Markdown i wizualizacja - podstawy

Poniższy segmnt kodu ustawia parametry globalnie dla wszystkich segmentów funkcją z biblioteki knitr knitr::opts\_chunk$set(). Zaznaczona w naglówku opcja include=FALSE oznacza, że ten fragment nie pojawi się w dokumencie wygenerowanym z tego pliku markdown.

# Segmenty tekstowe - formatowanie i najważniejsze informacje

Tekst zwykły

# nagłówek pierwszego poziomu

## nagłówek drugiego poziomiu

### nagłówek trzeciego poziomu

# Formatowanie tekstu[[1]](#footnote-24)

tekst[[2]](#footnote-25)

*kursywa* **pogrubienie** kod indeks górny 2 indeks dolny 2

# Nagłowek 1

## Nagłówek 2

listy [@girard1987]

* Element 1. listy wypunktowanej
* Element 2.
  + Element 2a
  + Element 2b

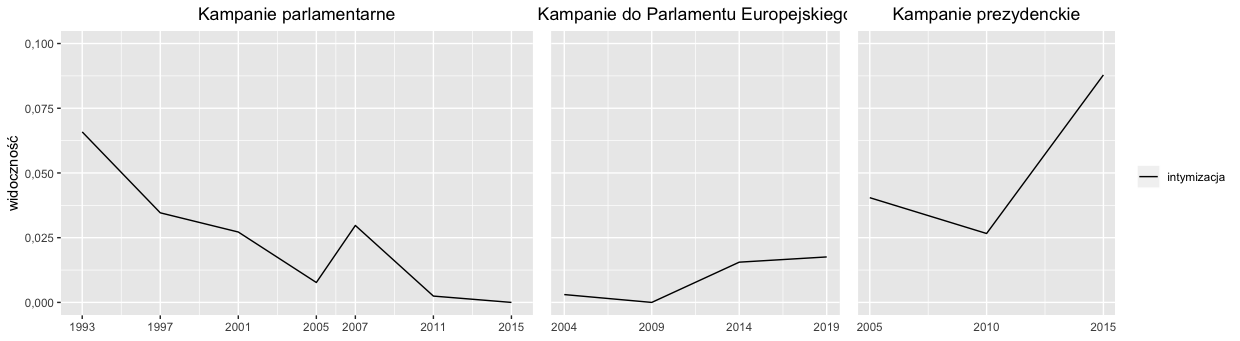
1. Element 1. listy numerowanej
2. Element 2. Numeracja zostanie automatycznie zwiększona w dokumencie wyjściowym.

block quotes

# Łącza i obrazy

<http://rmarkdown.rstudio.com>

[R Markdown](http://rmarkdown.rstudio.com)

Obraz z mojego dysku: 

# Tabele

# tabela

**tabela**

Tabele

| Prawy | Lewy | domyślny | Center |
| --- | --- | --- | --- |
| 12 | 12 | 12 | 12 |
| 123 | 123 | 123 | 123 |
| 1 | 1 | 1 | 1 |

data <- faithful[1:4, ]  
knitr::kable(data,  
 caption = "Tabela z kable")

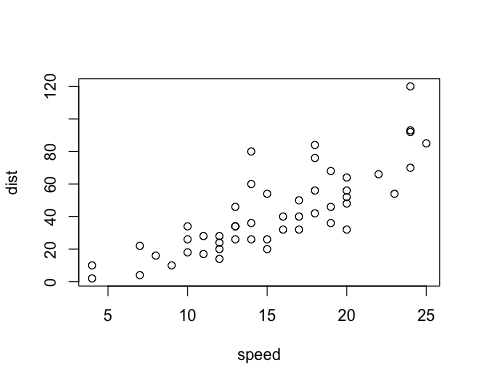
Tabela z kable

| eruptions | waiting |
| --- | --- |
| 3.600 | 79 |
| 1.800 | 54 |
| 3.333 | 74 |
| 2.283 | 62 |

Notatnik. Wykonanie kodu w notatniku sprawia że rezultaty pojawiają się pomiżej kodu.

Try executing this chunk by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Cmd+Shift+Enter*.

plot(cars)



Przy zapisywaniu notatnika plik HTML zawierajacy kod i jego efekty zostanie zapisany wraz z nim (klikając *Preview* lub skrótem *Cmd+Shift+K* można uzyskać widok preview).

Preview pokazuje wyrenderowaną kopię HTML. Inaczej niż *Knit*, *Preview* nie uruchamia kodu . W preview widać efekt ostatniego uruchomienia kodu w edytorze.

[@grolemund] [@grolemund] To nie jest wersja po renderze tylko pisana, stąd cytowania w tej formie

## Segmenty kodu

Ogólne informacje

Kod w tekście: Dwa dodać dwa to 4

Kod dodajemy *Insert Chunk* na pasku narzędzi albo skrótem *Cmd+Option+I* w Widnows CtrL zamiast cmd

## Opcje segmentów kodu

Opcje segementów są zapisywane w nagłówkach segmentu

``` {r } ```

Format tag=value

### nagłówki segmentów kodu (chunk headers)

* nagłówek musi być pisany jako jedna linia -unikać spacji, kropek i podkreślników w etykietach segmentu (chunk labels)

#### Etykieta segmentu

chunk label (n.p, my-chunk ). Tylko ona nie ma taga ., my-chunk, albo label = “my-chunk” Te etykiety muszą być unikalne, segementom bez numeru przypisywane są etykiety takie jak unnamed-chunk-i, gdzie i jest rosnącą liczbą.

### Opcje domyślne

knitr::opts\_chunk$set() zmienia domyślne wartości ustawień dla segmentów kodu .

knitr::opts\_chunk$set()

### Najważniejsze opcje

#### Kod

* eval(TRUE) czy wykonać fragment kodu czy nie, mozną takżę podać które wyrażenia wykonać za pomocą wektora liczb eval = c(1,3)

#### tekst

* echo (TRUE), czy kod ma być wyświetlony w dokumencie wyjściowym
* results:
  + asis, pisze rezultaty tak jak są surowy rezutlat bezpośrednio w dokumencie bez znaczników any markups.
  + markup, ze znacznikami
  + hide lub FALSE): ukrywa tekstowe.
* collapse czy skolapsować całe źródło i to co wychodzi do jednego bloku
* warning: czy zostawić ostrzeżenia w pliku wyjściowym, jeśli FALSE to nie
* error (TRUE), jeśli FALSE kod będzie się zatrzymywał na błędach, domyślnie jest na FALSE
* messageCzy zostawić komunikaty emitowane przez message()
* include czy efekt działania danego segmentu ma zawierać się dokumecie wyjściowym, jeśli FALSE nic nie zostanie zapisane w dokumencie wyjsciowym ale kod wciąż będzie wykonany ( na ile rozumiem w Rmarkdown nie ma różnicy między evaluate a execute)

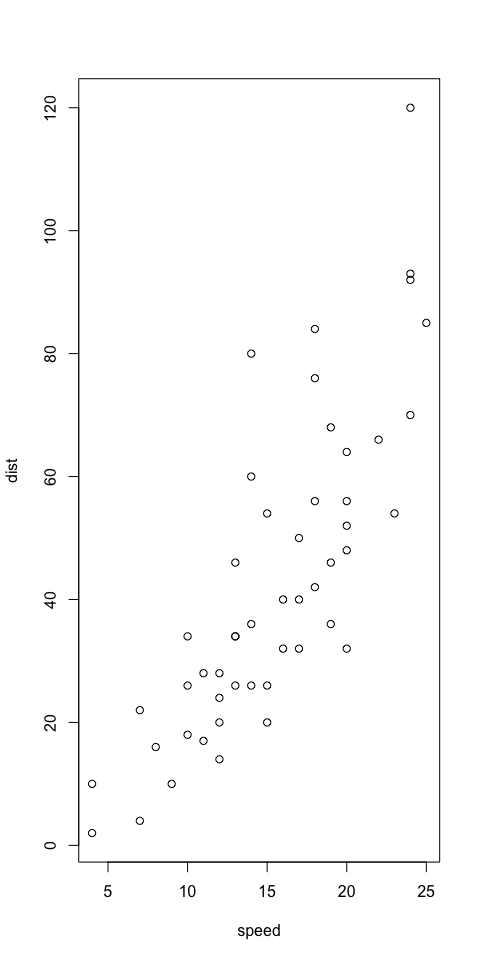
#### Związane z wykresami

* fig.path ścieżka do pliku, fig.path i chunk label zostają połączone i tworzą pełną ścieżkę pliku, może być ścieżka do folderu

np:

* fig.keep: jak wykresy z segmentu mają być zatrzymane, high, none, all, first ,last
* fig.show (asis) jak pokazać zaaranżować wykresy, tak jak jest, hold na końcu segmentu, animate - połącz wykresy w animacje jeśli jest ich wiele, hide
* dev: pdf dla latex i png dla html markdown, urządzenie graficzne pdf, png, svg, jpeg, tiff, cairo\_pdf, CairoJPEG, CairoPNG, svglite, gridSVG, ragg\_png, tikz, można sprawdzić knitr::auto\_exts, mogą być wektory np dev = c('pdf', 'png') ten wytworzy dwa wykresy, jeden w pdf drugi w png
* dev.args argumenty które można przekazać do urządzenia graficznego
* dpi( numeryczna)
* fig.height(), fig.width numeryczne wysokość i szerokosć w calach
* fig.asp stosunek wysokości do szerokości jeśli jest podany, wysokość kalkuluowana jest przez przemnożenie przez ten paramter
* fig.dim: wymiary wykresu, = c(szerokość, wysokość)
* fig.align: Possible values are default, left, right, and center.
* fig.retina: dla html, dla ekranów retina, ustawienie zazwyczaj na dwa zmienia dpi
* out.width:, out.height: szerokość i wysokość w dokumencie wyjściowym, moze być rózna od fizycznej, wykresy można skalować, może to być procent, w HTML pixele
* out.extra - dodatkowe opcje na przykład angle = 90 jako rotacja
* fig.cap podpis pod wykresem
* fig.alt

plot(cars) #komentarz



# jak umieścić kod z nagłówkiem

<https://bookdown.org/yihui/rmarkdown-cookbook/show-header.html>

### Bibliografia

1. Przypis numer 1 [@goffman1987] [↑](#footnote-ref-24)
2. tekst przypisy numerują się automatycznie, ale muszą mieć identyfikatory [↑](#footnote-ref-25)