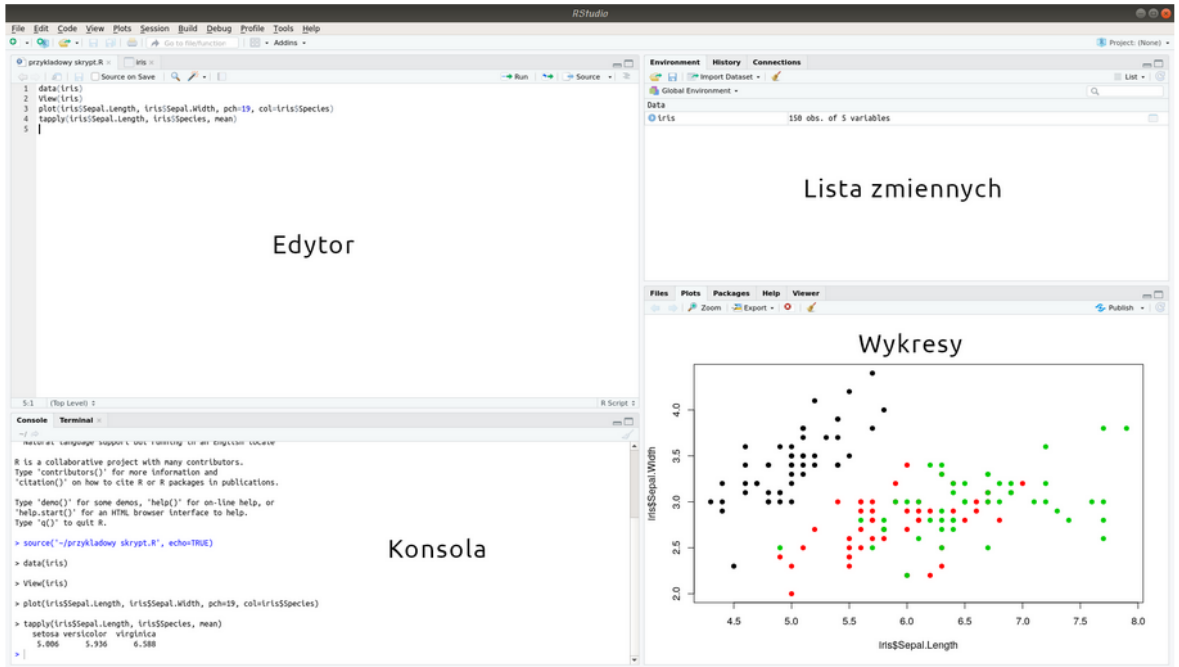
R – tego będziemy tutaj używać

- R to język programowania pozwalający na wykonanie realtywnie prosto obliczeń statystycznych
- Kod źródłowy R opublikowany jest na zasadach licencji GNU GPL
- Praca może odbywać się w konsoli
- Ale podczas zajęć najczęściej używać będziemy GUI – R-studio

Skąd pozyskać?

- R: https://cran.r-project.org/
- R-studio: https://rstudio.com/products/rstudio/download/

R-studio


RStudio
Edytor
Lista zmiennych
Konsola
Wykresy
iris$Sepal.Length
iris$Sepal.Width

R – podstawy

- Praca w trybie tekstowym
- Dodatkowe pakiety należy zainstalować: `install.packages("nazwa")`
- I uruchomić: `library(nazwa)`

R – podstawy

- Przypisanie: `nazwa_obiektu <- komenda`
- Przypisanie: `nazwa_obiektu = komenda`
- Operatory: `+,-,*,/,!,==`
- `:` pozwala zdefiniować zakres, np 1:20

Typy danych

- R rozróżnia typy danych, najbardziej podstawowym jest vector
- Wśród statystyków popularna również ramka danych: data frame
- Więcej o typach danych za tydzień

ggplot2

- Służy do przygotowania grafiki naukowej w R
- Wskazywana, jako jedna z mocnych stron R
- Opiera się na paradygmacie tidy
- Wbrew pozorom nie jest jedyną opcją do tworzenia wykresów

ggplot2 – składnia

- Budujemy wykres z klocków
- Składnia komend ggplot2 zawiera następujące elementy:
  1. Stworzenie pustego wykresu: ggplot()
  2. Dodanie co najmniej jednej warstwy, każda symbolizuje osobny typ wykresu, tworzymy ją najczęściej komendą zaczynającą się od `geom_` lub `stat_`
  3. Dodatkowe opcje (nieobowiązkowe), np. temat – wzorzec wyglądu
- Poszczególne elementy łączymy ze sobą plusami

warstwy – składnia

- Warstwa określa jaki typ wykresu nas interesuje, np. `geom_point()` zwraca wykres punktowy
- To, co ma się znaleźć na wykresie podajemy za pomocą estetyk (*aesthetics*)
- Estetyki określają m.in. położenie na osi x i na osi y, kolor, rozmiar i wypełnienie punktów.

Przykładowa komenda

```
ggplot(iris) + geom_point(aes(x=Sepal.Length,
y=Sepal.Width, col=Species))
```

- `ggplot(iris)` – pusty wykres, Wszystkie wartstwy pracowałyby ze zbiorem danych iris
- `geom_point()` – będzie wykres punktowy
- `aes(x=Sepal.Length, y=Sepal.Width, col=Species)` – estetyka dla wykresu punktowego. x i y to dane, które umieścimy. Punkty zostaną pokolorowane na podstawie wartości zmiennej Species.
- Wiemy, z jakiego zbioru pochodzą zmienne – zostało to przekazane w ggplot()