

Pogromcy Danych

Wizualizacja oraz modelowanie danych

Przemysław Biecek
Uniwersytet Warszawski

Wizualizacja danych

Przemysław Biecek @ Uniwersytet Warszawski

sezon 2 / odcinek 1

pogRomcy danych

- O czym jest ten odcinek
- Jak zrobić wykres w pakiecie ggplot2
- Co czyni wykres złym lub dobrym?

O czym jest ten odcinek

Sezon drugi składa się z dwóch części, jednej poświęconej wizualizacji i jednej poświęconej modelowaniu.

Seria o wizualizacji też jest podzielona na dwie części: narzędziową i teoretyczną.

W części narzędziowej pokazujemy praktycznie jak tworzyć wykresy w programie R. W części teoretycznej omawiamy podstawowe zagadnienia związane z wizualizacją danych.

Część narzędziowa wymaga pewnego komentarza. W programie R wykresy można tworzyć na wiele bardzo różnych sposobów. Dostępne są biblioteki i funkcje oparte o pakiet `graphics`, są też rozwiązania oparte o pakiet `grid` czyli `ggplot2` i `lattice`. Są w końcu rozwiązania opracowane dla grafiki interaktywnej, czyli `ggvis`, `rCharts` i inne. Różnych rozwiązań jest tak wiele, że na omawianie ich wszystkich nie wystarczyłoby życia.

Najbardziej zaawansowana i najbardziej dojrzała jest rodzina rozwiązań opartych o pakiet `ggplot2`. Wykresy przygotowane z pakietem `ggplot2` wyglądają też bardzo estetycznie, dlatego zdecydowałem się omówić tutaj ten właśnie pakiet.

Jak zrobić wykres w pakiecie `ggplot2`

Część narzędziowa składa się z czterech odcinków

- Jak zrobić wykres? - Wprowadzenie do pakietu `ggplot2`, tutaj przedstawimy kluczowe rozwiązania tego pakietu oraz pokażemy jak zrobić wykresy z prostymi geometriami,
- Wykresy agregaty - W tym odcinku zaprezentowane będą geometrie, które liczą agregaty z danych,
- Dodatki do wykresów - W tym odcinku pokażemy jak pracować z detalami na wykresie, jak poprawiać

poszczególne elementy wykresów, zmieniać kolory, wielkości, kształty i mieć pełną kontrolę nad tym co i jak wygląda.

- Jak zrobić mapę? - Ostatni odcinek serii narzędziowej pokazuje jak rysować kartogramy używając pakietu `ggplot2`.

Co czyni wykres złym lub dobrym?

Część teoretyczna składa się z pięciu odcinków. Każdy z nich to krótki film wprowadzający do określonego zagadnienia.

- Historia i współczesność - Przedstawia najciekawsze historyczne grafiki statystyczne oraz pokazuje jak dobre pomysły z historii znajdują nowe zastosowania dzisiaj,
- Iluzje - Przedstawia wybrane problemy ludzkiego aparatu widzenia, który wiele rzeczy robi dobrze, ale akurat precyzja nie jest jego silną stroną.
- Info-pomyłka - Przedstawia wykresy gubiące lub zniekształcające informacje. Ku przestrodze.
- Jak dobierać kolory? - Przedstawia wybrane problemy z doborem kolorów, oraz odpowiada na fundamentalne pytanie: Jak (oraz czy) dobierać kolory?
- Jak pokazywać różne liczby? - Przedstawia wybrane

problemy z prezentacją liczb różnych typów.

Każda z tych części to streszczenie jednego z rozdziałów z książki „Zbiór esejów o sztuce prezentacji danych” dostępnego pod adresem <http://biecek.pl/Eseje>. Do każdej części poza krótkim filmem wskazujemy też przykładowe inne źródła, w których można przeczytać więcej na dany temat.

Jak zrobić wykres w ggplot2?

Przemysław Biecek @ Uniwersytet Warszawski

*sezon 2 / odcinek 2
pogRomcy danych*

- O czym jest ten odcinek
- Zbiory danych
- Dlaczego warto poznać pakiet ggplot2?
- Dlaczego ggplot2 a nie standardowe pakiety R?
- Co różni te dwa wykresy?
- Co różni te dwa wykresy?
- Kiedy należy używać wykresu punktowego?
- Grupy na wykresie punktowym
- Jak zrobić wykres punktowy?
- Elementy wykresu punktowego
- Dodajemy mapowania - kształt punktu - shape
- Jak budować wykresy z ggplot2?
- Dodajemy mapowania - wielkość punktu - size
- Dodajemy mapowania - kolor punktu - color
- Wiele mapowań dla tej samej zmiennej

- Na jaką cechę mapować?
- Geometria `geom_text()` - etykiety tekstowe na wykresie
- Geometria `geom_text()` - pozycjonowanie napisu
- Składanie dwóch geometrii
- Kiedy stosować etykiety na wykresie?
- Globalne mapowania
- Lokalne mapowania
- Geometria `geom_line()` - wykres liniowy
- Kiedy stosować linie na wykresie?
- Geometria `geom_ribbon()` - wykres wstęga
- Gdzie szukać dalszych informacji
- Zadanie, sezon 2, odcinek 7

O czym jest ten odcinek

Dane są czymś abstrakcyjnym i patrząc w tabele liczb często trudno zauważać zależności pomiędzy różnymi zmiennymi.

Ale znacznie łatwiej dostrzec lub zrozumieć te zależności, jeżeli dane przedstawi się graficznie w poprawny sposób.

Pierwsza połowa sezonu 2 poświęcona jest wizualizacji danych. W tym odcinku poznamy pakiet graficzny `ggplot2`, pozwalający na tworzenie dobrze wyglądających i czytelnych wykresów.

W tym odcinku dowiemy się, że:

- W programie R wykresy można tworzyć z użyciem najróżniejszych pakietów. Dowiemy się, dlaczego w pierwszej kolejności warto poznać i dlaczego warto używać pakietu `ggplot2`.
 - Jednym z najpopularniejszych sposobów przedstawiania danych są wykresy punktowe. Pokażemy jak tworzyć takie wykresy.
 - Jednym z kluczowych pomysłów w pakiecie `ggplot2` są mapowania zmiennych na właściwości wykresu. Dokładnie omówimy na czym polega ten pomysł.
 - Poznamy kolejne rodzaje wykresów, takie jak wykres z etykietami lub wykres liniowy.
 - Pokażemy też jak składać różne warstwy w jeden czytelny wykres.
-

Zbiory danych

W tym odcinku będziemy pracować z dwoma zbiorami danych: `koty_ptaki` oraz `WIG`. Oba zbiory są dostępne w pakiecie `PogromcyDanych`.

Aby odtworzyć przykłady z tego odcinka potrzebny jest zainstalowany pakiet `PogromcyDanych`. Informacja o tym

jak go zainstalować jest umieszczona w odcinku trzecim pierwszego sezonu *Jak zainstalować R, RStudio oraz dodatkowe pakiety?*.

Funkcją `library()` włączamy pakiet. Funkcją `head()` wyświetlamy pierwsze sześć wierszy z każdego ze zbiorów danych.

```
## jeżeli pakiet jeszcze nie jest zainstalowany
## install.packages("PogromcyDanych")

## wczytujemy pakiet
library(PogromcyDanych)
## wyświetlamy pierwsze wiersze z interesującymi
head(koty_ptaki)

##      gatunek   waga  dlugosc predkosc habitat  zywoc
## 1     Tygrys    300      2.5        60    Azja
## 2       Lew     200      2.0        80  Afryka
## 3    Jaguar    100      1.7        90 Ameryka
## 4      Puma      80      1.7        70 Ameryka
## 5  Leopard      70      1.4        85    Azja
## 6   Gepard      60      1.4       115  Afryka

head(WIG)
```

```
##            Data Nazwa Kurs.otwarcia Kurs.maksymalny
## 1 2013-12-02   WIG          54627.26             5479.0
## 2 2013-12-03   WIG          54025.72             5402.0
## 3 2013-12-04   WIG          53222.49             5328.0
## 4 2013-12-05   WIG          52837.25             5290.0
## 5 2013-12-06   WIG          52837.58             5289.0
## 6 2013-12-09   WIG          53113.49             5318.0
##           Kurs.zamkniecia Zmiana Wartosc.obrotu.w.tygodniu
```

# # 1	53934.52	-1.41	640
# # 2	53276.83	-1.22	914
# # 3	52867.03	-0.77	968
# # 4	52597.13	-0.51	808
# # 5	52727.52	0.25	1012
# # 6	52881.29	0.29	591

Dlaczego warto poznać pakiet `ggplot2`?

Coraz więcej osób dostrzega potencjał wizualizacji danych zarówno w zastosowaniach biznesowych, w nauce, mediach czy życiu codziennym. Z tego powodu powstaje coraz więcej rozwiązań, programów i pakietów, pozwalających na prezentację danych.

My skupimy się na jednym, na pakiecie `ggplot2`. Dlaczego akurat nim? Jest ku temu kilka powodów.

- Zaawansowana analiza danych bardzo często ma miejsce w programie R. Wykorzystanie tego samego programu do analiz i do wizualizacji, ma tę zaletę, że nie musimy danych konwertować do nowego formatu, nie musimy poznawać dwóch języków opisu wizualizacji, możemy pracować w jednym spójnym środowisku. Z pakietów programu R, na dzień dzisiejszy, to właśnie `ggplot2` jest najbardziej

dojrzałym rozwiązaniem.

- Pakiet `ggplot2` pozwala na tworzenie grafiki *publication ready*, czyli wystarczająco dobrej by pokazać ją innym osobom w prezentacjach czy raportach biznesowych lub publikacjach naukowych, bez dodatkowych zabiegów upiększających.
 - Pakiet `ggplot2` wymusza myślenie o wykresach danych w kategorii zmiennych, które chcemy przedstawić a nie w kategorii szablonu, który chcemy wypełnić. Krzywa uczenia jest bardziej stroma, niż w przypadku „klikadeł”, ale z czasem nasze możliwości niepomiernie rosną.
-

Dlaczego `ggplot2` a nie standardowe pakiety R?

Wykorzystując zbiór danych `koty_ptaki` przedstawmy zależność wagi od prędkości dla wybranych gatunków. Wykorzystamy w tym celu dwa pakiety graficzne dostępne w programie R, pakiet `graphics` z funkcją `plot()` oraz pakiet `ggplot2` z funkcją `ggplot()`.

Poniższe instrukcje tworzą wykres z domyślnymi ustawieniami. To, w jaki sposób tworzą one wykres wyjaśnimy za cztery slajdy, na slajdzie o tytule *Elementy wykresu punktowego*. W tym miejscu, przyjrzyjmy się

wyłącznie wynikom, które obie instrukcje produkują.

Bez wczytywania dodatkowych pakietów, możemy korzystać z funkcji `plot()`

```
plot(koty_ptaki$waga, koty_ptaki$predkosc)
```

Funkcje graficzne, które będziemy omawiać, wymagają włączenia dodatkowego pakietu `ggplot2`. Jeżeli ten pakiet nie jest zainstalowany, należy go zainstalować poleciением `install.packages("ggplot2")`. W pakiecie `ggplot2` wykresy wygodnie tworzy się funkcją `ggplot`.

```
library(ggplot2)
ggplot(koty_ptaki, aes(x=waga, y=predkosc)) +
  geom_point()
```

Co różni te dwa wykresy?

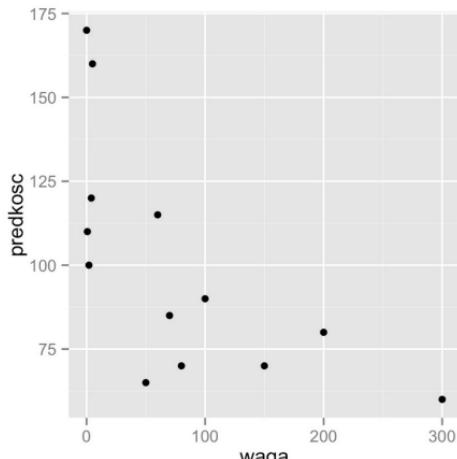
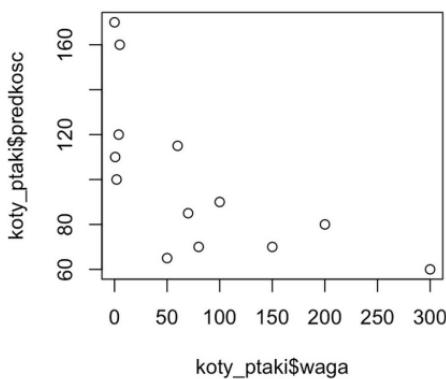
Dla kolejnych wierszy zbioru danych `koty_ptaki`, na wykresie przedstawiamy zależność pomiędzy zmiennymi `waga` a `predkosc`. Każdy wiersz opisany jest przez jeden punkt, przez co wykres ten nazywa się wykresem punktowym.

Na tym wykresie przedstawiane są duże koty i ptaki. Lżejsze gatunki to głównie ptaki, są one też zazwyczaj

szybsze niż duże koty. Więc przedstawiany wykres pokazuje, że dla tych danych, im cięższy gatunek, tym niższa jego prędkość maksymalna.

Pomimo jednak tego, że oba wykresy przedstawiają te same dane, a więc zależność przedstawiona na obu wykresach jest ta sama, to jednak wykresy te wyglądają różnie.

Przyjrzymy się obu wykresom oraz postarajmy się znaleźć jak najwięcej różnic i zastanówmy się, który sposób prezentacji jest lepszy.



Co różni te dwa wykresy?

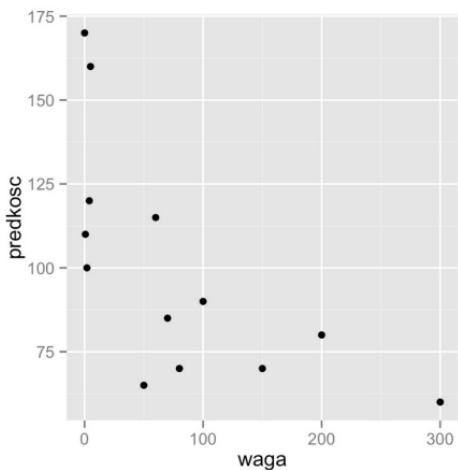
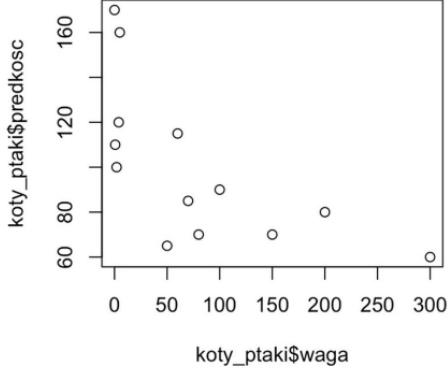
Oczywiście te wykresy różnią się wieloma elementami.

Różnice, które zauważyłem i uznałem za najistotniejsze to:

1. Na prawym wykresie widoczne są pionowe i poziome linie pomocnicze, ułatwiają one dokładniejsze odczytanie współrzędnych punktów niż w przypadku lewego wykresu.
2. Na prawym wykresie linie pomocnicze są białe na szarym tle, przez co nie dominują wykresu i są delikatnie przesunięte na drugi plan.
3. Jeżeli wykres umieszczamy na białej stronie internetowej to szare tło umożliwia szybką lokalizację wykresu.
4. Oba wykresy wypełniają obszar 10 x 10 cm, ale lewy wykres ma duże marginesy, przez co mniej miejsca pozostaje na prezentacje danych. Prawy wykres ma małe marginesy, przez co jest więcej miejsca na zaprezentowanie danych.
5. Etykiety na osiach na prawym wykresie są poziome, nie trzeba obracać głowy by je odczytać.
6. Etykiety na prawym wykresie są pomniejszone i wyszarzane, przesunięte na drugi plan. Wciąż są widoczne ale mniej dominują wykres, pozwalając przesunąć dane na pierwszy plan.
7. Nazwy na osiach są krótsze i czytelniejsze, nie ma członu koty_ptaki\$, który nie jest potrzebny.
8. Punkty zaznaczone są kropkami a nie pustymi okręgami, dzięki czemu zajmują mniej miejsca, więcej ich się zmieści na wykresie i łatwiej odczytać

współrzędne punktu (ale gdy punktów jest bardzo bardzo dużo to i tak zleją się w jeden kleks).

Używając pakietu `ggplot2` otrzymujemy przy fabrycznych ustawieniach czytelny wykres. Jeżeli chcemy go zmienić, możemy zmienić każdy jego element. Ale jeżeli nie chcemy, możemy skupić się na danych a nie upiększaniu wykresu.



Kiedy należy używać wykresu punktowego?

Wykres punktowy stosuje się najczęściej w sytuacji gdy dla pewnego zbioru obiektów (najczęściej opisanego wiersz po wierszu), chcemy przedstawić zależność

poniędzy dwiema zmiennymi liczbowymi/ilościowymi. Np. dla państw to może być zależność pomiędzy PKB i średnią długością życia, dla osób to może być wzrost i waga, dla samochodów to może być przebieg i cena.

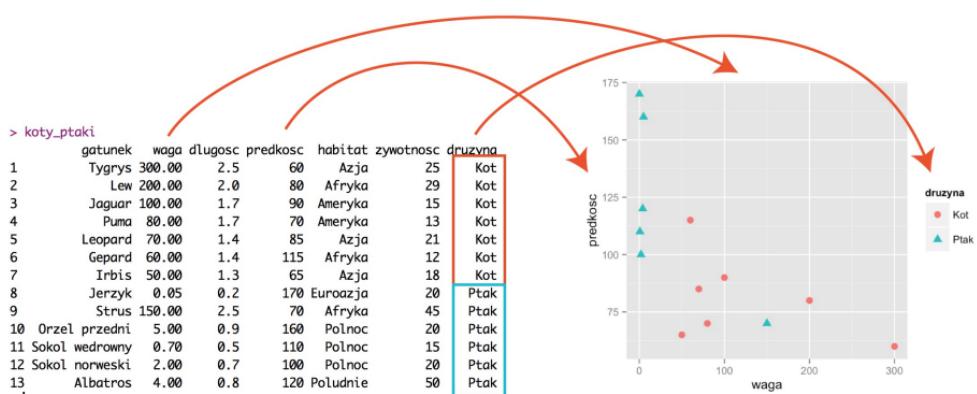
Dodatkowo:

1. Nie wiemy czy i jakiej zależności się spodziewać (czy i jak długość życia jest związana z PKB).
 2. Podejrzewamy, że dwie zmienne są zależne i chcemy przedstawić tę zależność (chcemy pokazać jak cena auta zmienia się z jego przebiegiem).
 3. Podejrzewamy, że w jednej zmiennej występują nietypowe obserwacje (np. bardzo duże) i chcemy zobaczyć jakim wartościom stwarzyszonej zmiennej one odpowiadają (np. czy osoby o bardzo wysokiej wadze to osoby bardzo wysokie czy nie).
-

Grupy na wykresie punktowym

Na wykresie punktowym można również zaznaczać grupy wierszy. W tym celu można przynależność do grupy oznaczyć kolorem, kształtem lub wielkością. Pozwala to dodatkowo na porównanie grup pod względem przedstawianych zmiennych (patrz rysunek poniżej). Taki wykres punktowy z grupami warto stosować gdy:

1. Nie wiemy czy grupy różnią się pod kątem danej pary zmiennych i chcemy to sprawdzić (np. czy koty różnią się ptaków jeżeli chodzi o wagę i prędkość).
2. Wiemy, że grupy się różnią i chcemy pokazać w jaki sposób (np. aby pokazać, że kobiety są średnio niższe i znacznie lżejsze niż mężczyźni).



Jak zrobić wykres punktowy?

Przyjrzyjmy się instrukcji, która tworzy wykres punktowy (na kolejnym slajdzie jest przedstawiony rozkład na części pierwsze).

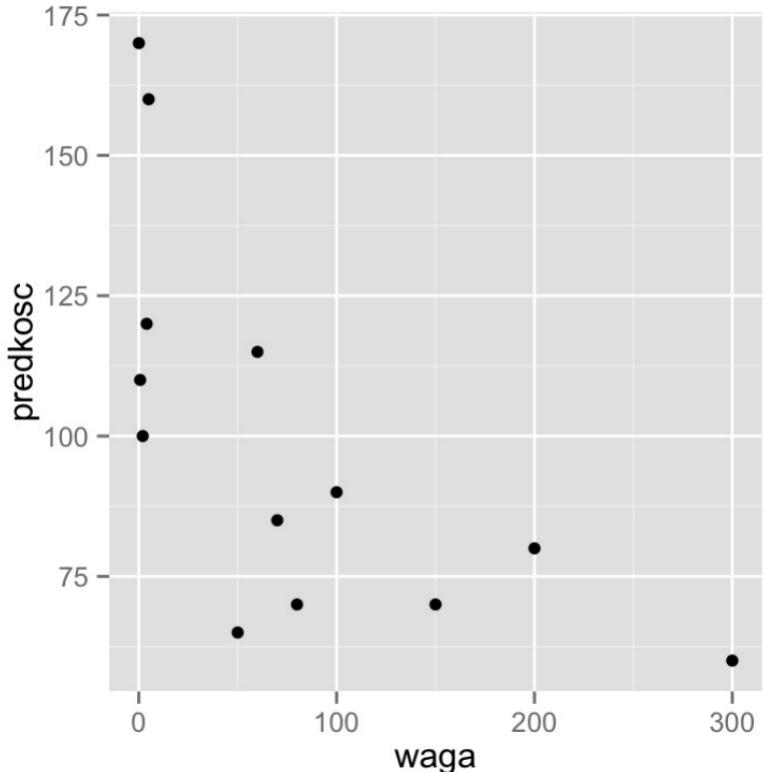
Przypomnijmy jak wyglądają dane. Oto pierwsze trzy wiersze ze zbioru `koty_ptaki`.

```
head(koty_ptaki, 3)
```

```
##      gatunek waga dlugosc predkosc habitat zywo
## 1    Tygrys   300      2.5       60     Azja
## 2      Lew    200      2.0       80   Afryka
## 3   Jaguar   100      1.7       90 Ameryka
```

Instrukcja rysująca wykres punktowy wygląda następująco.

```
ggplot(koty_ptaki, aes(x=waga, y=predkosc)) +
  geom_point()
```



Elementy wykresu punktowego

Przyjrzyjmy się instrukcji, która tworzy wykres punktowy.

```
ggplot(koty_ptaki, aes(x=waga, y=predkosc)) +  
  geom_point()
```

Trzy elementy są kluczowe w tej instrukcji.

- Funkcja `ggplot()` tworzy trzon wykresu. Opisuje ona jakie zmienne będą przedstawione na wykresie, ale nie określa w jaki sposób będą one przedstawione. Zazwyczaj przyjmuje ona dwa argumenty. Pierwszy to zbiór danych (w tym przypadku `koty_ptaki`). Drugi argument to lista par, określająca które zmienne i w jaki sposób mają być przedstawione na wykresie. Te pary nazywa się „mapowaniami”. Mapowania określają, które zmienne mają być przedstawione za pomocą jakich cech wykresu.
- Mapowania opisuje funkcja `aes()`. Argumenty tej funkcji to pary `cecha wykresu=zmienna`. Każda z tych par opisuje w jaki sposób określona zmienna ma być przedstawiona. W powyższym przypadku opis ten oznacza, że zmienna `waga` ma być przedstawiona przez współrzędną `x` a zmienna `predkosc` ma być przedstawiona przez współrzędną `y` wykresu.
- Trzon wykresu rozbudowuje się dodając kolejne

funkcje z użyciem operatora `+`. W tym przypadku funkcja `geom_point()` dodaje do wykresu warstwę przedstawiającą dane za pomocą punktów. Funkcje, o nazwach rozpoczynających się od `geom_` nazywamy geometriami, określając one w jaki sposób dane są prezentowane. Czy jako punkty (`geom_point()`), linie (`geom_line()`), obszary (`geom_area()`) czy jeszcze inaczej. W tym sezonie poznamy kilka geometrii, zbiór wszystkich dostępnych geometrii znaleźć można na stronie <http://docs.ggplot2.org>.

Dodajemy mapowania - kształt punktu - `shape`

Jak dotąd, nie widać jeszcze jak potężnym mechanizmem są mapowania. Zademonstrujmy to w kolejnym przykładzie, gdzie na wykresie zaznaczymy grupy punktów. Przyjmijmy, że chcemy przedstawić grupy opisane przez zmienną `druzyna`. Grupy punktów zaznaczymy różnymi kształtami, a więc cechą `shape`.

Zauważmy, że zmienne `waga`, `predkosc` i `druzyna` to kolumny występujące w zbiorze danych `koty_ptaki`. Zaś `x`, `y` i `shape` to cechy wykresu punktowego. Funkcja `aes()` wskazuje, które cechy wykresu przedstawiają które zmienne.

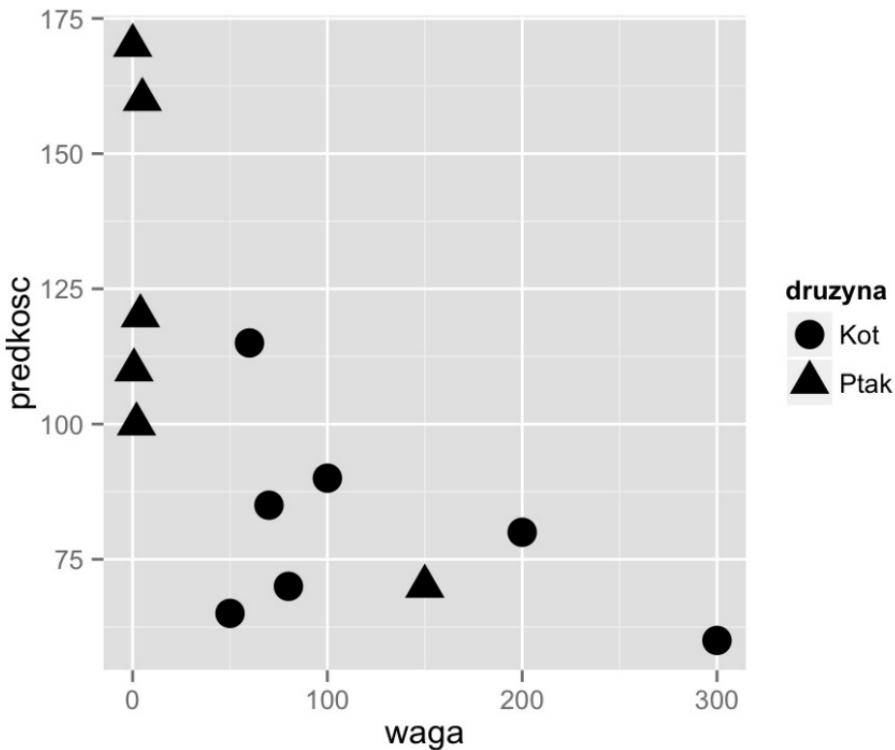
Przypomnijmy jakie zmienne są dostępne w zbiorze danych.

```
head(koty_ptaki, 2)
```

```
##   gatunek waga dlugosc predkosc habitat zywo
## 1  Tygrys  300      2.5       60     Azja
## 2      Lew   200      2.0       80   Afryka
```

Dodajemy do mapowań zmienną `druzyna`. Otrzymujemy wykres punktowy z grupami zaznaczonymi przez różne kształty punktów.

```
ggplot(koty_ptaki, aes(x=waga, y=predkosc, shape=drzyna))
  geom_point(size=5)
```



Jak budować wykresy z `ggplot2`?

Spójrzmy raz jeszcze na polecenie tworzące wykres i przeanalizujemy jak wygląda sposób myślenia przy tworzeniu wykresu.

```
ggplot(koty_ptaki, aes(x=waga, y=predkosc, shape=grupa)) +  
  geom_point(size=5)
```

W pierwszym kroku wybieramy zmienne, które chcemy przedstawić na wykresie. W naszym przypadku te zmienne to predkosc, waga oraz druzyne.

W kolejnym kroku wybieramy geometrię, która przedstawi wybrane dane. Kierujemy się tutaj rodzajem zmiennych. Przedstawiając zależność pomiędzy dwoma ilościowymi zmiennymi (waga i predkosc), naturalnym wyborem jest geometria punktowa. Zmienną jakościową (druzyne) potraktujemy jako zmienną grupującą.

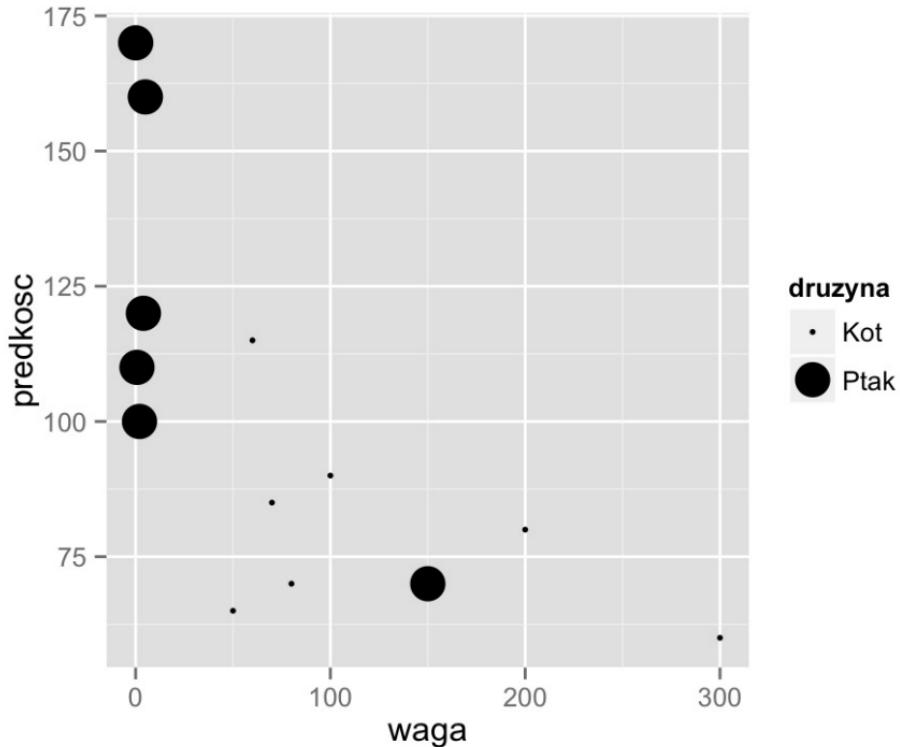
Kolejny krok to wybór cech geometrii, które mają przedstawić wybrane zmienne. W naszym przypadku zmienne predkosc i waga przedstawimy przez współrzędne x i y punktu. Zmienną punkt przedstawimy przez kształt punktu, a więc cechę shape.

Dodajemy mapowania - wielkość punktu - size

Jeżeli efekt oznaczania grup nam się nie podoba, możemy wykorzystać inną cechę do rozróżnienia grupy kotów od ptaków.

Na przykład wielkość punktu, czyli cechę size.

```
ggplot(koty_ptaki, aes(x=waga, y=predkosc, size=10)) +  
  geom_point()
```

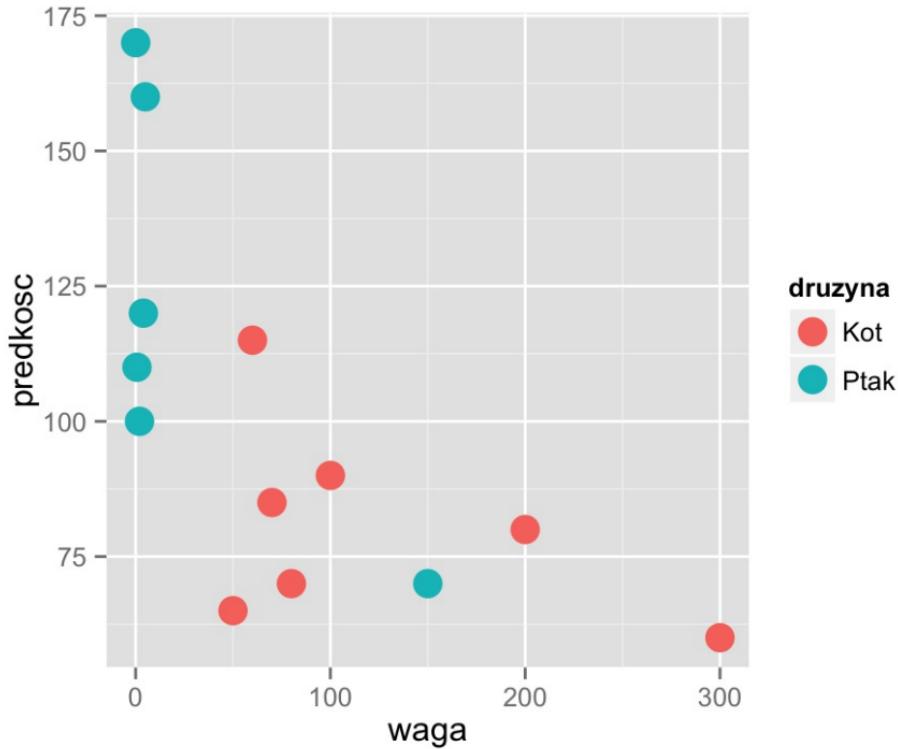


Dodajemy mapowania - kolor punktu - color

Jeżeli wynik wciąż jest niezadowalający, możemy próbować zmieniać kolejne cechy.

Na przykład kolor, czyli cechę `color` (dopuszczalna jest też brytyjska pisownia `colour`).

```
ggplot(koty_ptaki, aes(x=waga, y=predkosc, color=druzyна))  
  geom_point(size=5)
```



Wiele mapowań dla tej samej zmiennej

Zależność `cecha wykresu=zmienna` to relacja jeden do

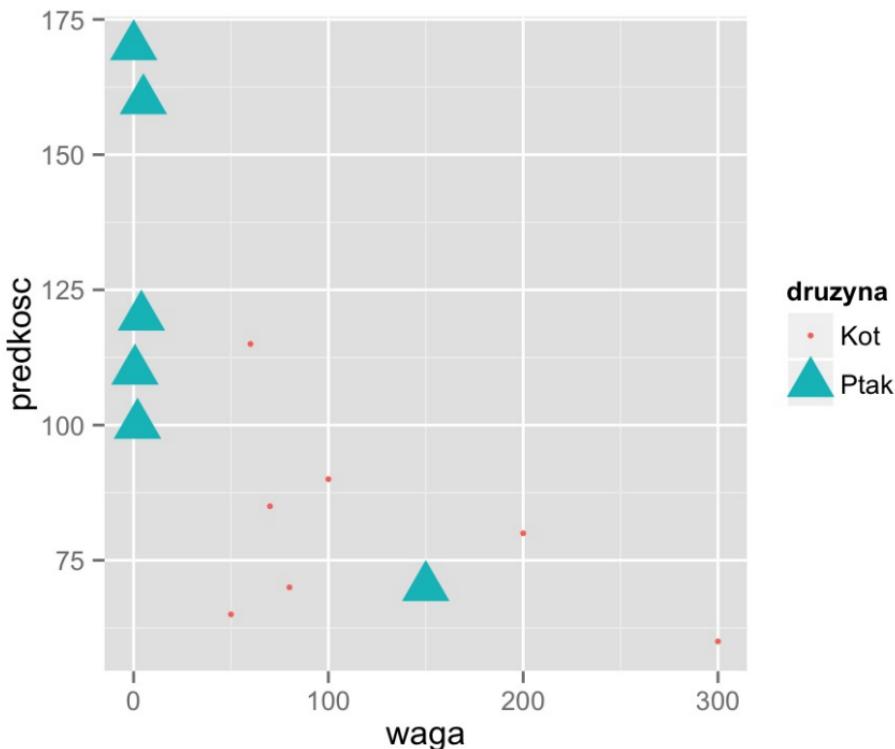
wielu. Oznacza to, że w opisie mapowań jedna cecha może być powiązana z tylko jedną zmienną. Ale jedna zmienna, może być powiązana z kilkoma cechami.

Czyli kolor punktów odpowiadając może tylko jednej kolumnie w zbiorze z danymi, kształt punktu też tylko jednej kolumnie w zbiorze z danymi i podobnie inne cechy. Ale jedna kolumna w zbiorze z danymi może być przedstawiana i przez kolor i przez kształt.

Na poniższym przykładzie jedna zmienna jest reprezentowana przez zarówno kolor, kształt jak i wielkość punktu.

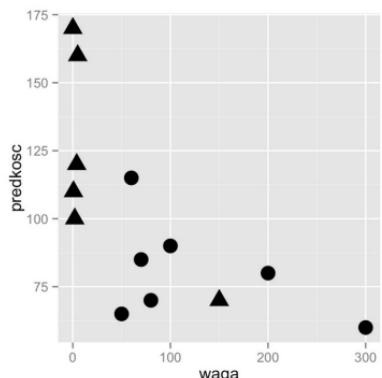
Zauważmy, że bez względu na to, jakie mapowania wybierzemy, legenda automatycznie dopasowuje się do treści wykresu.

```
ggplot(koty_ptaki, aes(x=waga, y=predkosc, color=druzyna, shape=druzyna, size=druzyna)) +  
  geom_point()
```

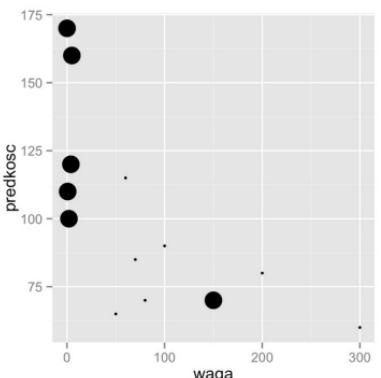


Na jaką cechę mapować?

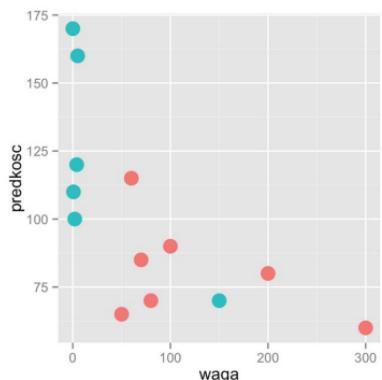
Zmieniając argumenty funkcji `aes()`, możemy wybierać sposób reprezentacji określonych zmiennych. Jak dotąd pokazaliśmy cztery z nich. Która jest najlepsza?



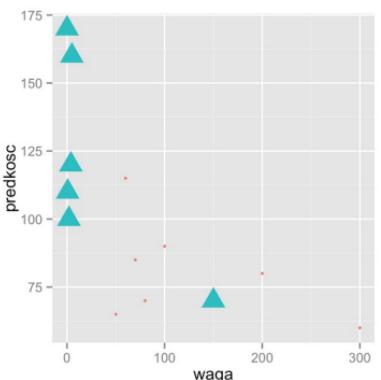
druzyna
● Kot
▲ Ptak



druzyna
● Kot
▲ Ptak



druzyna
● Kot
● Ptak



druzyna
● Kot
● Ptak

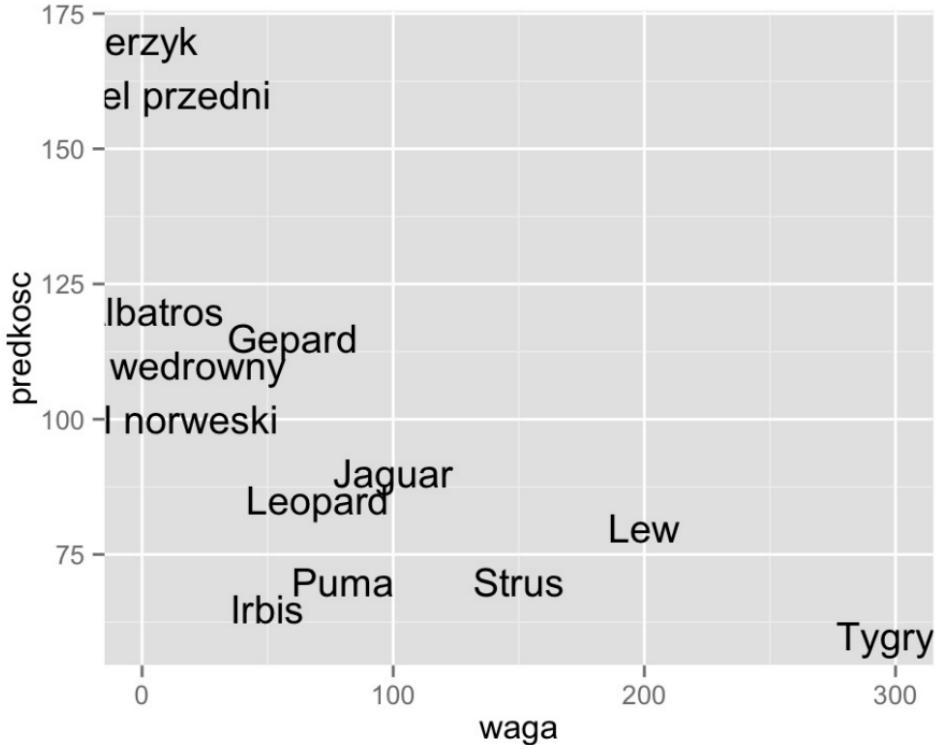
Geometria `geom_text()` - etykiety tekstowe na wykresie

Zobaczmy teraz, jak korzystać z innych geometrii. Za przykład posłuży nam wykres z napisami, który można wykonać używając geometrii `geom_text()`. Szczegółowy jej opis, wraz z przykładami użycia, znajduje się na stronie http://docs.ggplot2.org/current/geom_text.html. W

sekcji Aesthetics można listę cech, które można modyfikować. Aby skorzystać z tej geometrii należy określić przynajmniej trzy cechy: `x`, `y` i `label`.

A tak wygląda przykładowe wywołanie:

```
ggplot(koty_ptaki, aes(x= waga, y=predkosc, label= nome)) +  
  geom_text()
```



Geometria `geom_text()` -

pozycjonowanie napisu

Zauważmy, że niektóre napisy wychodzą poza wykres. Nie wygląda to najlepiej. Można ten problem naprawić na dwa sposoby. Jeden to rozszerzenie zakresu wartości na osi OX (dodanie marginesów), drugi to pozycjonowanie napisów względem ich lewego brzegu a nie środkowego (współrzędne `x` i `y` to domyślnie współrzędne środkowa napisu).

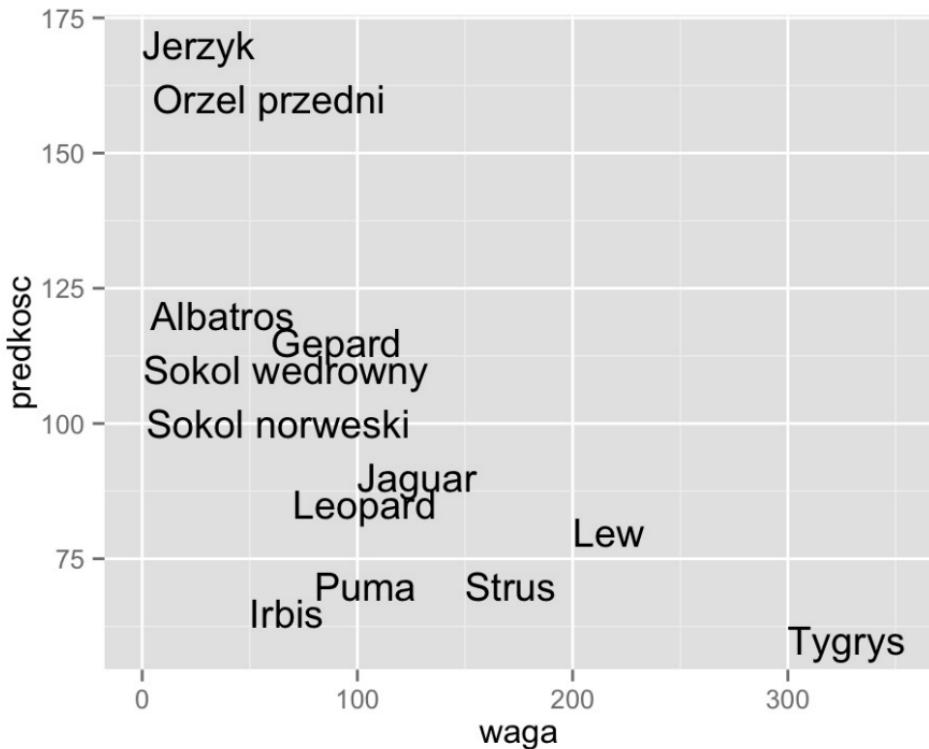
Spróbujmy tego drugiego rozwiązania. Czytając stronę z dokumentacją dla geometrii `geom_text()` można zauważyc, że za pozycjonowanie odpowiadają cechy `hjust` (w poziomie) i `vjust` (w pionie). Zobaczmy jak wyglądać będzie użycie tych argumentów.

```
ggplot(koty_ptaki, aes(x= waga, y=predkosc, la  
geom_text(hjust=0) + xlim(0,350)
```

Zauważmy, że argument `hjust` definiujemy wewnątrz funkcji `geom_text()` ale poza blokiem `aes()`. Nie mapuje on żadnej zmiennej na cechy wykresu ale przypisuje określonej ciesze stałą wartość (w tym przypadku 0).

Dodatkowo, aby na wykresie zmieściła się też etykieta Tygrys zmieniamy funkcję `xlim()` zakres prezentowany na osi OX dodając trochę pustego miejsca po prawej

stronie.

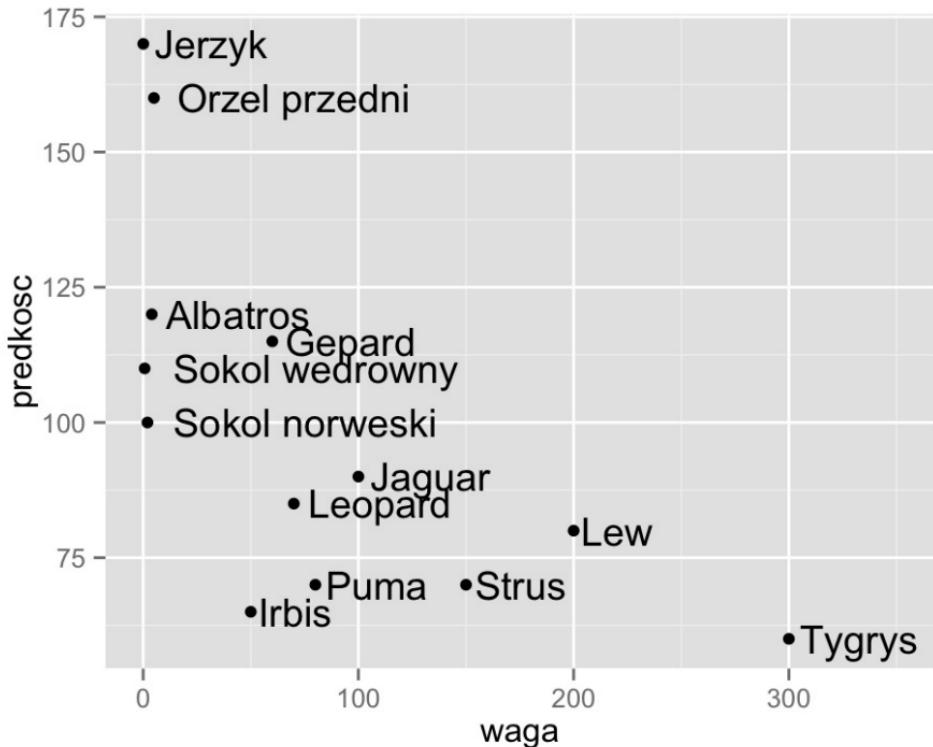


Składanie dwóch geometrii

Jedną z ciekawszych cech pakietu `ggplot2` jest to, że możemy dodawać kolejne warstwy z danymi dodając kolejne geometrie. Dzięki temu, można osiągnąć bardzo ciekawe efekty.

Na poniższym przykładzie do warstwy z napisami (`geom_text()`) dodajemy warstwę z punktami (`geom_point()`).

```
ggplot(koty_ptaki, aes(x= waga, y=predkosc, label= imie)) +  
  geom_text(hjust=-0.1) +  
  geom_point() + xlim(0,350)
```



Kiedy stosować etykiety na wykresie?

Wykres z etykietami stosować można w podobnych sytuacjach co wykres punktowy. Również w tym przypadku przedstawiamy zależność pomiędzy dwoma zmiennymi ilościowymi.

Stosowanie etykiet na wykresie ma wady i zalety. Do głównych zalet można zaliczyć:

1. Nazwy pozwalają zorientować się co reprezentuje dany punkt. Jest to przydatne, jeżeli wykres wykorzystujemy do identyfikacji *ciekawych* obserwacji, np. najbardziej odstających, najbardziej charakterystycznych.
2. Jeżeli etykiety są zbyt długie, nachodzą na siebie i przestają być czytelne, to można je skracać do kilku pierwszych liter np. funkcją `substr()`.
3. Jeżeli wykres przedstawia tylko kilka obserwacji, to nazwy lepiej *wypełniają* wykres, nie pozostawiając dużych pustych przestrzeni. A duże puste przestrzenie źle wyglądają.

A główne wady to:

1. Jeżeli napisów jest dużo, to na siebie nachodzą, przez co mogą być nieczytelne.
2. Jeżeli napisy są długie, to może nie być oczywiste, któremu punktowi odpowiadają.

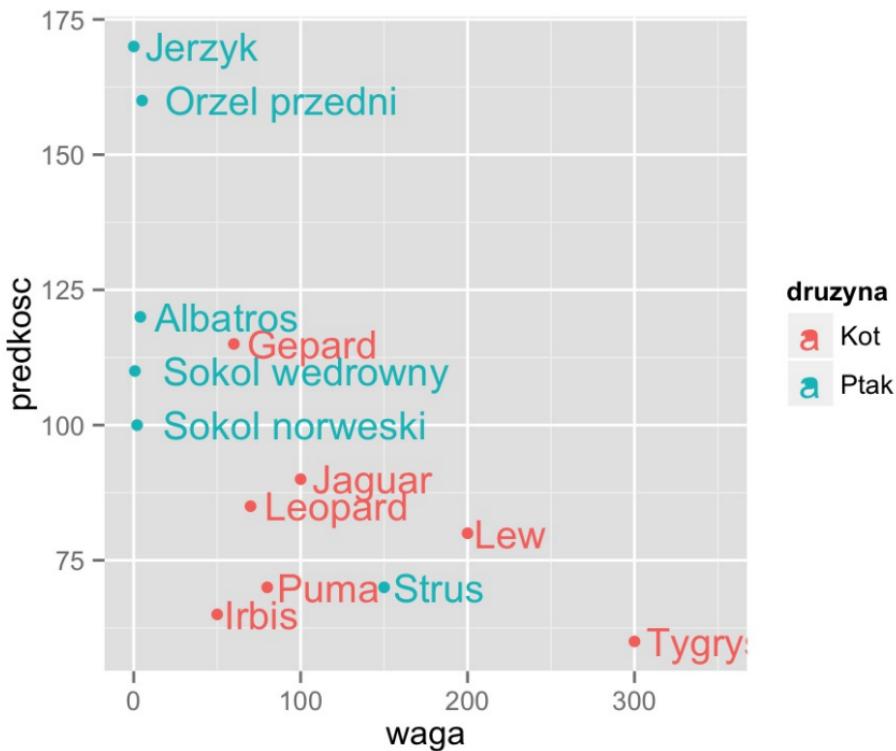
Podsumowując, geometria `geom_text()` jest świetnym sposobem by wyróżnić niewielką liczbę interesujących punktów.

Globalne mapowania

Zauważmy, że mapowania określone przez funkcję `aes()` wewnętrz funkcji `ggplot()` dotyczą wszystkich warstw.

Na poniższym przykładzie mapujemy zmienną `druzyna` na cechę `color` i to mapowanie dotyczy obu geometrii. I punkty i napisy będą miały kolory zależne od zmiennej `druzyna`.

```
ggplot(koty_ptaki, aes(x=waga, y=predkosc, label=druzyna)) +  
  geom_text(hjust=-0.1) +  
  geom_point() + xlim(0,350)
```

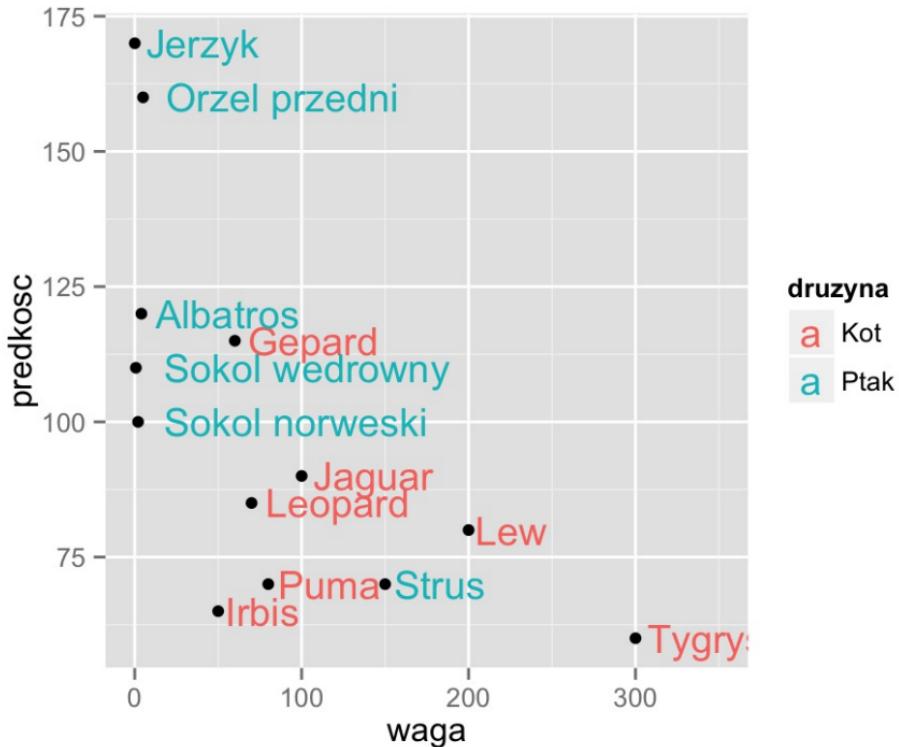


Lokalne mapowania

Jeżeli chcemy aby mapowanie dotyczyło tylko jednej warstwy, należy przenieść je do określonej geometrii.

Na poniższym przykładzie mapowanie zmiennej `druzyna` na `color` przypisane jest tylko do geometrii `geom_text()`. Wszystkie punkty mają ten sam kolor.

```
## kolor grupy jest określony tylko dla etykiety
ggplot(koty_ptaki, aes(x=waga, y=predkosc, label=imie)) +
  geom_text(hjust=-0.1, aes(color=druzyna)) +
  geom_point() + xlim(0,350)
```



Geometria geom_line() - wykres liniowy

Pierwsze zetknięcie z pakietem `ggplot` wymaga często

przestawienia się na nowy sposób myślenia o wykresach. Ale gdy już się w ten sposób wejdzie, to okazuje się, że poznawszy dwie geometrie, z kolejnych korzysta się już naprawdę prosto.

Przykładowo, geometria `geom_line()` służy do rysowania linii. Aby ją wykorzystać na wykresie, wystarczy ją dodać do wykresu operatorem `+`.

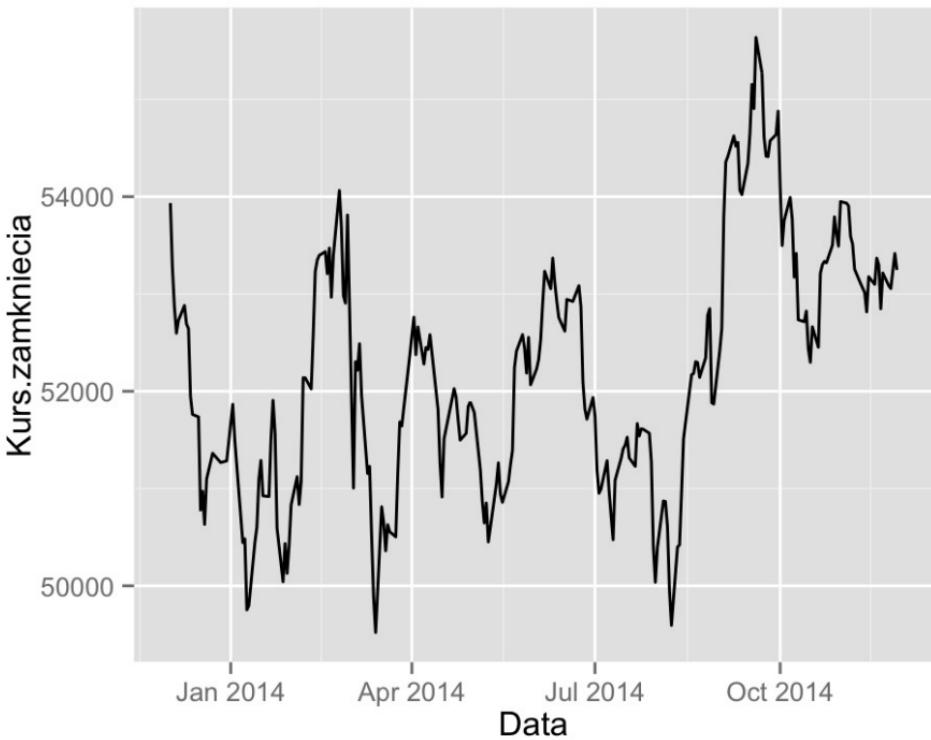
Przedstawimy tę geometrię na przykładzie danych o indeksie WIG. Poniżej dwie pierwsze linie z tego zbioru danych.

```
head(WIG, 2)
```

```
##           Data Nazwa Kurs.otwarcia Kurs.maksymalny
## 1 2013-12-02   WIG        54627.26            5479.0
## 2 2013-12-03   WIG        54025.72            5402.0
##   Kurs.zamkniecia Zmiana Wartosc.obrotu.w.tygodniu
## 1           53934.52   -1.41             6402.0
## 2           53276.83   -1.22             914.0
```

Współrzędną `x` przedstawimy datę notowania, a współrzędną `y` kurs zamknięcia. Przy okazji zauważmy, że jeżeli na cechę `x` będziemy mapować zmienną, która jest datą, to oś wykresu zostanie tak dobrana by poprawnie odwzorować wskazaną zmienną.

```
ggplot(WIG, aes(x=Data, y=Kurs.zamkniecia)) +
  geom_line()
```



Kiedy stosować linie na wykresie?

Z punktu widzenia programu R, to równie dobrze możemy te same dane (dwie zmienne liczbowe) przedstawić używając punktów jak i używając linii.

Jest jednak duża różnica w sposobie w jaki będziemy takie wykresy postrzegać.

Punkty nie mają kierunku, więc chmura punktów na wykresie pozwoli ocenić czy te punktu układają się wzdłuż jakiejś krzywej (jest zależność pomiędzy zmiennymi), czy tworzą kilka grup (są skupiska), czy występują wartości odstające.

Linie na wykresie mają kierunek. Patrząc na linie nasze oko ocenia ich względne długości oraz względne kierunki, szacując kąty pomiędzy liniami.

Z tego powodu, wykresów liniowych najlepiej używać gdy:

1. Chcemy przedstawić trend, pokazać jak pewna wartość zmienia się w czasie. Standardowo na osi poziomej x przedstawia się czas a na osi pionowej y wartość.
 2. Chcemy przedstawić tempo zmiany trendu, czy tempo wzrostu rośnie czy maleje (czy kąty są coraz większe czy nie)
 3. Chcemy pokazać zmienność pomiędzy sąsiednimi pomiarami. Ponieważ sąsiednie pomiary połączone są linią, będzie widać czy ta linia gwałtownie czy spokojnie zmienia kierunki.
-

Geometria `geom_ribbon()` - wykres

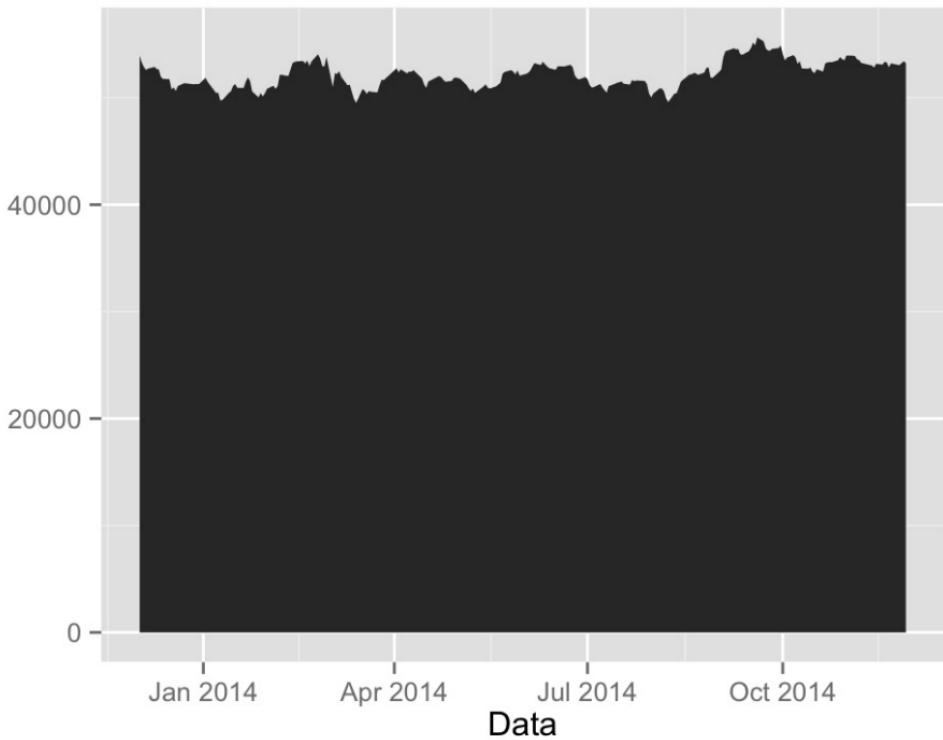
wstęga

Ostatnim przykładem w tym odcinku będzie geometria `geom_ribbon()` do rysowania wstęg.

Jest to o tyle ciekawy przykład, że wymagać będzie wskazania innej cechy niż `y`. Tak się bowiem składa, że wstęgi są opisane przedziałem, czyli cechami `ymin` i `ymax` (co można wyczytać z dokumentacji dostępnej na stronie <http://docs.ggplot2.org/>). Każdą z tych cech możemy połączyć ze zmienią lub przypisać do nich stałą wartość.

Na poniższym przykładzie rysujemy wstęgę rozpinającą się od zera (stała wartość `ymin=0`) do wartości indeksu WIG (`ymax=Kurs.zamkniecia`).

```
ggplot(WIG, aes(x=Data, ymin=0, ymax=Kurs.zamkniecia)  
      geom_ribbon())
```



Gdzie szukać dalszych informacji

Opanowaliśmy podstawy tworzenia wykresów z użyciem pakietu `ggplot2`. W kolejnych odcinkach przedstawimy więcej informacji, ale można ich też poszukać w poniższych źródłach.

- Bardzo interesująca prezentacja o projektowaniu wykresów, o nazwie *Figure Design* jest dostępna

pod adresem

<http://www.bioinformatics.babraham.ac.uk/training/F>

- Więcej informacji o tym jak budować wykresy w pakiecie `ggplot2` znaleźć można w rozdziale 4 książki „*Przewodnik po pakiecie R*”. Więcej informacji o tej książce <http://www.biecek.pl/R/>
- Wiele przydatnych wskazówek można również znaleźć w internetowej książce „*Cookbook for R*”
<http://www.cookbook-r.com/Graphs/>
- Niezła ściągawka jak korzystać z pakietu „`ggplot2`”
<http://www.rstudio.com/wp-content/uploads/2015/03/ggplot2-cheatsheet.pdf>
- Bardzo rozbudowana dokumentacja dla pakietu `ggplot2` dostępna jest na stronie „`ggplot2`”
<http://docs.ggplot2.org/current/index.html>
- Nic jednak nie zastąpi własnych eksperymentów z tworzeniem wykresów. Gdyby coś nie zadziałało jak trzeba, można szukać pomocy na forum do tego kursu lub w internecie, np. na forum
<http://stats.stackexchange.com/>.

- Przedstaw graficznie za pomocą wykresu punktowego zależność pomiędzy żywotnością (kolumna `zywotnosc`) a wagą zwierzęcia (kolumna `waga`).
- Zaznacz kolorem lub kształtem punktu informację czy przedstawiany jest ptak czy kot. Czy są różnice pomiędzy żywotnością a wagą dla kotów i ptaków?
- Użyj etykiet by odczytać który ptak i który kot żyją najdłużej.
- Używając geometrii wstęga (`geom_ribbon`) przedstaw kurs minimalny i maksymalny każdego dnia na podstawie danych ze zbioru `WIG`.

Przykładowe odpowiedzi znajdują się na stronie
http://pogromcydanych.icm.edu.pl/materials/2_modelowan

Wykresy, które prezentują podsumowania danych

Przemysław Biecek @ Uniwersytet Warszawski

*sezon 2 / odcinek 3
pogRomcy danych*

- O czym jest ten odcinek?
- Przygotowanie danych
- Geometria `geom_point()` - jednak nie do wszystkiego
- Geometria `geom_smooth()` - wykresy trendu
- Geometria `geom_smooth()` - wykresy trendu, błąd standardowy dla średniej
- Geometria `geom_smooth()` - składanie warstw
- Geometria `geom_smooth()` - składanie warstw
- Geometria `geom_smooth()` - wykresy trendu
- Geometria `geom_smooth()` - wykresy trendu
- Geometria `geom_smooth()` - wykresy trendu
- Zmienne jakościowe
- Geometria `geom_point()` ponownie nie daje rady
- Geometria `geom_boxplot()` - wykresy ramka - wąsy

- Geometria `geom_boxplot()` - wykresy ramka - wąsy
- Geometria `geom_boxplot()` - wykresy ramka - wąsy
- Geometria `geom_hist()` - histogram
- Geometria `geom_hist()` - histogram
- Geometria `geom_bar()` - wykresy słupkowe
- Geometria `geom_bar()` - wykresy słupkowe
- Geometria `geom_bar()` - wykresy słupkowe, wykresy warunkowe
- A jak policzyć dokładnie liczebności grup?
- Gdzie szukać dalej
- Zadanie, sezon 2, odcinek 8

O czym jest ten odcinek?

W poprzednim odcinku na wykresach pokazywaliśmy każdą obserwację / każdy wiersz ze zbioru danych. Informacje o kolejnych wierszach były przedstawiane za pomocą punktów, linii lub napisów.

Jednak, gdy dane są duże i zawierają setki lub setki tysięcy obserwacji, nie sposób przedstawić każdej z nich osobno. Gdy elementów na wykresie jest zbyt wiele, to wykres zamienia się w wielką płatanię punktów i krzywych. W takim przypadku często lepszym rozwiązaniem jest przedstawienie podsumowań / statystyk zamiast surowych danych.

W tym odcinku dowiemy się jak graficznie przedstawiać podsumowania danych.

- Jak tworzyć wykresy trendu, przedstawiające wygładzone średnie?
- Jak tworzyć wykresy ramka wąsy / wykresy pudełkowe, przedstawiające pięć liczb Tukeya?
- Jak tworzyć histogram, przedstawiający rozkład wartości?
- Jak tworzyć wykresy słupkowe, przedstawiające liczebności obserwacji?

W tym odcinku będziemy pracować ze zbiorem danych auta2012.

Przygotowanie danych

Zbiór danych o cenach ofertowych używanych aut wykorzystywaliśmy już w pierwszym sezonie. Te dane są szczegółowo opisane w pliku

http://pogromcydanych.icm.edu.pl/materials/1_przetwarzaj

Na potrzeby tego odcinka i prezentowanych tutaj przykładów, do pracy wybierzemy fragment zbioru danych o cenach ofertowych aut. Zawężimy tutaj naszą uwagę do obserwacji dotyczących Skody Octavii. Ze zbioru kolumn

wybierzemy pięć z informacją o roku produkcji, cenie i rodzaju paliwa.

Pełne dane są dostępne w pakiecie `PogromcyDanych`. Będziemy też korzystać z pakietu `dplyr`, który omawialiśmy w pierwszym sezonie. Funkcją `filter()` wybieramy interesujące nas wiersze, a funkcją `select()` interesujące nas kolumny.

```
library(PogromcyDanych)
skody <- auta2012 %>%
  filter(Marka == 'Skoda', Model ==
  select(Marka, Model, Rok.produkcji, Cena.w.PLN)
head(skody)

##      Marka    Model Rok.produkcji Cena.w.PLN
## 1 Skoda Octavia      2010     52750 olej
## 2 Skoda Octavia      2003     16800 olej
## 3 Skoda Octavia      2007     38900 olej
## 4 Skoda Octavia      2009     39999 olej
## 5 Skoda Octavia      2008     28500 olej
## 6 Skoda Octavia      2008     36800 olej
```

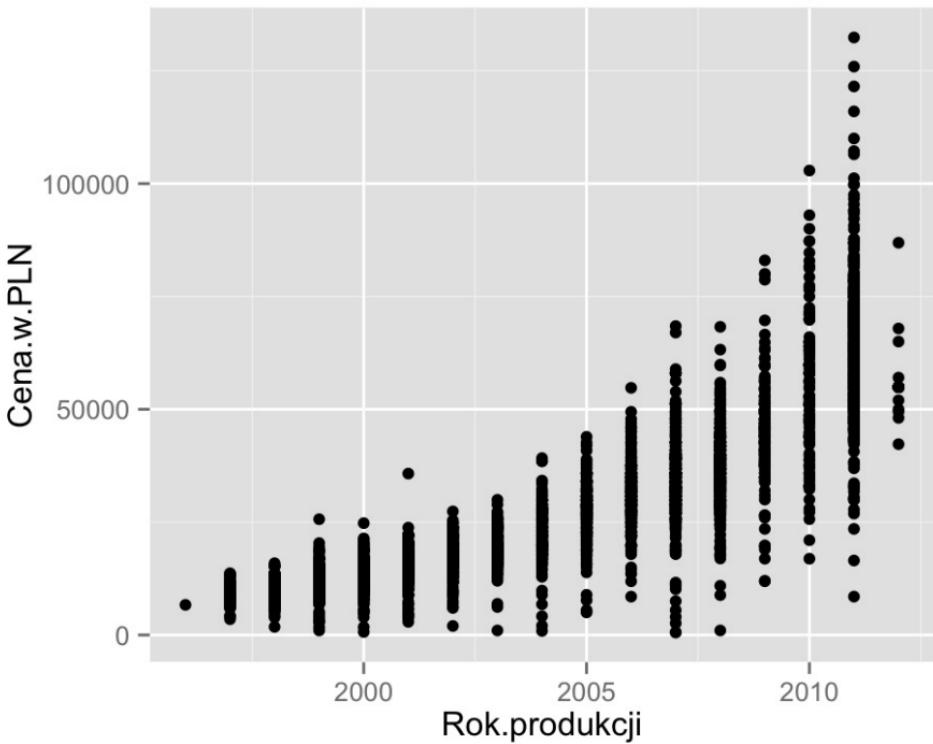
Geometria `geom_point()` - jednak nie do wszystkiego

Gdy chcemy przedstawić zależność, pomiędzy parą zmiennych, często wykorzystywaną jest geometria punktowa `geom_point()`. Jednak, gdy punktów jest wiele,

taka prezentacja może być mało czytelna. W takich przypadkach warto rozważyć geometrię `geom_smooth()` prezentującą wygładzoną lokalną średnią.

Porównajmy zależność pomiędzy ceną samochodu (na rok 2012) a rokiem produkcji. Jeżeli przedstawić te dane za pomocą punktów, to zauważymy, że jakaś zależność jest, ale trudno nam będzie ją dokładnie odczytać. Punktów jest tak wiele, że nakładają się na siebie i trudno się zorientować jakie ceny są typowe dla różnych roczników.

```
ggplot(skody, aes(x=Rok.produkcji, y = Cena.w.)  
  geom_point())
```



Geometria `geom_smooth()` - wykresy trendu

Zastąpienie `geom_point()` przez `geom_smooth()` powoduje, że rysowana jest wygładzona lokalna średnia zamiast punktów.

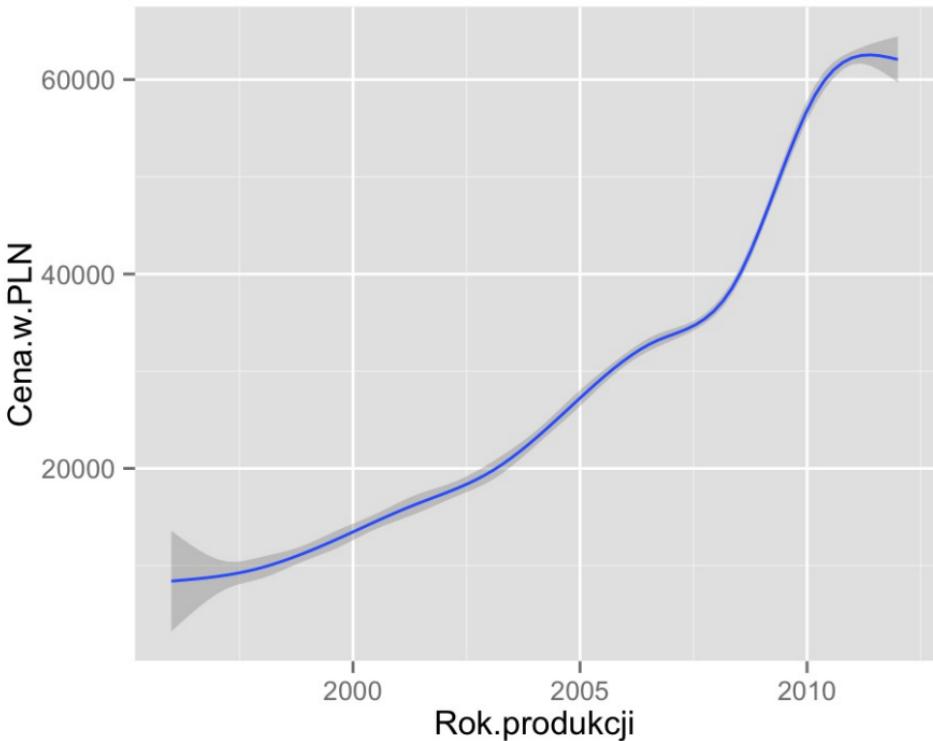
Jak dokładnie wyznaczana jest ta wygładzana średnia?

Jeżeli danych jest mniej niż 1000 obserwacji, to do wygładzania wykorzystywana jest funkcja `loess()` (Local Polynomial Regression Fitting), a jeżeli więcej niż 1000 obserwacji to funkcja `gam()` (Generalized Additive Models). Szczegóły każdej z tych metod można odczytać z plików pomocy dla funkcji `loess()` i `gam()`.

Geometria `geom_smooth()` rysuje również informacje o błędzie standardowym oceny średniej. Szary przedział wokół średniej przedstawia precyze oszacowania średniej. W miejscach gdzie ta średnia wyznaczana jest na małym zbiorze obserwacji błąd standardowy jest większy, dokładność oceny średniej jest mniejsza.

Jak więc wygląda zależność pomiędzy średnią ceną a rokiem produkcji (ceny na rok 2012)? Nowsze auta są oczywiście droższe.

```
ggplot(skody, aes(x=Rok.produkcji, y = Cena.w.)  
  geom_smooth())
```



Geometria `geom_smooth()` - wykresy trendu, błąd standardowy dla średniej

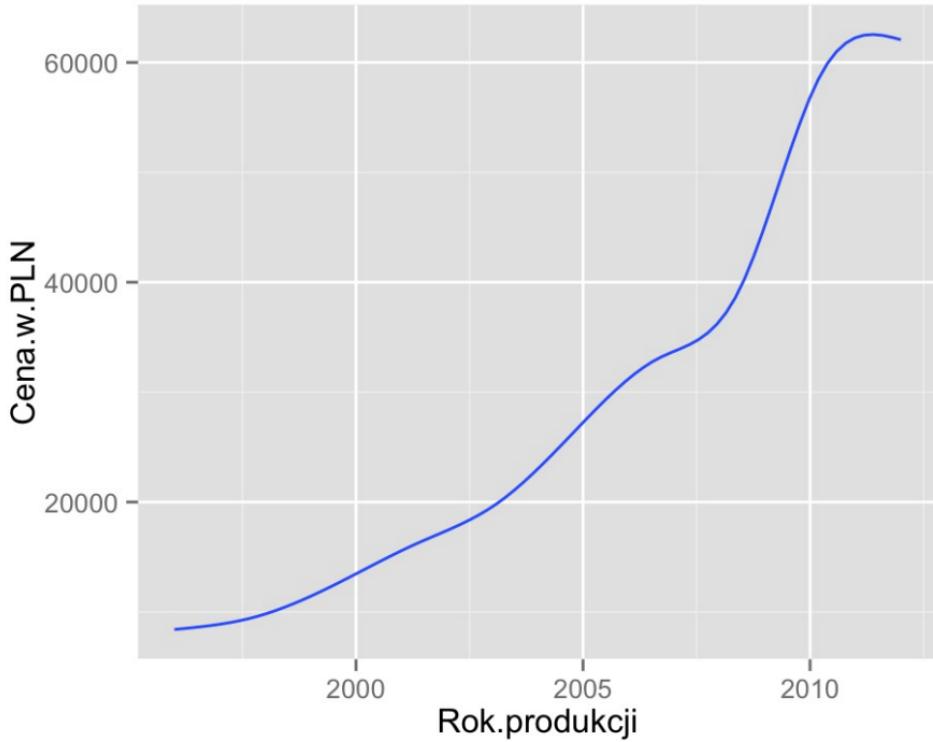
Parametry geometrii `geom_smooth()` przedstawione są na stronie http://docs.ggplot2.org/0.9.3.1/geom_smooth.html.

Zmieniając je możemy uzyskać inne stopnie wygładzania (argument `span` dla `loess`), różną grubość linii, różne

modele stosowane do wygładzenia itp.

Jednym z parametrów jest `se`, którym możemy włączać lub wyłączać rysowanie błędów standardowych. Często interesuje nas wyłącznie trend. Dodatkowe przedziały jedynie zaśmiecąją wykres. Można je wyłączyć dopisując argument `se=FALSE`.

```
ggplot(skody, aes(x=Rok.produkcji, y = Cena.w.)  
  geom_smooth(se=FALSE)
```

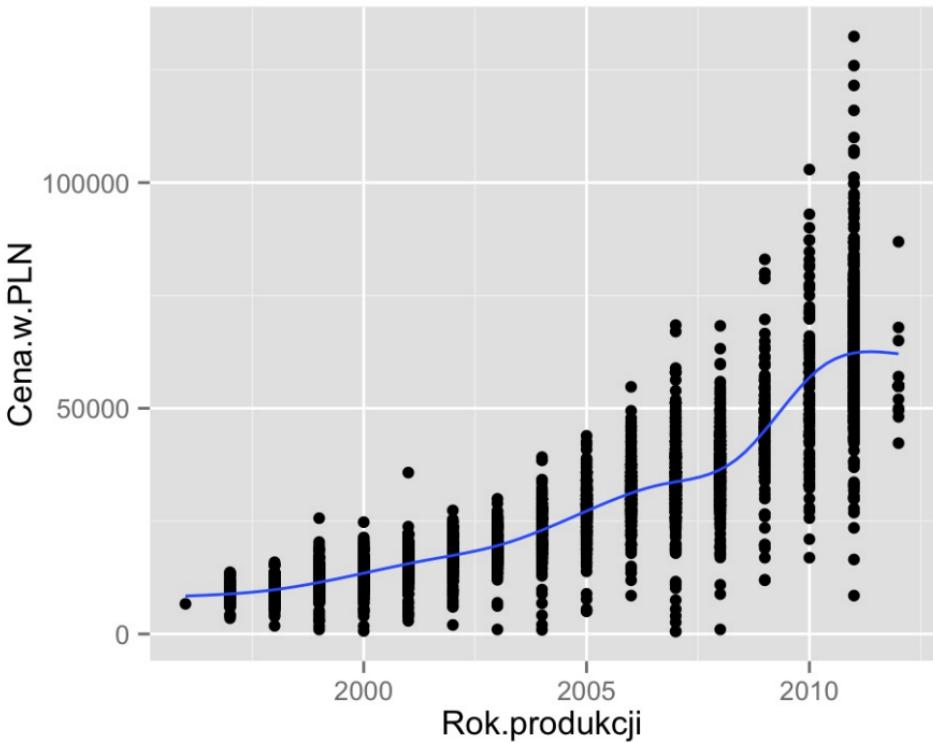


Geometria `geom_smooth()` - składanie warstw

W pakiecie `ggplot2` możemy składać na jednym wykresie warstwy z różnymi geometriami.

Przykładowo, możemy nałożyć na siebie warstwę pokazującą trendy i warstwę pokazującą pojedyncze punkty. W wielu sytuacjach to bardzo dobry sposób prezentacji. Pokazuje zarówno każdą z obserwacji na drugim planie jak i główny trend na planie pierwszym.

```
ggplot(skody, aes(x=Rok.produkcji, y=Cena.w.PL))  
  geom_point() +  
  geom_smooth(se=FALSE)
```



Geometria `geom_smooth()` - składanie warstw

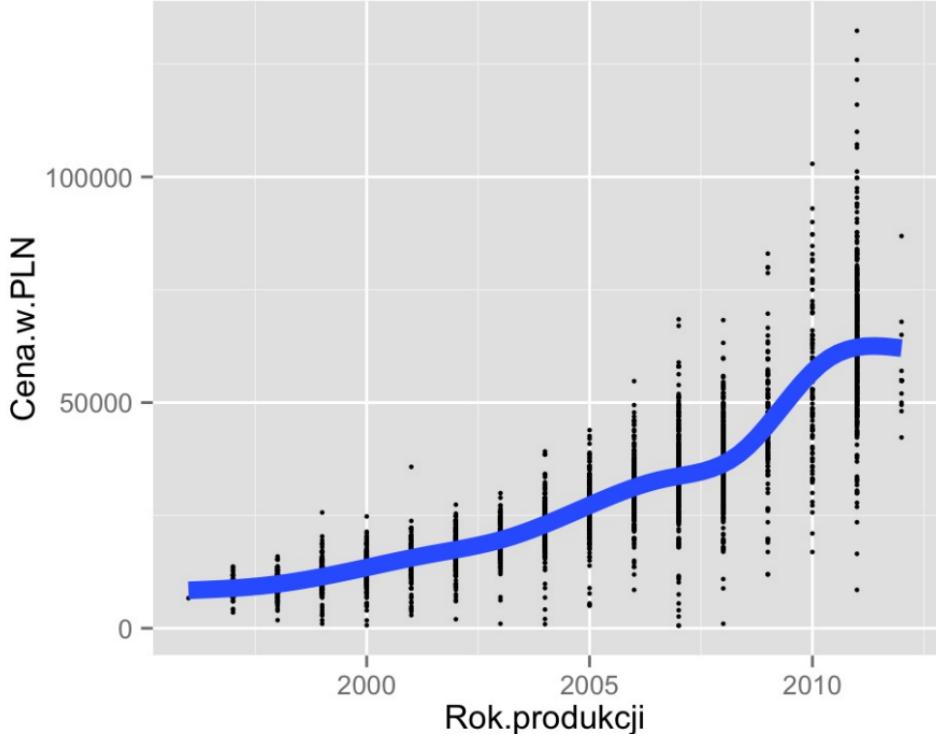
Mając na wykresie kilka warstw, często chcemy pewne uczynić bardziej wyraźnymi a inne mniej. Dwie cechy najsilniej podkreślające widoczność to wielkość i kolor.

Wykorzystamy cechę `size` aby podkreślić trend a

zmniejszyć widoczność warstwy z punktami. Pomniejszymy punkty i powiększymy linię trendu by jeszcze silniej zaznaczyć, która warstwa powinna być bardziej widoczna.

Dla każdej warstwy osobno określamy wielkość linii i punktów argumentem `size`.

```
ggplot(skody, aes(x=Rok.produkcji, y=Cena.w.PLN)
       geom_point(size=0.8) +
       geom_smooth(se=FALSE, size=3)
```



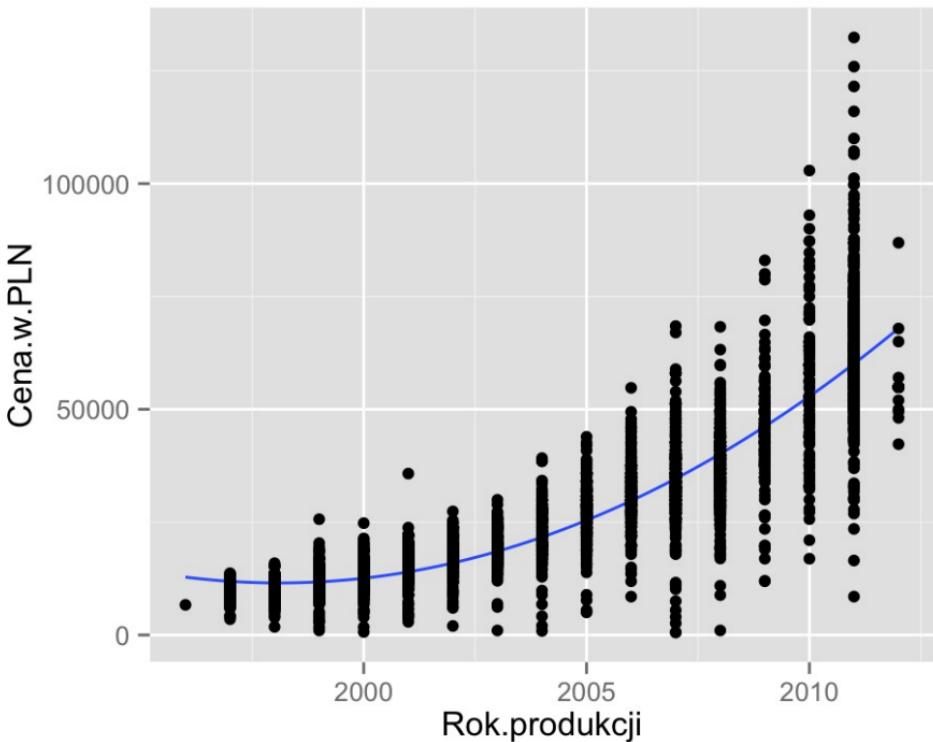
Geometria `geom_smooth()` - wykresy trendu

Inną użyteczną opcją geometrii `geom_smooth()` jest możliwość zmiany funkcji wygładzającej. Do wyboru jest między innymi regresja liniowa (parametr `method="lm"`), regresja logistyczna i modele uogólnione (parametr `method="glm"`), regresja odporna (parametr `method="rlm"`) oraz wymieniane już funkcje lokalnie wygładzające (parametr `method="gam"` lub `method="loess"`).

Dla modeli regresyjnych możemy dopasować nie tylko zależność liniową, ale również zależności wielomianowe, wykładnicze czy potęgowe. Opis struktury modelu można wyrazić parametrem `formula`, używając typowego R-owego formatu opisu formuł.

W poniższym przykładzie dodajemy trend kwadratowy do opisu zależności pomiędzy zmiennymi. Argumentem `formula=y~poly(x, 2)` określamy trend kwadratowy. Argumentem `method="lm"` określamy sposób liczenia współczynników dla tego trendu.

```
ggplot(skody, aes(x=Rok.produkcji, y=Cena.w.PL)  
  geom_smooth(se=FALSE, method="lm", formula=y~  
  geom_point())
```



Geometria `geom_smooth()` - wykresy trendu

Mówiąc o modelach wykładniczych, logarytmicznych lub potęgowych, warto rozważyć jeszcze inną ciekawą opcję, którą oferuje pakiet `ggplot2` - mianowicie transformacje osi.

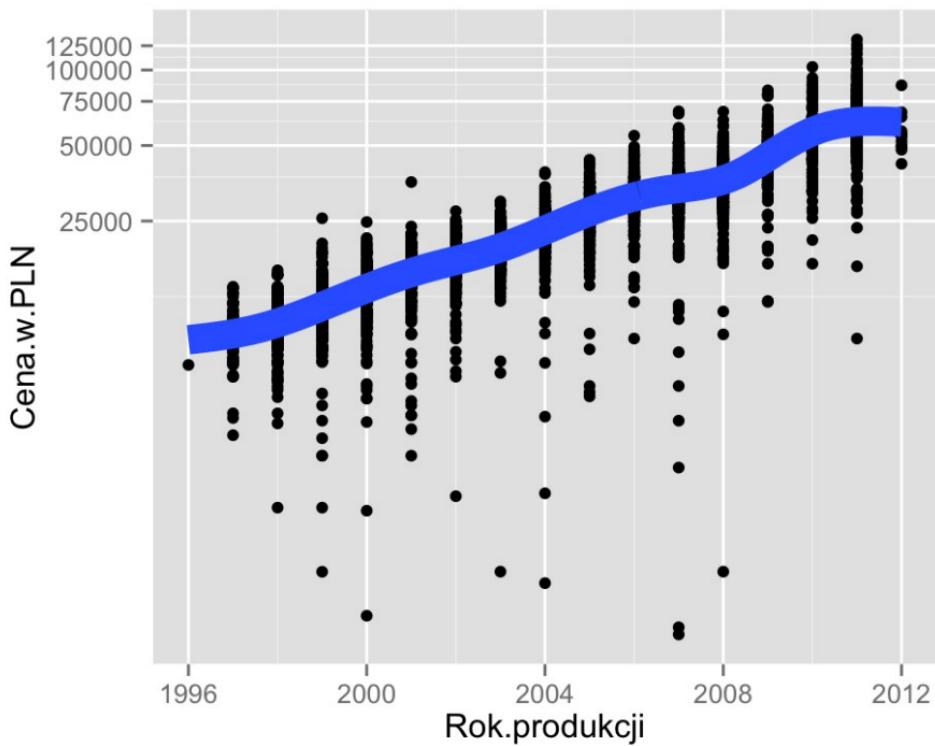
Jeżeli podejrzewamy, że zmiana ceny z roku na rok ma charakter spadku o x procent a nie o x złotówek (czyli, używając specjalistycznej terminologii, charakter multiplikatywny a nie addytywny) warto zobaczyć ten sam wykres z osią OY w skali wykładniczej.

Funkcje do zarządzania i modyfikacji osi mają prefix `coord_`. Wpisawszy go w RStudio, możemy nacisnąć klawisz TAB aby wyświetliła się lista wszystkich funkcji rozpoczynających się od tego słowa.

Funkcją do transformacji osi jest `coord_trans()`. Możemy jej użyć aby określić czy i jak poszczególne osie mają być transformowane.

Na poniższym przykładzie używamy funkcji `coord_trans()` aby zastosować transformację logarytmiczną dla osi OY.

```
ggplot(skody, aes(x=Rok.produkcji, y = Cena.w.) +  
  geom_point() +  
  geom_smooth(se=FALSE, size=5) +  
  coord_trans(y = "log10")
```



Na stronie

http://docs.ggplot2.org/current/coord_trans.html

przedstawiane są najpopularniejsze transformacje, takie jak pierwiastkowa, potęgowa czy logarytmiczna.

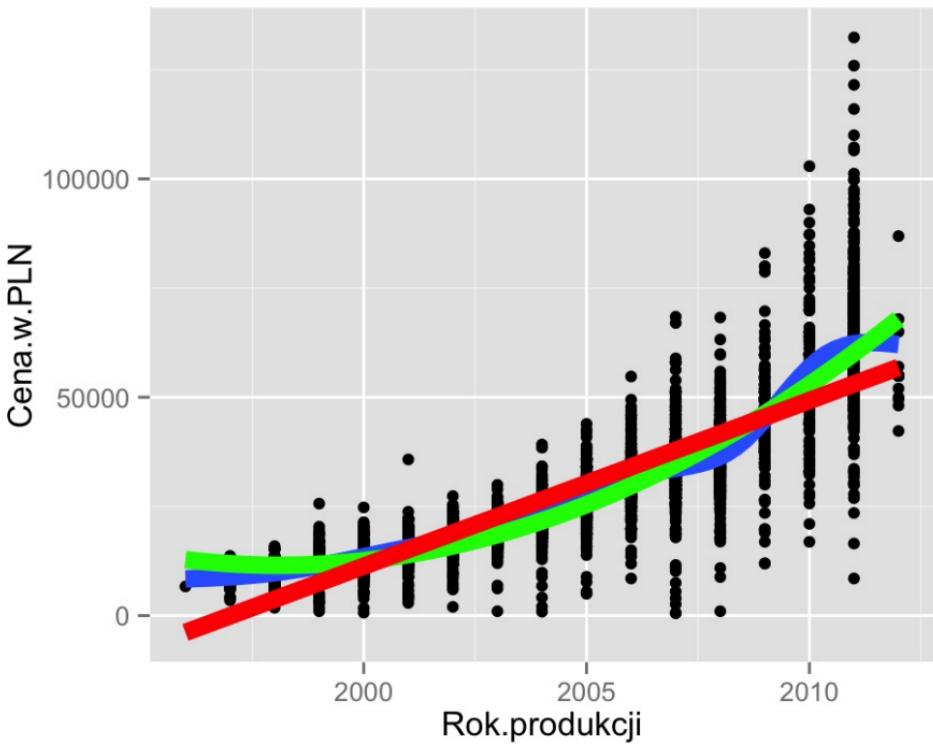
Geometria `geom_smooth()` - wykresy trendu

Bardzo często analizując trend, tak na prawdę nie wiemy z jakim trendem mamy do czynienia. Na wagę złota jest wtedy możliwość porównania różnych trendów na tle danych.

Pakiet `ggplot2` pozwala na tworzenie wykresu przez nakładanie na siebie warstw. Jest to bardzo użyteczne do przedstawienia obok siebie różnych sposobów wyznaczania trendu.

Wykres, który wykonaliśmy przedstawia zależność pomiędzy ceną a rokiem produkcji. Nałożmy na niego informację o trendzie liniowym, trendzie liczonym algorytmem GAM i trendzie kwadratowym. Każdy z tych trendów narysuje funkcja `geom_smooth()` z określonymi dodatkowymi argumentami.

```
ggplot(skody, aes(x=Rok.produkcji, y = Cena.w.)  
  geom_point() +  
  geom_smooth(se=FALSE, size=3) +  
  geom_smooth(method="lm", formula=y~poly(x, 2),  
  geom_smooth(method="lm", se=FALSE, size=3, co
```



Zmienne jakościowe

Geometria `geom_smooth()` była użyteczna do przedstawiania podsumowań dwóch zmiennych ilościowych.

Przyjrzymy się teraz geometriom do prezentowania zależności pomiędzy zmiennymi jakościowymi. Wpierw jednak przygotujmy dane w taki sposób, by mieć ciekawe

zmienne do przedstawiania.

Będziemy wciąż korzystać ze zbioru danych z cenami pięcioletnich samochodów marki Skoda, a za zmienne jakościowe wykorzystamy Model oraz Rodzaj.paliwa.

Ponownie, funkcjami filter() i select() wybierzymy interesujące nas wiersze i kolumny.

Poprzednio pracowaliśmy z samochodami w modelu Octavia, teraz będziemy pracować z autami pięcioletnimi. Stąd drobna zmiana w wywołaniu funkcji filter().

```
skody <- auta2012 %>%
  filter(Marka == 'Skoda', Rok.produkcji == 2007) %>%
  select(Marka, Model, Rok.produkcji, Cena.w.PLN)
```

##	Marka	Model	Rok.produkcji	Cena.w.PLN
## 1	Skoda	Fabia	2007	21850
## 2	Skoda	Octavia	2007	38900
## 3	Skoda	Octavia	2007	32500
## 4	Skoda	Octavia	2007	32500
## 5	Skoda	Superb	2007	36600
## 6	Skoda	Fabia	2007	14500

Geometria geom_point() ponownie nie daje rady

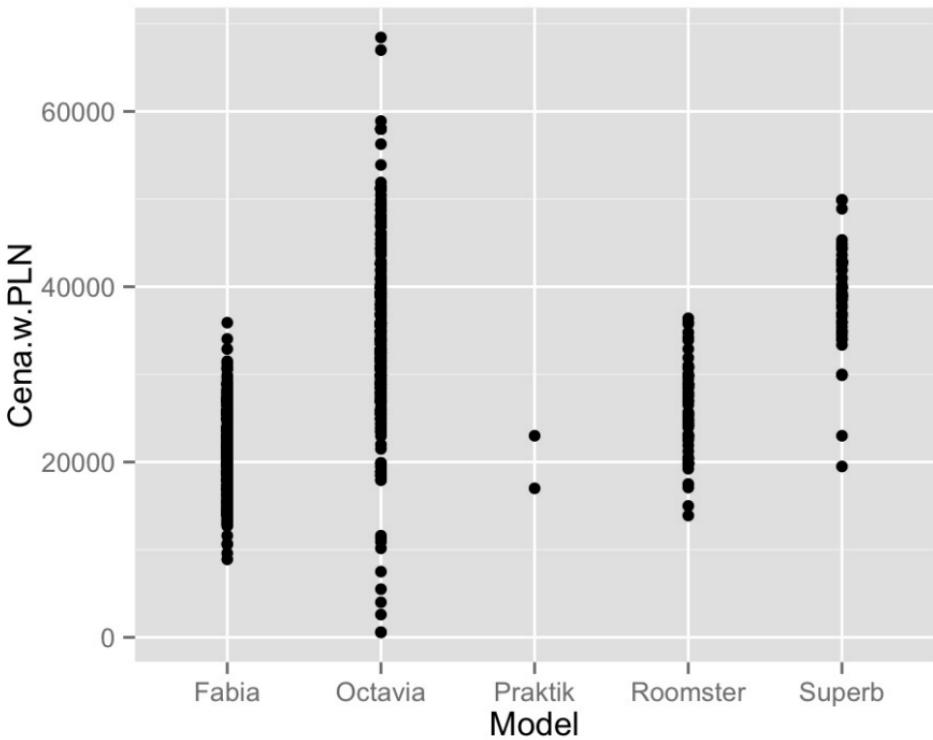
Czy różne modele pięcioletnich Skód mają różne ceny?

Pewnie tak. Ale jak różne są te średnie ceny oraz na ile są one zróżnicowane? Spróbujmy odpowiedzieć na to pytanie przedstawiając zależność pomiędzy ceną a modelem.

Zauważmy, że geometria `geom_point()` potrafi poradzić sobie z sytuacją, gdy jedna ze zmiennych nie jest zmienną ilościową ale przedstawia nazwę modelu. W poniższym przypadku następuje mapowanie takich wartości jak Octavia, Roomster, Superb na współrzędną x na wykresie. Domyślnie kolejność odpowiada kolejności alfabetycznej nazw tych czynników.

Czy to jednak czytelna prezentacja?

```
ggplot(skody, aes(x=Model, y = Cena.w.PLN)) +  
  geom_point()
```



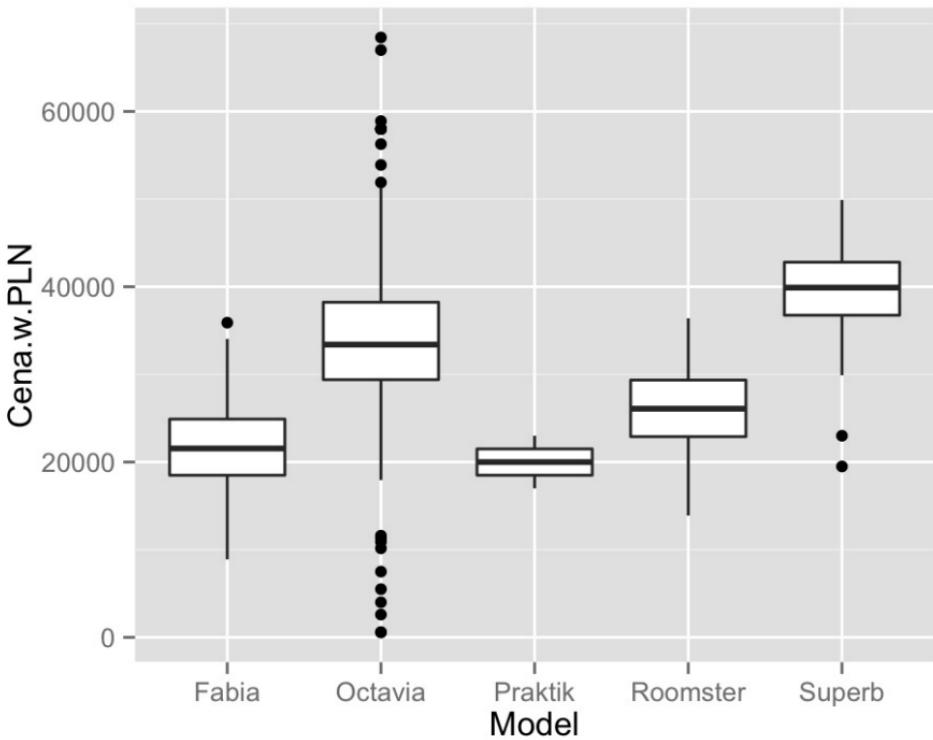
Geometria `geom_boxplot()` - wykresy ramka - wąsy

Patrząc na słupki punktów z poprzedniego slajdu trudno odczytać gdzie jest średnia, jaka jest zmienność cen oraz co porównywać pomiędzy modelami. Punktów jest tak dużo, że zlewają się ze sobą.

Do porównywania zmiennej liczbowej pomiędzy grupami, znacznie bardziej nadają się są wykresy pudełko - wąsy, przedstawiające tak zwaną piątkę Tukeya, czyli: minimalną obserwację z grupy, maksymalną, medianę (obserwację środkową) i dwa kwartyle. Te pięć liczb dzieli przedział zmienności cechy na cztery równoliczne przedziały. Od minimalnej wartości do dolnego kwartyla, od dolnego kwartyla do mediany, od mediany do górnego kwartyla i od górnego kwartyla do elementu maksymalnego.

Kwartyle i medianę prezentują dolny i górny brzeg ramki oraz środek ramki. Element minimalny i maksymalny to najniższa i najwyższa wartość na wykresie (odcinek lub punkt). Często najniższy i najwyższy element to odcinek (wąs), ale jeżeli jakieś obserwacje są daleko od brzegu pudełka (dalej niż 1.5 szerokości pudełka) to są one oznaczane osobnym punktem. Po co prezentowane są te punkty? Pomagają one w identyfikowaniu pojedynczych wartości odstających, gdy jedna lub kilka obserwacji jest bardzo daleko od pudełka (pudełko przedstawia środkową połowę obserwacji).

```
ggplot(skody, aes(x=Model, y = Cena.w.PLN)) +  
  geom_boxplot()
```



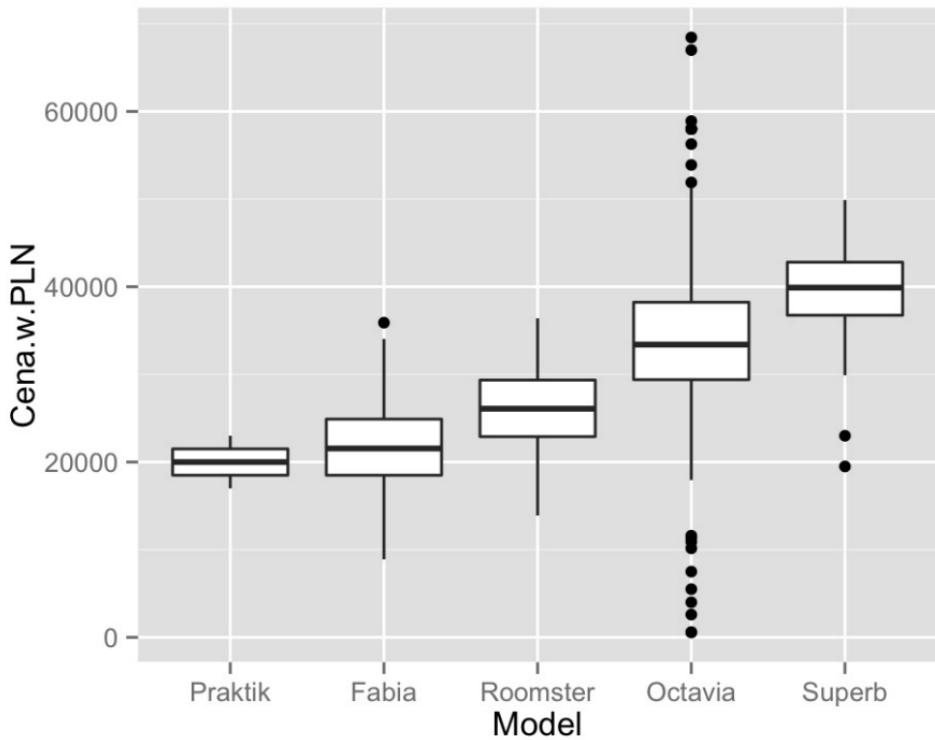
Geometria `geom_boxplot()` - wykresy ramka - wąsy

Kolejność na wykresach pudełkowych nie ma znaczenia. Ale często z powodów głównie estetycznych, chcemy uporządkować grupy od największej do najmniejszej. Aby to zrobić na wykresach ramka-wąsy, musimy wcześniej zmienić kolejność grup w zbiorze danych.

Służy do tego funkcja `reorder()`. Jako pierwszy argument przyjmuje wektor ze zmienną grupującą, drugim argumentem jest wektor wartości a trzecim argumentem funkcja. Kolejność grup jest zmieniana tak by odpowiadały kolejności wyników funkcji na tych grupach.

W poniższym przykładzie, poziomy czynnika Model są układane w ten sposób, by odpowiadały medianie (funkcja `median()`) ceny kolejnych modeli.

```
## zmień kolejność modeli z alfabetycznego na :
skody$Model <- reorder(skody$Model, skody$Cena
## ta sama instrukcja, prezentuje teraz inny -
ggplot(skody, aes(x=Model, y = Cena.w.PLN)) +
  geom_boxplot()
```



Geometria `geom_boxplot()` - wykresy ramka - wąsy

Opis parametrów funkcji `geom_boxplot()` jest przedstawiony na stronie http://docs.ggplot2.org/0.9.3.1/geom_boxplot.html.

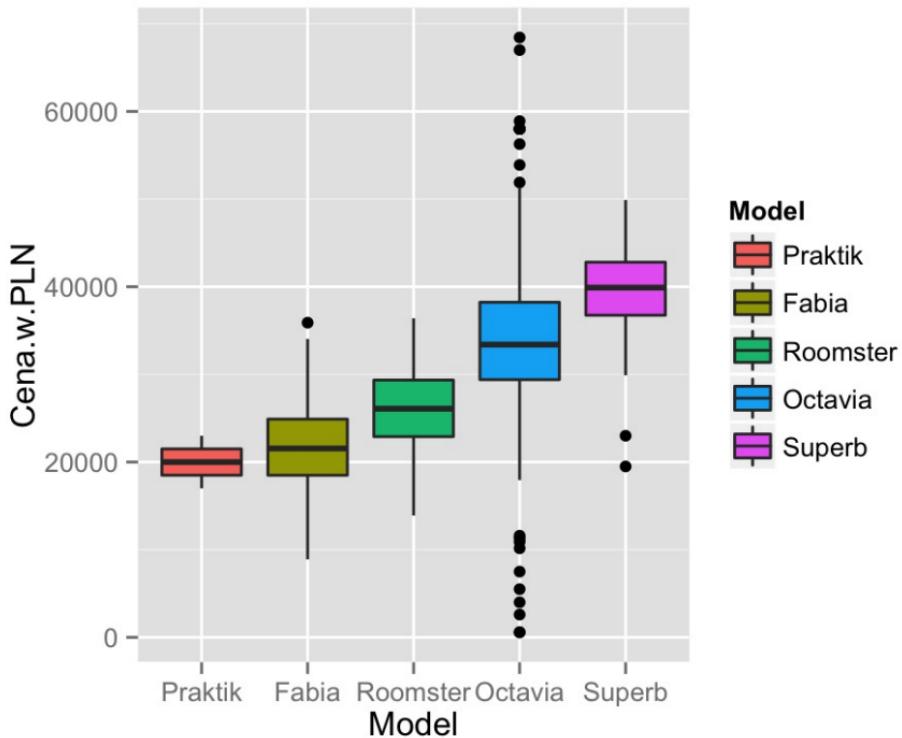
Jedną z ciekawszych opcji, pozwalającą na szybkie

wyróżnienie określonej grupy na wykresie jest użycie cechy wykresów pudełkowych wraz z kolorem wypełnienia.

Aby wypełnić wykresy pudełkowe różnymi kolorami wystarczy dodać mapowanie `fill=Model` wewnątrz funkcji `aes()`. Wynikiem ubocznym będzie legenda z opisem poszczególnych mapowań.

Same kolory nie wnoszą nowej informacji do wykresu, ale mogą ułatwić odnoszenie się do elementów wykresu. Można bowiem odwoływać się np. do niebieskiego pudełka zamiast mówić o pudełku dla samochodów modelu Skoda Octavia.

```
ggplot(skody, aes(x=Model, y = Cena.w.PLN, fill=Model)) +  
  geom_boxplot()
```



Geometria `geom_hist()` - histogram

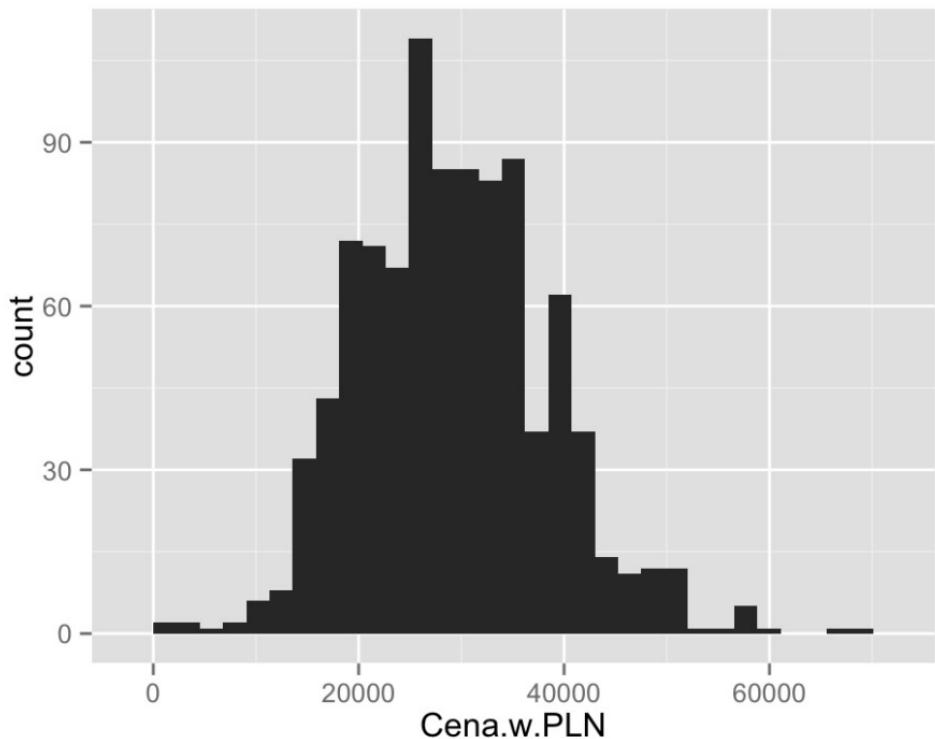
Wykres ramka - wąsy przedstawia pięć liczb opisujących wartości w grupie. Bardziej szczegółowym opisem, często pozwalającym na lepsze zrozumienie i przedstawienie zmienności w danych, jest histogram.

Histogram za pomocą długości słupków przedstawia liczebności (lub częstości) wartości w określonych

przedziałach. Definicja pokrętna, zobaczymy co to oznacza na przykładzie.

Mając wybrane pięcioletnie Skody, zobaczymy jak wygląda zmienność ich ceny.

```
ggplot(skody, aes(x=Cena.w.PLN)) +  
  geom_histogram()
```



Funkcja `geom_histogram()` szacuje na ile równie długich przedziałów podzielić zbiór wartości. W tym przypadku

cena została podzielona na około 30 przedziałów. Na wykresie przedstawiana jest liczba obserwacji, które zawierają się w określonym odcinku.

Co można dodatkowego odczytać z takiego wykresu? Np. ile jest mniej więcej samochodów w cenie poniżej 20 tysięcy lub powyżej 60 tysięcy. Takiej informacji nie wyciągnelibyśmy prosto z wykresu pudełko - wąsy.

Geometria `geom_hist()` - histogram

Prezentując dane za pomocą histogramu duże możliwości daje dodanie informacji o grupach obserwacji. Można to zrobić np. zmieniając kolor wypełnienia, a więc cechę `fill`.

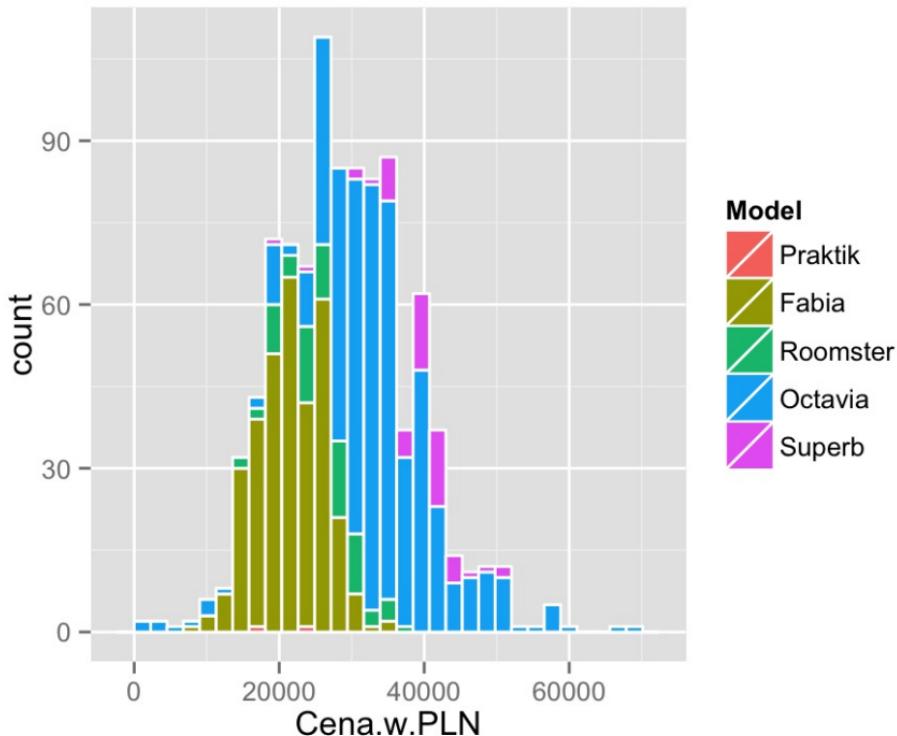
Na poniższym przykładzie kolor wypełnienia oznacza Model samochodu. W danych dla pięcioletnich Skód jest pięć różnych modeli aut. Każdy oznaczono innym kolorem. Wizualnie można teraz nie tylko ocenić w jakich przedziałach cenowych spotkać można najwięcej samochodów danej marki, ale również można oszacować liczbę aut danej marki w określonym przedziale.

Widzimy, że najpopularniejsze są Fabie i Oktavie, zajmują one najwięcej miejsca na wykresie. Dodatkowo cena 30

tysiący dobrze rozgranicza te dwa modele, Fabie są w większości tańsze a Oktavie droższe.

Aby zwiększyć czytelność poszczególnych grup dodaliśmy do wykresu białe obramowania poszczególnych słupków. Można to zrobić argumentem 'color='white'.

```
ggplot(skody, aes(x=Cena.w.PLN, fill=Model)) +  
  geom_histogram(color='white')
```



Geometria `geom_bar()` - wykresy słupkowe

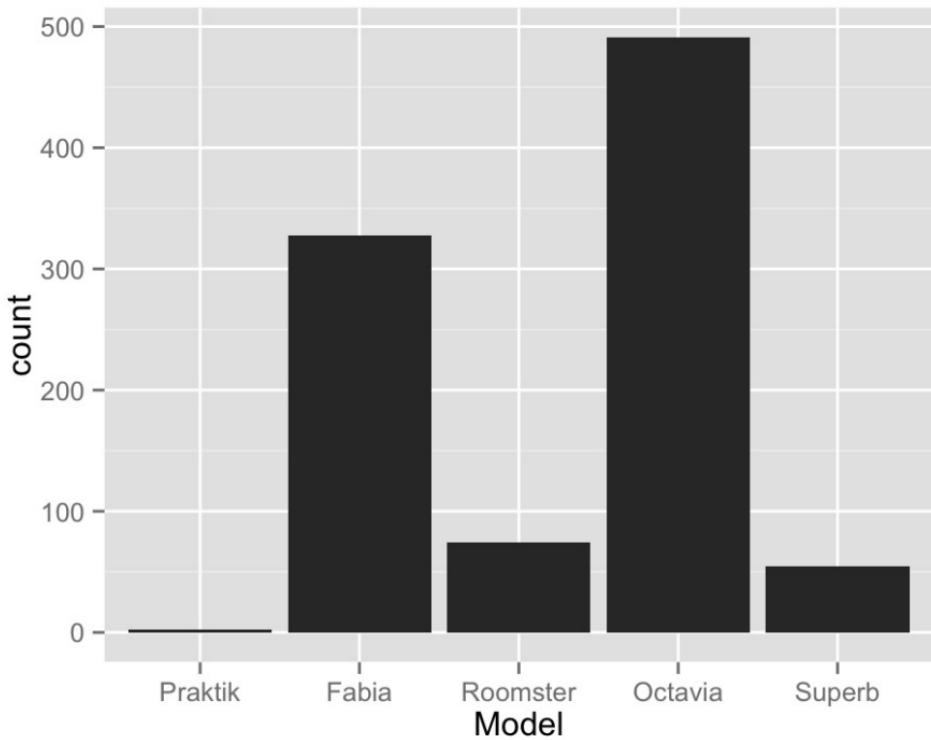
Jednym z częściej spotykanych wykresów są wykresy słupkowe. Czas więc i na nie. W pakiecie `ggplot2` można takie wykresy przygotować używając geometrii `geom_bar()`. Zlicza ona liczbę wystąpień poszczególnych grup i przedstawia te liczebności na wykresie za pomocą wysokości słupków.

Zarówno w użyciu jak i w wyglądzie, jest ona bardzo podobna do geometrii `geom_hist()`. To co je najbardziej różni, to że jedna operuje na zmiennych jakościowych (grupach) a inna na zmiennych ilościowych.

W geometrii `geom_bar()` wystarczy wskazać zmienną grupującą. Częstości dla poziomów tej zmiennej wyznaczone będą automatycznie i przedstawione na współrzędnej `y`.

Zobaczmy których modeli Skody jest najwięcej wśród oferowanych 5-letnich aut.

```
ggplot(skody, aes(x=Model)) +  
  geom_bar()
```



Geometria `geom_bar()` - wykresy słupkowe

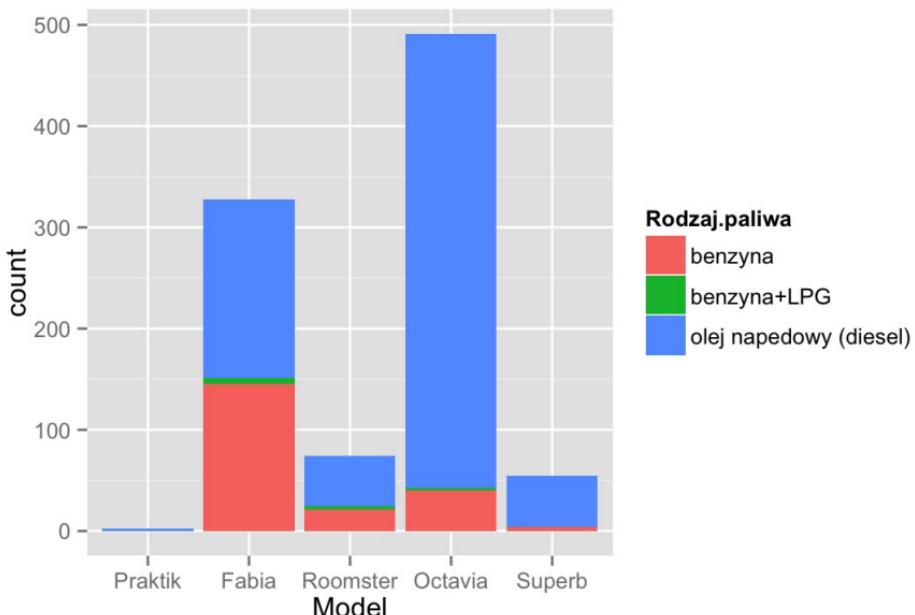
Podobnie jak w przypadku histogramu, tak i w przypadku wykresów słupkowych, możemy zaznaczyć na wykresie dodatkową zmienną grupującą. Najczęściej robi się to mapującą zmienną grupującą na kolor wypełnienia (cechę `fill`).

Spowoduje to automatyczne dodanie legendarnej do wykresu. Pozwoli też ocenić udział poszczególnych podgrup w grupach.

Na poniższym przykładzie dodamy informację o rodzaju paliwa. Dodatkowe mapowanie `fill=Rodzaj.paliwa` podzieli słupki na fragmenty o długościach proporcjonalnych do udziału ofert sprzedaży o różnym rodzaju paliwa.

Co ciekawego można z takiego wykresu odczytać? Skód z zainstalowaną instalacją gazową jest niewiele. Jeżeli szukamy samochodu z silnikiem diesla, to częściej można go spotkać u Skody Octavia i Superb niż Fabia czy Roomster.

```
ggplot(skody, aes(x=Model, fill=Rodzaj.paliwa))  
  geom_bar()
```



Geometria `geom_bar()` - wykresy słupkowe, wykresy warunkowe

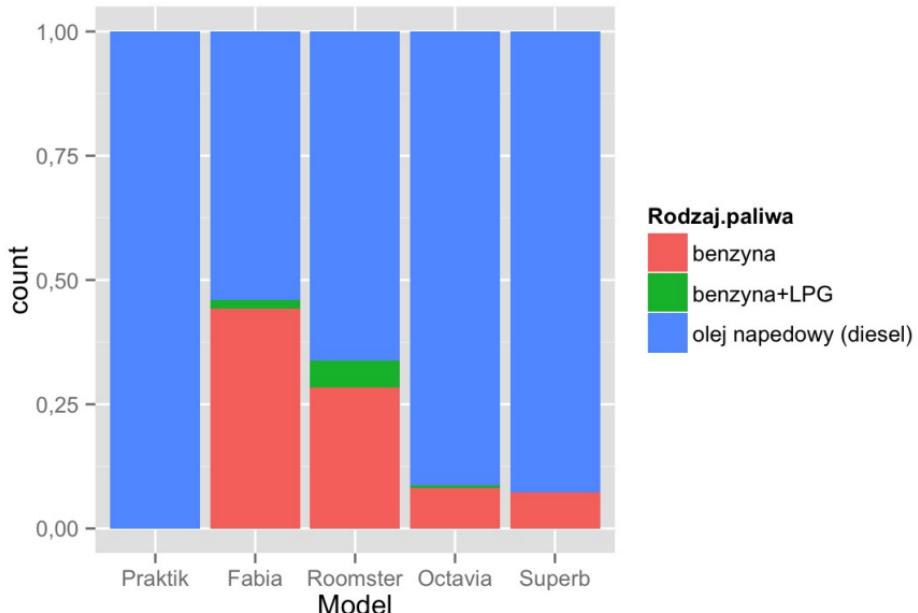
Jak dotąd histogram i wykresy paskowe pokazywały liczebności wierszy w poszczególnych grupach, np. modelach aut.

Gdy jednak zaczynamy umieszczać na wykresie informacje o kolejnych zmiennych, wygodnie jest móc uniezależnić się od wielkości tych grup.

Przykładowo, jeżeli interesuje nas względy udziału samochodów z silnikiem diesla, a więc stosunek samochodów zasilanych olejem w grupie Skoda Fabia, do wszystkich samochodów z grupy Skoda Fabia, to musimy istotne liczebności unormować.

Taką normalizację, można w geometrii `geom_bar()` wykonać dodając argument `position="fill"`.

```
ggplot(skody, aes(x=Model, fill=Rodzaj.paliwa))  
  geom_bar(position="fill")
```



A jak policzyć dokładnie liczebności grup?

Wykres pozwala nam na zbudowanie szybkiego wyobrażenia, jak duże są prezentowane wartości.

Ale trudno z niego bardzo dokładnie odczytać wartości liczbowe.

Aby wyliczyć dokładne częstotliwości, możemy wykorzystać poznane w pierwszym sezonie funkcje `group_by()` i `summarise()` z pakietu `dplyr`.

Dla każdej kombinacji grup Model / Rodzaj.paliwa wyznacz liczbę obserwacji.

```
skody %>%
  group_by(Model, Rodzaj.paliwa) %>%
  summarise(liczba = n())
```

```
## Source: local data frame [12 x 3]
## Groups: Model
## #          Model           Rodzaj.paliwa  liczba
## 1    Praktik olej napędowy (diesel)      2
## 2      Fabia                 benzyna    145
## 3      Fabia            benzyna+LPG      6
## 4      Fabia olej napędowy (diesel)   177
## 5   Roomster                 benzyna     21
## 6   Roomster            benzyna+LPG      4
```

## 7	Roomster	olej napędowy (diesel)	49
## 8	Octavia	benzyna	40
## 9	Octavia	benzyna+LPG	3
## 10	Octavia	olej napędowy (diesel)	448
## 11	Superb	benzyna	4
## 12	Superb	olej napędowy (diesel)	51

Gdzie szukać dalej

Podobnie jak dla poprzedniego odcinka, uzupełnieniem dla opisanych tematów będą następujące pozycje.

- Wiele przydatnych wskazówek można również znaleźć w internetowej książce „*Cookbook for R*”
<http://www.cookbook-r.com/Graphs/>
- Więcej informacji o tym jak budować wykresy w pakiecie ggplot2 można znaleźć w rozdziale 4 książki „*Przewodnik po pakiecie R*”. Więcej informacji o tej książce <http://www.biecek.pl/R/>
- Bardzo rozbudowana dokumentacja dla pakietu ggplot2 dostępna jest na stronie „*ggplot2*”
<http://docs.ggplot2.org/current/index.html>
- Wiele ciekawych przykładów z użyciem *ggplot2* można znaleźć na stronach
<http://www.ats.ucla.edu/stat/r/faq/> oraz

Zadanie, sezon 2, odcinek 8

- Wybierz samochody marki Volkswagen model Passat a następnie narysuj jak średnia cena zależy od roku produkcji za pomocą geometrii `geom_smooth()`.
- Wybierz samochody marki Volkswagen, narysuj jak średnia cena zależy od roku produkcji, różnymi kolorami przedstaw różne modele Volkswagena.
- Wybierz pięcioletnie auta marki Volkswagen i za pomocą wykresu ramka - wąsy przedstaw jak cena auta zależy od modelu.
- Dla wybranych pięcioletnich aut marki Volkswagen przedstaw w podziale na modele jaka część aut ma silnik diesla.

Przykładowe odpowiedzi znajdują się na stronie

http://pogromcydanych.icm.edu.pl/materials/2_modelowan

Pakiet `ggplot2` - praca nad detalami

Przemysław Biecek @ Uniwersytet Warszawski

*sezon 2 / odcinek 4
pogRomcy danych*

- O czym jest ten odcinek
- Adnotacje, napisy na osiach i tytuły
- Większa kontrola nad mapowaniami
- Większa kontrola nad mapowaniami
- Większa kontrola nad mapowaniami - zmienne ciągłe
- Większa kontrola nad mapowaniami - zmienne ciągłe
- Modyfikacja legendy wykresu
- Modyfikacja legendy wykresu
- Wiele wykresów w jednym oknie
- Wiele wykresów w jednym oknie - `vp=`
- Wiele wykresów w jednym oknie - `grid.layout`
- Wiele wykresów w jednym oknie - `grid.arrange`

- [Motyw graficzne](#)
- [Ciekawostki](#)
- [Co dalej](#)
- [Zadanie, sezon 2, odcinek 9](#)

O czym jest ten odcinek

Funkcje z pakietu `ggplot2` wyglądają dobrze przy domyślnych ustawieniach.

Ale tworząc bardziej złożoną grafikę, często pojawia się potrzeba zmiany jakiegoś detalu.

W tym odcinku nauczmy się:

- Jak dodać adnotacje do wykresu?
 - Jak kontrolować mapowanie dla cechy `shape` (kształt)?
 - Jak kontrolować mapowanie dla cechy `color` (kolor)?
 - Jak modyfikować legendę wykresu?
 - Jak zmieniać motyw graficzne wykresu?
-

Adnotacje, napisy na osiach i tytuły

Dobry tytuł wykresu i opisy osi to elementy, które

znacznie ułatwiają poprawne odczytanie wykresu. Ich wkład jest umniejszany ale w rzeczywistości są to bardzo ważne elementy.

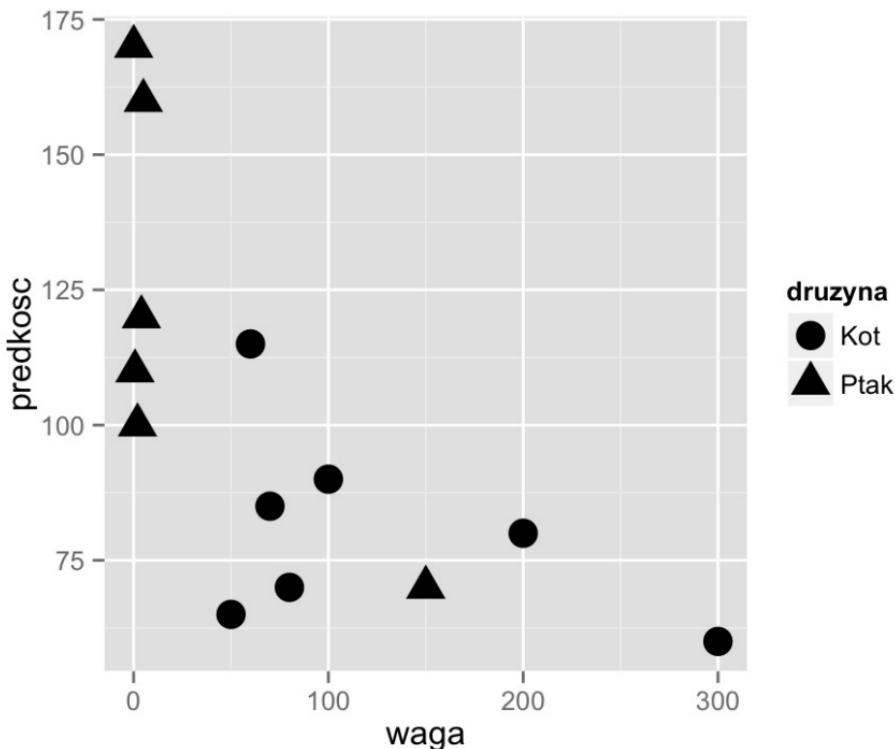
Dlaczego są one takie ważne? Ponieważ bez tytułu i opisu osi czytelnik może nie mieć pojęcia co wykres przedstawia. A jeżeli czytelnik nie wie co wykres przedstawia, to taki wykres jest bezużyteczny.

Dlaczego umniejszany? Ponieważ twórca wykresu zazwyczaj wie co wykres przedstawia i niekoniecznie ma ochotę i czas by dbać o czytelnika.

My będziemy tworzyć bardzo dobre wykresy, ponieważ na naszych wykresach wszystko będzie jasne i opisane.
Przyjrzyjmy się wykresowi z pierwszego odcinka i zastanówmy się, które elementy warto dodatkowo opisać?

Poświęć kilka minut by znaleźć przynajmniej pięć elementów, które warto lepiej opisać na tym wykresie.

```
library(PogromcyDanych)
ggplot(koty_ptaki, aes(x=waga, y=predkosc, shape=specie))
  geom_point(size=5)
```



Adnotacje, napisy na osiach i tytuły

Czy to było trudne zadanie?

Mogło być, ponieważ widzieliśmy ten wykres już wiele razy i wiemy co on przedstawia. Jeżeli jednak udało Ci się znaleźć elementy, które warto dodatkowo opisać, to świetnie, jesteś na dobrej ścieżce by krytycznie patrzeć na wykresy.

Oto co ja znalazłem:

- W opisach osi brakuje jednostek. Jest informacja ‘waga’ ale w jakich to jednostkach? Kilogramach, funtach?
- Podobny problem dotyczy prędkości, czy to kilometry na godzinę czy mile?
- Opis legendy ‘drużyna’ może i jest zabawny ale niewiele mówi. Jaka drużyna? Dla kogoś, kto nie skojarzy tych danych z wyścigami gatunków, ta nazwa nic nie mówi.
- O co chodzi z tym wykresem? Gdzie jest tytuł? Co on w ogóle przedstawia?
- Jakie gatunki reprezentują najbardziej skrajne punkty? Który kot i który ptak jest najszybszy?

Zobaczmy jak przygotować wykres z odpowiednimi adnotacjami.

Adnotacje, napisy na osiach i tytuły

W poniższym przykładzie do wykresu dodajemy cztery linijki, każda zmieniająca określony element wykresu.

1. Funkcja `ggtitle()` dodaje tytuł do wykresu a funkcją `theme()` zmieniamy wielkość tytułu wykresu.

Większy będzie wyraźniejszy.

2. Funkcjami `xlab()` i `ylab()` dodajemy opisy dla osi. Do opisów osi dodaliśmy jednostki, warto też by opisy zaczynały się dużą literą i miały polskie znaki.
3. Aby zmienić tytuł legendy należy określić argument `name` (nazwa). Zrobimy to używając funkcji opisującej mapowanie na rozmiar (`scale_shape_discrete()`). Funkcją `theme()` przesuwamy tę legendę na góre wykresu (domyślnie jest po prawej stronie).
4. Funkcją `geom_text()` dodajemy etykiety dla wybranych punktów na wykresie. Zaznaczyliśmy najszybszego kota i najszybszego ptaka.

Adnotacje, napisy na osiach i tytuły

A oto i wykres po dodaniu tych modyfikacji

Przed [lewy]

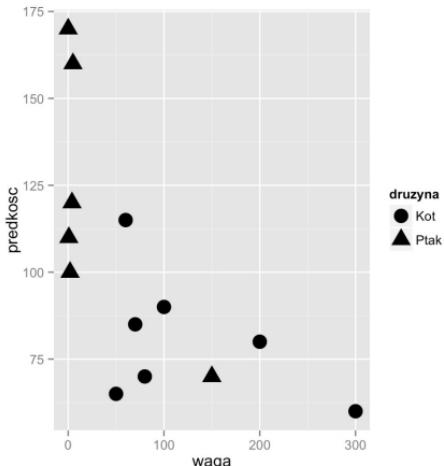
```
ggplot(koty_ptaki, aes(x=waga, y=predkosc, shape=rodzaj)) +  
  geom_point(size=5)
```

Po [prawy]

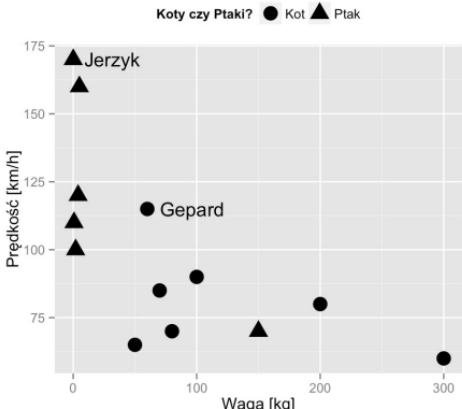
```

ggplot(koty_ptaki, aes(x=waga, y=predkosc, shape=druzyna))
  + geom_point(size=5) +
  # większy i czytelny tytuł
  ggtitle("Lżejszym łatwiej szybko biegać!") +
  # opisy osi
  xlab("Waga [kg]") + ylab("Prędkość [km/h]")
  # tytuł w legendzie
  scale_shape_discrete(name="Koty czy Ptaki?")
  # dodatkowe napisy na wykresie
  geom_text(data=koty_ptaki[c(6,8),], aes(label=imię,
                                             x=waga, y=predkosc))

```



Lżejszym łatwiej szybko biegać!



Większa kontrola nad mapowaniami

Mapowania to jedne z najciekawszych rozwiązań w pakiecie `ggplot2`. Wystarczy, że wskażemy jakie zmienne mają być przedstawione na wykresie, oraz jakie cechy wykresu mają te zmienne opisać, a program R automatycznie dobiera sposób prezentacji uwzględniając

typ zmiennej. Przy okazji sprawdzi czy przyjmuje ona kilka wartości czy wiele, czy jest zmienną ilościową czy jakościową itp.

Czasem jednak, chcielibyśmy zmienić sposób reprezentacji zmiennych, a więc zmienić domyślne mapowanie.

Możemy to zrobić korzystając z funkcji o nazwach zgodnych z szablonem `scale_***_yyy()`, gdzie `***` opisuje cechę na którą mapujemy (`shape/size/color/x/y`), a `yyy` opisuje sposób reprezentacji.

Przykładowo, funkcja `scale_shape_manual()` pozwala na „ręczne” zarządzanie mapowaniem zmiennej na kształty punktów. W tej funkcji argument `values=` pozwala na wskazanie jakimi kształtami mają być przedstawione poszczególne grupy, a argument `breaks=` pozwala wskazać wartości mapowanej zmiennej w legendzie.

W podobny sposób `scale_color_manual()` pozwala na ręczne (manualne) wskazanie kolorów, które mają kodować poszczególne wartości.

A mapując na cechy `x` i `y` (współrzędne) interesującym parametrem jest `expand=`, który pozwala na określenie jak duże mają być marginesy na osi.

Większa kontrola nad mapowaniami

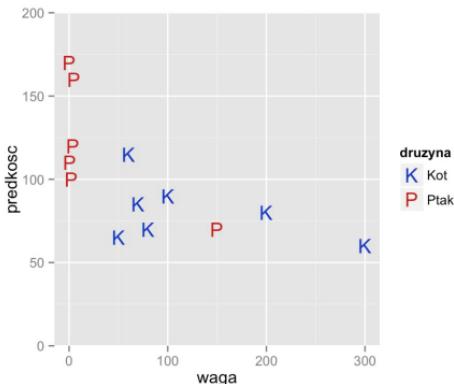
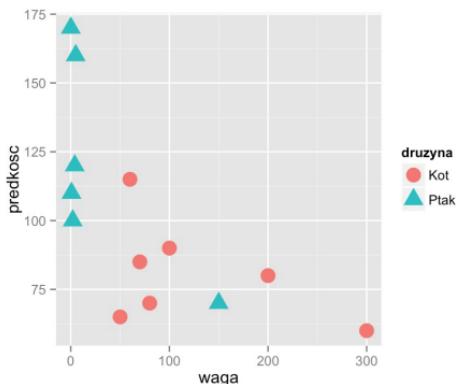
A oto i przykładowe zastosowanie wszystkich tych modyfikacji mapowań.

Przed [lewy]

```
ggplot(koty_ptaki, aes(x=waga, y=predkosc, shape=drzyna)) +  
  geom_point(size=5)
```

Po [prawy]

```
ggplot(koty_ptaki, aes(x=waga, y=predkosc, shape=drzyna)) +  
  geom_point(size=5) +  
  # określamy kształty poszczególnych grup K/P  
  scale_shape_manual(values=c("K", "P"), breaks=c(0, 100, 200), labels=c("Kot", "Ptak")) +  
  # określamy kolory (ciemnoniebieski i ciemnoczerwony)  
  scale_color_manual(values=c("blue3", "red3")) +  
  # dla osi OY usuwany marginesy poza zakresem  
  scale_y_continuous(limits=c(0, 200), expand=c(0, 0))
```



Większa kontrola nad mapowaniami - zmienne ciągłe

Zobaczmy jak wygląda kontrolowanie mapowania zmiennych ciągłych. Jako przykład wykorzystamy zmienną `zywotnosc`. Będziemy ją mapować na kolor i wielkość punktu.

Jeżeli nie podoba nam się domyślny dobór kolorów (dla zmiennych ciągłych od czarnego do niebieskiego), to funkcją `scale_color_gradient()` możemy określić pomiędzy jakimi wartościami `low=` i `high=` ma rozpinać się skala kolorów.

Dla wielkości punktu określamy jaki przedział żywotności ma być prezentowany `limits=` oraz na jaki zakres wielkości punktów ten przedział ma być przedstawiony `range=`.

Więcej informacji o możliwych dodatkowych parametrach znaleźć można w plikach pomocy dla funkcji `?scale_color_gradient` i `?scale_size_continuous`.

Większa kontrola nad mapowaniami - zmienne ciągłe

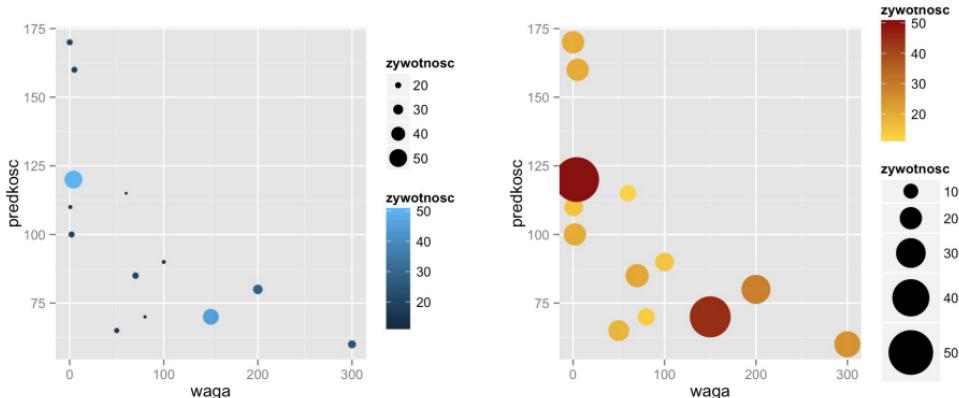
A oto i przykładowe zastosowanie wszystkich tych modyfikacji mapowań.

Przed [lewy]

```
ggplot(koty_ptaki, aes(x=waga, y=predkosc, size=zywotnosc)) +  
  geom_point()
```

Po [prawy]

```
ggplot(koty_ptaki, aes(x=waga, y=predkosc, size=zywotnosc)) +  
  geom_point() +  
  # zmiana zakresu wartości dla wielkości punktu  
  scale_size_continuous(range=c(5,15), limits=c(20,50))  
  # zmiana skali kolorów (ciągła skala, podaje zakres)  
  scale_color_gradient(low="gold", high="red4")
```



Modyfikacja legendy wykresu

Pakiet `ggplot2` automatycznie tworzy i rysuje legendę. W

wielu przypadkach to domyślne zachowanie jest wystarczające i legenda jest wystarczająco czytelna. Ale czasem chcemy umieścić tę legendę w innym miejscu, inaczej ją zatytułować czy pokolorować.

To też można zmienić. Najczęściej wystarczy zmodyfikować poszczególne argumenty motywu graficznego, używając funkcji `theme()`.

W poniższym przykładzie funkcja `theme()` służy do określenia koloru tła i obramowania w legendzie, wielkości tytułu legendy oraz jej pozycji. Funkcja `scale_shape_manual()` określa elementy mapowania, tytuł legendy i nazwę kluczy na legendzie.

Modyfikacja legendy wykresu

Przykład zmian w wyglądzie legendy.

Przed [lewy]

```
ggplot(koty_ptaki, aes(x=waga, y=predkosc, shape=mapa)) +  
  geom_point(size=5)
```

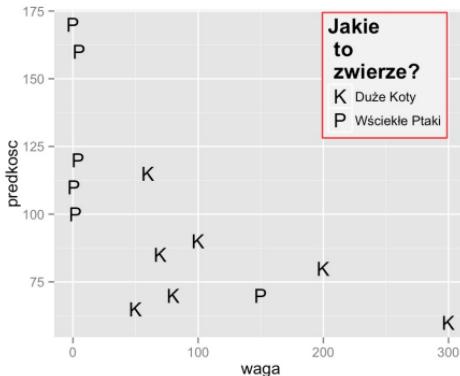
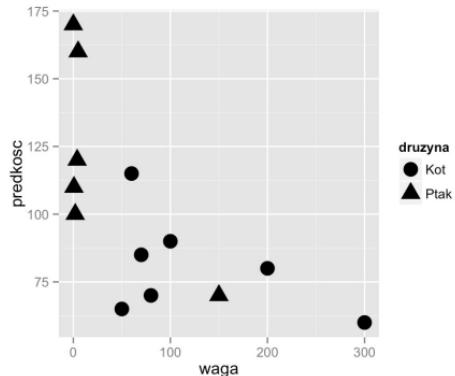
Po [prawy]

```
ggplot(koty_ptaki, aes(x=waga, y=predkosc, shape=mapa)) +  
  geom_point(size=5) +  
  # zmiana napisów w legendzie dotyczących map
```

```

scale_shape_manual(values=c("K", "P"),
                   labels=c("Duże Koty", "Wściele Ptaki"),
                   name="Jakie \n to \n zwierze?")
# dodanie szarego tła, większego tytułu oraz
theme(legend.background = element_rect(color= "#D9D9D9"),
      legend.title = element_text(size=15),
      legend.position=c(0.8,0.8))
# 0.8, 0.8 to współrzędne gdzie ma pojawić się legenda

```



Wiele wykresów w jednym oknie

Bardzo często chcemy umieścić kilka wykresów obok siebie. Poniżej pokażemy jak to zrobić.

W programie R wszystko jest obiektem. Oznacza to, że również wynik funkcji `ggplot()` jest obiektem, który można przypisać do zmiennej i później wielokrotnie wykorzystać.

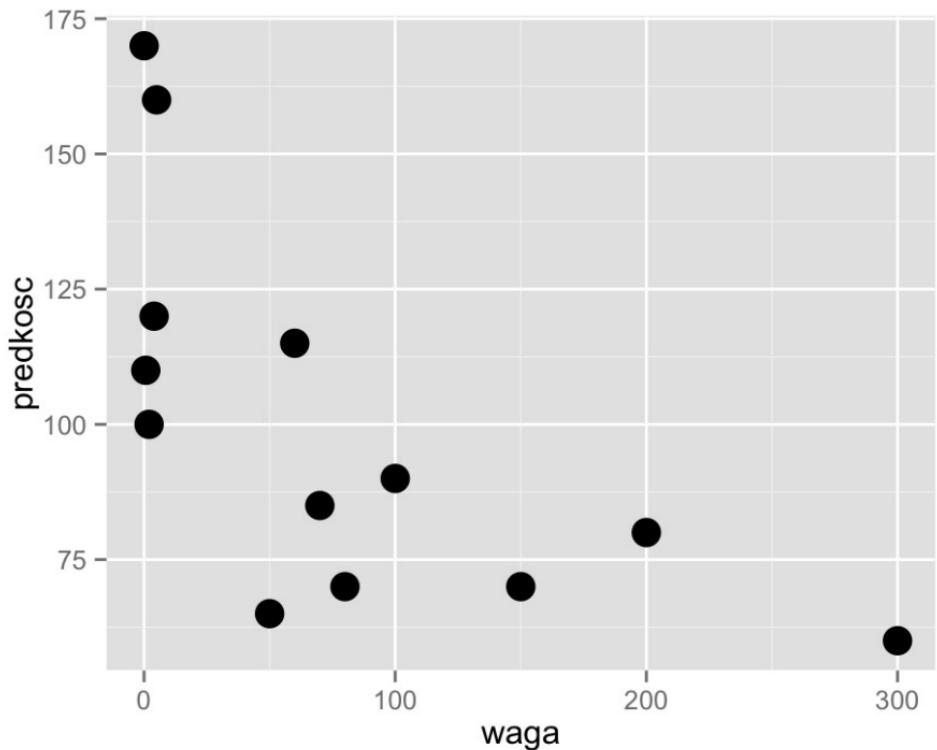
Zilustrujmy to przykładem. Poniższa instrukcja przypisze

definicje wykresu punktowego do zmiennej `pl_waga`.
Żaden wykres nie będzie narysowany chyba, że
wyświetlimy zawartość zmiennej `pl_waga`.

```
pl_waga <- ggplot(koty_ptaki, aes(x=waga, y=preco)) +  
  geom_point(size=5)
```

Aby wyświetlić wykres `pl_waga` można wykorzystać
funkcję `print()`.

```
print(pl_waga)
```



Wiele wykresów w jednym oknie - vp=

Jednym z argumentów funkcji `print()` dla wykresów, jest argument `vp=`. Pozwala on określić w którym miejscu ekranu wykres ma być rysowany.

Domyślnie zarysowany jest cały obszar o współrzędnych $[0,1] \times [0,1]$, ale możemy zażyczyć sobie by wykres rysowany był w mniejszym obszarze.

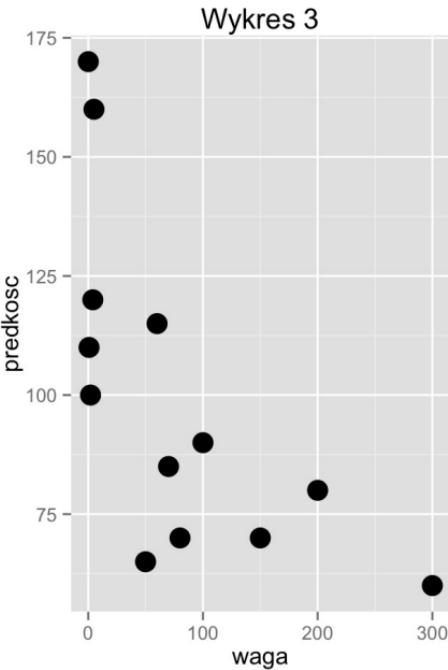
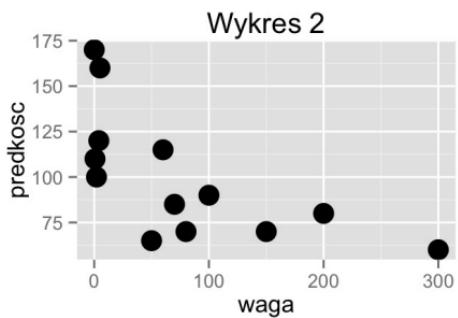
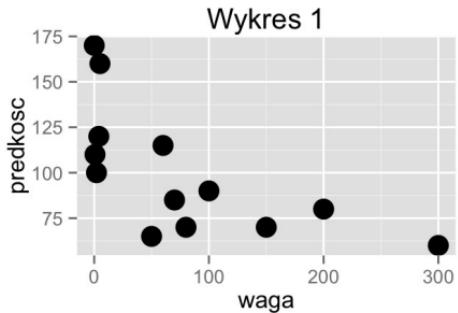
Jest to bardzo wygodne, jeżeli chcemy zmieścić kilka wykresów obok siebie.

Poniższy przykład rysuje ten sam wykres trzykrotnie, za każdym razem w innym miejscu i z innym tytułem.

Argumenty `x` i `y` określają środek zarysowanego obszaru, a `width` i `height` szerokość i wysokość zarysowanego prostokąta.

```
library(grid)
## rysujemy wykres w kwadracie o szerokości i wysokości 1
## w lewym górnym rogu
print(pl_waga + ggtitle("Wykres 1"),
      vp=viewport(x=0.25, y = 0.75, width=0.5,
## rysujemy wykres w kwadracie o szerokości i wysokości 1
## w lewym dolnym rogu
print(pl_waga + ggtitle("Wykres 2"),
      vp=viewport(x=0.25, y = 0.25, width=0.5,
## rysujemy wykres w prostokącie o szerokości 0.5 i wysokości 0.5
## po prawej stronie
```

```
print(pl_waga + ggtitle("Wykres 3"),  
      vp=viewport(x=0.75, y = 0.5, width=0.5, height=0.5))
```



Wiele wykresów w jednym oknie - `grid.layout`

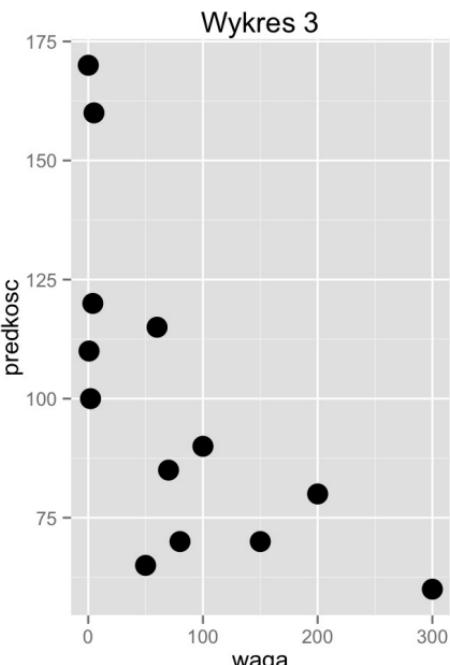
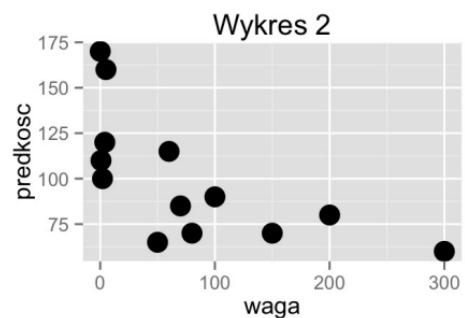
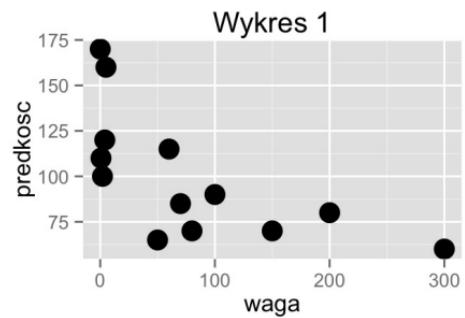
Argument `vp=` pozwala na narysowanie wykresu w dowolnym miejscu na ekranie.

Ale jeżeli chcemy rysować wykresy na regularnej siatce to wygodnie jest najpierw określić tę siatkę funkcją `grid.layout()`, a następnie wypełnić ją wykresami

wskazując pozycje siatki do zarysowania. Zobaczmy jak to zrobić.

Tworzymy opis siatki o rozmiarze dwa wiersze i dwie kolumny. Następnie wklejamy wykresy, do wybranych komórek poprzednio zdefiniowanej siatki.

```
pushViewport(viewport(layout = grid.layout(2, 2)
## rysujemy w komórce o współrzędnych 1x1
print(pl_waga + ggtitle("Wykres 1"),
      vp = viewport(layout.pos.row=1, layout.p
## rysujemy w komórce o współrzędnych 2x1
print(pl_waga + ggtitle("Wykres 2"),
      vp = viewport(layout.pos.row=2, layout.p
## rysujemy w komórce o współrzędnych 1:2x2
print(pl_waga + ggtitle("Wykres 3"),
      vp = viewport(layout.pos.row=1:2, layout
```



Wiele wykresów w jednym oknie - `grid.arrange`

Najprostszym sposobem na wyświetlenie kilku wykresów obok siebie (jeżeli tylko nie potrzebujemy żadnego wyrafinowanego rozmieszczenia) jest użycie funkcji `grid.arrange()` z pakietu `gridExtra`.

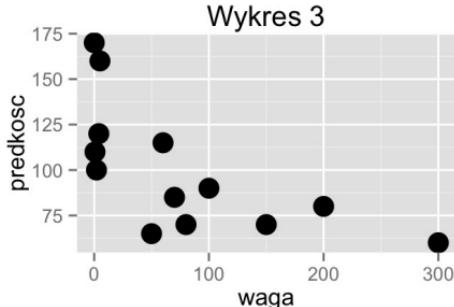
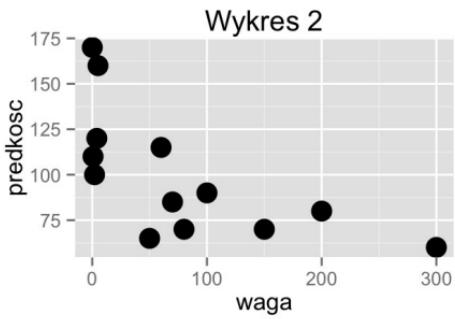
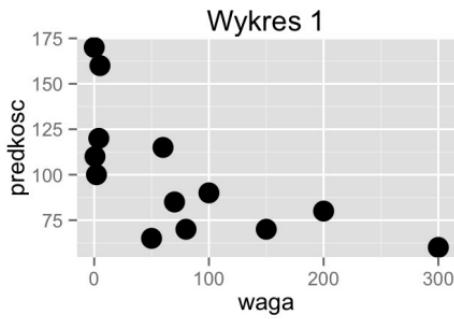
Wystarczy jako kolejne argumenty podać wykresy, które mają być wyświetlane. Funkcja `grid.arrange()` sama je uporządkuje. Argumentem `ncol` można określić w ilu

kolumnach mają być wyświetlane wykresy.

Uwaga! Funkcja `grid.arrange()` jest dostępna w pakiecie `gridExtra`. Ten pakiet nie jest dostępny w podstawowej dystrybucji R. Należy go doinstalować funkcją `install.packages()`.

```
library(gridExtra)
```

```
grid.arrange( pl_waga + ggttitle("Wykres 1") ,  
              pl_waga + ggttitle("Wykres 2") ,  
              pl_waga + ggttitle("Wykres 3") ,  
              ncol=2)
```



Motyw graficzne

Do tego miejsca mogło nam trochę obrzydnąć szare tło na wykresach.

Okazuje się, że w prosty sposób można zmienić wygląd całego wykresu, używając tzw. motywów, czyli kompletów ustawień graficznych. Wiele takich motywów dostępnych jest w pakiecie `ggthemes`.

Aby zmienić wygląd wykresu wystarczy dodać funkcję z motywem. Nazwy funkcji z motywami graficznymi zazwyczaj rozpoczynają się od `theme_`.

Poniżej przykład dla czterech motywów graficznych. Zachęcam do samodzielnego sprawdzenia jak wyglądają inne.

Przygotowujemy wykres punktowy, do którego dodamy motywy.

```
pl <- ggplot(koty_ptaki, aes(x=waga, y=predkos)  
  geom_point(size=5)
```

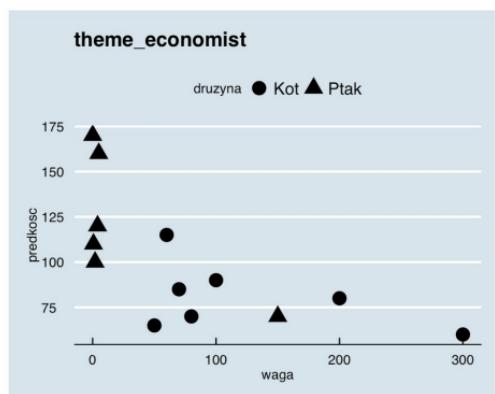
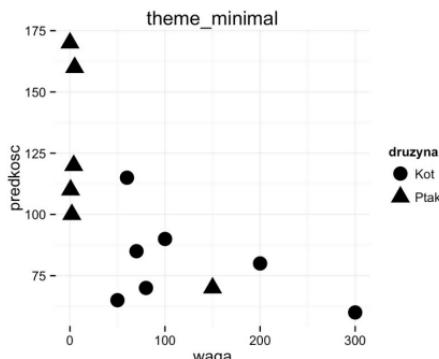
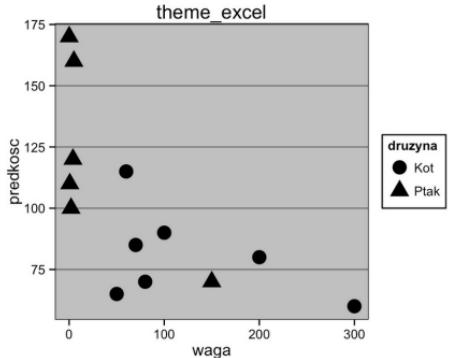
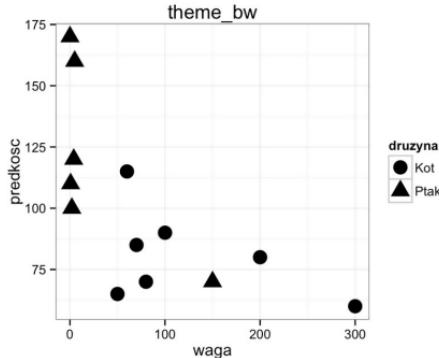
Włączamy pakiet z dodatkowymi motywami. Jeżeli nie jest zainstalowany, należy go wcześniej zainstalować funkcją `install.packages()`.

```
library(ggthemes)
```

```

## białe tło i szare linie pomocnicze
pl + theme_bw()
## wykres stylizowany na wykresy z Excela
pl + theme_excel()
## wykres z usuniętymi dodatkowymi elementami
pl + theme_minimal()
## wykres stylizowany na gazecie Economist
pl + theme_economist()

```

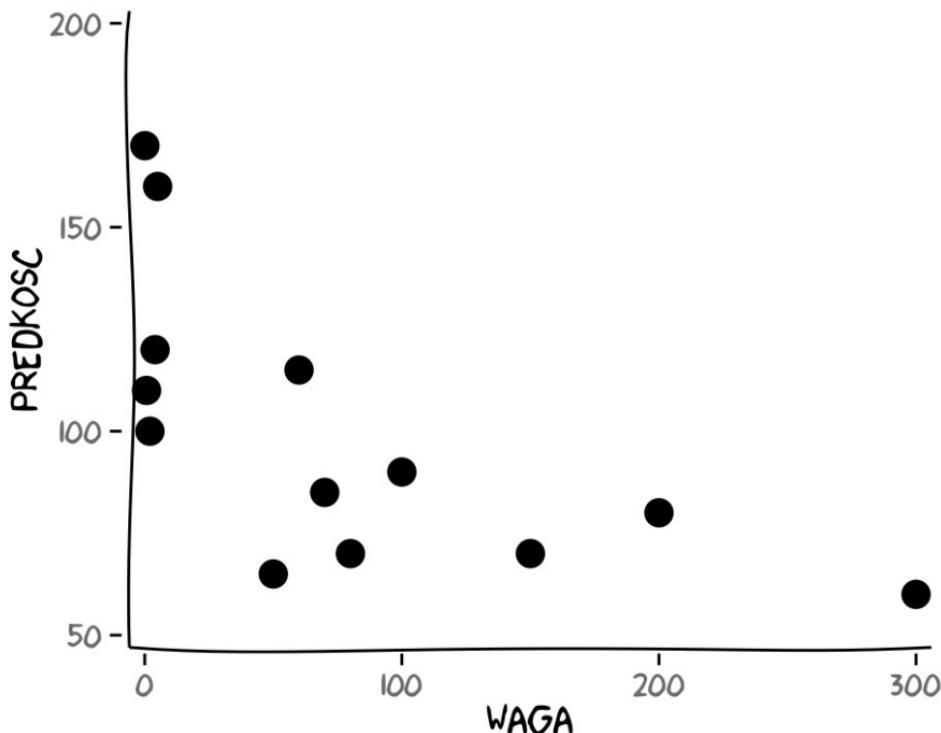


Ciekawostki

Elastyczność pakietu `ggplot2` jest olbrzymia. Można robić bardzo niestandardowe rzeczy.

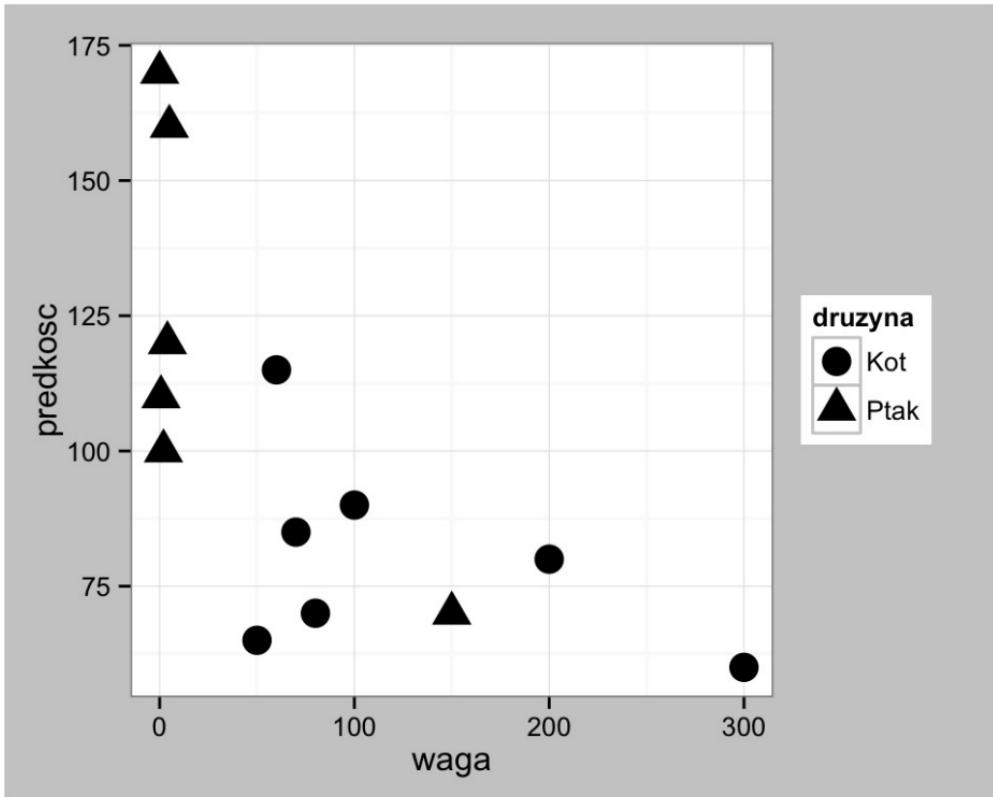
Przykładowo, możliwe jest tworzenie wykresów w stylu popularnych komiksów z serii XKCD <http://xkcd.com/>

```
## aby można było rysować wykresy jak w komiksach
## Jeżeli go jeszcze nie masz, najpierw zainstaluj
library(xkcd)
ggplot(koty_ptaki, aes(x=waga, y=predkosc)) +
  geom_point(size=5) + xkcdaxis(c(0,300),c(50,200))
```



A bazując na motywie `theme_bw()` można uzyskać inwersję domyślnych wykresów z pakietu `ggplot2`. Nie są to być może codziennie używane możliwości, ale pokazują, że zgłębiając pakiet `ggplot2` nie natknijemy się na barierę w stylu „tego się nie da zrobić”.

```
pl + theme_bw() + theme(plot.background = elem
```



Co dalej

- Informację o różnych nakładkach dla pakietu `ggplot2` znaleźć można na `easyGgplot2`
<http://www.sthda.com/english/wiki/easyggplot2>
- Niezła ściągawka jak korzystać z pakietu „`ggplot2`”
<http://www.rstudio.com/wp-content/uploads/2015/03/ggplot2-cheatsheet.pdf>
- Wklejając wykresy do Excela lub Worda może nam zależeć na wykresach, które można edytować. In a to jest metoda
<http://www.sthda.com/english/wiki/create-an-editable-graph-from-r-software>
- Wiele przydatnych wskazówek można również znaleźć w internetowej książce „*Cookbook for R*”
<http://www.cookbook-r.com/Graphs/> oraz w dokumencie *Effective tables and graphs in official statistics, Government Statistical Service*
<https://gss.civilservice.gov.uk/wp-content/uploads/2014/12/Effective-graphs-and-tables-in-official-statistics-version-1.pdf>
- W internecie znaleźć można wiele stron z poradami, jak robić dobre wykresy. Wybrane ciekawe listy takich porad to:
 - *Do's and Don'ts* http://stat545-ubc.github.io/block015_graph-dos-donts.html

- *Ten Simple Rules for Better Figures*
<http://www.ploscompbiol.org/article/info:doi/10>
- *Twenty rules for good graphics*
<http://robjhyndman.com/hyndsiht/graphics/>

Zadanie, sezon 2, odcinek 9

- Podobnie jak w poprzednim odcinku, wybierz samochody marki Volkswagen model Passat a następnie narysuj jak średnia cena zależy od roku produkcji za pomocą geometrii `geom_smooth()`. Następnie zobacz jak ten wykres będzie wyglądał z motywami `theme_bw()`, `theme_excel()` i `theme_economist()`.
- Zmień poniższy wykres, zamieniając skalę kolorów na od zielonego do czerwonego, kropki zamień na kwadraty a do wykresu dodaj tytuł (i odpowiednie etykiety osi).

```
ggplot(koty_ptaki, aes(x=waga, y=predkosc, size=geom_point())
```

Przykładowe odpowiedzi znajdują się na stronie
http://pogromcydanych.icm.edu.pl/materials/2_modelowan

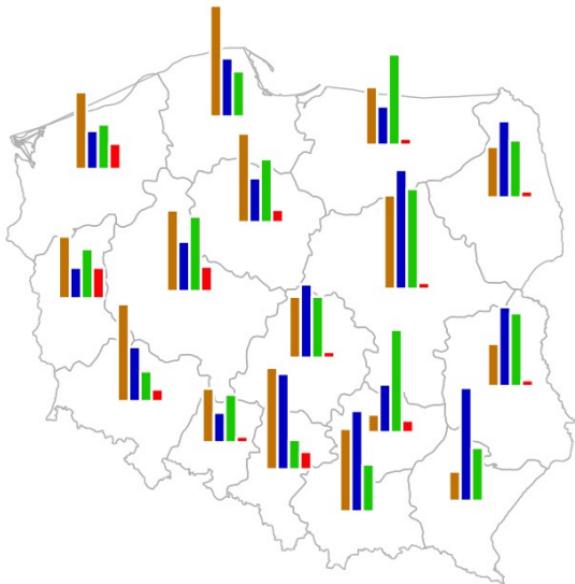
Pakiet **ggplot2** - mapy

Przemysław Biecek @ Uniwersytet Warszawski

sezon 2 / odcinek 5

pogRomcy danych

- O czym jest ten odcinek
- Zbiór danych
- Pliki z mapami / kształtami - shape files
- Korygujemy pliki z kształtami - shape files
- Rysowanie mapy województw
- Rysowanie mapy województw - `geom_map()`
- Rysowanie mapy
- Rysowanie mapy - pracujemy nad szczegółami
- Rysowanie mapy - zmienna jako wielkość punktu.
- Rysowanie mapy - zmienna jako długość paska
- Rysowanie mapy - zmienna jako długość paska



-
- [Co dalej](#)

O czym jest ten odcinek

Jednym z bardzo atrakcyjnych wizualnie sposobów prezentacji danych są kartogramy, czyli popularne „mapki”.

W tym odcinku nauczmy się

- Jak wczytywać dane o mapach?
- Jak przedstawiać na mapach informacje za pomocą

koloru wypełnienia?

- Jak przedstawiać na mapach informacje za pomocą wielkości punktów lub słupków?

W tym odcinku będziemy pracować ze zbiorem danych mandatySejmik2014, który jest dostępny w pakiecie PogromcyDanych.

Zbiór danych

W tym odcinku będziemy pracować na zbiorze danych o wynikach wyborów samorządowych z roku 2014 do sejmików.

Zbiór danych mandatySejmik2014 dla każdego z 16 województw pokazuje ile mandatów zdobyli kandydaci pod sztandarami SLD, PiS, PO, SLD i innymi.

W kolumnie ProcentWaznychGlosow przedstawiono procent ważnych oddanych głosów (w stosunku do osób uprawnionych do głosowania) a ostatnie dwie kolumny przedstawiają długość i szerokość geograficzną środka województwa.

```
## jeżeli pakiet nie jest zainstalowany, należy
## funkcją install.packages("PogromcyDanych")
library(PogromcyDanych)
```

```
## pierwsze sześć wierszy ze zbioru danych
head(mandatySejmik2014)

##      Województwo PSL PiS PO SLD Inne Pro
## 1 Dolnośląskie   5   9 16   2   4
## 2 Kujawsko-Pomorskie 10   7 14   2   0
## 3 Łódzkie        10  12 10   1   0
## 4 Lubelskie       12  13  7   1   0
## 5 Lubuskie        8   5 10   5   2
## 6 Małopolskie     8  17 14   0   0
##      lat
## 1 51,07988
## 2 53,02223
## 3 51,55958
## 4 51,24725
## 5 52,20549
## 6 49,84203
```

Pliki z mapami / kształtami - shape files

Aby przedstawiać informacje na mapach w programie R, należy wpierw wczytać obrysów map.

Takie dane są udostępnione w plikach o rozszerzeniu `.shp`, nazywanych z angielska *shape files* i z tej nazwy będę korzystał poniżej.

Pliki z kształtami województw Polski, które wykorzystam w tym odcinku, pochodzą z bazy danych GADM

<http://www.gadm.org> i mają pewne niewielkie błędy w kodowaniu, które ręcznie naprawimy.

Po pobraniu danych, w katalogu `dane/POL_adm` znajdują się kształty obszarów na różnych poziomach szczegółowości. Plik `POL_adm0.shp` to obrys całego kraju, `POL_adm1.shp` to obrysy województw a `POL_adm2.shp` to obrysy gmin.

Funkcją `readShapePoly()` wczytamy dane o obrysach województw do zmiennej `shp1`. Ale zanim zaczniemy pracować z tą funkcją, potrzebujemy wczytać kilka bibliotek. Jeżeli ich nie ma, należy je zainstalować.

```
library(maptools)
library(rgeos)
```

Wczytujemy dane o kształtach województw z plików *shape files*.

```
shp1 <- readShapePoly("POL_adm/POL_adm1.shp")
```

Korygujemy pliki z kształtami - shape files

Obiekt `shp1` to obiekt klasy `SpatialPolygonsDataFrame` i więcej o jego elementach, można dowiedzieć się z pliku pomocy `?SpatialPolygonsDataFrame`.

Przykładowo, zmienna `shp1@data` zawiera zbiór danych dołączony do kształtów, a w tym konkretnym pliku `shp1@data$VARNAME_1` zawiera nazwy województw.

Z uwagi na błąd w kodowaniu musimy w nim poprawić nazwę dla województwa Łódzkiego.

```
## W zmiennej VARNAME_1 znajdują się nazwy woj  
shp1@data$VARNAME_1 <- as.character(shp1@data$  
shp1@data$VARNAME_1  
  
## [1] "Wielkopolskie"                 "Kujawsko-Pomorskie"  
## [4] "L\xf3dzkie"                     "Dolnoslaskie"  
## [7] "Lubuskie"                      "Mazowieckie"  
## [10] "Podlaskie"                     "Pomorskie"  
## [13] "Podkarpackie"                  "Swietokrzyskie"  
## [16] "Zachodniopomorskie"
```

Poprawiamy kodowanie polskich znaków w województwie Łódzkim.

Uwaga! W zależności od pliku shp i kodowania, województwo Łódzkie będzie wyświetlało się na pozycji 1 lub 4. W poniższym przykładzie zakładamy, że województwo Łódzkie jest na pozycji 4, jeżeli jest inaczej należy zmienić indeks.

```
shp1@data$VARNAME_1[4] <- "Lodzkie"
```

Rysowanie mapy województw

Dane wczytane przez funkcję `readShapePoly()` mają specyfczną strukturę - listy wektorów, będących obrysami obszarów. Jedno województwo to nie zawsze obrys jednego obszaru, jeżeli mamy województwa nadmorskie, oraz chcemy uwzględnić wyspy, to na jedno województwo będą się składały też obrysy każdej z wysp.

Aby można było takie dane narysować w pakiecie `ggplot2` należy je wpierw przetransportować do formatu tabelarycznego. Najłatwiej to zrobić funkcją `fortify()`. Zmienia ona format danych na tabelaryczny. W poniższym przykładzie nowy format dla obrysów województw jest zapisany w zmiennej `shp1f`.

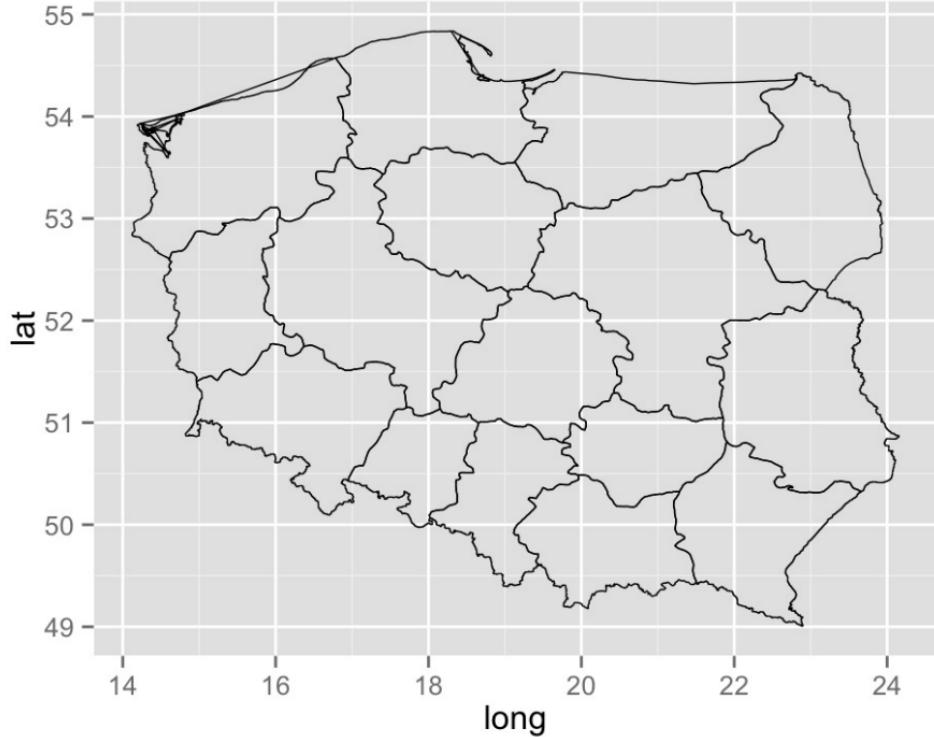
Pierwsze dwie kolumny tej tabeli to współrzędne geograficzne punktów na obrysie każdego województwa.

```
library(ggplot2)
## fortify zmienia format danych na zgodny z p:
shp1f <- fortify(shp1, region = "VARNAME_1")
head(shp1f, 4)
```

##	long	lat	order	hole	piece
## 1	16,16883	51,66089	1	FALSE	1 Dolno:
## 2	16,19236	51,65255	2	FALSE	1 Dolno:
## 3	16,22376	51,65711	3	FALSE	1 Dolno:
## 4	16,24667	51,66169	4	FALSE	1 Dolno:

Mając dane w formacie tabelarycznym, możemy je narysować, np. używając geometrii `geom_path`. Tak jak na poniższym przykładzie.

```
ggplot() +  
  geom_path(data=shplf, aes(x=long, y=lat, gro1
```



Rysowanie mapy województw - `geom_map()`

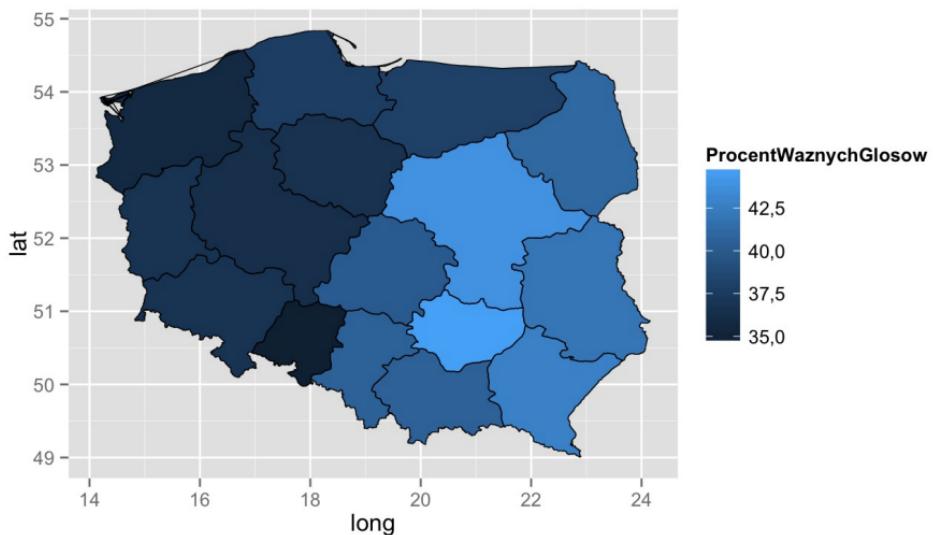
Geometrii `geom_path()` będziemy używać do rysowania obrysów województw.

Ale do zaznaczenia wartości liczbowych na mapie za pomocą koloru wypełnienia użyjemy geometrii `geom_map()`.

Argument `map=` tej geometrii wskazuje obiekt przedstawiający województwa (tutaj `shp1f`). Argument `data=` wskazuje na obiekt z danymi liczbowymi, czyli nasze dane o mandatach w sejmikach. W mapowaniach dla geometrii `geom_map()` określa się też cechę `map_id=` aby wskazać nazwy województw oraz `fill=` by wskazać która zmienna ma być przedstawiona kolorem wypełnienia.

Poniższy przykład rysuje kartogram, w którym informacja o procencie oddanych ważnych głosów przedstawiamy kolorem wypełnienia

```
## pusty zrąb wykresu
ggplot() +
  # rysowanie wypełnień
  geom_map(data=mandatySejmik2014, aes(map_id=id),
            map=shp1f) +
  # rysowanie obrysów na czarno
  geom_path(data=shp1f, aes(x=long, y=lat, group=id))
```



Rysowanie mapy

Z rysowaniem map związanego jest pojęcie projekcji, czyli sposobu odwzorowania trójwymiarowej powierzchni globu na dwuwymiarowej powierzchni wykresu.

Mapy można rysować w różnych projekcjach, nie ma jednej idealnej. Niektóre zachowują lepiej pola, inne kąty, inne długości odcinków. Listę projekcji z opisem właściwości można znaleźć pod adresem http://en.wikipedia.org/wiki/List_of_map_projections.

W programie R, w pakiecie `ggplot` możemy korzystać z większości popularnych projekcji. Aby wybrać którąś z

nich, wystarczy dodać wywołanie funkcji `coord_map()`.

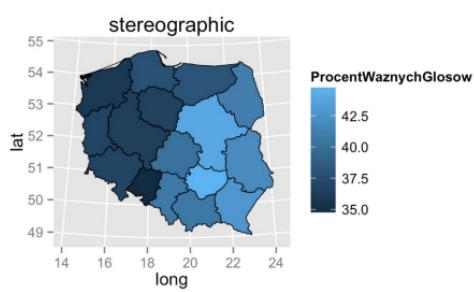
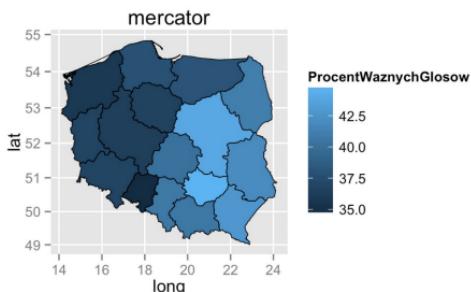
Poniżej dwa przykładowe wykresy, różniące się projekcjami. Ponieważ w skali globu obszar Polski nie jest duży, różnice pomiędzy mapami są raczej subtelne.

Przygotowujemy wykres i wczytujemy go do zmiennej `mapka`.

```
mapka <- ggplot() +  
  geom_map(data=mandatySejmik2014, aes(map_id=  
    map=shplf) +  
  geom_path(data=shplf, aes(x=long, y=lat, g:
```

Przekształcamy mapę, stosując różne projekcje.

```
## projekcja mercator (odwzorowanie walcowe równej  
## zachowuje kąty pomiędzy równoleżnikami a południkami  
mapka + coord_map(projection="mercator") + ggt:  
## projekcja stereographic (rzut stereograficzny)  
mapka + coord_map(projection="stereographic") -
```



Rysowanie mapy - pracujemy nad szczegółami

Narysowaliśmy już kilka kartogramów. Ale nie są one jeszcze najpiękniejsze. Jak je modyfikować?

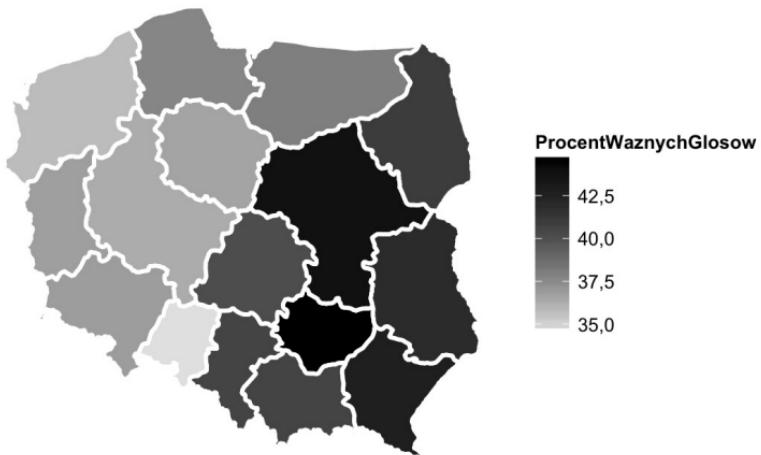
Mapy wykonane z pakietem `ggplot2` modyfikuje się podobnie jak inne wykresy w pakiecie `ggplot2`.

Na poniższym przykładzie przedstawiamy kilka przykładowych zmian. Funkcją `theme_bw()` zmieniamy domyślny motyw wykresu, funkcją

`scale_fill_gradient()` zmieniamy sposób reprezentacji zmiennej za pomocą koloru wypełniania, a dodatkowymi argumentami funkcji `theme()` usuwamy wszystkie dodatkowe elementy wykresu.

```
ggplot() +  
  geom_map(data=mandatySejmik2014, aes(map_id =  
    map=shplf) +  
  geom_path(data=shplf, aes(x=long, y=lat, group=  
# określamy projekcje mercator  
  coord_map(projection="mercator") +  
# zmieniamy motyw graficzny, usuwając szare tła  
  theme_bw() +  
# zmienią ilościową będziemy prezentować na mapie  
  scale_fill_gradient(low = "grey90", high = "black") +  
# dodatkowe elementy motywów usuwają wszystko  
  theme(axis.ticks = element_blank(),  
        panel.border = element_rect(colour = "black",  
        axis.title.x = element_text(size = 12, face = "bold"),  
        axis.title.y = element_text(size = 12, face = "bold"),  
        axis.text.x = element_text(size = 10),  
        axis.text.y = element_text(size = 10)) +  
# dodatkowe elementy motywów usuwają wszystko  
  theme(panel.grid.major = element_line(colour = "black"),  
        panel.grid.minor = element_line(colour = "black")) +  
# dodatkowe elementy motywów usuwają wszystko  
  theme(panel.background = element_rect(colour = "white",  
        fill = "white")) +  
# dodatkowe elementy motywów usuwają wszystko  
  theme(plot.background = element_rect(colour = "white",  
        fill = "white")) +  
# dodatkowe elementy motywów usuwają wszystko  
  theme(plot.title = element_text(size = 14, face = "bold")) +  
# dodatkowe elementy motywów usuwają wszystko  
  theme(plot.subtitle = element_text(size = 12, face = "bold")) +  
# dodatkowe elementy motywów usuwają wszystko  
  theme(plot.caption = element_text(size = 10, face = "bold")) +  
# dodatkowe elementy motywów usuwają wszystko  
  theme(plot.tag = element_text(size = 10, face = "bold"))
```

```
panel.grid.minor=element_blank(), pane  
axis.title.x = element_blank(), axis
```



Rysowanie mapy - zmienna jako wielkość punktu.

Nie zawsze przedstawianie danych za pomocą koloru wypełnienia to dobry pomysł.

Alternatywny sposób prezentacji informacji na mapie to np. wielkość punktu.

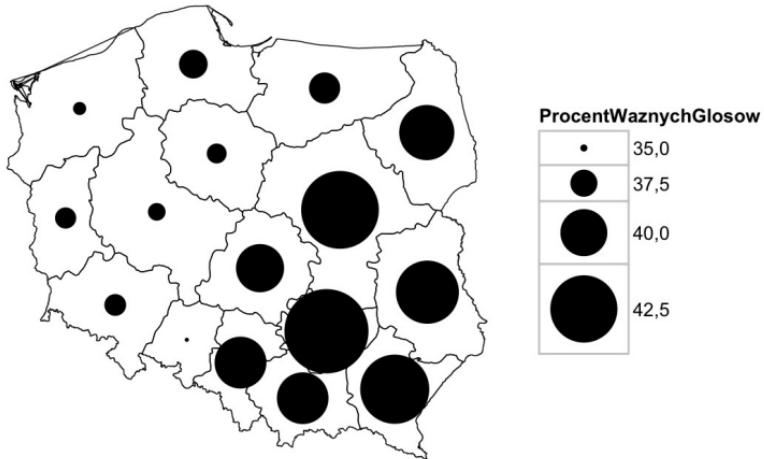
Do rysowania punktów można wykorzystać geometrię `geom_point()`, wskazując jako współrzędne punktu

środki województw, a wielkość punktu uzależniamy od zmiennej `ProcentWaznychGlosow`.

Aby punkty były bardziej widoczne powiększamy je modyfikując mapowanie za pomocą funkcji `scale_size_continuous()`.

Należy być ostrożnym z takimi zmianami, w tym przypadku różnica w wielkości punktów przerysowuje rzeczywiste różnice wynikające z liczb, które te wielkości prezentują.

```
ggplot() +  
  geom_point(data=mandatySejmik2014, aes(x=long, y=lat, size=ProcentWaznychGlosow)) +  
  geom_path(data=shp1f, aes(x=long, y=lat, group=group)) +  
  # określamy projekcje mercator  
  coord_map(projection="mercator") +  
  # zmieniamy motyw graficzny, usuwając szare tła  
  theme_bw() +  
  # zmienią ilościową będziemy prezentować na osi y  
  scale_size_continuous(range=c(1,20)) +  
  theme(axis.ticks = element_blank(), panel.grid.major=element_line(), panel.grid.minor=element_line(), axis.title.x = element_text(), axis.title.y = element_text(), axis.text.x = element_text(), axis.text.y = element_text(), axis.line = element_line(), panel.border=element_rect(), panel.background=element_rect())
```



Rysowanie mapy - zmienna jako długość paska

Jeszcze innym sposobem prezentacji danych liczbowych na mapach są długości słupków.

Aby dorysować informację o mandatach zdobytych przez PO wykorzystamy geometrię `geom_rect()`.

Należy do niej podać współrzędne `xmin` i `xmax` oznaczające początek i koniec słupka oraz `ymin` i `ymax` zależne od wartości zmiennej `PO`.

```
ggplot() +
```

```
geom_path(data=shplf, aes(x=long, y=lat, group=group), color="black", fill="white")
geom_rect(data=mandatySejmik2014, aes(xmin=lon, xmax=lon+0.05, ymin=lat, ymax=lat+0.05), fill="white", color="black", size=0.5)
# określamy projekcję mercator
coord_map(projection="mercator") +
# zmieniamy motyw graficzny, usuwając szare tła
theme_bw() +
theme(axis.ticks = element_blank(), panel.background = element_rect(fill="white", color="black", size=1), axis.title.x = element_blank(), axis.title.y = element_blank(), axis.text.x = element_blank(), axis.text.y = element_blank(), panel.grid.minor=element_blank(), panel.grid.major=element_line(color="black", size=0.5))
```



Rysowanie mapy - zmienna jako długość paska

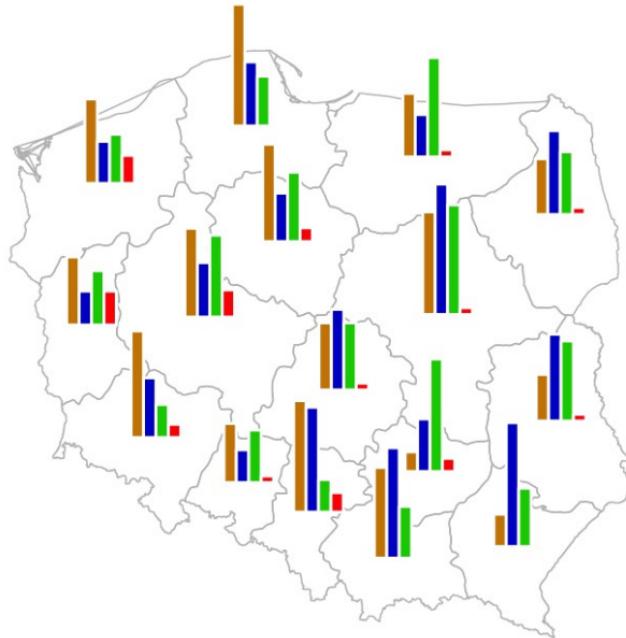
Jeżeli udało się narysować na mapie wyniki dla jednej partii, możemy dodać kolejne warstwy z wynikami dla kolejnych partii.

Na poniższym wykresie realizujemy ten pomysł.

Dodajemy słupki prezentujące liczbę zdobytych mandatów przez PiS, PSL i SLD, każdą partię oznaczamy innym kolorem. Łatwiej dzięki temu zauważać, która partia wygrała w którym województwie i z jaką przewagą.

```
ggplot() +  
  geom_path(data=shplf, aes(x=long, y=lat, group=  
    # warstwa z paskami, prezentująca liczbę mandatów  
    geom_rect(data=mandatySejmik2014, aes(xmin=lon, xmax=lon+  
      ymin=lat, ymax=lat+  
    # warstwa z paskami, prezentująca liczbę mandatów  
    geom_rect(data=mandatySejmik2014, aes(xmin=lon, xmax=lon+  
      ymin=lat, ymax=lat+  
    # warstwa z paskami, prezentująca liczbę mandatów  
    geom_rect(data=mandatySejmik2014, aes(xmin=lon, xmax=lon+  
      ymin=lat, ymax=lat+  
    # warstwa z paskami, prezentująca liczbę mandatów  
    geom_rect(data=mandatySejmik2014, aes(xmin=lon, xmax=lon+  
      ymin=lat, ymax=lat+  
    # inne graficzne parametry, tak jak w poprzednim wykresie  
    coord_map(projection="mercator") +  
    theme_bw() +  
    scale_size_continuous(range=c(1,20)) +
```

```
theme(axis.ticks = element_blank(),      pane
      axis.text.x = element_blank(),      axis
      panel.grid.minor=element_blank(),   pane
      axis.title.x = element_blank(),     axis
```



Co dalej

- Świecone omówienie możliwości prezentowania danych na mapach zaprezentowanie jest w tym artykule *ggmap: Spatial Visualization with ggplot2* <http://journal.r-project.org/archive/2013-1/kahle->

wickham.pdf

- Krótkim streszczeniem, z ciekawymi przykładami będzie ten blog *Making Maps in R*
<http://www.kevjohnson.org/making-maps-in-r/>
- Wiele przydatnych wskazówek można również znaleźć w internetowej książce „*Cookbook for R*”
<http://www.cookbook-r.com/Graphs/>

Historia i współczesność

Przemysław Biecek @ Uniwersytet Warszawski

sezon 2 / odcinek 6

pogRomcy danych

- [Historia wizualizacji danych](#)
- [Iluzje, percepcja obrazu i kolory](#)
- [Info-pomyłka](#)
- [Jak dobierać kolory?](#)
- [Jak pokazywać różne liczby?](#)

Pierwsze pięć odcinków poświęconych było pakietowi `ggplot2`. Znamy już narzędzia, pozwalające na tworzenie dobrych wykresów w programie R.

Teraz potrzebujemy informacji co charakteryzuje dobry wykres i na jakie pułapki uważyć tworząc wizualizacje.

Kolejne pięć podrozdziałów to linki do filmów umieszczonych na serwisie YouTube, które omawiają najistotniejsze zagadnienia dotyczące tworzenia wykresów.

Historia wizualizacji danych

Ten odcinek jest też dostępny w formacie video na kanale youtube <https://youtu.be/vyoCoXXIIA8>

Więcej informacji

- Esej o historii wizualizacji danych
<http://biecek.pl/Eseje/indexHistoria.html>
- Strona www Edwarda Tuftego, autora wielu książek o wizualizacji danych
<http://www.edwardtufte.com/tufte/>
- Bardzo krótka historia infografiki
<http://inspirationfeed.com/articles/design-articles/a-brief-history-of-infographics-and-data-visualization/>

Iluzje, percepcja obrazu i kolory

Ten odcinek jest też dostępny w formacie video na kanale youtube <https://youtu.be/6a7FTdykk9s>

Więcej informacji

- Esej o iluzjach optycznych
<http://biecek.pl/Eseje/indexObraz.html>
- Strona www Edwarda Tuftego, autora wielu książek o wizualizacji danych

<http://www.edwardtufte.com/tufte/>

- Lista dziesiątek iluzji optycznych
<http://www.michaelbach.de/ot/>

Info-pomyłka

Ten odcinek jest też dostępny w formacie video na kanale youtube <https://youtu.be/7AILrHsyAT8>

Więcej informacji

- Esej o błędnych wykresach
<http://biecek.pl/Eseje/indexPomylka.html>
- Strona www Edwarda Tuftego, autora wielu książek o wizualizacji danych
<http://www.edwardtufte.com/tufte/>
- Blog o śmieciowych wykresach
<http://junkcharts.typepad.com/>

Jak dobierać kolory?

Ten odcinek jest też dostępny w formacie video na kanale youtube <https://youtu.be/LoPIIuBuzbo>

Więcej informacji

- Esej o kolorach

<http://biecek.pl/Eseje/indexKolory.html>

- Strona www Edwarda Tuftego, autora wielu książek o wizualizacji danych
<http://www.edwardtufte.com/tufte/>
- Lista schematów kolorów <http://colorbrewer2.org/>

Jak pokazywać różne liczby?

Ten odcinek jest też dostępny w formacie video na kanale youtube <https://youtu.be/bbtJfM2PbIs>

Więcej informacji

- Esej o sposobach przedstawiania różnych liczb
<http://biecek.pl/Eseje/indexKuchnia.html>
- Strona www Edwarda Tuftego, autora wielu książek o wizualizacji danych
<http://www.edwardtufte.com/tufte/>
- Strona o interesujących historiach ukrytych w danych
<http://freakonomics.com/>

Próba, populacja, estymacja, testowanie

Przemysław Biecek @ Uniwersytet Warszawski

*sezon 2 / odcinek 7
pogRomcy danych*

- [O czym jest ten odcinek](#)
- [Kluczowe koncepcje - zależność](#)
- [Kluczowe koncepcje - losowość](#)
- [Kluczowe koncepcje - dokładność pomiaru](#)
- [Kluczowe koncepcje - próba a populacja](#)
- [Kluczowe koncepcje - estymacja i testowanie](#)
- [Spis treści](#)

O czym jest ten odcinek

Najbliższe odcinki przedstawią kilka wybranych technik analizy danych ilościowych i jakościowych.

Techniki te będą przedstawione na kilku poziomach.

1. Poziom narzędziowy. Dla każdej metody pokażemy na jednym lub dwóch przykładach, jak wykonać określoną analizę w programie R, oraz jak interpretować wyniki.
2. Poziom koncepcyjny. Dla każdej metody pokażemy w jaki sposób przedstawiane algorytmy działają, jakie kroki są wykonywane oraz po co są wykonywane.
3. Poziom problemowy. Dla każdej metody pokażemy w jaki sposób przedstawiane algorytmy są związane z problemem, który chcemy rozwiązać.

Aby sprawnie przedstawiać wybrane algorytmy analizy danych, musimy zapoznać się wpierw z kilkoma pojęciami / koncepcjami stojącymi za statystyczną analizą danych.

Kluczowe koncepcje - zależność

Jednym z pojęć wykorzystywanym podczas analizy danych jest *zależność*. Najczęściej dotyczy zależności pomiędzy dwoma cechami.

Kiedy mamy do czynienia z zależnością pomiędzy zmiennymi? Jeżeli dla różnych wartości jednej zmiennej z różną częstością obserwujemy wartości drugiej zmiennej.

Przykładowo, dla dorosłych osób jest zależność pomiędzy

wzrostem a płcią.

Zależności są zazwyczaj symetryczne, a więc:

- Mężczyźni są średnio wyżsi od kobiet, czyli gdy obserwujemy mężczyznę to częściej związane są z nim wyższe wartości pomiaru wzrostu,
- Wyższe osoby są częściej mężczyznami, czyli gdy obserwujemy wysoką osobę to częściej jest ona mężczyzną.

(Nie zawsze mamy do czynienia z symetrią).

Zauważmy że:

1. Zależność jest związana z pewną tendencją, ale nie daje 100% funkcyjnego związku. To znaczy, nie jest tak, że wszystkie osoby poniżej jakiegoś wzrostu są *zawsze* kobietami. Można spotkać bardzo wysokie kobiety i bardzo niskich mężczyzn ale nie przeczy to ogólnej prawidłowości, że średnio mężczyźni są wyżsi.
2. Zależność jest widoczna dla całej populacji, ale nie musi być prawdziwa dla podgrup. Gdybyśmy porównywali dwie grupy kobiet i mężczyzn i tak się złożyło, że jedną z tych grup byłyby siatkarki (wysokie kobiety) a drugą grupą byliby dżokeje (zazwyczaj niscy mężczyźni) dla tych grup nie

obserwowałyśmy zależności widocznej dla całej populacji. Jeżeli więc obserwujemy podgrupy to nie zawsze w tych podgrupach widoczna jest zależność charakterystyczna dla całej populacji.

3. Zależność może dotyczyć dwóch zmiennych ilościowych (np. waga i wzrost, wyższe osoby są zazwyczaj cięższe) jak i jakościowych (np. kolor włosów i kolor oczu).
4. Siła zależności pomiędzy zmiennymi stopniuje się. Dwie zmienne mogą być bardzo słabo zależne lub silnie zależne. Siła zależności odzwierciedla ilość informacji, którą zdobywamy o jednej zmiennej obserwując drugą.

Kluczowe koncepcje - losowość

Choć jesteśmy przyzwyczajeni do myślenia o wartościach jak do pewnych ustalonych liczbach, to nie zawsze jest to możliwe.

Przykładowo, czy można określić jaki jest średni wzrost mężczyzn w Polsce?

Gdybyśmy wybraли losowo 1000 mężczyzn moglibyśmy policzyć ich wzrost i policzyć z nich średnią. Ale gdybyśmy wybrali inne 1000 osób możemytrzymać inną średnią. Być może bardzo bliską do pierwszej średniej,

ale raczej nie identyczną. Tak więc na bazie próby możemy wnioskować coś o średnim wzroście ale to wnioskowanie obarczone jest pewną losowością/dokładnością wynikającą z przypadkowości wyboru tych 1000 osób.

Co jednak gdybyśmy zmierzyli wszystkich Polaków? Czy wtedy otrzymalibyśmy jedną stałą wartość za każdym razem? Też niekoniecznie. Rano jesteśmy wyżsi niż wieczorem. Średni wzrost zależy więc od godziny pomiaru. Jeżeli godzinę pomiaru wybieramy przypadkowo to możemy otrzymać różne średnie nawet dla tych samym osób.

Co jednak gdybyśmy zmierzyli wszystkich Polaków dokładnie o 10 rano? Czy wtedy otrzymalibyśmy jedną stałą wartość? Też nie, nie ma czegoś takiego jak uniwersalny zbiór wszystkich Polaków. W nocy ktoś umiera, ktoś się rodzi. Zmieniają się proporcje, być może nieznacznie ale zawsze, pomiędzy liczbą dzieci a dorosłych, a więc zbiór wszystkich Polaków nieustannie się zmienia a tym samym zmienia się ich średni wzrost.

Oczywiście te zmiany są nieduże, ale pokazują, że nie każda wielkość jest mierzalna ze 100% dokładnością. Niektóre wielkości, takie jak średni wzrost nieustannie fluktują i gdy o nich myślimy powinniśmy pamiętać o tych fluktuacjach.

Kluczowe koncepcje - dokładność pomiaru

Pewne cechy, takie jak wzrost są szacowane jedynie z określoną dokładnością ponieważ koncepcja średniego wzrostu ma taki charakter.

Ale są też cechy, które (w co wierzymy) mają pewną stałą, choć niekoniecznie znaną, wartość ale my możemy ją ocenić jedynie z pewną dokładnością.

Przykładowo, gdy ważymy się wagą, w chwili gdy stajemy na wadze mamy jakąś określoną masę. Ale w zależności od różnych czynników, takich jak temperatura, sprawność wagi, możemy otrzymać różne wyniki pomiaru. Standardowe wagi mają dokładność rzędu +- 2kg więc dla tej samej osoby na różnych wagach moglibyśmy otrzymać nieznacznie różne odczyty.

Dokładność pomiaru dotyczy też sytuacji, w której pomiaru dokonuje się nie mechanicznym urządzeniem, ale np. ankietą lub sondażem. Tak jak np. w przypadku pomiaru preferencji politycznych. W tym przypadku błąd pomiaru wynika z niemożności przeprowadzenia sondażu na wszystkich Polakach, ale konieczności przeprowadzenia sondażu na próbie o zazwyczaj niewielkiej wielkości.

Kluczowe koncepcje - próba a populacja

Zazwyczaj stosujemy narzędzia analizy danych bo poznać pewną ogólną prawidłowość. Uniwersalną regułę.

Gdy prowadzimy analizy skuteczności leku na 100 pacjentach robimy to nie po to by ocenić czy tych konkretnych 100 pacjentów zareaguje na terapię, ale by przenieść wnioski na większą populację. Chcemy wiedzieć czy lekarstwo jest skuteczne.

Zależy nam więc na tym by wnioski uzyskane z próby mogły być uogólnione na całą populację. Z drugiej strony zdajemy sobie sprawę z przypadkowości wynikającej z próby, która nie zawsze może odzwierciedlać zależności z całej populacji.

W analizie danych wiele uwagi poświęca się określeniu na ile wyniki są dokładne a na ile ich dokładność zależy od wielkości i reprezentatywności próby.

Kluczowe koncepcje - estymacja i testowanie

W klasycznej analizie danych często wyróżnia się dwa

rodzaje algorytmów: estymację (szacowanie) i testowanie.

Estymacja to działanie mające na celu oszacowanie na podstawie próby ile wynoszą pewne nieznane parametry populacji. Przykładowo,

- na bazie 1000 ankietowanych chcemy ocenić jakie jest poparcie dla partii X.
- na bazie partii 100 butów chcemy ocenić jak wygląda ich wytrzymałość na ścieranie.
- na przykładzie 1000 samochodów chcemy ocenić jak wygląda ich bezawaryjność.

Wynikiem jest oszacowanie określonej wartości oraz informacja o tym jak dokładny jest ten szacunek.

Testowanie, to działanie mające na celu ocenę czy pewna wartość w populacji spełnia określony warunek.

Przykładowo

- możemy testować czy poparcie dla partii X jest większe niż dla partii Y,
- możemy testować, czy średnia wytrzymałość butów pozwala na ich użytkownie przez 2 sezony,
- możemy testować, czy 80% samochodów nie zepsuje się przez najbliższe 5 lat.

Przedstawiając w kolejnych odcinkach różne zagadnienia analizy danych będziemy pokazywać metody służące do estymacji / oszacowania określonej wartości oraz do testowania określonych warunków dla tej wartości.

Spis treści

Materiały o modelowaniu, zostały podzielone na następujące odcinki

Wprowadzający do eksploracji danych

- eksploracja danych

Dwa odcinki poświęcone analizie trendu

- zmienne ilościowe - analiza trendu
- zmienne ilościowe - testowanie trendu

Odcinek poświęcony analizie średnich

- zmienne ilościowe - porównanie dwóch grup

Dwa odcinki poświęcone regresji liniowej, przedziałowej i multiplikatywnej

- zmienne ilościowe - regresja liniowa
- zmienne ilościowe - regresja przedziałowo liniowa

Dwa odcinki poświęcone zmiennym jakościowym

- zmienne jakościowe - tabele 2x2
- zmienne jakościowe - tabele kontyngencji

Eksploracja danych

Przemysław Biecek @ Uniwersytet Warszawski

sezon 2 / odcinek 8

pogRomcy danych

- O czym jest ten odcinek?
- Baza danych o filmach i serialach
- Seriale
- Tabela liczebności
- Breaking Bad
- Co się dzieje z ocenami odcinków?
- Co się dzieje z ocenami odcinków?
- Co się dzieje z ocenami odcinków?
- Sortowanie
- Jak oceniano ten serial?
- Pięć liczb - wykres pudełko wąsy
- Pięć liczb - wykres pudełko wąsy
- Pięć liczb - wykres pudełko wąsy
- Histogram
- Co może dać nam histogram czego nie dają inne miary?
- Podsumowanie instrukcji R
- Podsumowanie instrukcji R

- [Zadania](#)

O czym jest ten odcinek?

Zanim zacznie się budować modele, zanim rozpocznie się zaawansowaną analizę danych, wcześniej zawsze warto przyjrzeć się bliżej danym.

Takie przyglądarki się danym, często bez konkretnego celu, nazywa się eksploracją.

Pierwsze spojrzenie na dane. Sprawdzenie z jakimi zmiennymi, z jakimi zakresami zmiennych, jakimi relacjami pomiędzy zmiennymi mamy do czynienia.

Eksplorację często zaczyna się od przyjrzenia każdej ze zmiennych osobno, ewentualnie parami, bazując na prostych podsumowaniach graficznych lub tabelarycznych.

W tym odcinku pokażemy jak przedstawiać rozkład zmiennej ilościowej oraz zmiennej jakościowej, za pomocą kilku popularnych statystyk.

Dla zmiennej ilościowej (liczbowej) przedstawimy

- średnią, medianę,
- pięć liczb Tukeya,
- wykres ramka wąsy, histogram.

Dla zmiennej jakościowej (z kategoriami) przedstawimy

- tablice kontyngencji,
 - sortowanie.
-

Baza danych o filmach i serialach

W bazie danych o filmach IMDB (Internet Movie Database) znaleźć można różne ciekawe dane o serialach. W szczególności średnie oceny użytkowników wystawione kolejnym odcinkom serialu.

Przykładowo, na stronie

[http://www.imdb.com/title/tt0903747/episodes?
ref_=ttep_ql_4](http://www.imdb.com/title/tt0903747/episodes?ref_=ttep_ql_4)

znajdują się oceny serialu *Breaking Bad*.

Na potrzeby tego odcinka pobraliśmy dane z Internetu, oczyściliśmy je i udostępniliśmy w zbiorze danych `serialsIMDB` w pakiecie `PogromcyDanych`, który towarzyszy temu kursowi.

Cały zbiór danych jest dosyć duży, wszystkich odcinków wszystkich seriali jest ponad 20 tysięcy wierszy, każdy opisany przez 8 kolumn/zmiennych.

Wczytujemy zbiór danych, jeżeli nie jest zainstalowany to należy go zainstalować poleceniem

```
install.packages("PogromcyDanych").
```

```
library(PogromcyDanych)
```

Sprawdzamy wielkość zbioru danych.

```
dim(serialeIMDB)
```

```
## [1] 20122      8
```

Wyświetlmy pierwsze 6 wierszy. W kolejnych kolumnach jest nazwa serialu, nazwa odcinka, numer sezonu, numer odcinka, średnia ocena, liczba oddanych głosów, oraz identyfikator z bazy IMDB.

```
head(serialeIMDB)
```

##	id	serial	name	year	rating	votes	imdbID
## 1	1	Breaking Bad	P: Breaking Bad	2008	8.1	100000	tt0904943
## 2	2	Breaking Bad	Cat's in the Bag	2008	8.1	100000	tt0904943
## 3	3	Breaking Bad	...And the Bag's in the R	2008	8.1	100000	tt0904943
## 4	4	Breaking Bad	Cancer	2008	8.1	100000	tt0904943
## 5	5	Breaking Bad	Gray Matter	2008	8.1	100000	tt0904943
## 6	6	Breaking Bad	Crazy Handful of Nots	2008	8.1	100000	tt0904943

Serial

Każdy wiersz w zbiorze danych `serialeIMDB` opisuje oceny jednego odcinka dla każdego z przeszło 200 seriali.

Kolumna danych o nazwie `serial` jest zmienną jakościową, przyjmującą za wartości nazwy seriali. Listę poziomów zmiennej jakościowej można wyświetlić funkcją `levels()`.

Zapis `serialeIMDB$serial` oznacza odwołanie się do kolumny `serial` ze zbioru danych `serialeIMDB`.

Z jakimi serialami mamy do czynienia w tym zbiorze danych?

```
levels(serialeIMDB$serial)
```

```
## [1] "Breaking Bad"  
## [2] "Cosmos: A Space-Time Odyssey"  
## [3] "Planet Earth"  
## [4] "Game of Thrones"  
## [5] "True Detective"  
## [6] "The Wire"  
## [7] "Sherlock"  
## [8] "Cosmos"  
## [9] "The Sopranos"  
## [10] "Leyla ile Mecnun"  
## [11] "Firefly"  
## [12] "Arrested Development"  
## [13] "I, Claudius"  
## [14] "Avatar: The Last Airbender"  
## [15] "Life"  
## [16] "The Angry Video Game Nerd"  
## [17] "Dexter"  
## [18] "House of Cards"  
## [19] "Batman: The Animated Series"  
## [20] "Cowboy Bebop"
```

```
## [21] "Death Note"
## [22] "Freaks and Geeks"
## [23] "Friends"
## [24] "Rome"
## [25] "The Simpsons"
## [26] "Twin Peaks"
## [27] "Seinfeld"
## [28] "Monty Python's Flying Circus"
## [29] "Top Gear"
## [30] "Oz"
## [31] "House M.D."
## [32] "Attack on Titan"
## [33] "Undercover"
## [34] "Twilight Zone"
## [35] "Blackadder Goes Forth"
## [36] "The Daily Show with Jon Stewart"
## [37] "Deadwood"
## [38] "Only Fools and Horses...."
## [39] "QI"
## [40] "Doctor Who"
## [41] "South Park"
## [42] "Fawlty Towers"
## [43] "Archer"
## [44] "Dragon Ball Z"
## [45] "It's Always Sunny in Philadelphia"
## [46] "Six Feet Under"
## [47] "Suits"
## [48] "Fullmetal Alchemist Brotherhood"
## [49] "The Office"
## [50] "One Piece"
## [51] "Isler Güçler"
## [52] "The Shield"
## [53] "Battlestar Galactica"
## [54] "Downton Abbey"
## [55] "Black-Adder II"
```

```
## [56] "The Adventures of Sherlock Holmes"
## [57] "Curb Your Enthusiasm"
## [58] "Black Adder the Third"
## [59] "The Thick of It"
## [60] "Chappelle's Show"
## [61] "Spaced"
## [62] "Berserk"
## [63] "The Prisoner"
## [64] "The West Wing"
## [65] "The X-Files"
## [66] "Flight of the Conchords"
## [67] "Supernatural"
## [68] "Whose Line Is It Anyway?"
## [69] "Sons of Anarchy"
## [70] "Boardwalk Empire"
## [71] "The Walking Dead"
## [72] "Justified"
## [73] "Futurama"
## [74] "Adventure Time"
## [75] "Spartacus: War of the Damned"
## [76] "Mystery Science Theater 3000"
## [77] "Mad Men"
## [78] "Behzat Ç.: Bir Ankara Polisiyesi"
## [79] "The Legend of Korra"
## [80] "Fullmetal Alchemist"
## [81] "Dragonball"
## [82] "Father Ted"
## [83] "Black Mirror"
## [84] "The Colbert Report"
## [85] "The Originals"
## [86] "Friday Night Lights"
## [87] "Dragon Ball Z Kai"
## [88] "Dragon Ball"
## [89] "Star Trek: The Next Generation"
## [90] "Code Geass: Lelouch of the Rebellion"
```

```
## [91] "Peep Show"
## [92] "Shameless"
## [93] "Community"
## [94] "Modern Family"
## [95] "Homicide: Life on the Street"
## [96] "Jeeves and Wooster"
## [97] "The Big Bang Theory"
## [98] "Entourage"
## [99] "Louie"
## [100] "The Newsroom"
## [101] "Coupling"
## [102] "Garth Marenghi's Darkplace"
## [103] "Black Books"
## [104] "I'm Alan Partridge"
## [105] "Carnivàle"
## [106] "Samurai Champloo"
## [107] "Luther"
## [108] "Agatha Christie's Poirot"
## [109] "The Muppet Show"
## [110] "How I Met Your Mother"
## [111] "The Venture Bros."
## [112] "The Bugs Bunny Show"
## [113] "Summer Heights High"
## [114] "Lost"
## [115] "The IT Crowd"
## [116] "Prison Break"
## [117] "Naruto: Shippûden"
## [118] "Police Squad!"
## [119] "Neon Genesis Evangelion"
## [120] "Parks and Recreation"
## [121] "Mr. Bean"
## [122] "Terriers"
## [123] "Justice League"
## [124] "Red vs. Blue: The Blood Gulch Chronicle"
## [125] "Homeland"
```

```
## [126] "Young Justice"
## [127] "Scrubs"
## [128] "Vikings"
## [129] "The Bridge"
## [130] "Red Dwarf"
## [131] "Late Night with Conan O'Brien"
## [132] "MythBusters"
## [133] "Hellsing Ultimate"
## [134] "Ghost in the Shell: Stand Alone Comp."
## [135] "Rurouni Kenshin"
## [136] "X-Men"
## [137] "Fringe"
## [138] "Extras"
## [139] "24"
## [140] "I Love Lucy"
## [141] "Regular Show"
## [142] "Orange Is the New Black"
## [143] "Stargate SG-1"
## [144] "Hannibal"
## [145] "The Guild"
## [146] "FLCL"
## [147] "All in the Family"
## [148] "Boston Legal"
## [149] "M*A*S*H"
## [150] "Borgen"
## [151] "Trailer Park Boys"
## [152] "Psych"
## [153] "Californication"
## [154] "The Inbetweeners"
## [155] "Trigun"
## [156] "Family Guy"
## [157] "Foyle's War"
## [158] "American Horror Story"
## [159] "Misfits"
## [160] "The Avengers: Earth's Mightiest Heroes"
```

```
## [161] "Pushing Daisies"
## [162] "Blue Mountain State"
## [163] "Southland"
## [164] "Invader ZIM"
## [165] "The Adventures of Pete & Pete"
## [166] "Person of Interest"
## [167] "The League"
## [168] "Call the Midwife"
## [169] "My So-Called Life"
## [170] "Star Trek"
## [171] "Broadchurch"
## [172] "Life on Mars"
## [173] "Orphan Black"
## [174] "Veronica Mars"
## [175] "'Allo 'Allo!"
## [176] "The Wonder Years"
## [177] "The Mighty Boosh"
## [178] "Are You Afraid of the Dark?"
## [179] "Avrupa yakasi"
## [180] "Farscape"
## [181] "Rescue Me"
## [182] "Green Wing"
## [183] "Courage the Cowardly Dog"
## [184] "White Collar"
## [185] "Castle"
## [186] "The Boondocks"
## [187] "Almost Human"
## [188] "The Andy Griffith Show"
## [189] "Utopia"
## [190] "Get Smart"
## [191] "Party Down"
## [192] "The Adventures of Tintin"
## [193] "Metalocalypse"
## [194] "The Black Adder"
## [195] "Arrow"
```

```
## [196] "Ninjago: Masters of Spinjitzu"  
## [197] "Robin of Sherwood"  
## [198] "The Good Wife"
```

Tabela liczebności

Kolumna `serial` w `IMDB$serial` zawiera nazwy seriali. Te nazwy powtarzają się dla każdego serialu tyle razy, ile odcinków jest opisanych w zbiorze danych. Użyteczną funkcją do eksploracji zmiennych jakościowych jest tabela częstości. Przedstawia ona jak często określona wartość występuje w wektorze.

Tabelę częstości można wyznaczyć funkcją `table()`. Wynik tej funkcji można wyświetlić na ekranie lub, jak w poniższym przykładzie, zapisać do zmiennej.

```
tabela <- table(serialIMDB$serial)  
## pierwsze 6 wartości z wektora  
head(tabela)
```

```
##  
##                                Breaking Bad Cosmos: A Space  
##                                         62  
##                                Game of Thrones  
##                                         40
```

Tabelę liczebności dla 200 seriali trudno się przegląda. Jeżeli interesują nas najczęstsze lub najrzadsze wartości

to aby je odczytać warto tabelę liczebności posortować (funkcja `sort()`).

Po posortowaniu w sposób malejący (argument `decreasing = TRUE`) najczęstsze wartości znajdują się na początku a najrzadsze na końcu posortowanego wektora.

Ponad 500 odcinków dla Simpsonów? Robi wrażenie.

```
sort(tabela, decreasing = TRUE)
```

```
##                                     The Daily Show with Jon Stewart      1814
##                                     Doctor Who                      819
##                                     Dragon Ball Z                  589
##                                     Naruto: Shippûden       397
##                                     MythBusters                 258
##                                     M*A*S*H                     251
##                                     Family Guy                   240
##                                     Whose Line Is It Anyway?    214
##                                     Supernatural                208
##                                     All in the Family            208
##                                     The Office
```

##		201
##	Mystery Science Theater	3000
##		197
##	Adventure Time	
##		182
##	I Love Lucy	
##		181
##	House M.D.	
##		176
##	The Big Bang Theory	
##		173
##	Dragon Ball Z Kai	
##		168
##	Twilight Zone	
##		156
##	Dragon Ball	
##		153
##	The Angry Video Game Nerd	
##		141
##	Get Smart	
##		138
##	Futurama	
##		124
##	Homicide: Life on the Street	
##		122
##	The Muppet Show	
##		120
##	Lost	
##		118
##	It's Always Sunny in Philadelphia	
##		106
##	Boston Legal Re	
##		101
##	Fringe	
##		100

Dexter
96

Sons of Anarchy
95

Justice League
91

The Shield
89

The Sopranos
86

Mad Men
85

Californication
84

Prison Break
81

White Collar
81

Behzat Ç.: Bir Ankara Polisiyesi
76

X-Men
76

The Guild
72

Agatha Christie's Poirot
70

Arrested Development
68

Avrupa yakası
68

Trailer Park Boys
66

Fullmetal Alchemist Brotherhood
65

Six Feet Under

##		63
##	Avatar: The Last Airbender	62
##	Metalocalypse	62
##	The Wire	60
##	Arrow	58
##	Oz	56
##	The Boondocks	55
##	Louie	53
##	Shameless	52
##	Fullmetal Alchemist	51
##	Peep Show	49
##	Homeland	48
##	Young Justice	46
##	Downton Abbey	43
##	İsler Güçler	41
##	Blue Mountain State	40
##	The Adventures of Tintin	39
##	Death Note	37

##	Spartacus: War of the Damned	
##		34
##	Ninjago: Masters of Spinjitzu	
##		34
##	Ghost in the Shell: Stand Alone Complex	
##		33
##	Borgen	
##		30
##	Coupling	
##		28
##	Invader ZIM	
##		28
##	Cowboy Bebop	
##		27
##	House of Cards	
##		26
##	Neon Genesis Evangelion	
##		26
##	Hannibal	
##		26
##	Berserk	
##		25
##	The Newsroom	
##		25
##	The Thick of It	
##		24
##	Robin of Sherwood	
##		24
##	Rome	
##		22
##	Pushing Daisies	
##		22
##	The Bridge	
##		20
##	Party Down	

##		20
##	Vikings	
##		19
##	Freaks and Geeks	
##		18
##	The Inbetweeners	
##		18
##	The Prisoner	
##		17
##	Terriers	
##		16
##	Luther	
##		15
##	Spaced	
##		14
##	Cosmos	
##		13
##	The Adventures of Sherlock Holmes	
##		13
##	Broadchurch	
##		13
##	Fawlty Towers	
##		12
##	Utopia	
##		12
##	Sherlock	
##		11
##	True Detective	
##		8
##	Black Mirror	
##		7
##	Blackadder Goes Forth	
##		6
##	Black Adder the Third	
##		6

Breaking Bad

Cały zbiór danych o serialach ma ponad 20 tysięcy wierszy. Praca na tak dużym zbiorze danych, o wszystkich serialach jest z różnych powodów trudna. Między innymi dlatego, że nie jesteśmy w stanie ogarnąć wzrokiem wszystkich 20 tysięcy wierszy.

Na potrzeby dalszych przykładów wybierzmy z tego zbioru danych jeden serial, któremu przyjrzymy się bliżej. Na tych slajdach będzie to *Breaking Bad*, ale warto samodzielnie poeksperymentować również z innymi serialami.

Funkcją `filter()` wybieramy tylko wiersze dla serialu *Breaking Bad*.

```
BreakingBad <- filter(serialIMDB, serial == "I
```

O ilu odcinkach dostępna jest informacja dla tego serialu?

```
dim(BreakingBad)
```

```
## [1] 62 8
```

A jak wyglądają dane dla serialu *Star Trek: The Next Generation*?

```
filter(serialeIMDB, serial == "Star Trek: The Next Generation")
```

#	id	serial
## 1	1	Star Trek: The Next Generation
## 2	2	Star Trek: The Next Generation
## 3	3	Star Trek: The Next Generation
## 4	4	Star Trek: The Next Generation
## 5	5	Star Trek: The Next Generation
## 6	6	Star Trek: The Next Generation
## 7	7	Star Trek: The Next Generation
## 8	8	Star Trek: The Next Generation
## 9	9	Star Trek: The Next Generation
## 10	10	Star Trek: The Next Generation
## 11	11	Star Trek: The Next Generation
## 12	12	Star Trek: The Next Generation
## 13	13	Star Trek: The Next Generation
## 14	14	Star Trek: The Next Generation
## 15	15	Star Trek: The Next Generation
## 16	16	Star Trek: The Next Generation
## 17	17	Star Trek: The Next Generation
## 18	18	Star Trek: The Next Generation
## 19	19	Star Trek: The Next Generation
## 20	20	Star Trek: The Next Generation
## 21	21	Star Trek: The Next Generation
## 22	22	Star Trek: The Next Generation
## 23	23	Star Trek: The Next Generation
## 24	24	Star Trek: The Next Generation
## 25	25	Star Trek: The Next Generation
## 26	26	Star Trek: The Next Generation
## 27	27	Star Trek: The Next Generation
## 28	28	Star Trek: The Next Generation
## 29	29	Star Trek: The Next Generation

30 30 Star Trek: The Next Generation
31 31 Star Trek: The Next Generation
32 32 Star Trek: The Next Generation
33 33 Star Trek: The Next Generation
34 34 Star Trek: The Next Generation
35 35 Star Trek: The Next Generation
36 36 Star Trek: The Next Generation
37 37 Star Trek: The Next Generation
38 38 Star Trek: The Next Generation
39 39 Star Trek: The Next Generation
40 40 Star Trek: The Next Generation
41 41 Star Trek: The Next Generation
42 42 Star Trek: The Next Generation
43 43 Star Trek: The Next Generation
44 44 Star Trek: The Next Generation
45 45 Star Trek: The Next Generation
46 46 Star Trek: The Next Generation
47 47 Star Trek: The Next Generation
48 48 Star Trek: The Next Generation
49 49 Star Trek: The Next Generation
50 50 Star Trek: The Next Generation
51 51 Star Trek: The Next Generation
52 52 Star Trek: The Next Generation
53 53 Star Trek: The Next Generation
54 54 Star Trek: The Next Generation
55 55 Star Trek: The Next Generation
56 56 Star Trek: The Next Generation
57 57 Star Trek: The Next Generation
58 58 Star Trek: The Next Generation
59 59 Star Trek: The Next Generation
60 60 Star Trek: The Next Generation
61 61 Star Trek: The Next Generation
62 62 Star Trek: The Next Generation
63 63 Star Trek: The Next Generation
64 64 Star Trek: The Next Generation

65 65 Star Trek: The Next Generation
66 66 Star Trek: The Next Generation
67 67 Star Trek: The Next Generation
68 68 Star Trek: The Next Generation
69 69 Star Trek: The Next Generation
70 70 Star Trek: The Next Generation
71 71 Star Trek: The Next Generation
72 72 Star Trek: The Next Generation
73 73 Star Trek: The Next Generation The I
74 74 Star Trek: The Next Generation The I
75 75 Star Trek: The Next Generation
76 76 Star Trek: The Next Generation
77 77 Star Trek: The Next Generation
78 78 Star Trek: The Next Generation
79 79 Star Trek: The Next Generation
80 80 Star Trek: The Next Generation
81 81 Star Trek: The Next Generation
82 82 Star Trek: The Next Generation
83 83 Star Trek: The Next Generation
84 84 Star Trek: The Next Generation
85 85 Star Trek: The Next Generation
86 86 Star Trek: The Next Generation
87 87 Star Trek: The Next Generation
88 88 Star Trek: The Next Generation
89 89 Star Trek: The Next Generation
90 90 Star Trek: The Next Generation
91 91 Star Trek: The Next Generation
92 92 Star Trek: The Next Generation
93 93 Star Trek: The Next Generation
94 94 Star Trek: The Next Generation
95 95 Star Trek: The Next Generation
96 96 Star Trek: The Next Generation
97 97 Star Trek: The Next Generation
98 98 Star Trek: The Next Generation
99 99 Star Trek: The Next Generation

```
## 100 100 Star Trek: The Next Generation
## 101 101 Star Trek: The Next Generation
## 102 102 Star Trek: The Next Generation
## 103 103 Star Trek: The Next Generation
## 104 104 Star Trek: The Next Generation
## 105 105 Star Trek: The Next Generation
## 106 106 Star Trek: The Next Generation
## 107 107 Star Trek: The Next Generation
## 108 108 Star Trek: The Next Generation
## 109 109 Star Trek: The Next Generation
## 110 110 Star Trek: The Next Generation
## 111 111 Star Trek: The Next Generation
## 112 112 Star Trek: The Next Generation
## 113 113 Star Trek: The Next Generation
## 114 114 Star Trek: The Next Generation
## 115 115 Star Trek: The Next Generation
## 116 116 Star Trek: The Next Generation
## 117 117 Star Trek: The Next Generation
## 118 118 Star Trek: The Next Generation
## 119 119 Star Trek: The Next Generation
## 120 120 Star Trek: The Next Generation
## 121 121 Star Trek: The Next Generation
## 122 122 Star Trek: The Next Generation
## 123 123 Star Trek: The Next Generation
## 124 124 Star Trek: The Next Generation
## 125 125 Star Trek: The Next Generation
## 126 126 Star Trek: The Next Generation
## 127 127 Star Trek: The Next Generation
## 128 128 Star Trek: The Next Generation
## 129 129 Star Trek: The Next Generation
## 130 130 Star Trek: The Next Generation
## 131 131 Star Trek: The Next Generation
## 132 132 Star Trek: The Next Generation
## 133 133 Star Trek: The Next Generation
## 134 134 Star Trek: The Next Generation
```

135 135 Star Trek: The Next Generation
136 136 Star Trek: The Next Generation
137 137 Star Trek: The Next Generation
138 138 Star Trek: The Next Generation
139 139 Star Trek: The Next Generation
140 140 Star Trek: The Next Generation
141 141 Star Trek: The Next Generation
142 142 Star Trek: The Next Generation
143 143 Star Trek: The Next Generation
144 144 Star Trek: The Next Generation
145 145 Star Trek: The Next Generation
146 146 Star Trek: The Next Generation
147 147 Star Trek: The Next Generation
148 148 Star Trek: The Next Generation
149 149 Star Trek: The Next Generation
150 150 Star Trek: The Next Generation
151 151 Star Trek: The Next Generation
152 152 Star Trek: The Next Generation
153 153 Star Trek: The Next Generation
154 154 Star Trek: The Next Generation
155 155 Star Trek: The Next Generation
156 156 Star Trek: The Next Generation
157 157 Star Trek: The Next Generation
158 158 Star Trek: The Next Generation
159 159 Star Trek: The Next Generation
160 160 Star Trek: The Next Generation
161 161 Star Trek: The Next Generation
162 162 Star Trek: The Next Generation
163 163 Star Trek: The Next Generation
164 164 Star Trek: The Next Generation
165 165 Star Trek: The Next Generation
166 166 Star Trek: The Next Generation
167 167 Star Trek: The Next Generation
168 168 Star Trek: The Next Generation
169 169 Star Trek: The Next Generation

```
## 170 170 Star Trek: The Next Generation
## 171 171 Star Trek: The Next Generation
## 172 172 Star Trek: The Next Generation
## 173 173 Star Trek: The Next Generation
## 174 174 Star Trek: The Next Generation
## 175 175 Star Trek: The Next Generation
## 176 176 Star Trek: The Next Generation
##      ocena glosow     imdbId
## 1       6.9    3051 tt0092455
## 2       6.5    1149 tt0092455
## 3       5.1    1061 tt0092455
## 4       6.3     974 tt0092455
## 5       7.5    1072 tt0092455
## 6       6.3     881 tt0092455
## 7       5.9     973 tt0092455
## 8       6.9     887 tt0092455
## 9       6.9     946 tt0092455
## 10      6.2     863 tt0092455
## 11      7.5     943 tt0092455
## 12      7.7     963 tt0092455
## 13      5.5     871 tt0092455
## 14      7.5     954 tt0092455
## 15      6.0     864 tt0092455
## 16      6.2     846 tt0092455
## 17      6.8     818 tt0092455
## 18      7.0     836 tt0092455
## 19      7.3     808 tt0092455
## 20      7.2     868 tt0092455
## 21      6.9     828 tt0092455
## 22      6.7     929 tt0092455
## 23      6.6     806 tt0092455
## 24      8.1     966 tt0092455
## 25      7.4     871 tt0092455
## 26      5.7     902 tt0092455
## 27      7.1     873 tt0092455
```

##	28	8.2	1039	tt0092455
##	29	6.2	904	tt0092455
##	30	6.4	804	tt0092455
##	31	6.9	818	tt0092455
##	32	6.4	796	tt0092455
##	33	8.1	892	tt0092455
##	34	8.9	1311	tt0092455
##	35	6.0	860	tt0092455
##	36	7.6	817	tt0092455
##	37	6.6	885	tt0092455
##	38	7.5	859	tt0092455
##	39	6.2	740	tt0092455
##	40	6.7	804	tt0092455
##	41	8.9	1186	tt0092455
##	42	6.4	758	tt0092455
##	43	6.3	787	tt0092455
##	44	6.0	764	tt0092455
##	45	7.5	810	tt0092455
##	46	7.7	802	tt0092455
##	47	3.3	974	tt0092455
##	48	6.6	804	tt0092455
##	49	7.4	831	tt0092455
##	50	7.7	915	tt0092455
##	51	8.0	1005	tt0092455
##	52	6.5	778	tt0092455
##	53	7.5	808	tt0092455
##	54	7.7	800	tt0092455
##	55	6.3	758	tt0092455
##	56	6.5	749	tt0092455
##	57	8.4	925	tt0092455
##	58	7.4	807	tt0092455
##	59	6.8	773	tt0092455
##	60	8.5	996	tt0092455
##	61	6.8	753	tt0092455
##	62	9.1	1459	tt0092455

##	63	8.3	1036	tt0092455
##	64	8.2	837	tt0092455
##	65	7.4	790	tt0092455
##	66	7.2	824	tt0092455
##	67	7.3	792	tt0092455
##	68	7.8	836	tt0092455
##	69	7.7	818	tt0092455
##	70	8.1	868	tt0092455
##	71	6.6	738	tt0092455
##	72	6.9	731	tt0092455
##	73	9.3	1499	tt0092455
##	74	9.2	1360	tt0092455
##	75	8.1	988	tt0092455
##	76	7.8	805	tt0092455
##	77	6.5	758	tt0092455
##	78	7.8	879	tt0092455
##	79	6.7	752	tt0092455
##	80	8.3	851	tt0092455
##	81	7.8	842	tt0092455
##	82	6.8	752	tt0092455
##	83	6.0	712	tt0092455
##	84	8.1	862	tt0092455
##	85	8.1	818	tt0092455
##	86	7.2	797	tt0092455
##	87	8.3	890	tt0092455
##	88	8.0	810	tt0092455
##	89	7.1	748	tt0092455
##	90	7.2	757	tt0092455
##	91	6.7	720	tt0092455
##	92	8.0	818	tt0092455
##	93	7.4	797	tt0092455
##	94	8.2	924	tt0092455
##	95	7.2	744	tt0092455
##	96	6.5	706	tt0092455
##	97	7.7	718	tt0092455

##	98	7.1	737	tt0092455
##	99	8.4	788	tt0092455
##	100	8.3	776	tt0092455
##	101	8.5	1150	tt0092455
##	102	7.8	747	tt0092455
##	103	7.2	695	tt0092455
##	104	7.8	809	tt0092455
##	105	7.4	826	tt0092455
##	106	8.3	816	tt0092455
##	107	8.3	804	tt0092455
##	108	7.3	721	tt0092455
##	109	6.5	646	tt0092455
##	110	6.6	682	tt0092455
##	111	6.3	632	tt0092455
##	112	6.8	655	tt0092455
##	113	8.3	809	tt0092455
##	114	7.5	691	tt0092455
##	115	7.2	698	tt0092455
##	116	6.6	735	tt0092455
##	117	8.9	1133	tt0092455
##	118	7.8	756	tt0092455
##	119	6.2	630	tt0092455
##	120	7.4	775	tt0092455
##	121	6.1	662	tt0092455
##	122	8.6	957	tt0092455
##	123	8.2	763	tt0092455
##	124	9.3	2099	tt0092455
##	125	8.4	842	tt0092455
##	126	8.3	855	tt0092455
##	127	7.2	671	tt0092455
##	128	5.8	642	tt0092455
##	129	8.5	891	tt0092455
##	130	7.6	713	tt0092455
##	131	7.5	715	tt0092455
##	132	7.3	738	tt0092455

##	133	7.6	741	tt0092455
##	134	7.4	643	tt0092455
##	135	8.3	832	tt0092455
##	136	8.8	895	tt0092455
##	137	8.5	864	tt0092455
##	138	6.1	642	tt0092455
##	139	7.9	713	tt0092455
##	140	8.8	1013	tt0092455
##	141	7.6	676	tt0092455
##	142	7.3	668	tt0092455
##	143	7.9	757	tt0092455
##	144	7.4	709	tt0092455
##	145	7.9	771	tt0092455
##	146	8.2	786	tt0092455
##	147	6.9	629	tt0092455
##	148	7.2	604	tt0092455
##	149	7.6	654	tt0092455
##	150	8.6	801	tt0092455
##	151	8.2	711	tt0092455
##	152	7.8	712	tt0092455
##	153	6.6	628	tt0092455
##	154	6.2	603	tt0092455
##	155	7.8	687	tt0092455
##	156	7.9	678	tt0092455
##	157	7.6	665	tt0092455
##	158	6.4	665	tt0092455
##	159	7.4	651	tt0092455
##	160	6.4	608	tt0092455
##	161	7.5	641	tt0092455
##	162	8.8	866	tt0092455
##	163	8.4	756	tt0092455
##	164	7.1	613	tt0092455
##	165	5.1	721	tt0092455
##	166	8.5	859	tt0092455
##	167	7.7	660	tt0092455

## 168	6.2	671	tt0092455
## 169	6.7	614	tt0092455
## 170	7.3	703	tt0092455
## 171	6.4	650	tt0092455
## 172	6.8	578	tt0092455
## 173	6.5	575	tt0092455
## 174	6.6	586	tt0092455
## 175	7.9	630	tt0092455
## 176	8.4	5607	tt0092455

Co się dzieje z ocenami odcinków?

Co ciekawego możemy zrobić z danymi dla jednego serialu?

Z pewnością interesującą informacją są oceny danego serialu. Ta informacja dostępna jest w kolumnie `ocena` i można się do niej dostać odwołując się do tej kolumny operatorem `$`.

Ale przecież każdy odcinek podobał się inaczej, jeden bardziej inny mniej. Porównajmy trzy sposoby przedstawienia informacji o ocenach. Czy któryś z tych sposobów prezentacji liczb pozwala na zauważenie czegoś ciekawego?

Odcinki są uporządkowane zgodnie z datą emisji.

[HINT] Od odcinka 45 oceny odcinków się poprawiają.

Na którym sposobie prezentacji widać to najwyraźniej? A na którym najmniej wyraźnie. Dlaczego?

1. Sposób pierwszy: wektor liczb

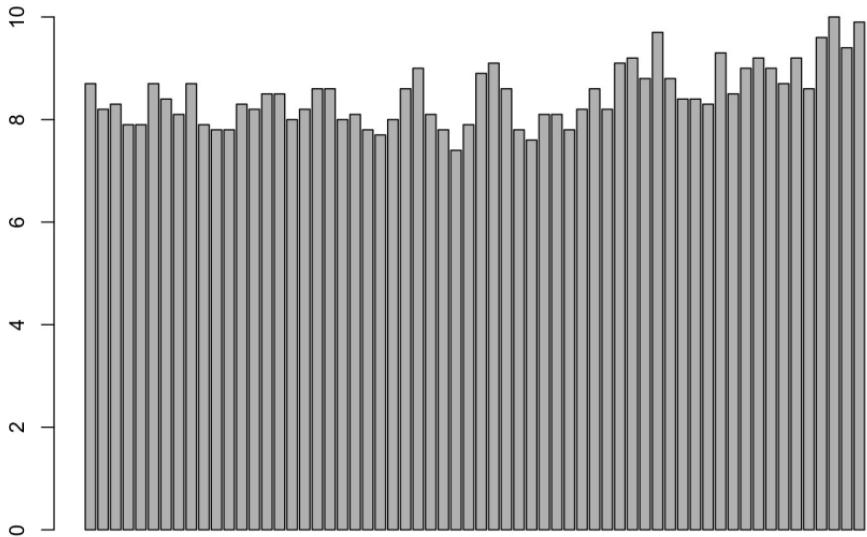
```
BreakingBad$ocena
```

```
## [1] 8.7 8.2 8.3 7.9 7.9 8.7 8.4 8.1
## [18] 8.2 8.6 8.6 8.0 8.1 7.8 7.7 8.0
## [35] 7.8 7.6 8.1 8.1 7.8 8.2 8.6 8.2
## [52] 8.5 9.0 9.2 9.0 8.7 9.2 8.6 9.1
```

Co się dzieje z ocenami odcinków?

2. Sposób drugi: wykres słupkowy, wysokość słupka odpowiada ocenie.

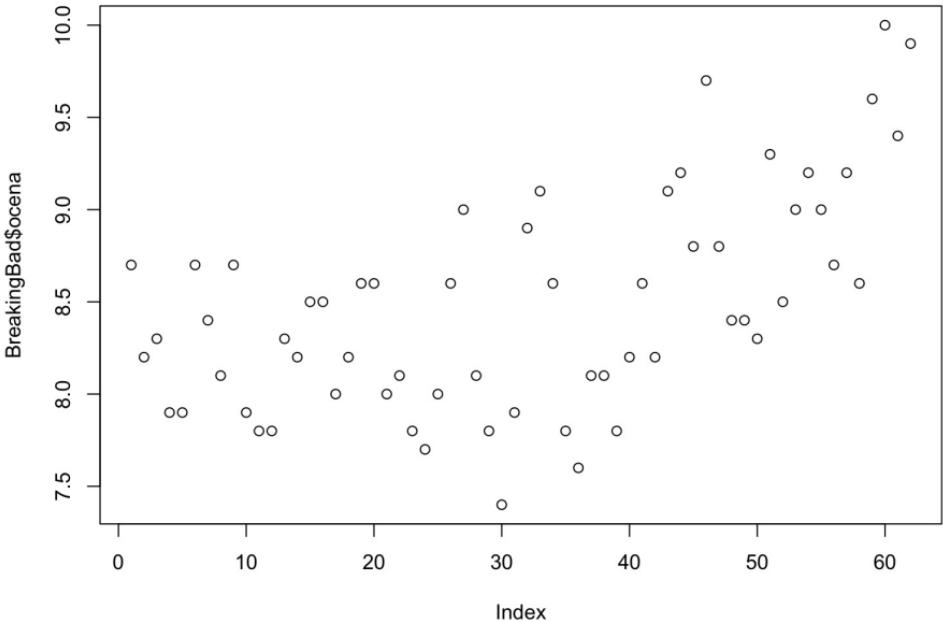
```
barplot(BreakingBad$ocena)
```



Co się dzieje z ocenami odcinków?

3. Sposób trzeci: wykres punktowy, pozycja punktu odpowiada ocenie odcinka.

```
plot(BreakingBad$ocena)
```



Sortowanie

Często interesujące obserwacje to skrajne obserwacje.

Ale jak najłatwiej się zorientować które odcinki uzyskują skrajne oceny?

Najłatwiej posortować odcinki po ocenie i wyświetlić taki posortowany zbiór danych. Na początku będą te o skrajnie niskich ocenach, na końcu te o skrajnie wysokich ocenach.

```
arrange(BreakingBad, ocena)
```

#	id	serial	ocena
## 1	30	Breaking Bad	Open I
## 2	36	Breaking Bad	Green]
## 3	24	Breaking Bad	Brea
## 4	11	Breaking Bad	I
## 5	12	Breaking Bad	Kafkae
## 6	23	Breaking Bad	Thirty-Eight
## 7	29	Breaking Bad	Corri
## 8	35	Breaking Bad	Cance
## 9	39	Breaking Bad	Gray Ma
## 10	4	Breaking Bad	Bit by a Dead
## 11	5	Breaking Bad	Ab:
## 12	10	Breaking Bad	No
## 13	31	Breaking Bad	Seven Thirty-S
## 14	17	Breaking Bad	Caballo sin No
## 15	21	Breaking Bad	I See
## 16	25	Breaking Bad	Bullet Po
## 17	8	Breaking Bad	Sho
## 18	22	Breaking Bad	Cat's in the Ba
## 19	28	Breaking Bad	Negro Y
## 20	37	Breaking Bad	Man
## 21	38	Breaking Bad	Probler
## 22	2	Breaking Bad	...And the Bag's in the I
## 23	14	Breaking Bad	Peel
## 24	18	Breaking Bad	Fift
## 25	40	Breaking Bad	A No-Rough-Stuff-Type
## 26	42	Breaking Bad	Mad:
## 27	3	Breaking Bad	Hazard
## 28	13	Breaking Bad	
## 29	50	Breaking Bad	
## 30	7	Breaking Bad	
## 31	48	Breaking Bad	
## 32	49	Breaking Bad	

## 33 15	Breaking	Bad	Better Call
## 34 16	Breaking	Bad	4 Days
## 35 52	Breaking	Bad	Bl
## 36 19	Breaking	Bad	Ph
## 37 20	Breaking	Bad	S
## 38 26	Breaking	Bad	Box C
## 39 34	Breaking	Bad	Herr
## 40 41	Breaking	Bad	Rabid
## 41 58	Breaking	Bad]
## 42 1	Breaking	Bad	Crazy Handful of Not
## 43 6	Breaking	Bad	Gr:
## 44 9	Breaking	Bad	Bi
## 45 56	Breaking	Bad	End :
## 46 45	Breaking	Bad	Live Free o:
## 47 47	Breaking	Bad	Half Mea:
## 48 32	Breaking	Bad	One Mi:
## 49 27	Breaking	Bad	Say My
## 50 53	Breaking	Bad	Blood M
## 51 55	Breaking	Bad	Full Mea
## 52 33	Breaking	Bad	
## 53 43	Breaking	Bad	Crawl S
## 54 44	Breaking	Bad	Gliding Ove
## 55 54	Breaking	Bad	Confess:
## 56 57	Breaking	Bad	Dead Fre
## 57 51	Breaking	Bad	Granite S
## 58 61	Breaking	Bad	To 'haj:
## 59 59	Breaking	Bad	Face
## 60 46	Breaking	Bad	F
## 61 62	Breaking	Bad	Ozyman
## 62 60	Breaking	Bad	

Jak oceniano ten serial?

Dla serialu Breaking Bad mamy dostępną informację o 62 odcinkach.

Każdy z nich ma określoną średnią ocenę, ale czasem chcielibyśmy myśleć o wszystkich odcinkach jak o pewnej całości. Czy cały serial jest dobry czy nie?

Jak podsumować oceny z 62 odcinków?

Dwa najpopularniejsze podejścia to średnia oraz mediana (czyli element środkowy). Można je prosto wyznaczyć funkcjami `mean()` i `median()`.

```
mean(BreakingBad$ocena)
```

```
## [1] 8.480645
```

```
median(BreakingBad$ocena)
```

```
## [1] 8.4
```

Mediana to wartość, która po posortowaniu całego wektora znajduje się w środku (lub dla parzystej liczby wartości, jest to średnia z dwóch środkowych wartości).

Więcej informacji niż sama średnia czy mediana, niesie tak zwanych pięć liczb Tukeya. Można je wyznaczyć (razem ze średnią arytmetyczną) funkcją `summary()`. Te pięć liczb przedstawia:

- Min ocenę najgorszego odcinka,
- Max ocenę najlepszego odcinka,
- Median ocenę środkowego odcinka (jeżeli jest nieparzysta liczba odcinków) lub średnią z dwóch środkowych odcinków (jeżeli jest parzysta liczna odcinków),
- 1st Qu. i 3rd Qu., czyli oceny ćwiartkowe (połowa obserwacji od minimum do mediany i od mediany do maksimum).

Pięć liczb Tukeya dzieli wartości na cztery równoliczne grupy.

```
summary(BreakingBad$ocena)
```

```
##      Min. 1st Qu.   Median     Mean 3rd Qu.    n 
## 7.400   8.025   8.400   8.481   8.800   10
```

Pięć liczb - wykres pudełko wąsy

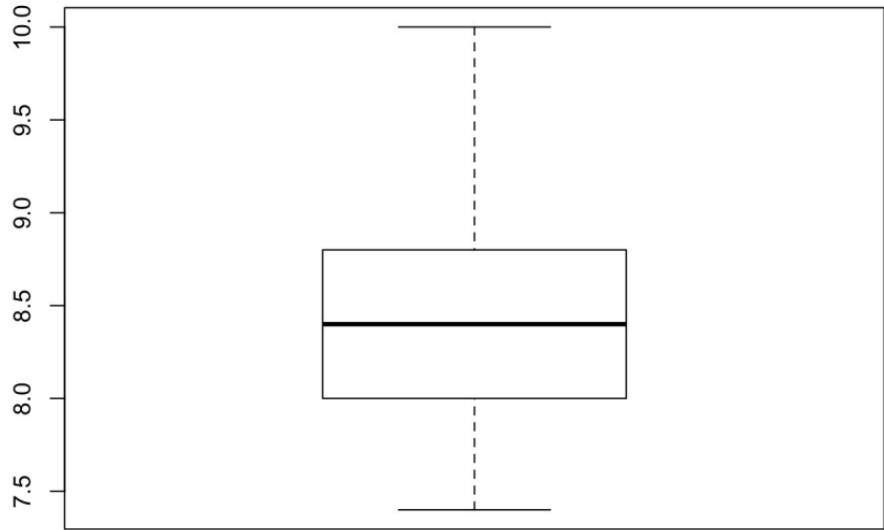
Pięć liczb Tukeya można przedstawić graficznie. Służy do tego wykres nazywany pudełko-wąsy lub ramka-wąsy lub z angielskiego boxplot.

Jest na tym wykresie pięć poziomych odcinków i każdy odpowiada jednej z pięciu liczb Tukeya.

Pomiędzy sąsiednimi poziomymi liniami mieści się tyle

samo wartości. Jeżeli jednej z tych odcinków jest dłuższy niż pozostałe, to oznacza że wartości w nim są rzadsze niż w innych.

```
boxplot(BreakingBad$ocena )
```



Pięć liczb - wykres pudełko wąsy

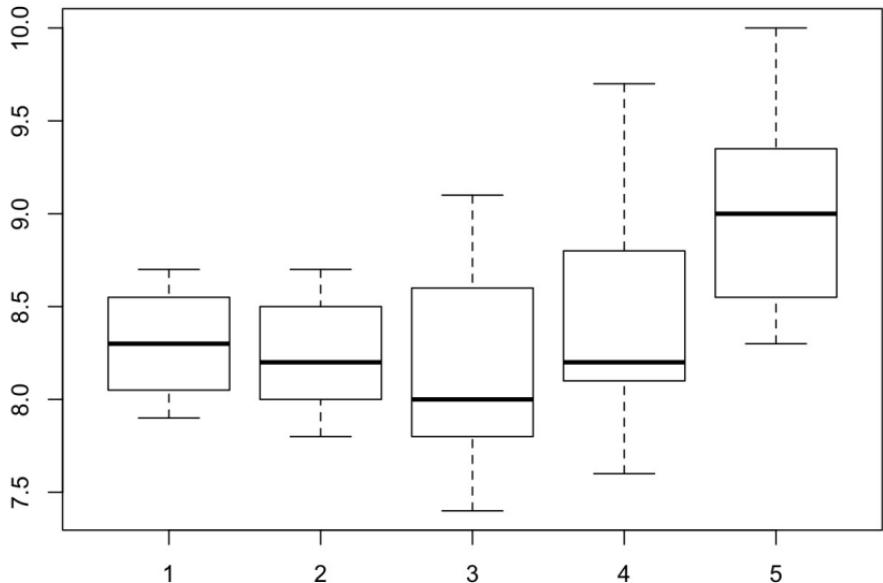
Zaletą wykresu pudełko - wąsy jest to, że pozwala łatwiej i lepiej porównać grupy obserwacji.

Zobaczmy jak oceny odcinków różnią się pomiędzy sezonami.

Jak to zrobić w programie R? W funkcji `boxplot()` użyjemy znaku formuły `~` aby określić, że chcemy przedstawiąć oceny w podziale na sezony. Lewa strona tego symbolu określa zmienną którą chcemy podsumować, a prawa strona określa grupy, które chcemy porównać.

Co możemy odczytać z takiej reprezentacji danych? Naprawdę wiele rzeczy. Spróbujmy zmierzyć się z tym zagadnieniem przez chwilę samodzielnie. Na poniższym wykresie znajdzmy trzy ciekawe obserwacje.

```
BreakingBad$sezon <- droplevels(BreakingBad$sezon)
boxplot(BreakingBad$ocena ~ BreakingBad$sezon)
```

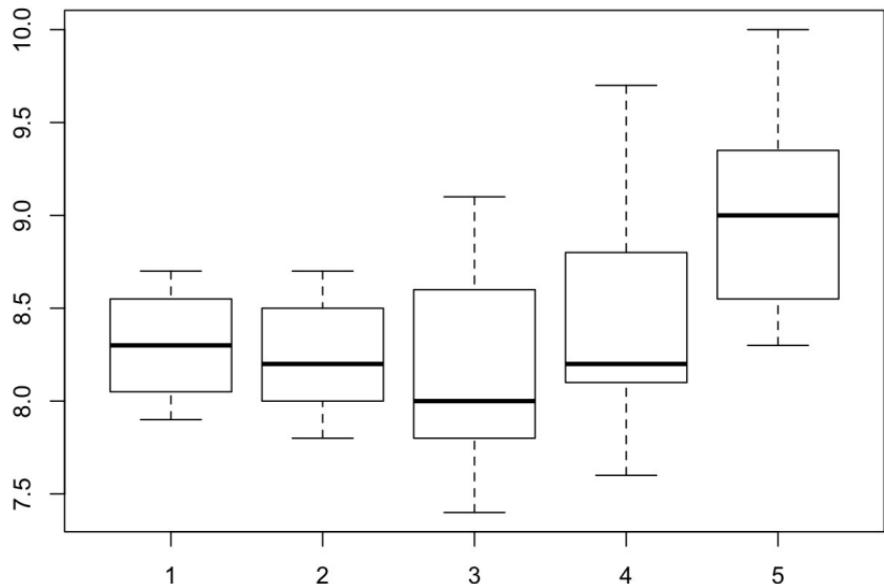


Pięć liczb - wykres pudełko wąsy

Co ciekawego udało się znaleźć? Dla mnie najciekawsze obserwacje to:

1. Najlepsze oceny ma sezon 5. Jego mediana jest powyżej całego pudełka dla sezonów 1 i 2, co oznacza, że ponad połowa odcinków z sezonu 5 była wyżej oceniana niż najlepszy z odcinków z pierwszych dwóch sezonów.

2. Najsłabsze wyniki notuje sezon 3. W nim to znajduje się odcinek najgorzej oceniany. Również dla tego sezonu obserwujemy najniższą medianę.
3. Największą zmienność obserwuje się dla sezonu 4. Był w nim i bardzo słaby odcinek (jak na ten serial) i bardzo dobry odcinek.



Histogram

Jak inaczej można przedstawić rozkład ocen?

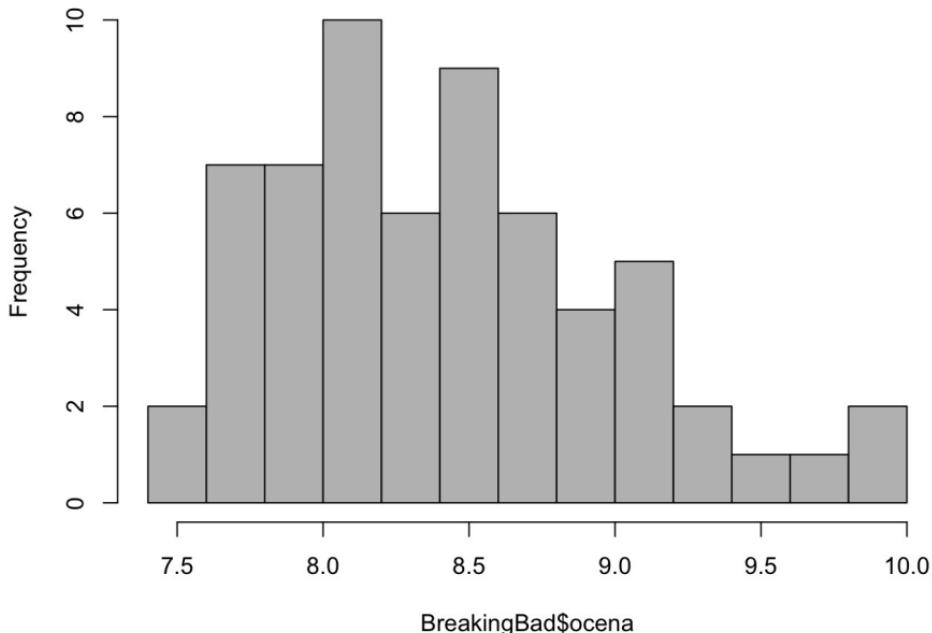
Interesującą alternatywą jest histogram. Zasada prezentacji danych jest następująca:

Dzielimy zakres wartości na pewną liczbę (poniżej na `breaks = 10`) przedziałów o równej długości. Następnie w każdym przedziale sprawdzamy ile wartości do niego wpada. Na wykresie oznaczamy wysokością słupka liczbę obserwacji, które wpadły do określonego przedziału.

Im wyższe słupki, tym większe zagęszczenie obserwacji w danym przedziale.

```
hist(BreakingBad$ocena, breaks = 10, col="grey")
```

Histogram of BreakingBad\$ocena



Co może dać nam histogram czego nie dają inne miary?

Histogram jest bardziej skomplikowanym wykresem niż wykres pudełkowy.

Pytanie czy ta komplikacja niesie użyteczną informację?

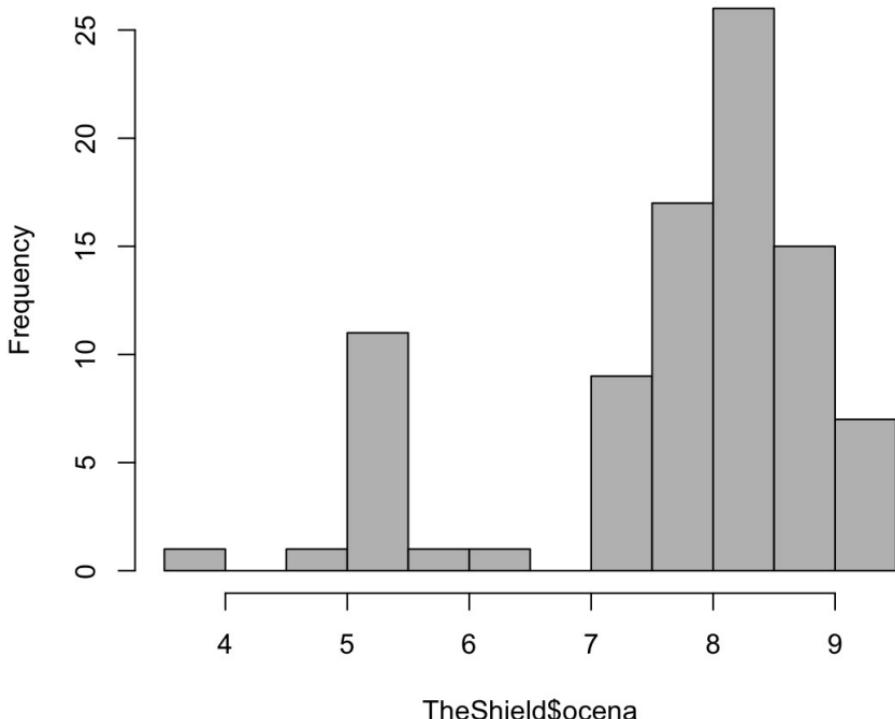
Tak! Łatwo to zauważyc dla serialu *The Shield*. Patrząc na histogram dostrzeżemy, że jest grupa odcinków o ocenach

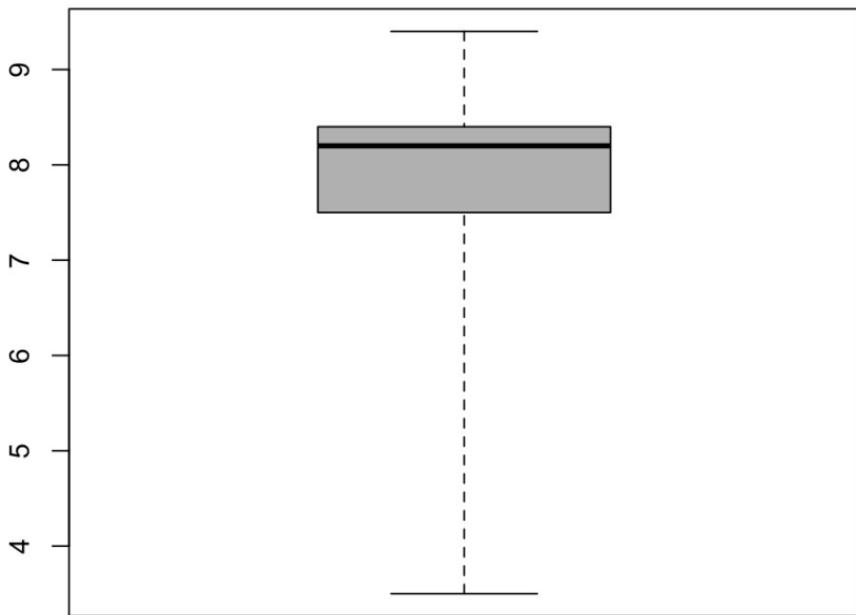
w okolicy 5-5.5, jest też grupa odcinków o ocenach w przedziale 7.5-9. Za to nie ma odcinków o ocenie pomiędzy 5.5 a 7.

Takiej informacji nie sposób odczytać ze zwykłego wykresu ramka-wąsy.

```
TheShield <- filter(serialeIMDB, serial == "The  
hist(TheShield$ocena, 10, col="grey")  
boxplot(TheShield$ocena, col="grey", range = 10
```

Histogram of TheShield\$ocena





Podsumowanie instrukcji R

W tym odcinku omawialiśmy funkcje służące do eksploracji liczbowej i graficznej danych ilościowych.

```
## pakiet z danymi  
library(PogromcyDanych)
```

```
## wielkość danych i kilka pierwszych wierszy
dim(serialeIMDB)
head(serialeIMDB)

## wartości zmiennych jakościowych
levels(serialeIMDB$serial)

## jak wyznaczyć tabelę liczebności i ja posortować
tabela <- table(serialeIMDB$serial)
sort(tabela, decreasing = TRUE)

## wybieramy tylko dane dotyczące serialu BreakingBad
BreakingBad <- filter(serialeIMDB, serial == "I")
## o ile odcinkach dostępna jest informacja dla
dim(BreakingBad)

## różne sposoby oglądania informacji o odcinkach
BreakingBad$ocena
barplot(BreakingBad$ocena)
plot(BreakingBad$ocena)
```

Podsumowanie instrukcji R

W tym odcinku omawialiśmy funkcje służące do eksploracji liczbowej i graficznej danych ilościowych.

```
## sortowanie danych
arrange(BreakingBad, ocena)

## podsumowanie danych o ocenach
summary(BreakingBad$ocena)
```

```
## wykres pudełkowy  
boxplot(BreakingBad$ocena )  
  
## wykres pudełkowy dla grup  
BreakingBad$sezon <- droplevels(BreakingBad$sezon)  
boxplot(BreakingBad$ocena ~ BreakingBad$sezon)  
  
## histogram a więc pełna informacja o rozkładzie  
hist(BreakingBad$ocena, 10, col="grey")  
  
## kiedy histogram zdradza dodatkowe informacje  
TheShield <- filter(serialeIMDB, serial == "The Shield")  
hist(TheShield$ocena, 10, col="grey")  
boxplot(TheShield$ocena, col="grey", range = 10)
```

Zadania

- Wybierz serial *Friday Night Lights* zobacz jak wygląda popularność tego serialu. Czy jest bardziej popularny niż *Breaking Bad*?
- Zobacz jak popularność tego serialu zmieniała się w czasie. Czy zaobserwowałeś coś interesującego?
- Zamiast prezentować informacje o ocenach odcinków przedstaw informacje o liczbie oddanych głosów (popularności) odcinków. Czy te same statystyki nadają się równie dobrze do prezentacji informacji o ocenie i o liczbie głosów?

Trendy w danych

Przemysław Biecek @ Uniwersytet Warszawski

sezon 2 / odcinek 9

pogRomcy danych

- O czym jest ten odcinek?
- Baza danych o filmach i serialach
- Jak wygląda trend w serialach?
- Jak wygląda trend w serialach?
- Jak szukać trendu liniowego?
- Jak wyznaczyć współczynniki trendu liniowego w programie R
- Jak wyznaczyć współczynniki trendu liniowego w programie R?
- Jaka to relacja?
- Jak wygląda rozwój trendu w ocenach dla Twojego ulubionego serialu?
- Podsumowanie instrukcji R
- Zadania
- Jak wygląda trend w serialach - Bonus

O czym jest ten odcinek?

Analizując dane, często trafiamy na sytuację, gdy mamy sekwencje liczb i chcemy ocenić czy widać w niej trend.

Analizując zużycie energii możemy sprawdzać czy występuje rosnący trend zapotrzebowania na energię. Jeżeli pracujemy z danymi o sprzedaży biletów może interesować nas trend w liczbie sprzedanych biletów.

Jednak nawet gdy występuje widoczny trend, bardzo rzadko wartości układają się idealnie w linię prostą. Zazwyczaj trend jest zanurzony w dużych fluktuacjach.

Jak więc oszacować z jakim trendem mamy do czynienia?

Tutaj pojawia się statystyka. Używając takich narzędzi jak regresja liniowa czy nieliniowa możemy ocenić jakie są parametry trendu. Dla uproszczenia w tym odcinku skupimy się na trendzie liniowym.

Oczywiście nie zawsze tendencje układają się w liniowy wzrost lub liniowy spadek. Potraktujemy więc prace z trendem liniowym jako pierwszy krok, zanim zaczniemy dopasowywać bardziej złożone zależności.

W tym odcinku, na przykładzie danych z serwisu IMDB (Internet Movie Database), pokażemy jak działa procedura wyznaczania trendu w danych.

Baza danych o filmach i serialach

W bazie danych z informacjami o filmach IMDB (Internet Movie Database) znaleźć można dane o serialach. Wśród tych danych znajdują się średnie oceny internautów wystawione kolejnym odcinkom serialu.

Przykładowo, na stronie

[http://www.imdb.com/title/tt0903747/episodes?
ref_=ttep_ql_4](http://www.imdb.com/title/tt0903747/episodes?ref_=ttep_ql_4)

znajdują się oceny serialu *Breaking Bad*.

Dane o ocenach zostały pobrane z Internetu i przygotowane tak, że teraz są dostępne w zbiorze danych `serialsIMDB` w pakiecie `PogromcyDanych`, który towarzyszy temu kursowi.

Na przykładzie tych danych poćwiczymy dopasowanie trendu liniowego do danych.

```
library(PogromcyDanych)
## pierwsze sześć wierszy ze zbioru danych serialsIMDB
head(serialsIMDB)

##      id      serial          title
## 1    1  Breaking Bad  Cat's in the Bag
## 2    2  Breaking Bad ...And the Bag's in the R...
## 3    3  Breaking Bad          Cancer
## 4    4  Breaking Bad          Gray Ma...
## 5    5  Breaking Bad  Crazy Handful of Not...
## 6    6  Breaking Bad
```

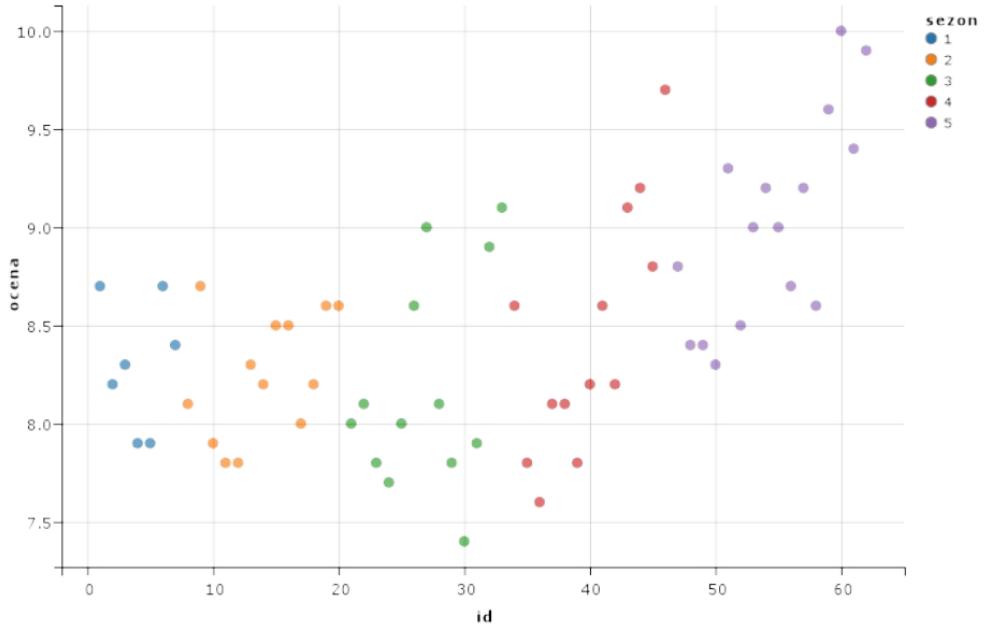
```
##      imdbId  
## 1 tt0903747  
## 2 tt0903747  
## 3 tt0903747  
## 4 tt0903747  
## 5 tt0903747  
## 6 tt0903747
```

Jak wygląda trend w serialach?

Mając taki zbiór danych, przyjrzyjmy się jak wyglądają oceny dla serialu *Breaking Bad*.

Poniższy wykres przedstawia średnie oceny (na osi OY) dla kolejnych odcinków (rozłożonych wzdłuż osi OX).

Kolejne sezony są oznaczane różnymi kolorami. Czy patrząc na te wyniki widzimy jakąś globalną tendencję / trend?

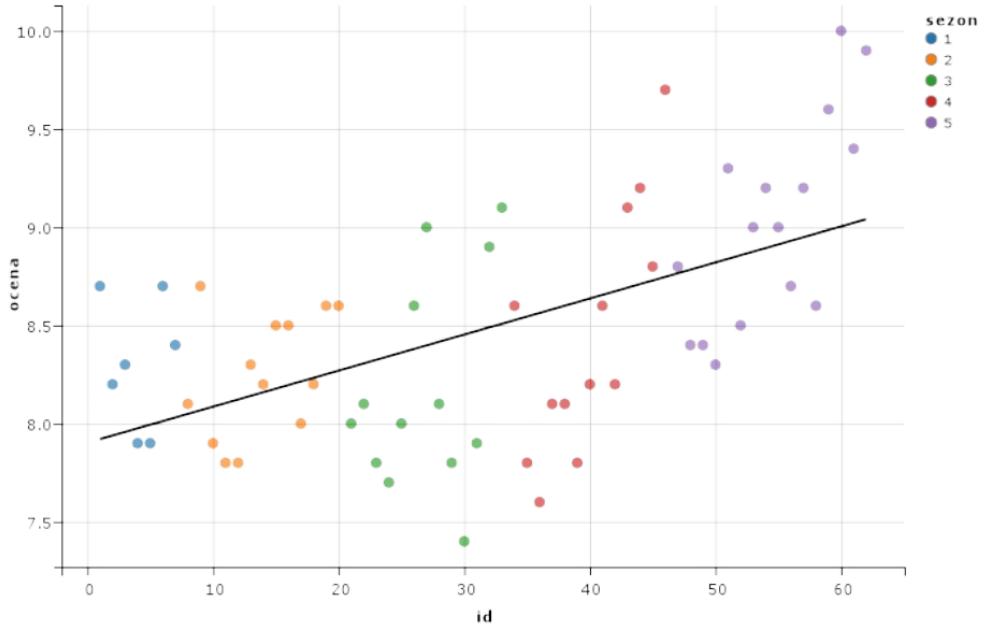


Jak wygląda trend w serialach?

Dodajmy do wykresu linię trendu liniowego, najlepiej dopasowanego do danych.

Patrząc na trend łatwiej zauważyc wzrostową tendencję ocen. Co więcej można odczytać tempo średniego wzrostu ocen.

Ale jak ta linia trendu została wyznaczona? Co to znaczy „najlepiej dopasowany do danych”?



Jak szukać trendu liniowego?

Problem znajdowania trendu liniowego można przedstawić na wiele sposobów. Poniżej przedstawimy go jako zagadnienie optymalizacji.

Trend liniowy można przedstawić za pomocą równania liniowego $(a_0 + a_1 * \text{numerOdcinka})$.

Wartości (a_0) i (a_1) są nieznane i należy je wyznaczyć z danych.

Ale jak je wyznaczyć?

Chcielibyśmy, żeby linia trendu była jak najbliżej danych.

Ale co to znaczy jak najbliżej?

Jest wiele sposobów określania miary bliskości.

Najczęściej używany jest następujący:

- Dla odcinka i , czyli punktu o współrzędnych $(\langle i \rangle, \langle \text{ocena}_i \rangle)$ odległość od lini trendu liczymy jako $\sqrt{\text{niedopasowanie}_i(a_0, a_1) = (\text{ocena}_i - (a_0 + a_1 * i))^2}$
- Niedopasowanie całej krzywej trendu (oznaczane zazwyczaj jako RSS, skrót od *Residual Sum of Squares*) liczymy jako sumę niedopasowań dla każdego punktu $\sum_i \text{niedopasowanie}_i = \sum_i (\text{ocena}_i - (a_0 + a_1 * i))^2$
- Oceny współczynników linii trendu $\langle \hat{a}_0 \rangle$ i $\langle \hat{a}_1 \rangle$ wyznacza się w ten sposób by minimalizowały niedopasowanie $\langle \text{RSS}(a_0, a_1) \rangle$.

Aby wyprowadzić wzór na te współczynniki dobrze znać przynajmniej rachunek różniczkowy. Wystarczy wtedy policzyć pochodne po $\langle a_i \rangle$ i przyrównać je do zera (co nie jest trudne bo mamy do czynienia z wielomianem drugiego stopnia, przykładowe wyprowadzenie jest przedstawione na wikipedii

http://en.wikipedia.org/wiki/Simple_linear_regression).

Aby wyznaczyć wartości tych współczynników w programie R wystarczy wykorzystać funkcję `lm()`.

Jak wyznaczyć współczynniki trendu liniowego w programie R

Przygotujmy zbiór danych `tylkoBreakingBad` w którym będą tylko wartości zmiennych `id` i `ocena` dla odcinków z serialu `Breaking Bad`.

Wykorzystamy funkcje `filter()` i `select()`, więcej o tym jak one działają można przeczytać w pierwszym sezonie Pogromców.

```
## wybierz z danych tylko dwie kolumny i dla jednego serialu
tylkoBreakingBad <- serialeIMDB %>%
  filter(serial == "Breaking Bad") %>%
  select(id, ocena)
```

Pierwsze 6 wierszy z nowo przygotowanego zbioru danych.

```
head(tylkoBreakingBad)
```

```
##     id  ocena
## 1     1   8.7
## 2     2   8.2
## 3     3   8.3
```

```
## 4 4 7.9  
## 5 5 7.9  
## 6 6 8.7
```

Jak wyznaczyć współczynniki trendu liniowego w programie R?

Do wyznaczenia trendu liniowego wykorzystujemy funkcję `lm()` (nazwa od *linear model* czyli model liniowy).

Funkcja `lm()` jako pierwszy argument przyjmuje formułę opisującą pomiędzy którymi zmiennymi chcemy wyznaczyć relację, a jako drugi argument wskazuje zbiór danych.

Centralnym symbolem w formule jest znak `~` (tylda). Po jego lewej stronie wskazuje się zmienną którą chcemy opisać (tak zwaną zmienną wyjaśnianą), a po prawej stronie podaje się zmienną wyjaśniającą.

My chcemy opisać średnią ocenę jako funkcję numeru odcinka. Wybieramy tylko dane dla *Breaking Bad*.

```
lm(ocena~id, tylkoBreakingBad)
```

```
##  
## Call:  
## lm(formula = ocena ~ id, data = tylkoBreakin
```

```
##  
## Coefficients:  
## (Intercept) id  
## 7.90222 0.01836
```

Wynik funkcji `lm()` jest listą z wieloma polami.
Współczynniki modelu znajdują się w elemencie o nazwie `coef`.

```
lm(ocena~id, tylkoBreakingBad)$coef  
  
## (Intercept) id  
## 7.90222105 0.01836267
```

Wartość `(Intercept)` to ocena współczynnika a_0 a `id` to ocena współczynnika a_1 .

Jaka to relacja?

Tak więc dla serialu `Breaking Bad` linia, która minimalizuje błąd RSS ma równanie $\text{ocena} = 7.90222 + 0.01836 * \text{id}$

Jak czytać to równanie?

Za `id` należy podstawać numer odcinka aby odczytać oszacowanie średniej oceny danego odcinka wynikającej z dopasowanego trendu liniowego.

Przykładowo, dla odcinka pierwszego `id=1`, a więc

oszacowanie startowej oceny z wykorzystaniem trendu liniowego to $\lfloor \text{ocena_1} = 7.90222 + 0.01836 * 1 = 7.92058 \rfloor$

Zmienna $\langle a_1 \rangle$ opisuje średnią zmianę oceny co odcinek. Zgodnie z oszacowanymi wartościami, średnia zmiana oceny wynosi 0.01836.

I tak po 62 odcinkach, dopasowany trend szacuje popularność końówki sezonu na

$$\lfloor \text{ocena_}\{62\} = 7.90222 + 62 * 0.01836 = 9.04054 \rfloor$$

Korzystając z tych średnich, musimy oczywiście mieć na uwadze, że linia prosta to pewne przybliżenie danych, ale niekoniecznie idealne.

Jak dopasowywać bardziej złożone modele? O tym napiszemy w kolejnych odcinkach.

Jak wygląda rozwój trendu w ocenach dla Twojego ulubionego serialu?

Do badania trendu można wykorzystać internetową interaktywną aplikację dostępną pod adresem <http://beta.icm.edu.pl/IMDB/>.

Podsumowanie instrukcji R

W tym odcinku omawialiśmy funkcje służące do wyznaczania trendu liniowego.

```
## wykres z ocenami odcinków, pakiet ggvis
library(PogromcyDanych)
library(ggvis)
serialeIMDB %>%
  filter(serial == "Breaking Bad") %>%
  mutate(sezon = droplevels(sezon)) %>%
  ggvis(x = ~id, y = ~ocena, fill = ~sezon) %
    layer_text(text := ~nazwa, opacity=0, fontweight="bold")
    layer_points(fillOpacity:=0.8)

## wykres z trendem, pakiet ggvis
serialeIMDB %>%
  filter(serial == "Breaking Bad") %>%
  mutate(sezon = droplevels(sezon)) %>%
  ggvis(x = ~id, y = ~ocena, fill = ~sezon) %
    # ewentualnie group_by(sezon)
    layer_text(text := ~nazwa, opacity=0, fontweight="bold")
    layer_points(fillOpacity:=0.8) %>%
    layer_model_predictions(model = "lm")

## wybierz tylko dwie kolumny i wiersze dla jednego serialu
tylkoBreakingBad <- serialeIMDB %>%
  filter(serial == "Breaking Bad") %>%
  select(id, ocena)
## pierwsze 6 wierszy z nowo przygotowanego zbioru
head(tylkoBreakingBad)

## wyznacz trend liniowy dla serialu Breaking Bad
```

```
lm(ocena~id, tylkoBreakingBad)
```

```
## odczytaj z trendu liniowego współczynniki t:  
lm(ocena~id, tylkoBreakingBad)$coef
```

Zadania

- Znajdź serial o najsilniej rosnącym trendzie dotyczącym ocen (ponieważ oceny są ograniczone z góry przez 10 to może nie być serial o najwyższej średniej ocenie)
- Znajdź serial o najsilniejszym trendzie spadkowym

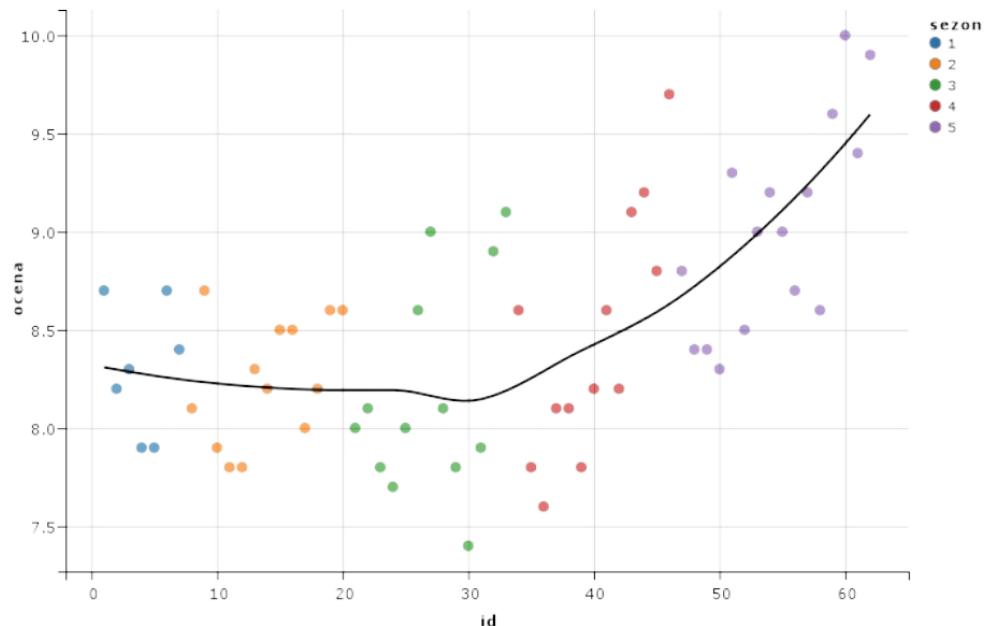
Jak wygląda trend w serialach - Bonus

Do tego miejsca mogło się zrodzić wrażenie, że być może trend liniowy to nie jedyny a może nawet nie najlepszy wybór.

Poniższa ilustracja prezentuje wyniki trendu liczonego w sposób bardziej lokalny, dzięki czemu, można zauważyć, że inna była zmiana ocen do około 30 odcinka (końca trzeciego sezonu), widać było lekką spadkową tendencję, a inna była zmiana po 30 odcinku (coraz to wyższe oceny).

Jak modelować takie, bardziej złożone zależności?

Pokażemy w ostatnim odcinku tego sezonu „Co dalej?”.



Trendy - testy statystyczne

Przemysław Biecek @ Uniwersytet Warszawski

sezon 2 / odcinek 10

pogRomcy danych

- O czym jest ten odcinek?
- Zbiór danych do pracy
- Dane i trend liniowy
- Czy wzrost ocen dla Breaking Bad to przypadek?
Test permutacyjny
 - Krok 1: Postaw hipotezę, określ co chcesz porównać
 - Krok 2: Określ miarę wielkości
- Czy wzrost ocen dla Breaking Bad to przypadek?
Test permutacyjny
 - Krok 3: Porównaj zaobserwowany trend z trendem w sytuacji gdyby nie było zależności
- Czy wzrost ocen dla Breaking Bad to przypadek?
Porównanie wartości
- Czy wzrost ocen dla Breaking Bad to przypadek? P-wartość
- A co z Przyjaciółmi?
- A co z Przyjaciółmi? Testowanie

- [A co z Przyjaciółmi? Testowanie](#)
- [A co z Przyjaciółmi? P-wartość](#)
- [Podsumowanie testowania](#)
- [Podsumowanie instrukcji R](#)
- [Zadania](#)

O czym jest ten odcinek?

W poprzednim odcinku poznaliśmy metodę wyznaczania współczynników (a_0) i (a_1) trendu liniowego.

Dzięki temu, możemy np. ocenić czy obserwowana wartość (np. ocena odcinka) średnio rośnie czy maleje.

Czasem jednak ocena współczynnika (a_1) , opisująca tempo wzrostu jest bardzo bliska wartości (0) . Możemy wtedy zadać dodatkowe pytanie, czy obserwowany wzrost/spadek nie jest wynikiem przypadkowych fluktuacji?

Jak odpowiedzieć na to pytanie? Kiedy obserwowany trend może być wynikiem przypadkowego ułożenia wartości, a kiedy jest wynikiem prawdziwej tendencji?

Tutaj ponownie pojawia się statystyka. Używając takich narzędzi jak *test dla współczynnika trendu* możemy ocenić na ile ocena współczynnika (a_1) jest istotnie

różna od 0 (a więc jej znak jest nieprzypadkowy).

W tym odcinku, na przykładzie danych z serwisu IMDB (Internet Movie Database), pokażemy jak działa procedura testowania trendu w danych.

Zbiór danych do pracy

W bazie danych o filmach IMDB (Internet Movie Database) znaleźć można dane o serialach. Dla każdego odcinka serialu dostępne są średnie oceny użytkowników.

Przykładowo, na stronie

[http://www.imdb.com/title/tt0903747/episodes?
ref_=ttep_ql_4](http://www.imdb.com/title/tt0903747/episodes?ref_=ttep_ql_4)

znajdują się oceny serialu *Breaking Bad*.

Dane te zostały pobrane z Internetu i przygotowane tak, że teraz są dostępne w zbiorze danych `serialsIMDB` w pakiecie `PogromcyDanych`, który towarzyszy temu kursowi.

Wybierzymy z tego zbioru danych tylko wartości dla serialu `Breaking Bad`.

```
library(PogromcyDanych)
## wybieramy dane o jednym serialu, dwie kolumny
tylkoBreakingBad <- serialsIMDB %>%
```

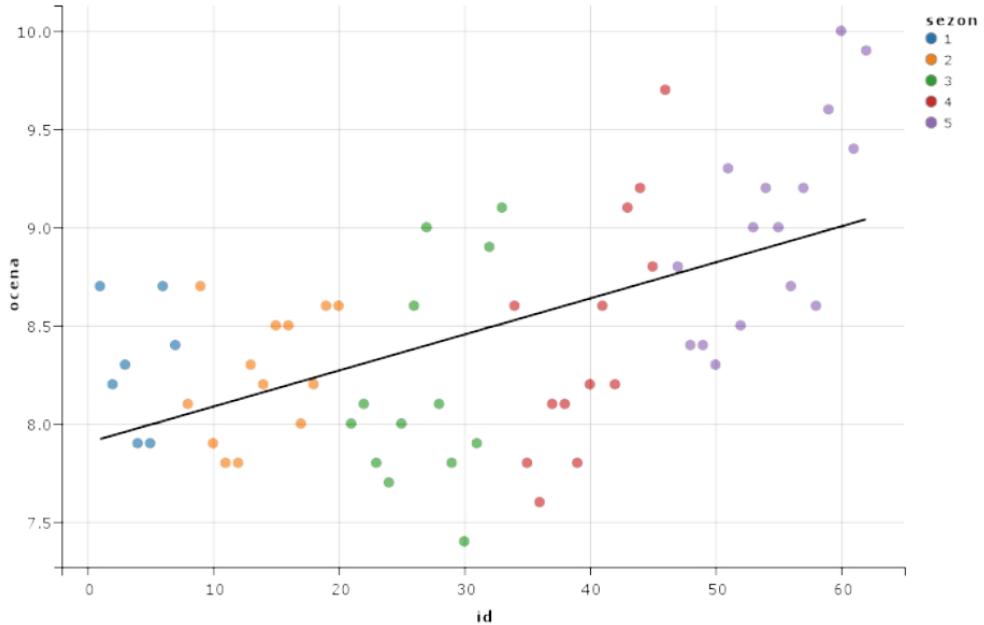
```
filter(serial == "Breaking Bad") %>% select(:  
## wyświetlamy pierwsze wiersze  
head(tylkoBreakingBad)  
  
##   id ocena  
## 1 1 8.7  
## 2 2 8.2  
## 3 3 8.3  
## 4 4 7.9  
## 5 5 7.9  
## 6 6 8.7
```

Dane i trend liniowy

W poprzednim odcinku pokazaliśmy jak wyznaczyć współczynniki trendu liniowego z użyciem funkcji `lm()`. Przypomnijmy, abytrzymać punkt startowy trendu, oraz tempo zmiany co odcinek, należało wykorzystać instrukcję.

```
lm(ocena~id, tylkoBreakingBad)$coef  
  
## (Intercept)      id  
## 7.90222105  0.01836267
```

Dla przypomnienia, graficzna ilustracja danych o tym serialu.



Czy wzrost ocen dla Breaking Bad to przypadek? Test permutacyjny

Dla serialu Breaking Bad widzimy wzrostową tendencję wynoszącą ponad 0.018.

Na wykresie wygląda to jak stabilny trend rosnący, ale czy poza oceną wzrokową możemy liczbowo ocenić czy ten trend nie jest przypadkowy?

Czy taka tendencja może nas przekonać, że faktycznie odcinki w tym serialu są coraz lepsze (a w każdym razie

coraz wyżej oceniane)?

Statystycy opracowali wiele testów aby badać istotność trendu / tendencji. W większości przypadków te testy dają zbliżone wyniki, więc zamiast przedstawiać ich leksykon wyjaśnimy na jednym przykładzie na czym polega testowanie. Przedstawimy tak zwany test permutacyjny do badania trendu.

Pomysł na ocenę, czy trend ma istotną tendencję wzrostową, składa się z trzech kroków. Przedstawmy jeden po drugim.

Krok 1: Postaw hipotezę, określ co chcesz porównać

Będziemy testować hipotezę (=przypuszczenie), że współczynnik wzrostu jest istotnie różny od zera, czyli od braku trendu.

Tę hipotezę, sformujmy w następujący sposób: *Czy obserwowana wartość współczynnika wzrostu jest wystarczająco duża, by uznać ją za istotną?*

Krok 2: Określ miarę wielkości

Ustalmy, że wielkość współczynnika trendu będziemy

mierzyć za pomocą wartości bezwzględnej tego współczynnika. Zarówno bowiem silny ujemny jak i silny dodatni trend uznamy za różny od zera.

Wartość bezwzględną można wyznaczyć funkcją `abs()`.

```
(wspolczynnikiTrendu <- lm(ocena~id, tylkoBreal  
## (Intercept) id  
## 7.90222105 0.01836267  
  
## moduł współczynnika wzrostu  
(modulBreakingBad <- abs(wspolczynnikiTrendu[2])  
## id  
## 0.01836267
```

Czy wzrost ocen dla Breaking Bad to przypadek? Test permutacyjny

Krok 3: Porównaj zaobserwowany trend z trendem w sytuacji gdyby nie było zależności

Wciąż nie wiemy, czy `0.0183` to duży wzrost czy mały.

Musimy go z czymś porównać. Najlepiej by było porównać ją z wartościami, które byśmy obserwowali gdyby nie było prawdziwego trendu.

Ale skąd wiadomo, ile wynosiłby ten współczynnik gdyby nie było trendu? Możemy to oszacować w następujący sposób

- [I] Wymieszamy losowo wartości ocen. Dzięki temu otrzymamy sekwencje ocen o której wiemy, że nie ma w niej żadnego systematycznego trendu (losowo je wymieszaliśmy), a ewentualna wielkość współczynnika trendu wynika z losowych fluktuacji,
- [II] Policzymy jak duży jest trend dla tych nowo wylosowanych grup,
- [III] Powtórzmy kroki [I] i [II] wielokrotnie, aby zobaczyć jak duże wartości może przyjmować trend w sytuacji gdy jest on wynikiem losowych fluktuacji. Powtórzmy te kroki wiele wiele razy, np. 99 999 razy.

Liczenie wartości trendu dla losowo pomieszanych danych jest realizowane przez poniższy kod.

```
## funkcja replicate() powtarza drugie wyrażenie:
trendy <- replicate(99999, {
  # wymieszaj losowo wartości, robi to funkcja
  tylkoBreakingBad$wymieszczone <- sample(tylkoB

  # wyznacz moduł współczynnika wzrostu dla losów
  wspolczynnikiTrendu <- lm(wymieszczone~id, tylko)
  abs(wspolczynnikiTrendu[2])
})
## modułBreakingBad jest też jedną z możliwych
```

```
trendy <- c(trendy, modulBreakingBad)
## wyświetl pierwsze 20 z tak otrzymanych wartości
head(trendy, 20)

##           id           id           id
## 8.017930e-03 3.817582e-03 5.937901e-03 1.500
##           id           id           id
## 5.560172e-03 3.631236e-03 7.050943e-05 7.051
##           id           id           id
## 3.928383e-04 2.039737e-03 8.491350e-03 6.970
##           id           id           id
## 2.155574e-03 3.812546e-03 4.683841e-04 4.770
```

Czy wzrost ocen dla Breaking Bad to przypadek? Porównanie wartości

Możemy teraz wartość `modulBreakingBad` porównać z wektorem wartości referencyjnych, które są w wektorze `trendy`.

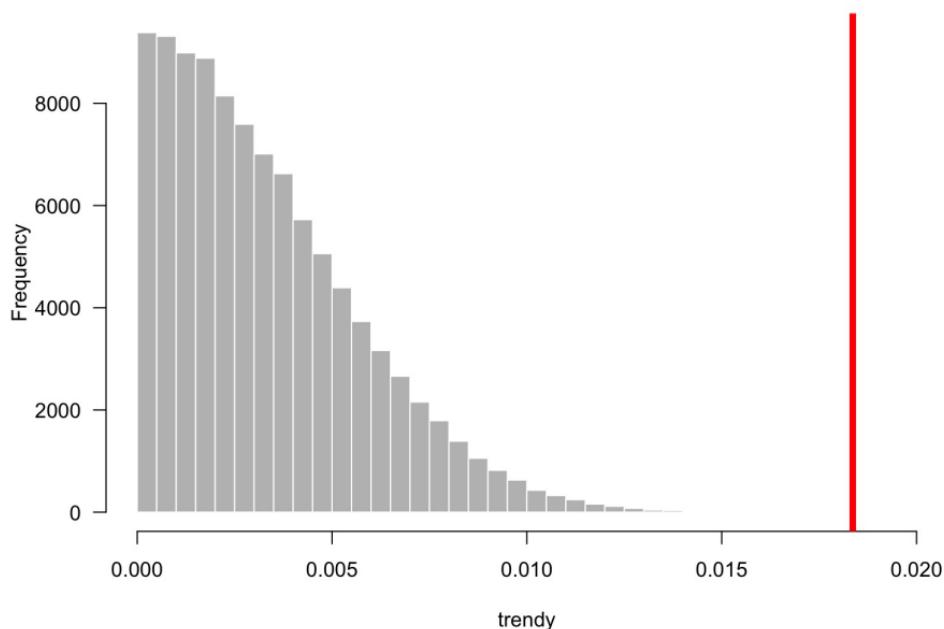
O wektorze `trendy` można myśleć jak o wielkościach wzrostów, jakich spodziewać się możemy gdy próby różnią się wyłącznie przypadkową zmiennością. Gdyby `modulBreakingBad` był znaczaco większy niż wartości z wektora `trendy`, to byłby to silny argument za tym, że trend rosnący w ocenie odcinków *Breaking Bad* nie jest przypadkowy.

Porównajmy na wykresie wielkości różnic dla losowych

fluktuacji (na szaro) z obserwowaną różnicą (na czerwono).

Jak widzimy, obserwowana wartość wzrostu ocen dla serialu *Breaking Bad* jest znacznie większa niż wzrosty, które byłoby widać, gdyby nie było żadnej prawdziwej tendencji.

```
## jak wyglądają przypadkowe współczynniki  
hist(trendy, 50, col="grey", main="", las=1, b  
## dorysuj na czerwono współczynnik dla Breakin  
abline(v=modulBreakingBad, col="red", lwd=5)
```



Czy wzrost ocen dla Breaking Bad to przypadek? P-wartość

Wykres pomaga w zobaczeniu jak wyglądają przypadkowe współczynniki na tle obserwowanej wartości.

Ale aby podjąć decyzje lepiej mieć jedną liczbę. Jaką?

Policzmy jaka część trendów dla losowych fluktuacji jest większa lub równa niż obserwowany trend dla *Breaking Bad*. Ta wartość nazywa się *wartością p* lub *p-wartością*.

```
## p-wartość  
mean(trendy >= modulBreakingBad)  
  
## [1] 1e-05
```

Tylko 0.00001? Czyli, gdy zaburzaliśmy losowo oceny to nie udało się uzyskać tak dużego wzrostu jak w przypadku serialu *Breaking Bad*. Jedna wartość, przez którą obserwowana średnia różni się od zera, to wynik trendu obserwowanego bez permutacji, który dodaliśmy do wektora `trendy`.

To porównanie, pokazujące jak dużą wartością `modulBreakingBad` na tle przypadkowych wartości zapisanych w wektorze `trendy` jest przekonującym

argumentem, że obserwowany wzrost ocen dla *Breaking Bad* nie jest przypadkowy.

A co z Przyjaciółmi?

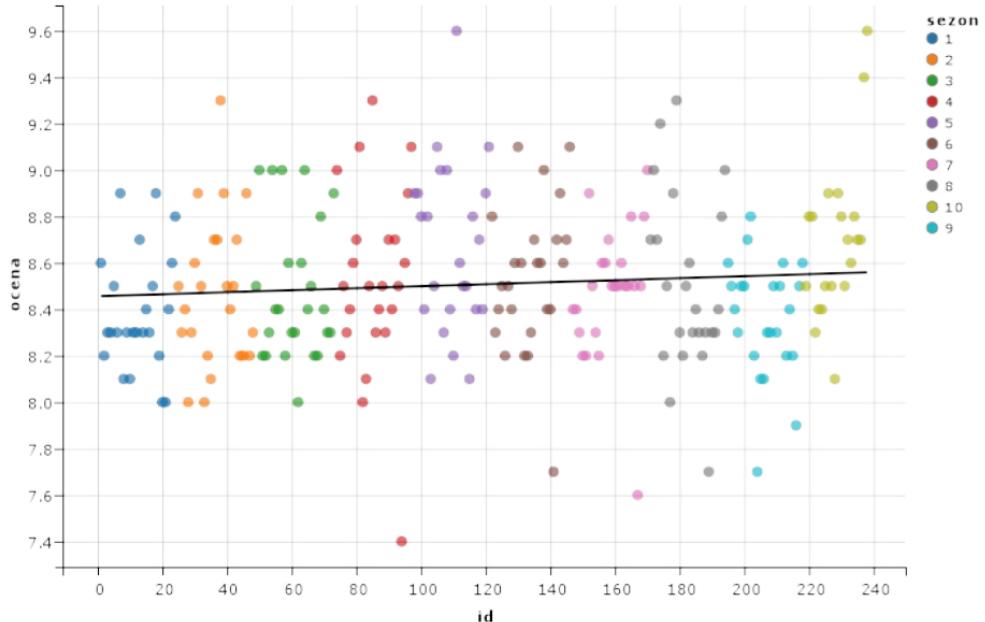
A jak wygląda trend dla serialu *Friends*?

Co do wartości to wynosi niewiele ponad 0.000432, czyli jest również dodatni, ale mniejszy niż dla *Breaking Bad*.

Hipoteza, którą chcemy sprawdzić, to czy ten trend istotnie większy od zera?

```
tylkoFriends <- serialsIMDB %>% filter(serial =  
  ## wyznaczmy współczynniki trendu liniowego dla  
  lm(ocena~id, tylkoFriends)$coef  
  
##   (Intercept)           id  
## 8.4562918838 0.0004325633
```

Oraz poglądowy rysunek prezentujący ten trend.



A co z Przyjaciółmi? Testowanie

Zweryfikujemy hipotezę o istotności trendu w ten sam sposób co dla serialu *Breaking Bad*.

Przeprowadźmy te same obliczenia, tylko że dla serialu *Przyjaciele*. Wyniki porównamy w ten sam sposób co poprzednio.

```
## trend dla serialu Friends
wspolczynnikiTrendu <- lm(ocena~id, tylkoFriends)
(modulFriends <- abs(wspolczynnikiTrendu[2]))
```

```
##                  id
## 0.0004325633
```

```
## losowe współczynniki trendu w liczbie 99 99!
trendy <- replicate(99999, {
  tylkoFriends$wymieszane <- sample(tylkoFriends$id)
  wspolczynnikiTrendu <- lm(wymieszane~id, tylkoFriends)
  abs(wspolczynnikiTrendu[2])
})
## modulFriends jest też jedną z możliwych pere
trendy <- c(trendy, modulFriends)
head(trendy, 20)
```

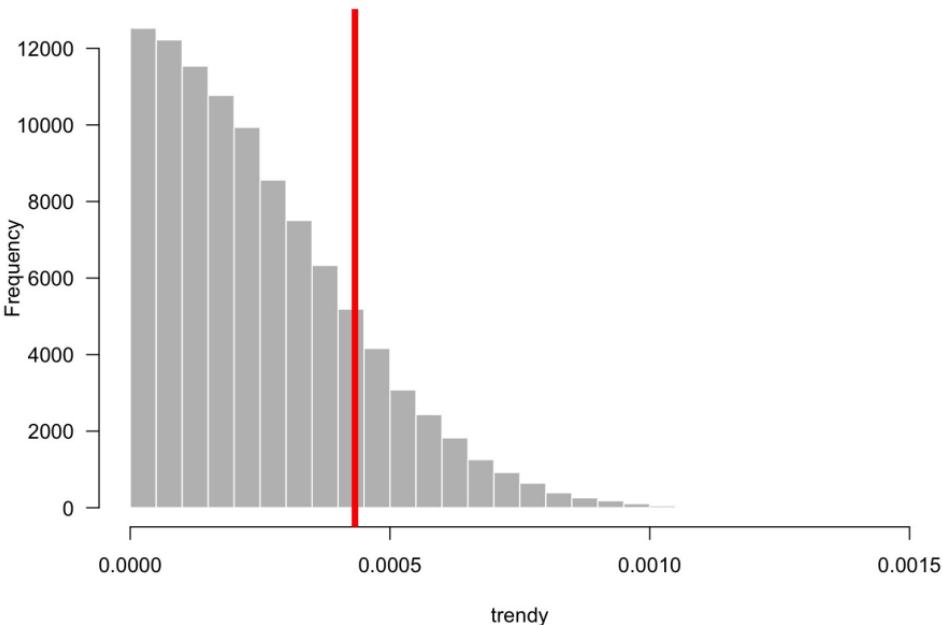
```
##           id           id           id
## 6.128610e-05 2.310357e-04 2.321039e-04 6.571
##           id           id           id
## 1.639192e-04 4.232613e-05 1.934718e-04 8.184
##           id           id           id
## 2.939685e-04 1.731766e-04 2.523990e-04 3.622
##           id           id           id
## 4.235729e-04 7.019195e-04 6.903031e-05 2.508
```

Mamy wyznaczony referencyjny wektor wzrostów trendy. Jak na jego tle wygląda obserwowane 0.00043?

A co z Przyjaciółmi? Testowanie

Wykres z wartościami referencyjnymi i wzrostem dla serialu Przyjaciele.

```
hist(trendy, 50, col="grey", main="", las=1, border="black")
abline(v=modulFriends, col="red", lwd=5)
```



A co z Przyjaciółmi? P-wartość

Już graficzna ocena sytuacji, pokazuje, że mamy do czynienia z inną zależnością niż dla *Breaking Bad*. Tym razem dosyć często się zdarza, żeby losowe permutacje wektora ocen owocowały większym współczynnikiem wzrostu niż ten, który obserwujemy dla serialu Przyjaciele.

Jak opisać te różnice jedną liczbą? Jak często obserwuje się wartość większą lub równą.

```
## p-wartość  
mean(trendy >= modulFriends)  
## [1] 0.17086
```

Dla serialu *Friends* nawet gdy losujemy wartości ocen, a więc mamy czystko przypadkowe sekwencje, to w ponad 17 procentach przypadków obserwujemy trend większy niż 0.00043. Czy to oznacza, że rosnący trend jest czysto przypadkowy?

Pewności nie ma, ale raczej jesteśmy skłonni uznać, że obserwowany trend nie jest istotnie różny od zera.

Za próg przyjmuje się często 0.05 (głównie z powodów historycznych, ale już tak się utarło). Co znaczy tyle, że uznaje się trend za nieprzypadkowy jeżeli jest mniejszy o co najwyżej 5% wartości, które można by przypadkowo zaobserwować (5% wartości to jedna na 20).

Przyjmując za próg 0.05, p-wartość którą obserwujemy dla trendu dla sezonu Przyjaciele jest wyższa, więc raczej przyjmiemy hipotezę, że ten trend nie jest istotnie różny od zera.

Podsumowanie testowania

Wyniki, które otrzymaliśmy dla serialu *Breaking Bad* są

inne niż dla serialu *Friends*.

W przypadku serialu *Breaking Bad* próbowaliśmy 99 999 razy losowo pomieszać dane by na pomieszanych zaobserwować wzrost większy, niż w przypadku oryginalnej kolejności. To się nie udało, co przekonało nas, że obserwowany trend dla *Breaking Bad* jest *istotnie większy od przypadkowego (=0)*.

W przypadku serialu *Friends* próbowaliśmy tego samego. Z tą różnicą, że udawało się dosyć często w wyniku losowych permutacji uzyskać współczynniki trendu większe, niż dla oryginalnych danych.

P-wartością nazywamy statystykę, opisującą jaka część losowo wygenerowanych trendów jest większa, niż trend w oryginalnych danych. Przyjęliśmy też, że jeżeli p-wartość będzie mała, np. mniejsza niż 0.05 (arbitralny wybór, ale się przyjęło) to uznamy, że trend jest istotny.

Tyle, że dla serialu *Friends* obserwowana p-wartość jest znacznie większa niż 0.05, przez co uznamy, że trend nie jest istotnie różny od przypadkowego ($=0$). Innymi słowy trudno rozstrzygnąć czy trend dla Przyjaciół rośnie czy maleje.

Podsumowanie instrukcji R

W tym odcinku omawialiśmy funkcje służące do testowania trendu liniowego.

```
library(PogromcyDanych)
## wybieramy dane o jednym serialu, dwie kolumny
tylkoBreakingBad <- serialsIMDB %>% filter(serial == "Breaking Bad")
## wyświetlamy pierwsze wiersze
head(tylkoBreakingBad)

## liczymy współczynniki trendu
lm(ocena~id, tylkoBreakingBad)$coef

(wspolczynnikiTrendu <- lm(ocena~id, tylkoBreakingBad))
## moduł / wartość bezwzględna współczynnika wzrostu dla losowej
(modulBreakingBad <- abs(wspolczynnikiTrendu[2]))

## funkcja replicate() powtarza drugie wyrażenie
trendy <- replicate(99999, {
  # wymieszaj losowo wartości, robi to funkcja
  tylkoBreakingBad$wymiesiane <- sample(tylkoBreakingBad$id)
  # wyznacz moduł współczynnika wzrostu dla losowej
  wspolczynnikiTrendu <- lm(wymiesiane~id, tylkoBreakingBad)
  abs(wspolczynnikiTrendu[2])
})
## modulBreakingBad jest też jedną z możliwych
trendy <- c(trendy, modulBreakingBad)
## wyświetl pierwsze 20 z tak otrzymanych wartości
head(trendy, 20)

## jak wyglądają przypadkowe współczynniki
hist(trendy, 50, col="grey", main="", las=1, border="black")
## dorysuj na czerwono gdzie jest współczynnik wzrostu
abline(v=modulBreakingBad, col="red", lwd=5)

## p-wartość dla tetstu
```

```
mean(trendy >= modulBreakingBad)
```

Zadania

- Znajdź serial o istotnym ujemnym trendem, to znaczy takim, którego oceny maleją.
- Znajdź serial (inny niż *Breaking Bad*) o istotnie dodatnim trendzie.
- [Trudne] Policz dla ilu serialu ich oceny rosną a dla ilu maleją i te wzrosty lub spadki są istotnie większe niż wynikające z przypadku.

Testy równości średnich

Przemysław Biecek @ Uniwersytet Warszawski

sezon 2 / odcinek 12

pogRomcy danych

- O czym jest ten odcinek?
- Baza danych o filmach i serialach
- Czy serial Friends jest lepszy niż serial Breaking Bed?
- Graficzne zestawienie
- Testowanie średnich - algorytm
 - Krok 1: Postaw hipotezę, określ co chcesz porównać
 - Krok 2: Określ miarę wielkości różnic
- Testowanie średnich - algorytm
 - Krok 3: Porównaj zaobserwowaną wielkość różnicy z różnicą w sytuacji gdyby grupy nie były różne
- Czy serial Friends jest lepszy niż serial Breaking Bed?
- Czy serial Friends jest lepszy niż serial Breaking Bed? - p-wartość
- Czy serial Friends jest lepszy niż serial Sherlock?

- Czy serial Friends jest lepszy niż serial Sherlock?
- Czy serial Friends jest lepszy niż serial Sherlock? - p-wartość
- Które dwa seriale różnią się istotnie?
- Zadania

O czym jest ten odcinek?

Analizując dane, często natrafiamy na sytuację, gdy mamy do porównania dwie grupy obserwacji i chcemy sprawdzić, czy te grupy różnią się między sobą.

W przypadku danych medycznych to mogą być pacjenci biorący lek A i lek B, a pytanie, które nas może nurtować to czy skuteczność obu leków jest porównywalna czy też jedne z nich jest istotnie skuteczniejszy. Analizując uprawę zbóż możemy mieć dwie odmiany i pytanie czy jedna uprawa daje wyższe plony niż druga. Podobne przykłady można mnożyć.

Zazwyczaj gdy w jednej grupie policzymy średnią określonej cechy i w drugiej grupie policzymy średnią tej cechy, to któraś z tych średnich jest większa. Ale czy różnica pomiędzy tymi średnimi jest istotna? A może jej wielkość jest pomijalna?

Jako przykład wyobraźmy sobie następującą sytuację.

Przypuśćmy, że porównuje średnie dzienne temperatury we Wrocławiu i średnie dzienne temperatury w Warszawie, i będę robił to dzień w dzień przez kolejnych 30 dni. Przypuśćmy, że średnia we Wrocławiu jest wyższa o około 0,5 stopnia Celsjusza od średniej w Warszawie. Ale czy ta różnica wynika z losowych fluktuacji temperatury czy też jest odzwierciedleniem pewnej ogólnej prawidłowości, że we Wrocławiu jest średnio cieplej?

W odpowiedzi na podobnie sformułowane pytania pojawia się statystyka. Używając takich narzędzi jak testy dla dwóch prób możemy pomóc sobie w odpowiedzi na pytanie, czy obserwowane różnice są czysto przypadkowe, czy też nie.

W tym odcinku, na przykładzie danych z serwisu IMDB (Internet Movie Database), pokażemy jak działa test permutacyjny, uniwersalne narzędzie do porównywania dwóch grup.

Baza danych o filmach i serialach

W bazie danych o filmach IMDB (Internet Movie Database) znaleźć można dane o serialach w tym średnie oceny użytkowników wystawione kolejnym odcinkom

serialu.

Przykładowo, na stronie

http://www.imdb.com/title/tt0108778/epdate?ref_=ttep_ql_4 znajdują się oceny serialu *Friends* a na

stronie http://www.imdb.com/title/tt0903747/epdate?ref_=ttep_ql_4 oceny serialu *Breaking Bad*.

Dane te zostały pobrane z Internetu i przygotowane tak, że teraz są dostępne w zbiorze danych `serialsIMDB` w pakiecie `PogromcyDanych`, który towarzyszy temu kursowi.

```
library(PogromcyDanych)
head(serialsIMDB)
```

```
##      id      serial          title
## 1    1  Breaking Bad  Cat's in the Bag
## 2    2  Breaking Bad ...And the Bag's in the R...
## 3    3  Breaking Bad           Cancer
## 4    4  Breaking Bad        Gray Man
## 5    5  Breaking Bad  Crazy Handful of Nots
##          imdbId
## 1 tt0903747
## 2 tt0903747
## 3 tt0903747
## 4 tt0903747
## 5 tt0903747
## 6 tt0903747
```

Czy serial Friends jest lepszy niż serial Breaking Bed?

Mając zbiór danych o serialach, możemy porównać oceny odcinków dwóch seriali.

Na początek wyciągnijmy ze zbioru danych o wszystkich serialach dwa podzbiory, osobno oceny dla serialu *Friends* i oceny dla serialu *Breaking Bad*

Oceny dla serialu Friends

```
ocenyFriends      <- serialsIMDB[serialsIMDB$se:  
summary(ocenyFriends)
```

	Min.	1st Qu.	Median	Mean	3rd Qu.	1
##	7.400	8.300	8.500	8.508	8.700	9

Oceny dla serialu Breaking Bad

```
ocenyBreakingBad <- serialsIMDB[serialsIMDB$se:  
summary(ocenyBreakingBad)
```

	Min.	1st Qu.	Median	Mean	3rd Qu.	1
##	7.400	8.025	8.400	8.481	8.800	10

Średnia w jednym serialu to 8.508 a w drugim 8.481.

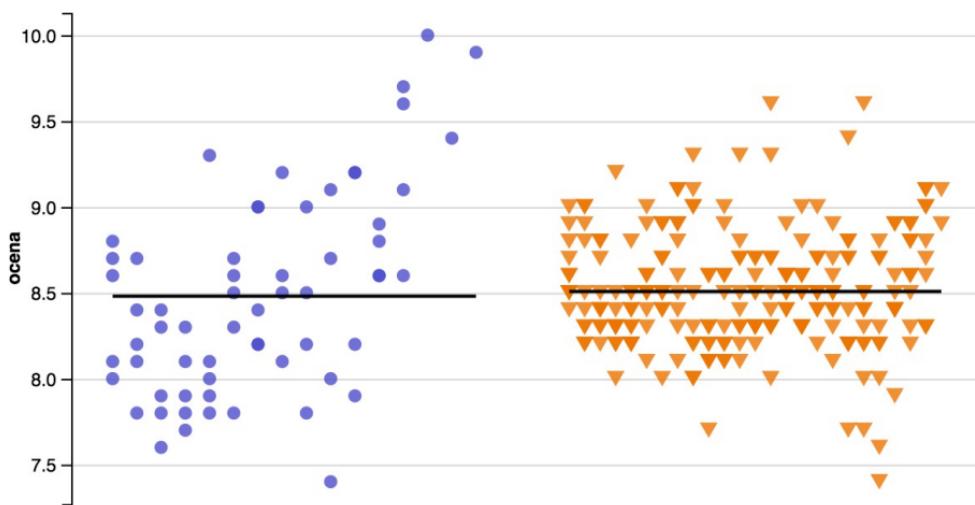
Czy to duża różnica czy mała?

Graficzne zestawienie

Pokażmy jak oceny dla obu seriali wyglądają gdy się je zestawi obok siebie porządkując oceny wzduż wspólnej osi.

Po lewej *Breaking Bad* po prawej *Friends*. Każdy punkt to ocena jednego odcinka.

Czy obserwowana różnica jest duża? A czy jest istotna statystycznie?



Testowanie średnich - algorytm

Statystycy opracowali wiele testów aby porównać

wartości w dwóch grupach. W większości przypadków te testy dają zbliżone wyniki, więc zamiast przedstawiać ich listę wyjaśnimy sposób działania na jednym przykładzie. Przedstawimy tak zwany test permutacyjny do porównywania średnich.

Pomysł na porównanie dwóch grup składa się z trzech kroków. Przedstawmy je jeden po drugim.

Krok 1: Postaw hipotezę, określ co chcesz porównać

Będziemy testować hipotezę (=przypuszczenie), czy wartości w jednym zbiorze są istotnie większe niż wartości w drugim zbiorze.

Tę hipotezę, sformujmy w następujący sposób: *Czy obserwowana różnica pomiędzy średnimi ocenami jest wystarczająco duża, by uznać ją za istotną (mieć pewność, która grupa jest większa)?*

Krok 2: Określ miarę wielkości różnic

Ustalmy, że wielkość różnicy pomiędzy grupami będziemy mierzyć za pomocą wartości bezwzględnej z różnicy pomiędzy średnimi w obu grupach.

```
## [1] 8.480645  
mean(ocenyFriends)  
## [1] 8.507983  
## moduł różnicy średnich  
(obserwowana_roznica <- abs(mean(ocenyBreaking  
## [1] 0.02733803
```

Testowanie średnich - algorytm

Krok 3: Porównaj zaobserwowaną wielkość różnicy z różnicą w sytuacji gdyby grupy nie były różne

Wciąż nie wiemy, czy `0.026` to duża różnica czy mała. Musimy ją z czymś porównać. Najlepiej by było porównać ją z wartościami, które byśmy obserwowali gdyby te dwie grupy się nie różniły.

Ale skąd wiadomo jakie byłyby różnice gdyby obie grupy się nie różniły? Możemy to oszacować w następujący sposób

- [I] Wymieszamy losowo wartości w obu grupach zachowując liczebność grup. Dzięki temu otrzymamy dwie grupy o których wiemy, że się nie różnią

istotnie (są losowo wymieszane) a jedynie różnice wynikają z losowych fluktuacji,

- [II] Policzymy jak duża jest różnica dla tych nowo wylosowanych grup,
- [III] Powtórzmy kroki [I] i [II] wielokrotnie (np. 99 999 razy) aby zobaczyć jak duże wartości może przyjmować ta różnica w sytuacji gdy obie grupy nie różnią się od siebie.

Liczenie różnic na próbach różniących się jedynie losową fluktuacją jest realizowane przez poniższy kod.

```
## połączony wektor ocen z obu seriali
ocenyRazem <- c(ocenyFriends, ocenyBreakingBad)
## ile pierwszych ocen pochodzi z Friends
liczbaFriends <- length(ocenyFriends)
## wynikiem funkcji replicate będzie 99999 wartości
roznice <- replicate(99999, {
  # wymieszaj losowo wartości pomiędzy grupami
  wymieszane <- sample(ocenyRazem)
  # losowo dobrane próby, pierwsze 'liczbaFriends'
  oceny1 <- wymieszane[1:liczbaFriends]
  oceny2 <- wymieszane[-(1:liczbaFriends)]
  # różnica pomiędzy losowymi grupami
  abs(mean(oceny1) - mean(oceny2))
})
## jedną z permutacji jest brak permutacji, dodaj
roznice <- c(roznice, obserwowana_roznica)
## wyświetli 20 różnic zaobserwowanych w przypadku
head(roznice, 20)

## [1] 0.03162104 0.05805096 0.02530496 0.053
```

```
## [7] 0.03568718 0.01535647 0.01942261 0.0391  
## [13] 0.04788561 0.03140417 0.04563567 0.0458  
## [19] 0.01942261 0.07206560
```

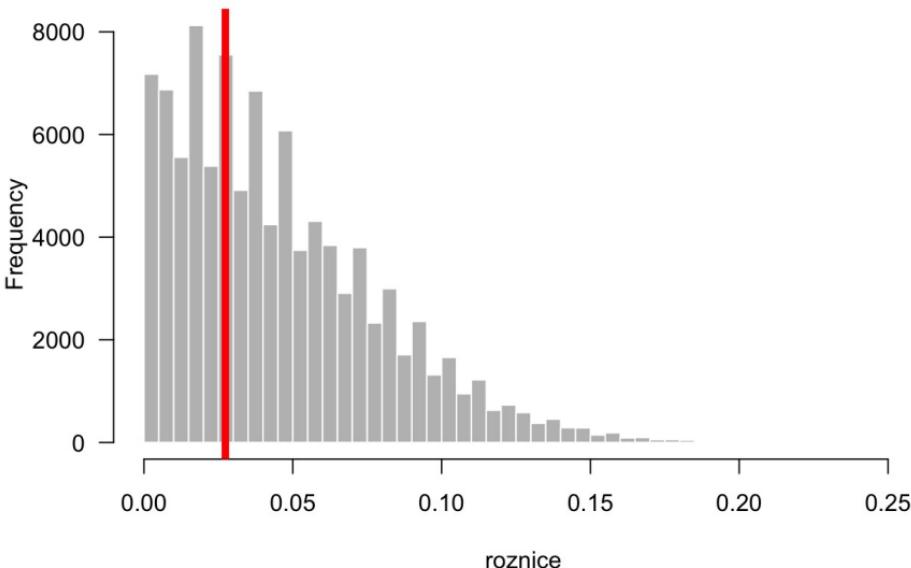
Czy serial Friends jest lepszy niż serial Breaking Bad?

Możemy teraz wartość `obserwowana_roznica` porównać z wektorem wartości `roznice`.

Wektor `roznice` opisuje jakich różnic spodziewać się możemy, gdy próby różnią się wyłącznie losowymi fluktuacjami. Gdyby `obserwowana_roznica` była znaczaco większa niż wartości z wektora `różnice`, to byłby to silny argument za tym, że różnica w średnich nie jest przypadkowa.

Porównajmy na wykresie wielkości różnic dla losowych fluktuacji (na szaro) z obserwowaną różnicą (na czerwono).

```
## wektor referencyjnych różnic po losowym pod:  
hist(roznice, 50, col="grey", main="", las=1, lwd=2)  
## obserowana różnica w średnich pomiędzy Friends  
abline(v=obserwowana_roznica, col="red", lwd=5)
```



Czy serial Friends jest lepszy niż serial Breaking Bad? - p-wartość

Wykres pomaga w zrozumieniu na ile obserwowana różnica średnich różni się od tych, wygenerowanych przez losowe fluktuacje.

Ale aby podjąć decyzje lepiej mieć jedną liczbę.
Policzmy jaka część różnic dla losowych fluktuacji jest większa lub równa niż obserwowana różnica. Taka częstość jest nazywana p-wartością.

```
## p-wartość  
mean(roznice >= obserwowana_roznica)  
## [1] 0.64292
```

Gdy próby są losowo wymieszane, a więc ich średnie, teoretycznie są równe. Czy wiedząc, że w wyniku losowych przypisać aż w 64% przypadków obserwuje się większą różnicę niż ta pomiędzy *Friends* a *Breaking Bad*, czy uznamy różnicę pomiędzy tymi dwoma serialami za istotną?

Pewnie nie. To jaki próg przyjmiemy zależy od konkretnego zastosowania, ale często za próg wybiera się 0.05. Z powodów historycznych.

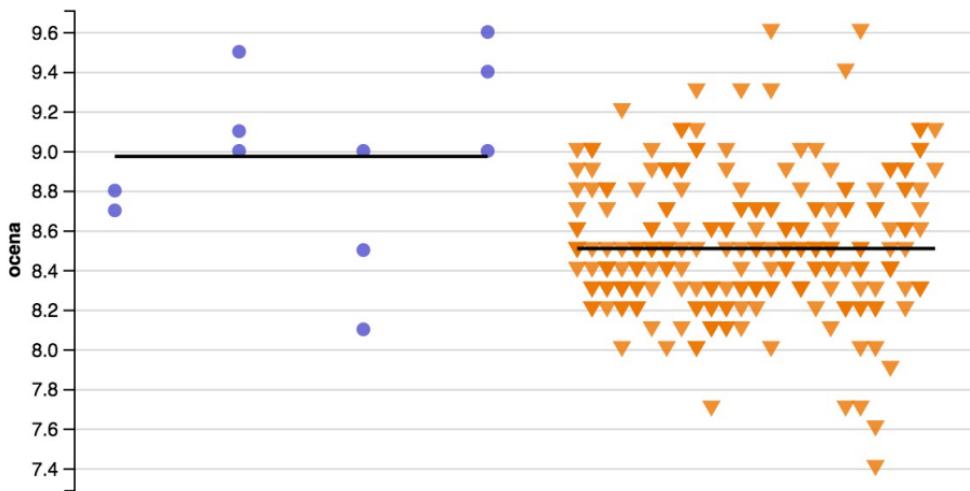
W każdym razie, jeżeli otrzymana p-wartość jest mniejsza niż 0.05, to przyjmiemy, że obserwowana wartość jest istotnie różna od przypadkowej ($=0$). Jeżeli otrzymana p-wartość jest większa równa 0.05 to przyjmiemy, że nie ma istotnych statystycznie różnic.

W przypadku part *Friends* i *Breaking Bad*, ta różnica była niewielka i raczej uznamy, że nie istotna statystycznie.

Czy serial **Friends** jest lepszy niż serial **Sherlock**?

Porównajmy teraz dwa inne seriale.

Zacznijmy od wykresu, po lewej prezentujemy serial Sherlock po prawej Friends.



Czy serial Friends jest lepszy niż serial Sherlock?

Różnica średnich pomiędzy tymi dwoma serialami wynosi 0.48. Wygląda na dużo, ale czy mamy jakiś argument, że jest to więcej niż przypadkowa fluktuacja?

```
ocenyFriends      <- serial$oceny[serial$se:ocenySherlock      <- serial$oceny[serial$se:# moduł różnicy średnich  
#(obserwowana_roznica <- abs(mean(ocenyFriends)
```

```
#> [1] 0.4647441
```

Powtórzmy test permutacyjny dla tych dwóch seriali. Tym razem obserwowana różnica jest wyższa niż praktycznie wszystkie wartości wygenerowane przez test permutacyjny.

```
## połączony wektor ocen z obu seriali
ocenyRazem <- c(ocenyFriends, ocenySherlock)
## ile pierwszych ocen pochodzi z Friends
liczbaFriends <- length(ocenyFriends)
## permutacyjny test
liczbaFriends <- length(ocenyFriends)
roznice <- replicate(99999, {
  wymieszane <- sample(ocenyRazem)
  oceny1 <- wymieszane[1:liczbaFriends]
  oceny2 <- wymieszane[-(1:liczbaFriends)]
  # różnica pomiędzy losowymi grupami
  abs(mean(oceny1) - mean(oceny2))
})
## dodajemy obserwowaną różnicę
roznice <- c(roznice, obserwowana_roznica)
```

Czy serial **Friends** jest lepszy niż serial **Sherlock**? - p-wartość

Policzywszy różnice dla losowych fluktuacji, możemy zestawić je graficznie z różnicą średnich dla Sherlocka i Friends.

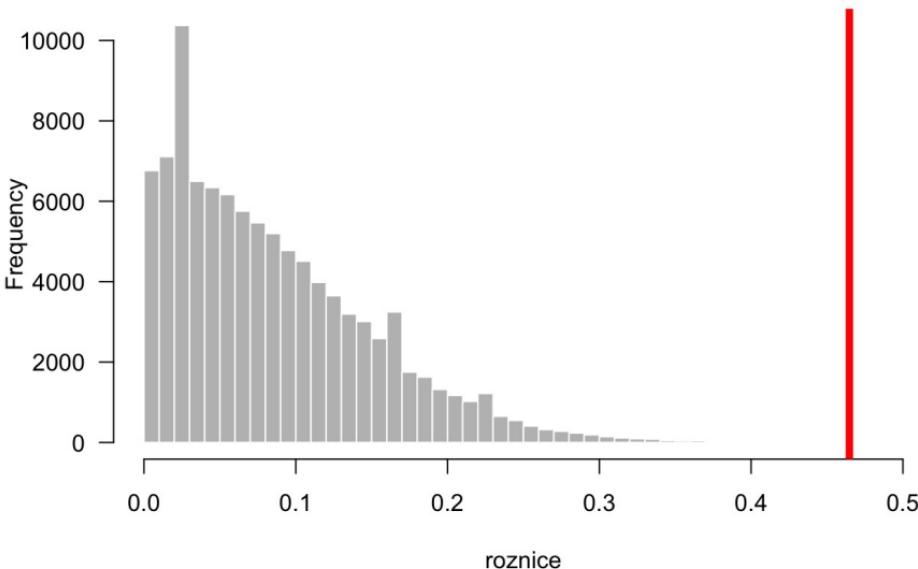
Tym razem p-wartość jest bardzo mała, obserwowana różnica średnich pomiędzy serialami jest znacznie większa niż te, które widać dla losowego podziału na grupy.

Uznamy więc, że Sherlock ma istotnie wyższą średnią ocen niż Friends.

```
## p-wartość dla różnicy średnich
mean(roznice >= obserwowana_roznica)

## [1] 2e-05

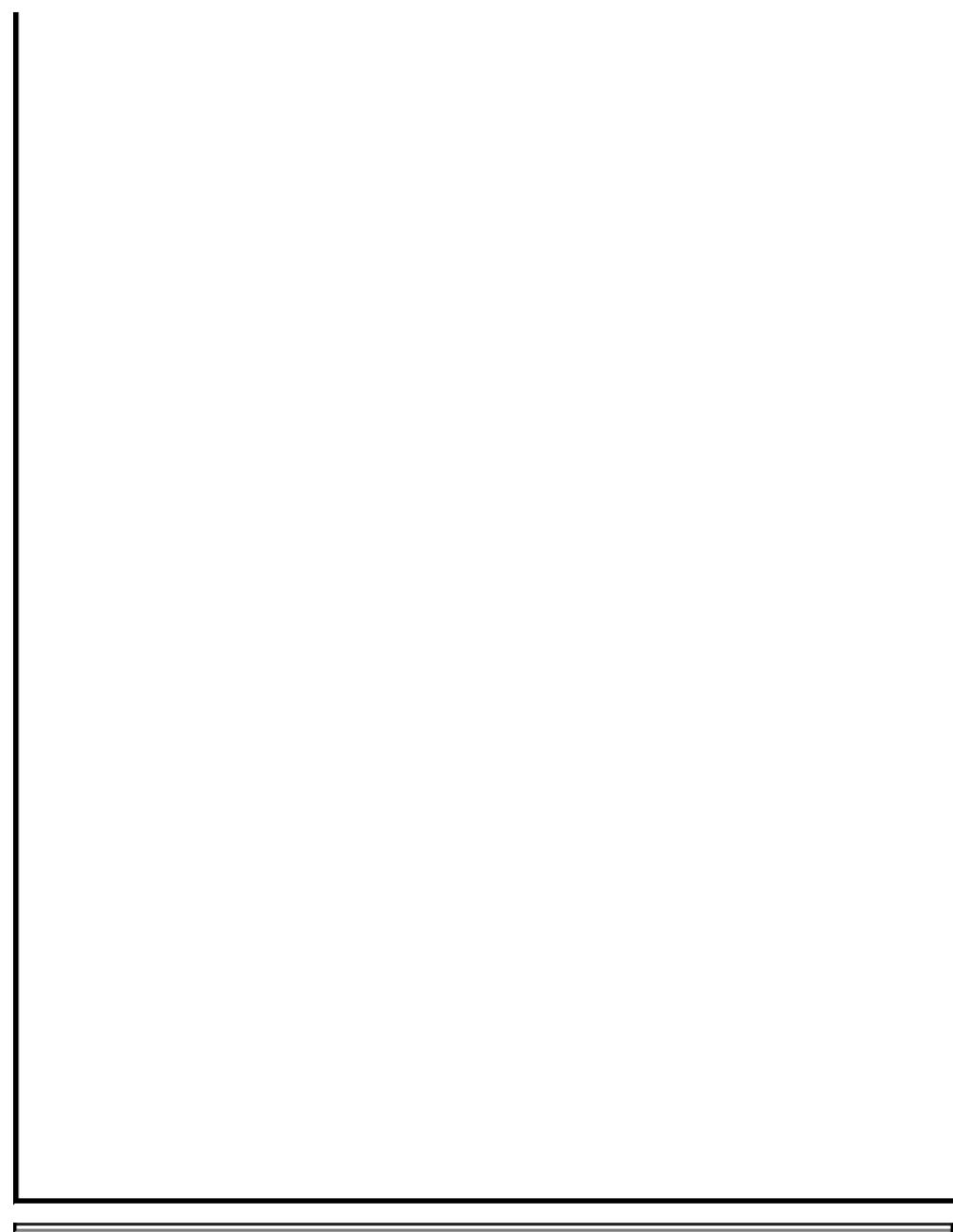
## wektor referencyjnych różnic po losowym podziale
hist(roznice, 50, col="grey", main="", las=1, border="black")
## obserwowana różnica w średnich pomiędzy Friends i Sherlockiem
abline(v=obserwowana_roznica, col="red", lwd=5)
```



Które dwa seriale różnią się istotnie?

Używając poniższej aplikacji można porównać dowolne dwa seriale oraz ocenić, czy różnica jest statystycznie istotnie różna.

Moved to: <http://mi2.mini.pw.edu.pl:8080/SmarterPoland>



Zadania

- Znajdź dwa seriale (inne niż Friends i Sherlock), które różnią się istotnie, przyjmując za próg istotności 0.05.
- Znajdź dwa seriale (inne niż Friends i Breaking Bad), które nie różnią się istotnie, przyjmując za próg istotności 0.05.

Regresja prosta

Przemysław Biecek @ Uniwersytet Warszawski

sezon 2 / odcinek 8

pogRomcy danych

- Wprowadzenie
- Kim był Francis Galton?
- Czy wzrost rodziców się liczy?
- Czy wzrost rodziców się liczy?
- Bez szumu mało widać
- Jaka to zależność?
- Jaka to zależność?
- Model liniowy
- Ocena współczynników
- Ocena współczynników
- Jak wygląda zależność pomiędzy wzrostem dziecka a rodzica
- Jak wygląda zależność pomiędzy wzrostem dziecka a rodzica - wykres
- Który model wybrać?
- Zadania

Wprowadzenie

Jak myślicie, czy wzrost jest dziedziczny?

Czy dzieci wysokich rodziców są wysokie?

Pewnie tak, ale mam znajomego, który jest niewysoki, 165cm, ale ma bardzo wysokie dziecko.

Czy to wyjątek czy reguła?

Raczej nie reguła, dzieci wysokich rodziców zazwyczaj są wyższe niż średnia (nawet jeżeli nie jest to dziedziczenie przez geny ale inne poza genetyczne czynniki).

Więc jak to jest, jaka część wzrostu dziecka jest zależna od wzrostu rodzica?

Dziś zmierzymy się z tym pytaniem.

Przyjrzymy się bardzo znanemu zbiorowi danych, który przedstawia zależność pomiędzy wzrostem rodzica a dziecka.

Zaczniemy od przyjrzenia się temu, jak wygląda zależność, następnie odpowiadając na pytanie jaka część wzrostu jest odziedzicjalna a na koniec spróbujemy oszacować jaki będzie wzrost moich dzieci.

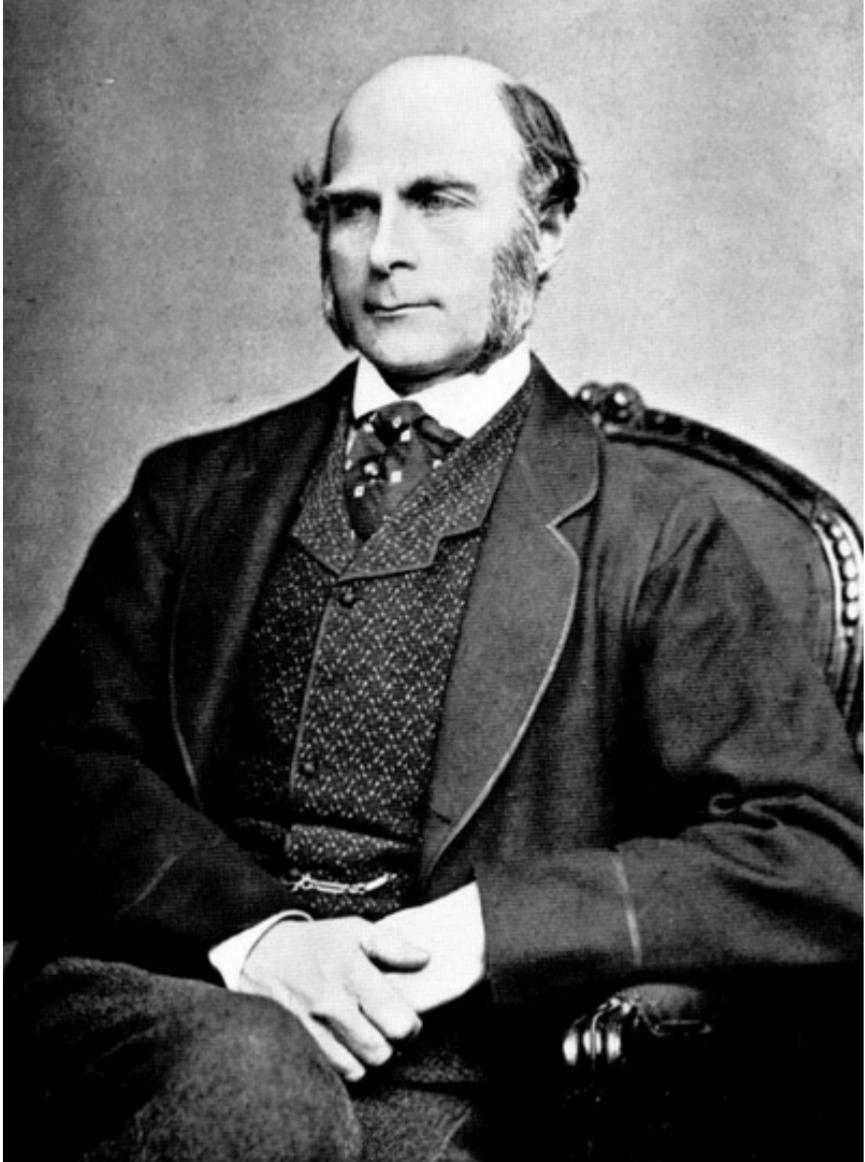
Kim był Francis Galton?

Podstawy matematyczne liniowych metod analizy danych powstały na przełomie XVIII i XIX wieku. Metody te były z początku wykorzystywane w nawigacji oraz astronomii. Dzisiaj najczęściej modele liniowe określa się terminem regresja.

Termin regresja został użyty po raz pierwszy przez Sir Francis Galtona jeszcze w XIX wieku. Francis Galton był człowiekiem renesansu, zajmującym się zarówno antropologią, statystyką, jak i wieloma innymi naukami. W 1886 roku opublikował pracę „Regression towards mediocrity in hereditary stature” w *The Journal of the Anthropological Institute of Great Britain and Ireland*, w której przedstawił wyniki badań nad dziedziczeniem wzrostu.

Galton zauważył, że synowie bardzo wysokich ojców są średnio wyżsi niż synowie niższych ojców, ale ich średni wzrost jest niższy niż średni wzrost ojców. Podobnie wzrost synów niskich ojców jest bliższy średniej w populacji. To zjawisko nazwał „dążeniem do przeciętności” lub „tendencją do przeciętności” (w języku angielskim ten współczynnik nazywa się *regression toward the mean*).

Galton zaproponował równanie opisujące zależność między wzrostem synów i ojców lub równoważnie opisujące regresję wzrostu z pokolenia na pokolenie w kierunku wartości przeciętnej (dziś nazwalibyśmy to równaniem regresji).



Czy wzrost rodziców się liczy?

Naszą przygodę z regresją liniową również rozpoczęmy od danych zebranych przez Galtona. Zbiór danych nazywa się `galton` i jest dostępny w pakiecie `PogromcyDanych`. Zobaczmy kilka pierwszych obserwacji z tego zbioru.

```
## w tym pakiecie znajduje się zbiór danych ga:  
library(PogromcyDanych)  
## pierwsze 6 wierszy  
head(galton)
```

	syn	rodzic
## 1	156.7	179.1
## 2	156.7	174.0
## 3	156.7	166.4
## 4	156.7	163.8
## 5	156.7	162.6
## 6	158.0	171.4

W kolumnie `rodzic` przedstawiony jest średni ważony wzrost rodziców (średnia wzrostu ojca i 1,08 wzrostu matki), a w kolumnie `syn` wzrost dorosłego syna. Wzrosty podane są w centymetrach (oryginalnie Galton przedstawał wzrost w calach, dla nas jednak wygodniejsze są centometry).

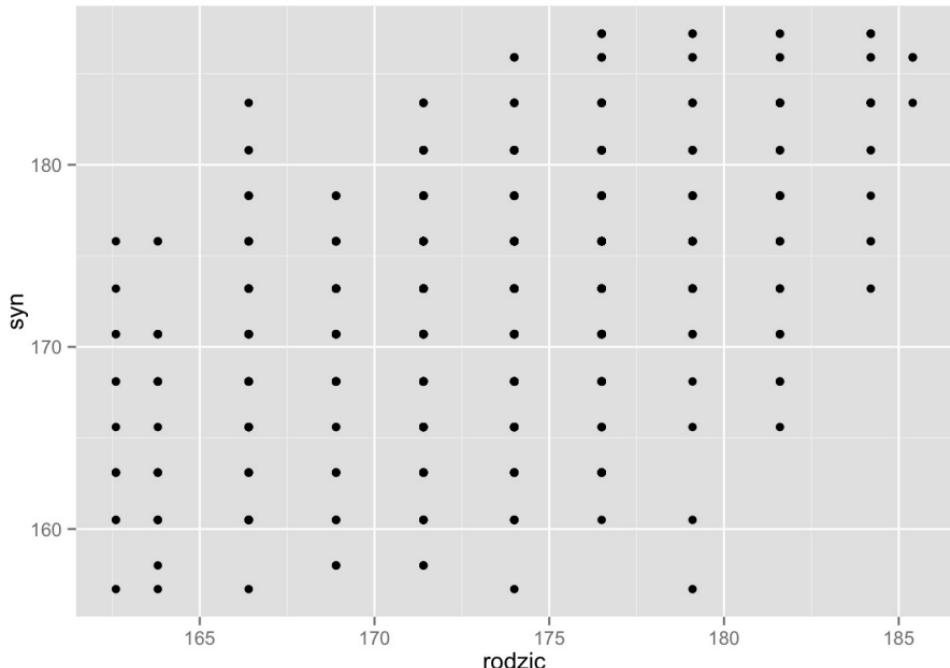
Czy wzrost rodziców się liczy?

Pytanie, które interesowało Galtona, jak również interesuje nas, to czy jest zależność pomiędzy wzrostem rodziców a dziecka.

Modelowanie zawsze powinno rozpoczynać się od wykresu. Zobaczmy więc jak wygląda zależność pomiędzy wzrostem dziecka a średnim ważonym wzrostem rodzica.

Do przedstawienia zależności wykorzystujemy pakiet `ggplot2` i funkcję `ggplot()`. Więcej informacji o tym, jak z tej funkcji korzystać znajduje się w pierwszej części tego sezonu (poświęconej wizualizacji).

```
ggplot(galton, aes(x=rodzic, y=syn)) +  
  geom_point()
```



Bez szumu mało widać

Ponieważ dane są zbierane z dokładnością do połowy cala (około 12mm), punkty dla różnych wierszy pokrywają się. Jedna kropka może oznaczać wiele par (wzrost rodzica, wzrost dziecka).

Oglądając takie dane (zbieranie z niewielką precyją), na potrzeby prezentacji warto je zaburzyć małym zastrzykiem szumu. Zrobimy to argumentem `position = "jitter"`.

```
ggplot(galton, aes(x=rodzic, y=syn)) +  
  geom_point(position = "jitter")
```

syn

180

170

160

165

170

175

180

185

rodzic

Jaka to zależność?

Wygląda na to, że jakaś zależność występuje, ale jaka? Jedną z możliwości jest policzenie średniego wzrostu dziecka dla każdej grupy średniego wzrostu rodziców.

Używając operatora `%>%` (szerzej omówionego w pierwszym sezonie), średni wzrost dziecka dla każdej grupy wzrostów rodziców wyznacza się następująco.

```
srednie <- galton %>%
  group_by(rodzic) %>%
```

```
  summarise(srednia = mean(syn))  
## wyświetl te średnie  
średnie  
  
## Source: local data frame [11 x 2]  
##  
##    rodzic   srednia  
## 1   162.6 165.8786  
## 2   163.8 166.1000  
## 3   166.4 169.4273  
## 4   168.9 170.3231  
## 5   171.4 171.6692  
## 6   174.0 172.7639  
## 7   176.5 174.5082  
## 8   179.1 176.7721  
## 9   181.6 178.0070  
## 10  184.2 182.5842  
## 11  185.4 185.2750
```

Jaka to zależność?

Narysujmy na niebiesko wektor średnich „w grupach wzrostów rodziców”. W tle pokażemy wszystkie obserwacje ze zbioru danych.

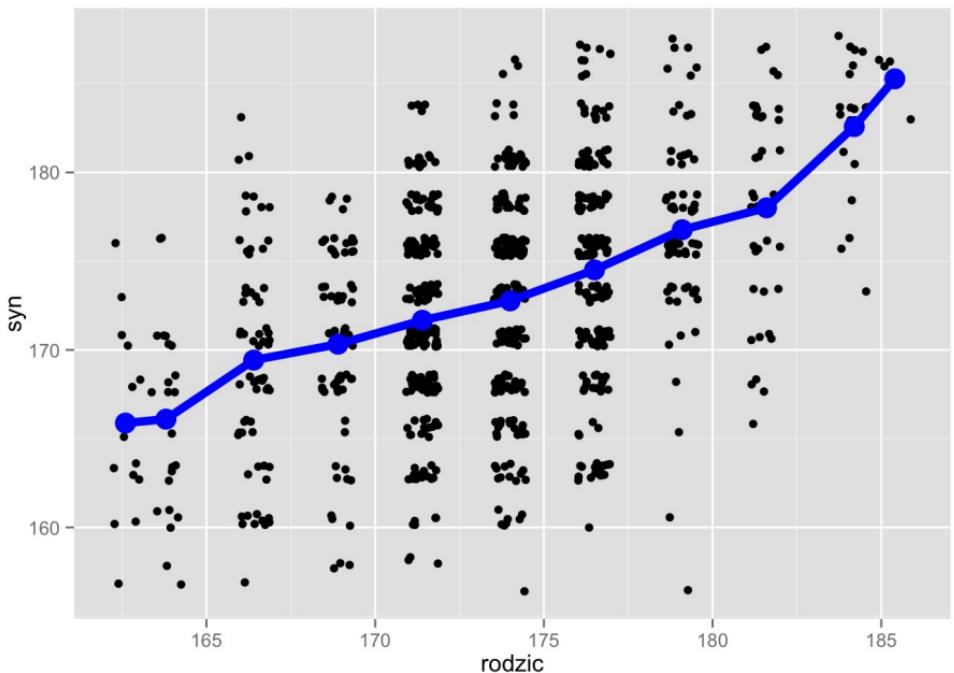
Pięknie! Widzimy, że im wyżsi rodzice tym średnio wyższe dzieci.

Jednak posługiwanie się samymi średnimi jest dosyć niewygodne. Aby opisać zależność posługując się średnimi w każdej grupie, należałoby podać średnią dla

każdego możliwego wzrostu rodziców.

Co więcej średnie w grupach bardzo wysokich lub bardzo niskich rodziców wyznaczane są na bazie jedynie tylko kilku obserwacji, a więc obarczone są mniejszą dokładnością wynikającą z małej próby.

```
ggplot(galton, aes(x=rodzic, y=syn)) +  
  geom_point(position = "jitter") +  
  ## poniższa funkcja dorysowuje średnie w grupach  
  geom_line(data=srednie, aes(x=rodzic, y=srednia))  
  geom_point(data=srednie, aes(x=rodzic, y=srednia), size=2)
```



Model liniowy

Wygodniej by było opisać zależność pomiędzy wzrostem rodzica a średnim wzrostem dziecka za pomocą mniejszej liczby parametrów, niż pamiętanie średniego wzrostu dzieci dla każdego możliwego wzrostu rodzica.

Założymy, że można tę zależność opisać zależnością liniową, a więc zależnością

$$\text{średni.wzrost.dziecka} = b_0 + \text{średni.wzrost.rodziców} * b_1$$

gdzie b_0 i b_1 to pewne nieznane wartości.

Metoda regresji liniowej pozwala wyznaczyć te nieznane wartości. Dokładny opis algorytmu wyznaczania tych wartości jest opisany w kolejnym punkcie, poniżej przyjrzyjmy się, jak można to zrobić w programie R.

Ocena współczynników

Proces wyznaczania nieznanych wartości w oparciu o model i dane nazywa się estymacją.

W programie R, do estymacji można wykorzystać funkcję `lm()` (skrót od *linear model*).

Dla modelu $\text{średni.wzrost.dziecka} = b_0 + \text{średni.wzrost.rodziców} * b_1$

lub używając nazw zmiennych z naszego zbioru danych \ (syn = b_0 + rodzic * b_1)

Funkcja estymująca/szacująca wartości współczynników \ (b_0) i \ (b_1) wygląda następująco.

```
## osoby lubiące operator %>% mogą też napisać
## galton %>% lm(syn ~ 1 + rodzic, data = .)
lm(syn ~ 1 + rodzic, data = galton)

##
## Call:
## lm(formula = syn ~ 1 + rodzic, data = galton)
##
## Coefficients:
## (Intercept)      rodzic
##       60.8130        0.6463
```

Funkcja `lm()` przyjmuje dwa argumenty. Pierwszy to formuła opisująca przyjęty model, drugi argument wskazuje zbiór danych.

Formuła składa się z lewej strony i prawej, rozdzielonej znakiem `~`. Lewa strona opisuje zmienną, którą chcemy opisać, tzw. zmienną objaśnianą, a prawa strona składa się z zmiennych, które wykorzystujemy do opisania lewej strony, czyli zmiennych objaśniających.

Zapis `1 + rodzic` oznacza, że średni wzrost dziecka chcemy opisać za pomocą wyrażenia $(1 * b_0 + \text{rodzic} * b_1)$, w którym należy wyznaczyć (b_0) i (b_1) .

Ocena współczynników

Wynikiem funkcji `lm()` jest szczegółowy opis dopasowanego modelu. Jeżeli interesują nas jedynie oceny (b_0) i (b_1) , możemy się odwołać do tych ocen przez funkcję `coef()`.

```
modelSynRodzic <- lm(syn ~ 1 + rodzic, data=galton)
coef(modelSynRodzic)
```

```
## (Intercept)      rodzic
##   60.812990     0.646291
```

A więc otrzymujemy model

$$\text{średni.wzrost.dziecka} = 60.813 + \text{wzrost.rodzica} * 0.6463$$

W formule możemy pominąć człon `1+`. Jest on domyślnie dodawany do formuły, dlatego taki sam wynik otrzymamy stosując następującą instrukcję.

```
## wartość 1 dodawana jest automatycznie
modelSynRodzic <- lm(syn ~ rodzic, data=galton)
coef(modelSynRodzic)
```

```
## (Intercept)      rodzic  
## 60.812990      0.646291
```

Jak wygląda zależność pomiędzy wzrostem dziecka a rodzica

W powyższym toku rozumowania rozważaliśmy kilka alternatywnych modeli. Podsumujmy je tutaj.

1. Model, w którym średni wzrost dziecka jest równy średniemu ważonemu wzrostowi rodzica, \ (średni.wzrost.dziecka = wzrost.rodziców\)
2. Model, w którym nie ma zależności pomiędzy wzrostem rodziców a dzieci, a więc, że dla każdego wzrostu rodziców, średni wzrost dzieci jest taki sam, \ (średni.wzrost.dziecka = b_0\)
3. Model, w którym wzrost dziecka jest liniowo zależny od wzrostu rodziców \ (średni.wzrost.dziecka = b_0 + wzrost.rodziców * b_1\)
4. Model, w którym średni wzrost dziecka jest różny dla każdego wzrostu rodzica (niezależnie w każdej grupie rodziców liczymy średni wzrost dzieci).

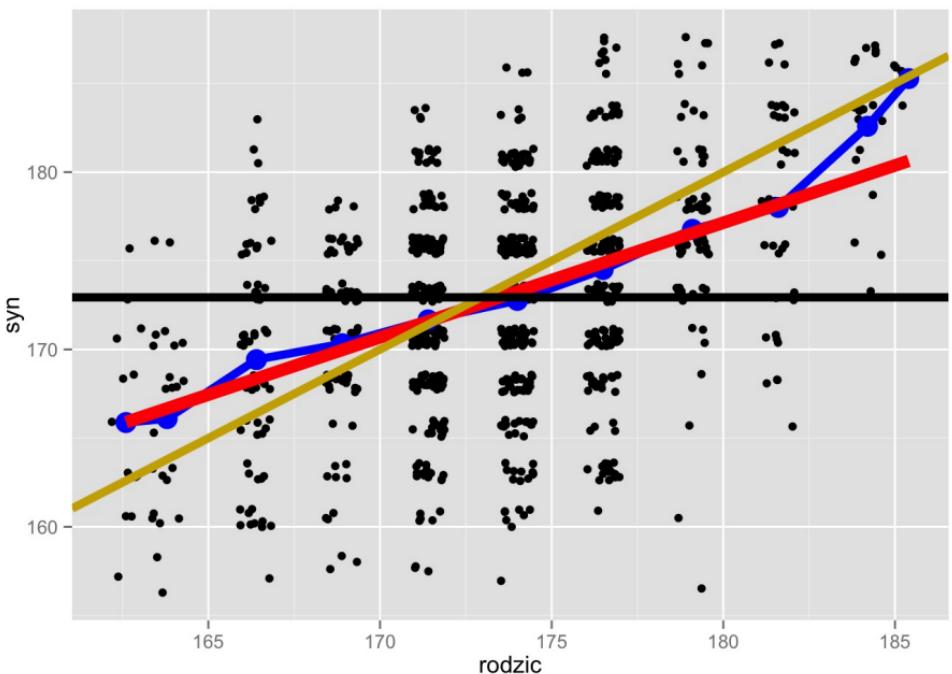
Modele te zostały uporządkowane ze względu na liczbę nieznanych parametrów. Pierwszy z powyższych modeli

nie ma żadnego parametru, kolejny ma jeden parametr, kolejny dwa a ostatni 11 parametrów.

Jak wygląda zależność pomiędzy wzrostem dziecka a rodzica - wykres

Pokażmy jak każdy z tych czterech modeli wygląda na tle danych. Oznaczenia kolorów: pierwszy model to kolor złoty, drugi - czarny, trzeci - czerwony, czwarty - niebieski.

```
ggplot(galton, aes(x=rodzic, y=syn)) +  
  geom_point(position = "jitter") +  
  # model 4, dla każdej grupy wzrostu rodziców  
  geom_line(data=srednie, aes(x=rodzic, y=srednia)) +  
  geom_point(data=srednie, aes(x=rodzic, y=srednia)) +  
  # model 3, do danych dopasowujemy trend liniowy  
  geom_smooth(method="lm", se=FALSE, size=3, color="red") +  
  # model 2, wzrost dziecka jest taki jak wzrost rodzica  
  geom_abline(size=2, color="gold3") +  
  # model 1, wzrost dziecka nie zależy od wzrostu rodzica  
  geom_abline(size=2, slope=0, intercept=mean(syn))
```



Który model wybrać?

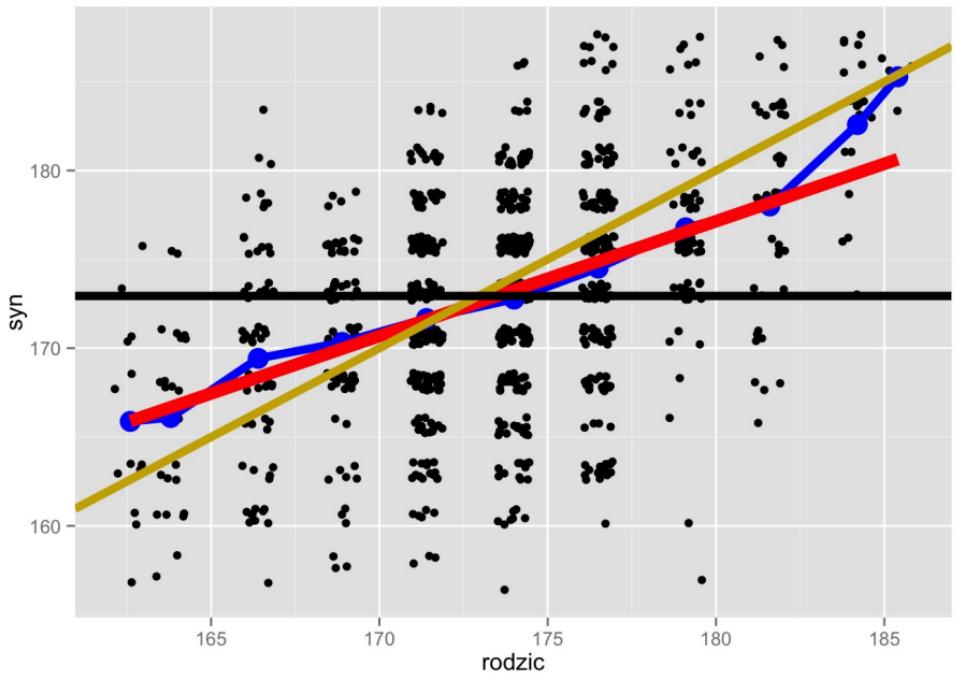
Zależność wzrostu pomiędzy dzieckiem a rodzicem można opisywać różnymi modelami. Który wybrać?

Z jednej strony zbyt proste modele (1 i 2) nie opisują dobrze danych, widać na poniższym wykresie, że występuje dla nich systematyczny błąd.

Z drugiej strony, zbyt złożone modele (4) powodują, że trudno nam zrozumieć całą relację. Co z tego, że mamy średnią dzieci dla rodziców o wzroście 170 i 172 cm,

skoro nie potrafimy tej wiedzy rozszerzyć na rodziców o wzroście 171 cm?

Praca z modelowaniem to wybieranie modelu, który do danych pasuje dobrze, ale nie za dobrze. Często (nie zawsze) sprawdza się model liniowy.



Zadania

- Dla każdego wzrostu rodzica ze zbioru danych galton wyznacz o ile średnio niższe są dzieci od rodziców

- Mając wyznaczone parametry $b0$ i $b1$ policz oceny średniego wzrostu dziecka dla każdego średniego wzrostu rodzica
- Wyznacz różnice pomiędzy ocenami z modelu liniowego a średnimi licznymi osobno dla każdej grupy rodziców
- W pakiecie `PogromcyDanych` udostępniony jest również zbiór danych `pearson` zebrany przez Pearsona. W tym zbiorze danych zebrane są wzrosty ojców i synów.

Dla tego zbioru danych wyznacz model regresji liniowej oraz narysuj zbiór danych z zaznaczoną krzywą regresji liniowej.

Czy parametry regresji liniowej w zbiorze Pearsona (z wysokością ojca) różnią się od parametrów ze zbioru Galtona (ze średnią wysokością rodziców)?

Regresja multiplikatywna

Przemysław Biecek @ Uniwersytet Warszawski

sezon 2 / odcinek 11

pogRomcy danych

- Regresja liniowa - kontynuacja
- Wczytanie danych
- Jaka to zależność?
- Jaka to zależność?
- Jeżeli nie liniowa, to jaka?
- Może logarytmiczna / multiplikatywna
- Logarytmiczna / multiplikatywna w R
- Zależność kawałkami liniowa
bez wyrazu wolnego
- Zestawienie modeli
- Zadania

Regresja liniowa - kontynuacja

W poprzednim odcinku przedstawialiśmy model dla liniowej zależności pomiędzy wzrostem dzieci a rodziców.

Ale analizując rzeczywiste dane, często możemy spotkać sytuacje w których zależność pomiędzy zmiennymi nie wygląda na liniową.

Również w takich sytuacjach regresja jest użytecznym narzędziem, ponieważ często zależności można sprowadzić do liniowych wykonując transformacje zmiennych.

Jak?

O tym poniżej.

W tej części będziemy bazować na przykładach dotyczących cen samochodów. Zaczniemy więc pracę od wczytania danych.

Wczytanie danych

W pakiecie `PogromcyDanych` dostępny jest zbiór danych `auta2012`, na którym już pracowaliśmy w pierwszym sezonie. Znajdują się w nim oferty sprzedaży aut bardzo różnych marek.

W tym odcinku podejmiemy się analizy statystycznej tego zbioru danych. Ale analizowanie wszystkich ofert w jednym worku nie ma sensu. Dlatego dalsze analizy przeprowadzimy wyłącznie na podzbiorze danych dotyczącym marki Volkswagen.

Dlaczego tej? Jest ona najpopularniejsza w zebranych ogłoszeniach, jest dla niej najwięcej obserwacji. A co nie bez znaczenia, również wyniki wychodzą ciekawe.

Każdy wiersz tego zbioru danych opisuje jedną ofertę sprzedaży auta. Każda kolumna opisuje jedną cechę / właściwość tej oferty.

```
## w tym pakiecie znajduje się zbiór danych aut
library(PogromcyDanych)
## funkcja filter pozostawia wybrane wiersze ze
volkswagen <- auta2012 %>%
  filter(Marka == "Volkswagen")
## pierwsze dwa wiersze
head(volkswagen, 2)
```

	Cena	Waluta	Cena.w.PLN	Brutto.netto	KM
## 1	79900	PLN	79900	netto	175
## 2	13600	PLN	13600	brutto	NA
	Liczba.drzwi	Pojemnosc.skokowa	Przebieg.w		
## 1	4/5	2500	157		
## 2	4/5	1900	1950		
	Rok.produkcji		Kolor	Kraj.akti	
## 1	2007	niebieski-metallic			
## 2	2001				
	Kraj.pochodzenia	Pojazd.uszkodzony	Skrzyn:		

```
## 1          Polska
## 2          Niemcy
##           Status.pojazdu.sprowadzonego
## 1
## 2 przygotowany do rejestracji / oplacony
##
## 1 ABS, el. szyby, el. lusterka, klimatyzacja
## 2
```

Jaka to zależność?

Zobaczmy jak wygląda zależność ceny od wieku Volkswagena.

W naszym zbiorze danych nie ma zmiennej wiek, jest jedynie rok produkcji. Wiedząc, że te oferty były zbierane w styczniu 2012 roku łatwo nam będzie wyznaczyć wiek auta z równania: wiek = 2012 - rok produkcji.

Zacznijmy więc od wyznaczenia zmiennej wiek (funkcja `mutate()` w poniższym przykładzie), a następnie usunięcia aut ponad 20 letnich (funkcja `filter()`, bardzo starych aut jest niewiele, są nietypowe i będą nam przeszkadzać w modelowaniu).

```
mlodeVolkswageny <- volkswagen %>%
  mutate(Wiek = 2012 - Rok.produkcji) %>%
  filter(Wiek < 20)
```

Wykorzystamy funkcję `lm()`, którą poznaliśmy w

poprzednim odcinku, do wyznaczenia liniowej zależności pomiędzy ceną a wiekiem auta.

```
modelCenaWiek <- lm(Cena.w.PLN ~ Wiek, data=mlodeVolkswageny)
## współczynniki modelu
coef(modelCenaWiek)

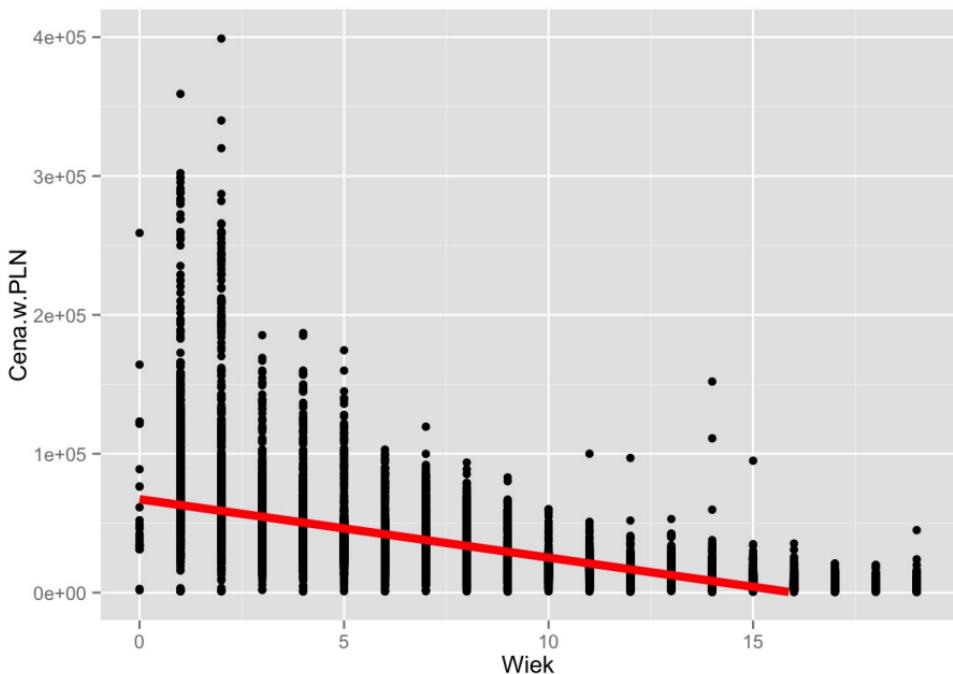
## (Intercept)           Wiek
##   67429.597     -4219.158
```

Jaka to zależność?

Mając współczynniki, możemy ocenić, że średnia cena nowo używanego auta według modelu liniowego to 67430 oraz co roku auto traci średnio -4220 pln na wartości. Te dwie liczby są użyteczne, ale jeszcze więcej można odczytać z wykresu.

Narysujemy więc obie zmienne dorysowując do nich linię trendu liniowego.

```
ggplot(mlodeVolkswageny, aes(y=Cena.w.PLN, x = Wiek)) +
  geom_point() +
  geom_smooth(method="lm", col="red", size=2) +
  # funkcja ylim() ograniczamy zakres na osiach
  ylim(0, 400000)
```



Gdyby bezrefleksyjnie przyjąć wyniki estymacji, okazałoby się, że zależność pomiędzy ceną a wiekiem opisana jest przez równanie

$$\backslash(\text{Średnia.cena} = 67640.352 - 4236.639 * \text{Wiek}\backslash)$$

Nieufność powinna w nas wzbudzić obserwacja, że podstawiając do tego wzoru za zmienną wiek wartość 16 lub więcej lat, okaże się, że otrzymujemy oszacowanie średniej ceny ujemne. Jest to jeden z sygnałów, że coś jest nie tak z naszym modelem.

Co możemy z tym zrobić?

Jeżeli nie liniowa, to jaka?

Gdy się zastanowić, łatwo dojść do wniosku, że auta tracą na wartości najwięcej w pierwszych latach, gdy są najdroższe.

Starsze auta są już tańsze i mniej tracą na wartości. Nie dotyczy to antyków, ale analizowane volkswageny nie są antykami (dlatego mamy tylko auta młodsze niż 20 lat).

Model, który możemy rozważyć, to model w którym co roku auta tracą pewien procent swojej ceny.

Taką zależność można opisać jako

$$\backslash(Cena.w.roku.i+1 = Cena.w.roku.i * (1-procent) = Cena.w.roku.i * wsp\backslash)$$

i równoważnie

$$\backslash(Cena.w.roku.i = Cena.w.roku.0 * (1-procent)^i = Cena.w.roku.0 * wsp^i\backslash)$$

Takie zależności nazywamy multiplikatywnymi. Można je sprowadzić do zależności liniowych logarytmując obie strony (Tak, logarytmy się przydają).

$$log(Cena.w.roku.i) = log(Cena.w.roku.0) + log(wsp) * i$$

czyli korzystając ze nazw zmiennych ze zbioru danych

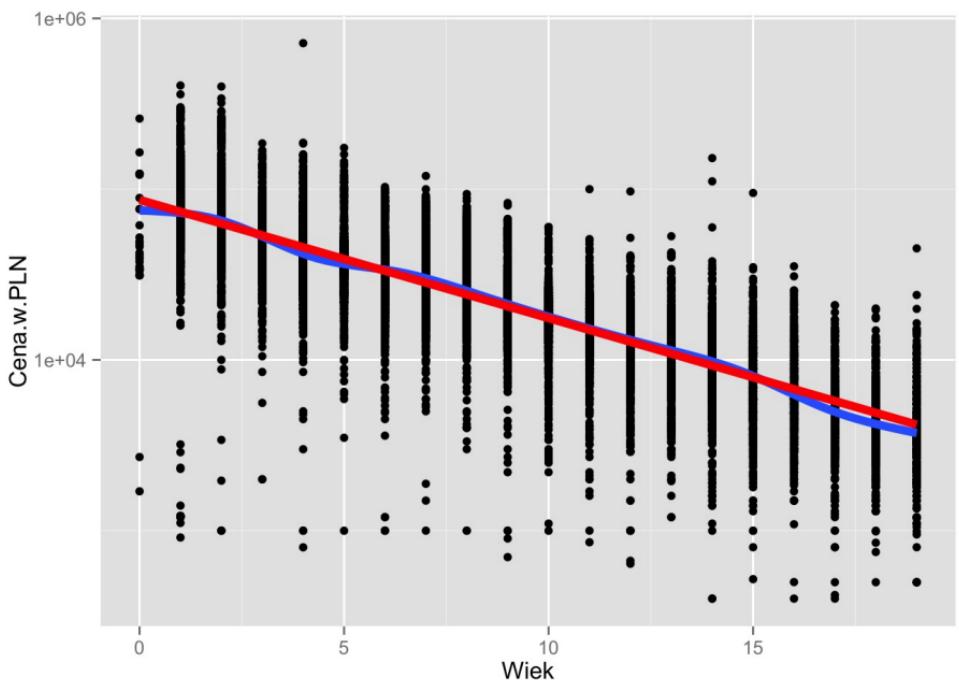
$$\log(\text{Cena.w.PLN}) = b_0 + b_1 * \text{Wiek}$$

A to już wygląda jak model liniowy!

Może logarytmiczna / multiplikatywna

Zobaczmy najpierw, jak te dane wyglądać będą, jeżeli ceny przedstawimy w skali logarytmicznej. Dodając funkcję `scale_y_continuous()` wprowadzamy transformacje na osi OY (w tym przypadku logarytmiczną). Do punktów dorysujemy linię regresji korzystając z funkcji `geom_smooth()`.

```
ggplot(mlodeVolkswageny, aes(y=Cena.w.PLN, x =  
  geom_point() +  
  geom_smooth(size=2) + geom_smooth(method="lm"  
  scale_y_continuous(trans="log10")  
  
## geom_smooth: method="auto" and size of large
```



Ciekawe. Po zlogarytmowaniu danych, zależność pomiędzy logarytmem ceny a wiekiem wygląda na idealnie liniową.

Logarytmiczna / multiplikatywna w R

Aby zbudować model dla logarytmu ceny należy wpierw musimy ten logarytm policzyć

```
## dodajemy nową kolumnę logCena.w.PLN  
mlodeVolkswageny <- mlodeVolkswageny %>%  
  mutate(logCena.w.PLN = log10(Cena.w.PLN))
```

```
## budujemy model na logarytmach
modelCenaWiek <- lm(logCena.w.PLN ~ Wiek, data=
wsp <- coef(modelCenaWiek)
wsp

## (Intercept)           Wiek
## 4.93723746 -0.06927818
```

Jeżeli przejść teraz ze skali logarytmicznej (logarytm dziesiętny) na oryginalną skalę, to okaże się, że średnia cena wyjściowa to $(10^{4.9372375}) = 8.6544099 \times 10^4$.

Drugi współczynnik wynosi -0.0692782 , co oznacza, że cena w kolejnym roku to 0.8525538 ceny z poprzedniego roku, czyli przeciętnie, corocznie auto traci 15% swojej ceny.

Zależność kawałkami liniowa

Są sytuacje, w których zależność pomiędzy dwiema zmiennymi można opisać nie tyle przez linię prostą co przez łamana, czyli krzywą prostą na odcinkach.

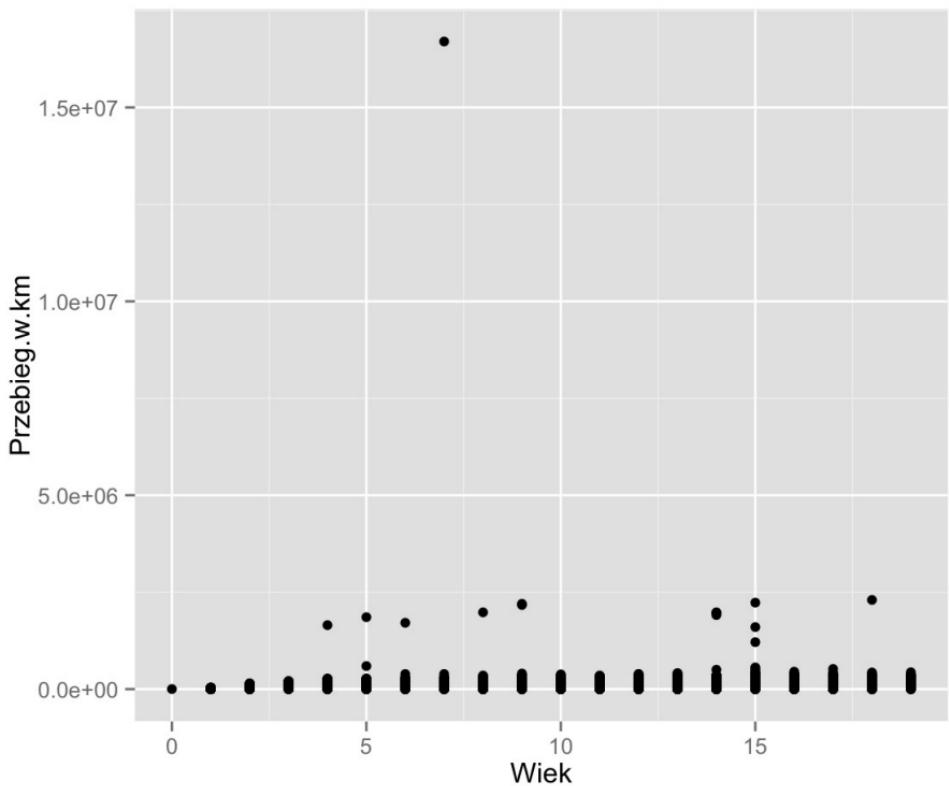
Za przykład takiej zależności posłuży nam zależność pomiędzy przebiegiem a wiekiem auta.

Zanim będziemy kontynuować jakiekolwiek modelowanie musimy wpierw przyjrzeć się danym.

W modelowaniu obowiązuje żelazna zasada *śmieci na wejściu — śmieci na wyjściu* (ang. *garbage in, garbage out*). Bez względu na to jakie metody będziemy wykorzystywać, jeżeli w danych są błędy, to nasze modelowanie prowadzić może do bezsensownych wniosków.

Zobaczmy jak wygląda przebieg i wiek aut.

```
ggplot(mlodeVolkswageny, aes(y=Przebieg.w.km, :  
    geom_point()
```



Przyglądając się dokładniej temu wykresowi, okazuje się, że są wśród naszych ofert auta o absurdalnym przebiegu ponad 15 milionów kilometrów. To ewidentny błąd.

Zanim zaczniemy modelować, musimy oczyścić dane. Usuńmy wszystkie wiersze, których przebieg przekracza czterysta tysięcy kilometrów.

```
mlodeVolkswageny <-      auta2012 %>%
  filter(Marka == "Volkswagen") %>%
  mutate(Wiek = 2012 - Rok.produkcji) %>%
```

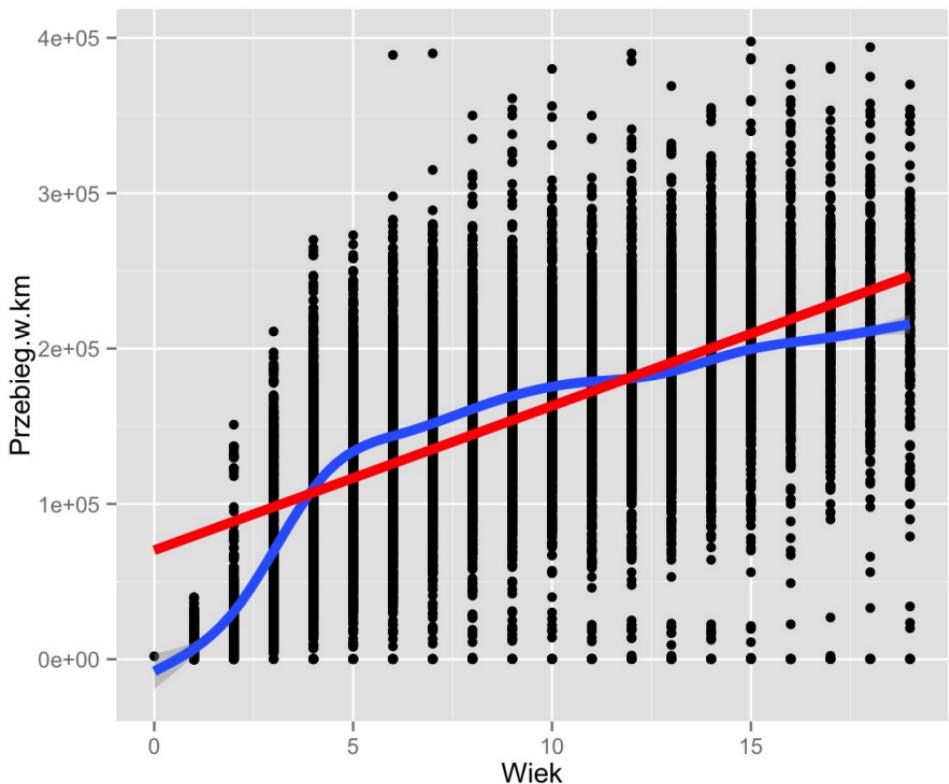
```
filter(Wiek < 20) %>%  
filter(Przebieg.w.km < 400000)
```

Zależność kawałkami liniowa

Po oczyszczeniu danych narysujmy zależność pomiędzy przebiegiem a wiekiem raz jeszcze.

Kolorem czerwonym zaznaczamy linię trendu regresji prostej, kolorem niebieskim zaznaczamy wygładzoną średnią w latach.

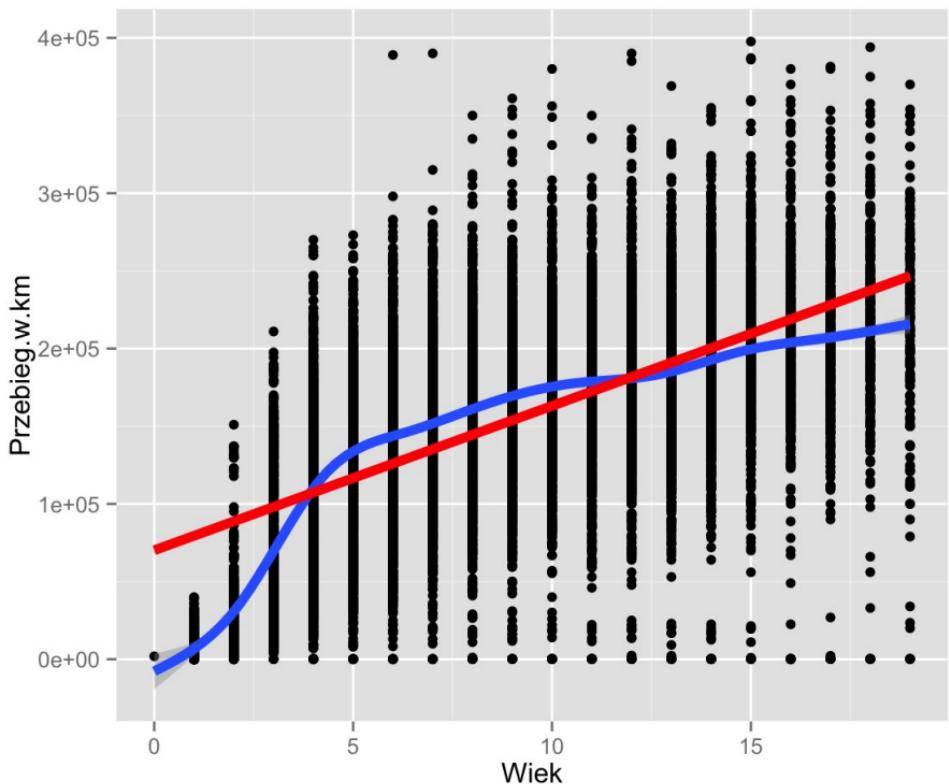
```
ggplot(mlodeVolkswageny, aes(y=Przebieg.w.km, :  
  geom_point() +  
  geom_smooth(size=2) + geom_smooth(method="lm")
```



Trend wygładzony i liniowy odstają od siebie znacząco. Co to oznacza? Być może zależność w danych wcale nie jest liniowa.

Niebieska linia, to linia wygładzonego trendu, lokalnie przybliżającego dane. Gdy się jej przyjrzeć, okazuje się, że ma ona bardzo ciekawe zachowanie.

Zależność kawałkami liniowa



Przez pierwsze pięć lat auta gwałtownie przybierają na liczniku, a po piątym roku tempo zwiększenia się licznika znaczco spada.

Czy volkswagenami starszymi niż 5 lat mniej się jeździ? Jest to jakieś wytlumaczenie, ale inne, bardziej prawdopodobne, jest takie że starsze auta są rzadziej serwisowane, a przez to ich przebiegi często są wartościami deklarowanymi.

Zależność kawałkami liniowa

W jaki sposób zamodelować taką łamana krzywą?

Potrzebujemy modelu, który będzie miał inną krzywą nachylenia w przedziale do 5 lat i ponad 5 lat.

Można taki model zbudować w następujący sposób

$$\text{Średni przebieg} = b_0 + b_1 * \text{wiek}[0+] + b_2 * \text{wiek}[5+]$$

gdzie $\text{wiek}[5+]$ to zmienna przyjmująca wartość 0, jeżeli wiek jest mniejszy niż 5 oraz wartość $\text{wiek}-5$ gdy wiek jest większy niż 5.

Stosując taką dodatkową zmienną, zależność pomiędzy wiekiem a przebiegiem będzie opisana przez współczynnik b_1 gdy wiek jest w przedziale 0-5 lat oraz przez współczynnik $b_1 + b_2$ gdy wiek jest wyższy niż 5 lat.

```
mlodeVolkswageny <- mlodeVolkswageny %>%
  mutate(Wiek0 = ifelse(Wiek >= 0, Wiek, 0),
         Wiek5 = ifelse(Wiek >= 5, Wiek - 5, 0))

## model z nowymi zmiennymi
M1 <- lm(Przebieg.w.km ~ Wiek0 + Wiek5, data=m1)
##
```

```
## Call:  
## lm(formula = Przebieg.w.km ~ Wiek0 + Wiek5,  
##  
## Coefficients:  
## (Intercept)          Wiek0          Wiek5  
##           -31518         34400        -28540
```

Zauważmy, że powyższy model szacuje przebieg dla nowych aut na b_0 . Bardziej naturalnym modelem jest taki, który dla nowych aut będzie prognozował przebieg zero. Można to osiągnąć usuwając wyraz wolny.

Zależność kawałkami liniowa bez wyrazu wolnego

Po usunięciu wyrazu wolnego mamy do czynienia z modelem.

$$\text{Średni przebieg} = b1 * wiek[0+] + b2 * wiek[5+]$$

Co w programie R można zapisać jako

```
M2 <- lm(Przebieg.w.km ~ Wiek0 + Wiek5 - 1, dat  
M2
```

```
##  
## Call:  
## lm(formula = Przebieg.w.km ~ Wiek0 + Wiek5  
##  
## Coefficients:
```

```
## Wiek0    Wiek5  
## 27481   -21230
```

Jak odczytać ten model?

Dla aut o wieku poniżej 5 lat obserwuje się średnie zwiększenie przebiegu w tempie 27481km na rok, ale już dla aut starszych niż 5 lat obserwuje się wydłużanie średniego przebiegu w tempie 6251km na rok.

Na obiekcie klasy `lm()` można wykonywać interesujące operacje. Przykładowo funkcją `summary()` można zbadać czy elementy tego modelu są istotnie różne od zera (ostatnia kolumna poniższego zestawienia to p-wartość).

```
summary(M2)
```

```
##  
## Call:  
## lm(formula = Przebieg.w.km ~ Wiek0 + Wiek5 +  
##  
## Residuals:  
##      Min       1Q   Median       3Q      Max  
## -224919 -31894     843    31412   245345  
##  
## Coefficients:  
##             Estimate Std. Error t value Pr(>|t|)  
## Wiek0    27480.8      131.2   209.38 <2e-16 :  
## Wiek5   -21229.7      224.0   -94.78 <2e-16 :  
## ---  
## Signif. codes:  0 '****' 0.001 '***' 0.01 '**'  
##
```

```
## Residual standard error: 51130 on 18164 deg
## Multiple R-squared:  0.9086, Adjusted R-squa
## F-statistic: 9.028e+04 on 2 and 18164 DF,  1
```

Zestawienie modeli

Rozważaliśmy jak dotąd kilka różnych modeli.

Przedstawmy te modele graficznie i zobaczymy, który wygląda na bliższy danych.

Aby pokazać jak wygląda model regresji łamanej wykorzystamy do funkcję `predict()`, która dla zadanego wieku wyznacza średni przebieg zgodnie z określonym modelem.

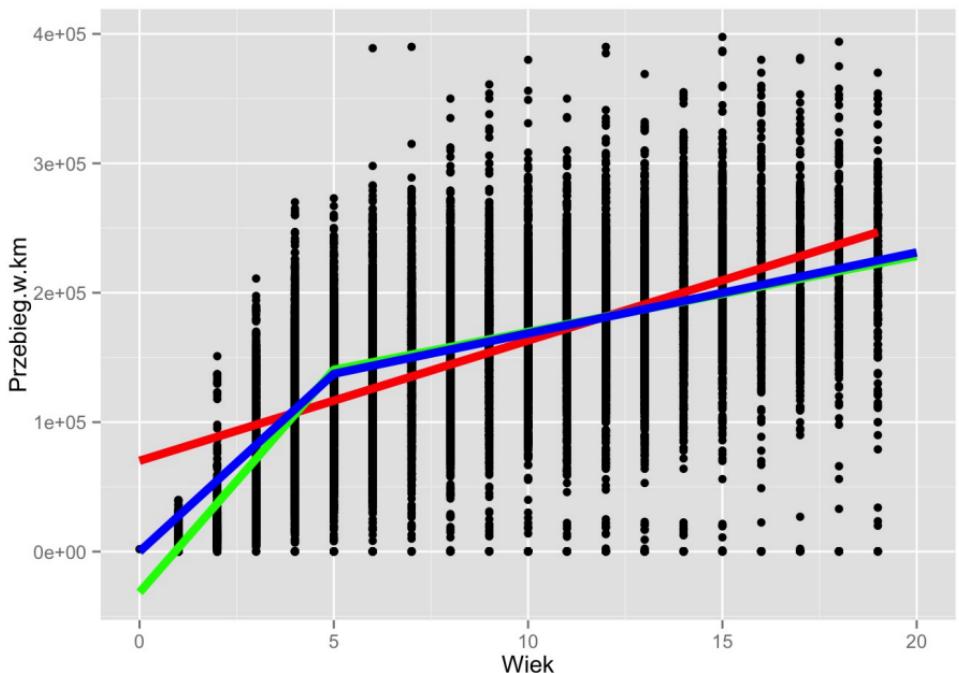
Funkcja `predict()` jako argument `newdata` przyjmuje obiekt `data.frame`, który powinien mieć takie same nazwy kolumn jak zmienen występujące w modelu.

Z przedstawianych modeli, najsensowniej wygląda niebieska krzywa, czyli model `M2`.

```
## sztuczny zbiór danych, na potrzeby pokazywania
topred <- data.frame(Wiek0 = c(0, 5, 20), Wiek1 =
topred$M1 <- predict(M1, newdata = topred)
topred$M2 <- predict(M2, newdata = topred)

## graficznie przedstawiamy dane i dorysowane krzywe
ggplot(mlodeVolkswageny, aes(y=Przebieg.w.km, x=Wiek)) +
  geom_point(data = mlodeVolkswageny, aes(x=Wiek0, y=Przebieg.w.km))
  geom_line(data = topred, aes(x=Wiek1, y=M1))
  geom_line(data = topred, aes(x=Wiek1, y=M2))
```

```
geom_point() +  
geom_smooth(method="lm", col="red", size=2)  
geom_line(data= topred, aes(y=M1, x = Wiek0),  
geom_line(data= topred, aes(y=M2, x = Wiek0))
```



Zadania

- Wykonaj modelowanie z użyciem modelu multiplikatywnego dla innej marki.
- Jeżeli model multiplikatywny wygląda na uzasadniony, policz ile procent corocznie traci na cenie auto danej marki.
- Zobacz jak wygląda zależność pomiędzy ceną a

wiekiem dla innych marek. Dziesięć najczęstszych marek w tym zbiorze danych to

Volkswagen, Opel, Ford, Renault, Audi, Merce

- Wykonaj modelowanie z użyciem łamanej regresji dla innych marek.

Zależność pomiędzy dwiema zmiennymi binarnymi

Przemysław Biecek @ Uniwersytet Warszawski

*sezon 2 / odcinek 13
pogRomcy danych*

- O czym jest ten odcinek
- Jak opisać zależność dwóch zmiennych binarnych
- Po co binaryzować
- Wczytanie danych
- Jak wygląda tabela kontyngencji
- Jak wygląda tabela z wartościami brzegowymi
- Jak wygląda tabela z procentami
- Jak wyglądają szanse
- Jak wyglądają ilorazy szans
- Wykres mozaikowy
- Wczytanie danych
- Czy przerzuty pogarszają rokowanie
- Czy przerzuty pogarszają rokowanie

- [Czy przerzuty pogarszają rokowanie](#)
- [Zadania](#)

O czym jest ten odcinek

Jak badać czy dwie zmienne są zależne?

Co oznacza zależność?

W tym odcinku rozważymy najprostszą zależność dla pary zmiennych - zależność pomiędzy parą zmiennych binarnych.

W tym odcinku nauczymy się:

- jak budować i jak czytać tablice kontyngencji dla pary zmiennych
- jak przedstawiać graficznie tabelę kontyngencji
- jak statystycznie testować zależność pomiędzy parą zmiennych.

Do ilustracji tych zagadnień wykorzystamy dwa zbiory danych. Pierwszy, to dane z Diagnozy Społecznej, dostępne w tabeli `diagnoza` a drugi o nowotworze piersi, dostępne w tabeli `TCGA_BRCA`. Oba dostępne w pakiecie `PogromcyDanych`.

Jak opisać zależność dwóch zmiennych binarnych

Zmienne binarne, to zmienne, które przyjmują po dwie wartości.

Zmienne takie można prosto scharakteryzować. Wystarczy napisać ile razy wystąpiła pierwsza wartość, ile razy druga i w ten sposób dwie liczby opisują obserwacje.

Typową zmienną binarną jest płeć ze swoimi dwoma poziomami/wartościami. Zmiennymi binarnymi są też zmienne logiczne przyjmujące wartości TRUE/FALSE.

W praktyce, dla zmiennych binarnych wprowadza się trzecią możliwą wartość, czyli *brak danych* (NA, ang.). Nawet takie zmienne z brakami danych, które de facto przyjmują trzy różne wartości, nazywa się binarnymi.

Zmienne jakościowe przyjmujące więcej poziomów lub zmienne ciągłe można binaryzować i często wykonuje się tę operację. Jeżeli mamy zmienną opisującą kolor włosów i w zbiorze danych jest dziesięć różnych kolorów, ale większość osób to brunaci/brunetki, wtedy na potrzeby analiz wygodnie może być zamienić zmienną z kilkoma poziomami na binarną. Np. odpowiadającą na pytanie: czy to brunet (a więc po połączeniu pozostałych poziomów w

jeden ‘nie brunet/ka’). Jeżeli mamy zmienną ciągłą - wzrost, możemy ją zbinaryzować dzieląc wzrost na dwa przedziały, np. do 170 cm i powyżej. Zamiast ciśnienia użyteczna może być informacja czy ciśnienie jest w normie czy jest podwyższone, zamiast wykształcenia informacja czy z maturą czy bez itp.

Po co binaryzować

W tym miejscu powinno pojawić się pytanie *po co?* Po co redukować więcej informacji do dwóch poziomów?

Główną zaletą jest łatwość opisu. Jedną zmienną binarną opisać można za pomocą dwóch liczb.

Relację pomiędzy dwoma zmiennymi binarnymi opisać można za pomocą czterech liczb. A więc za pomocą tak zwanej tablicy kontyngencji.

Zmienna 1: A Zmienna 1: B

Zmienna 2: X \(\{1,1\}\) \(\{n_{1,1}\}\)

Zmienna 2: Y \(\{1,2\}\) \(\{n_{1,2}\}\)

Zależność pomiędzy dwoma zmiennymi binarnymi opisują cztery liczby, \(\{1,1\}\) opisuje liczbę przypadków w których zmienna 1 przyjmuje wartość A a zmienna 2

wartość X, $\backslash(n_{\{1,2\}}\backslash)$ opisuje liczbę przypadków w których zmienna 1 przyjmuje wartość A a zmienna 2 wartość Y, i tak dalej.

Wczytanie danych

Zobaczmy jak ten opis wygląda dla danych diagnoza z pakietu PogromcyDanych. Aby te dane wczytać, wystarczy włączyć pakiet funkcją library() (wcześniej trzeba go zainstalować).

W zbiorze danych diagnoza dostępnych jest kilkudziesiąt kolumn, opisujących odpowiedzi na różne interesujące pytania. W tym odcinku przyjrzymy się tylko kilku wybranym kolumnom, opisującym imię, wiek, płeć, ukończoną edukację oraz odpowiedź na pytanie „Co jest według Pana ważniejsze w życiu?” (zmienna w kolumnie gp29) z możliwymi odpowiedziami *Poczucie sensu* i *Przyjemności*.

```
library(PogromcyDanych)
diagnoza[1:6, c("imie_2011", "wiek2013", "plec"

##    imie_2011     wiek2013      plec
## 1    WERONIKA          4   kobieta
## 2    ERNEST         13  mężczyzna  zasadnicze za...
## 3    SYLWIA         23   kobieta      wy...
## 4    MARIOLA        27   kobieta      wy...
## 5    WACŁAW         78  mężczyzna  poc...
```

```
#> 6 PIOTR gp29
## 1 <NA>
## 2 POCZUCIE SENSU
## 3 PRZYJEMNOSCI
## 4 POCZUCIE SENSU
## 5 POCZUCIE SENSU
## 6 POCZUCIE SENSU
```

48 mężczyzna zasadnicze za

Jak wygląda tabela kontyngencji

W zbiorze danych `diagnoza` mamy dwie binarne zmienne, `plec`, czyli płeć respondenta i `gp29`, czyli odpowiedź na pytanie co jest ważniejsze w życiu.

Zobaczmy jak wygląda tablica kontyngencji, zestawiająca te dwie zmienne.

```
table(diagnoza$plec, diagnoza$gp29)
```

	PRZYJEMNOSCI	POCZUCIE SENSU
mężczyzna	5114	6544
kobieta	5144	9421

Przyjrzyjmy się tym wynikom

- 5114 mężczyzn odpowiedziało, że ważniejsze są przyjemności,
- 6544 mężczyzn odpowiedziało, że ważniejsze jest poczucie sensu,

- 5144 kobiet odpowiedziało, że ważniejsze są przyjemności,
- 9421 kobiet odpowiedziało, że ważniejsze jest poczucie sensu.

Domyślne zachowanie funkcji `table()` jest taki, że pomija wartości brakujące. Aby je też wyświetlić, należy dodać argument `useNA = "always"`.

```
table(diagnoza$plec, diagnoza$gp29, useNA = "all")
```

	PRZYJEMNOSCI	POCZUCIE	SENSU	<NA>
##				
##	mężczyzna	5114	6544	669
##	kobieta	5144	9421	550
##	<NA>	0	0	40

Jak wygląda tabela z wartościami brzegowymi

Zarówno kobiety jak i mężczyźni częściej deklarują, że ważniejsze jest poczucie sensu. Ale czy są różnice pomiędzy tymi płciami i jak duże?

Funkcją `addmargins()` można dodać sumę wartości w wierszach i kolumnach.

```
plecSens <- table(diagnoza$plec, diagnoza$gp29)  
addmargins(plecSens)
```

		PRZYJEMNOSCI	POCZUCIE	SENSU	S
##	mężczyzna	5114		6544	116!
##	kobieta	5144		9421	1450
##	Sum	10258		15965	2622

Jak wygląda tabela z procentami

Przeliczmy liczebności na frakcje. Jaki procent mężczyzn odpowiedziało przyjemności a jaka poczucie sensu?

Funkcją `prop.table()` można policzyć frakcje.

Dodatkowy argument określa, czy mają być to proporcje w wierszach (1), kolumnach (2) czy całej macierzy (pusty argument).

```
plecSens <- table(diagnoza$plec, diagnoza$gp29)
## frakcje w wierszach
prop.table(plecSens, 1)
```

```
##
```

	PRZYJEMNOSCI	POCZUCIE	SENSU
##	0.4386687		0.5613313
##	0.3531754		0.6468246

```
## frakcje w całej tabeli
prop.table(plecSens)
```

```
##
```

	PRZYJEMNOSCI	POCZUCIE	SENSU
##	0.1950196		0.2495519
##	0.1961637		0.3592648

Jak wyglądają szanse

Pracując ze zmiennymi binarnymi, często zamiast prawdopodobieństw korzysta się z tak zwanych szans (ang. odds).

Co to jest szansa?

Szansa wystąpienia zjawiska X to prawdopodobieństwo wystąpienia X podzielone przez prawdopodobieństwo nie wystąpienia X. Czyli jeżeli prawdopodobieństwo oznaczymy symbolem $\backslash(p\backslash)$ to otrzymamy

$$\backslash[\text{odds} = \backslash\frac{\{p\}}{\{1-p\}} \backslash]$$

W przeciwnieństwie do prawdopodobieństwa, szansa może przyjmować wartości większe od 1.

Policzmy szanse, że kobieta lub mężczyzna odpowie *Przyjemności* na pytanie o to co ważniejsze.

```
frakcje <- prop.table(plecSens, 1)  
frakcje[,1]/frakcje[,2]
```

```
## mężczyzna    kobieta  
## 0.7814792  0.5460142
```

Jak wyglądają ilorazy szans

Jaka jest zaleta w używaniu szans nad prawdopodobieństwem?

Główną zaletą jest możliwość operowania na ilorazach szans. Iloraz szans przedstawia ile razy szansa w jednej grupie jest większa od szansy w dużej grupie.

$$\text{oddsRatio}_{A/B} = \text{odds}_A / \text{odds}_B = \frac{p_A}{(1-p_A)} \cdot \frac{(1-p_B)}{p_B}$$

Przedstawmy iloraz szans na przykładzie płci i odpowiedzi na pytanie o wartości ważne w życiu.

Policzymy ilu krotnie szansa na odpowiedź *Przyjemności* jest większa u mężczyzn, niż u kobiet.

Jeżeli iloraz szans wynosi 1 to znaczy, że szanse w obu grupach są równe. Gdyby szansa na odpowiedź *Przyjemności* była taka sama w obu grupach, to uznalibyśmy, że te dwie zmienne nie są zależne (wyniki jednej nie zależą od drugiej).

Jeżeli szansa jest mniejsza lub większa niż jeden, wtedy pojawia się zależność.

W przykładzie dla Diagnozy Społecznej, iloraz szans wynosi 1.4312, co oznacza, że szansa, że dla losowo wylosowanego mężczyzny, szansa że odpowie

Przyjemności jest o 43% większa niż u kobiety.

```
frakcje <- prop.table(plecSens, 1)
szanse <- frakcje[,1]/frakcje[,2]
szanse[1] / szanse[2]

## mężczyzna
## 1.431243
```

Wykres mozaikowy

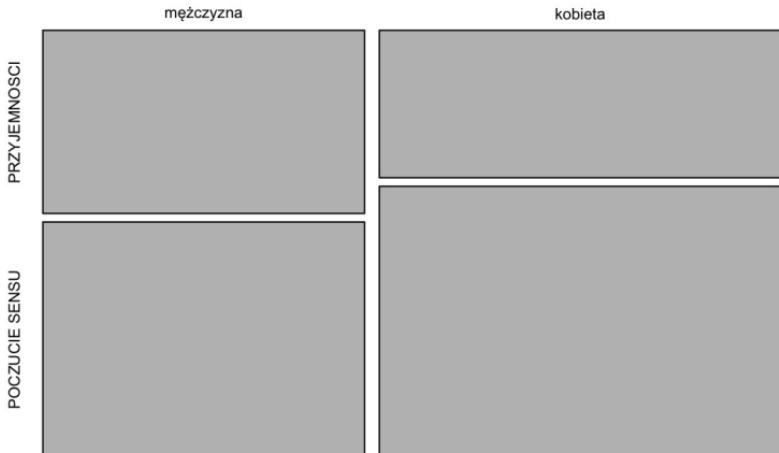
Tabele kontyngencji zazwyczaj przedstawia się za pomocą wykresu mozaikowego.

W programie R wystarczy w tym celu wykorzystać funkcję `mosaicplot()`.

Szerokość słupków odzwierciedla względne proporcje kobiet i mężczyzn w zbiorze danych. Każda z kolumn jest podzielona na długości słupków proporcjonalne do odpowiedzi na pytanie o to co ważne osobno dla każdej płci.

```
mosaicplot(plecSens, main="Odpowiedzi na pytanie")
```

Odpowiedzi na pytanie o to co ważne w życiu



Wczytanie danych

W zbiorze danych `diagnoza` jest ponad 38 tysięcy obserwacji. Dla tak dużej populacji, nawet bardzo małe różnice okazują się istotnie różne.

Dlatego przykład dotyczący testowania przedstawimy na innym zbiorze danych, zbiorze `TCGA_BRCA` z wybranymi zmiennymi dla raka piersi. W tym zbiorze danych mamy wyniki dla 999 pacjentów.

Kolejne kolumny tego zbioru danych opisują takie cechy

jak: mutacja genu P53, płeć pacjenta, informację czy pacjent wciąż żyje, informacja ile dni po operacji doszło do zgonu oraz informacja czy w okresie jednego roku pojawił się guz / przerzuty.

```
head(TCGA_BRCA)
```

```
##                      TP53     plec czy.zyje dni.do..  
## 1          Wild type female    live  
## 2          Other   female    live  
## 3          Wild type female    live  
## 4          Wild type female    live  
## 5 Missense_Mutation female  dead  
## 6          Wild type female    live
```

Czy przerzuty pogarszają rokowanie

Zobaczmy czy jest prawdopodobieństwo zgonu (zmienna `czy.zyje`) zależy od występowania przerzutów (zmienna `czy.nowy.guz`).

Zacznijmy od wyznaczenia tablicy kontyngencji.

```
(przerzuty_przezycia <- table(TCGA_BRCA$czy.nowy.guz,  
                                TCGA_BRCA$czy.zyje))
```

```
##  
##          dead  live  
##  no      17   342  
##  yes     23    20
```

Przedstawmy tę tabelę graficznie

```
mosaicplot(przerzuty_przezycia)
```



Czy przerzuty pogarszają rokowanie

Policzmy teraz frakcje osób, które przeżyły i które zmarły osobno w grupie osób bez i z przerzutami.

```
(frakcje <- prop.table(przerzuty_przezycia, 1))  
##                                     dead      live  
##   0.8500000000000001  0.14999999999999998
```

```
##    no  0.04735376  0.95264624  
##    yes 0.53488372  0.46511628
```

Policzmy szanse zgonu w grupie osób bez przerzutów i z przerzutami.

```
(szanse <- frakcje[,1]/frakcje[,2])
```

```
##          no        yes  
## 0.0497076 1.1500000
```

Iloraz szans wynosi 0.049, co znaczy, że w grupie badanych szansa na zgon osób bez przerzutów to mniej niż 5% szansy na zgon osób z przerzutami. Równoważnie można powiedzieć, że osoby z przerzutami mają szansę na zgon 24 razy większe.

```
## bez przerzutów do przerzutów  
szanse[1] / szanse[2]
```

```
##          no  
## 0.043224
```

```
## z przerzutami do bez przerzutów  
szanse[2] / szanse[1]
```

```
##          yes  
## 23.13529
```

Czy przerzuty pogarszają rokowanie

Iloraz szans równy 23 wydaje się być istotnie większy niż 1 (co jak wiemy, oznacza brak zależności).

Aby przetestować czy ta zależność pomiędzy przerzutami a przeżyciem jest istotna statystyczna można przeprowadzić test.

Dla tabel 2 na 2 najbardziej popularnym rozwiązaniem jest użycie testu Fishera. Podobnie jak dla testu dla regresji i testu dla średnich (które omawialiśmy cztery odcinki wcześniej), też test Fishera opiera się na badaniu, jak często przypadkowo iloraz byłby większy niż obserwowany (przy założeniu że tyle samo osób ma przerzuty i tyle samo osób umiera).

Czyli p-wartość wyznaczona w tym teście odpowiada prawdopodobieństwu, że obserwowany iloraz szans jest większy niż obserwowany w sytuacji gdyby pomiędzy zmiennymi nie było żadnych zależności.

```
fisher.test(przerzuty_przezycia)
```

```
## 
## Fisher's Exact Test for Count Data
## 
## data: przerzuty_przezycia
## p-value = 2.155e-15
## alternative hypothesis: true odds ratio is 1
## 95 percent confidence interval:
## 0.0186201 0.1006088
```

```
## sample estimates:  
## odds ratio  
## 0.0439938
```

W tym przykładzie p-wartość jest rzędu $(2 * 10^{-15})$, tak więc gdyby zmienne nie były zależne, byłoby nieprawdopodobne by zaobserwować tak duży iloraz szans.

Zadania

1. W zbiorze danych `diagnoza` wiek respondenta jest w zmiennej o nazwie `wiek2013`. Wyznacz zależność pomiędzy wiekiem podzielonym na dwie grupy, poniżej i powyżej 30 roku życia, a odpowiedziami na pytanie o to co w życiu ważne.
2. W zbiorze danych `TCGA_BRCA` w drugiej kolumnie jest płeć pacjenta. Wyznacz zależność pomiędzy zmiennymi `plec` a `czy.zyje`. Przetestuj tą zależność testem Fishera.

Zależność pomiędzy dwiema zmiennymi jakościowymi

Przemysław Biecek @ Uniwersytet Warszawski

*sezon 2 / odcinek 14
pogRomcy danych*

- [O czym jest ten odcinek](#)
- [Wczytanie danych](#)
- [Jak wygląda tabela kontyngencji dla wykształcenia](#)
- [Jak wygląda tabela kontyngencji dla wieku](#)
- [Proporcje](#)
- [Wykresy mozaikowe](#)
- [Więcej kolumn / wierszy](#)
- [Czy jest zależność?](#)
- [Czy jest zależność?](#)
- [Zadanie](#)

O czym jest ten odcinek

W poprzednim odcinku pokazaliśmy jak badać zależność pomiędzy dwoma zmiennymi binarnymi.

A co w sytuacji gdy zmienne nie są binarne? Temu tematowi poświęcony jest ten odcinek.

W tym odcinku nauczmy się:

- jak budować i jak czytać tablice kontyngencji dla pary zmiennych jakościowych
- jak przedstawiać graficznie tabelę kontyngencji
- jak statystycznie testować zależność pomiędzy parą zmiennych.

Do ilustracji tych zagadnień wykorzystamy dwa zbiory danych. Pierwszy, to dane z Diagnozy Społecznej, dostępne w tabeli `diagnoza` a drugi to dane o nowotworze piersi, dostępne w tabeli `TCGA_BRCA`. Oba dostępne w pakiecie `PogromcyDanych`.

Wczytanie danych

W zbiorze danych `diagnoza` dostępnych jest kilkadziesiąt kolumn, opisujących odpowiedzi na różne interesujące pytania. W tym odcinku przyjrzymy się tylko kilku wybranym kolumnom, opisującym imię (kolumna `imie_2011`), wiek (`wiek2013`), płeć (`plec`), ukończoną

edukację (eduk4_2013) oraz odpowiedź na pytanie „Co jest według Pana ważniejsze w życiu?” (zmienna w kolumnie gp29) z możliwymi odpowiedziami *Poczucie sensu* i *Przyjemności*.

```
library(PogromcyDanych)
diagnoza[1:6, c("imie_2011", "wiek2013", "plec"

##      imie_2011 wiek2013      plec
## 1    WERONIKA        4  kobietka
## 2    ERNEST         13  m'żczyzna zasadnicze za...
## 3    SYLWIA         23  kobietka          wy...
## 4    MARIOLA         27  kobietka          wy...
## 5    WACŁAW          78  m'żczyzna          poc...
## 6    PIOTR           48  m'żczyzna zasadnicze za...

##                      gp29
## 1                  <NA>
## 2    POCZUCIE SENSU
## 3  PRZYJEMNOSCI
## 4    POCZUCIE SENSU
## 5    POCZUCIE SENSU
## 6    POCZUCIE SENSU
```

Jak wygląda tabela kontyngencji dla wykształcenia

W poprzednim odcinku przyglądalismy się parze zmiennych plec, czyli płeć respondenta i zmiennej gp29, czyli odpowiedź na pytanie co jest ważniejsze w życiu.

W tym odcinku przyjrzymy się zależności pomiędzy

poziomem edukacji (zmienna `eduk4_2013`) a odpowiedziami na pytanie o to co jest ważniejsze w życiu.

Zobaczmy jak wygląda tablica kontyngencji, zestawiająca te dwie zmienne.

```
table(diagnoza$eduk4_2013, diagnoza$gp29)
```

	PRZYJEMNOSC:
## BD/ND/FALA	(
## podstawowe i niższe	1872
## zasadnicze zawodowe/gimanzjum	3662
## średnie	2981
## wyższe i policealne	1707

Zmienna `eduk4_2013` poza czterema poziomami edukacji ma też piąty poziom `BD/ND/FALA`, który w tych danych nie występuje. Najłatwiej go usunąć funkcją `droplevels()`.

```
(edukacjaTab <- table(droplevels(diagnoza$eduk4_2013)))
```

	PRZYJEMNOSC:
## podstawowe i niższe	1872
## zasadnicze zawodowe/gimanzjum	3662
## średnie	2981
## wyższe i policealne	1707

W każdej grupie wykształcenia respondenci częściej deklarują, że ważniejsze jest poczucie sensu.

Ale czy są różnice pomiędzy grupami wykształcenia i czy są one duże?

Jak wygląda tabela kontyngencji dla wieku

Zobaczmy jak wygląda zależność pomiędzy wiekiem a odpowiedziami na pytanie o sens w życiu.

Wiek jest zmienną ciągłą, zamienimy więc ją na zmienną jakościową o czterech poziomach. Aby zdyskretyzować zmienną ciągłą na dyskretną wykorzystamy funkcję `cut()`. Za granice przedziałów wiekowych wybierzemy 0, 25, 40, 60 i 110 lat.

```
diagnoza$wiek2013_4g <- cut(diagnoza$wiek2013,  
table(diagnoza$wiek2013_4g)
```

```
##  
##   (0,25]   (25,40]   (40,60]   (60,110]  
##   11049     7770     10414     9051
```

Przedstawmy teraz jak wygląda zależność pomiędzy wiekiem a odpowiedziami na pytanie o to co jest ważniejsze w życiu?

```
wiekTab <- table(diagnoza$wiek2013_4g, diagnoza  
wiekTab
```

	PRZYJEMNOSCI	POCZUCIE	SENSU
## (0, 25]	1816		1887
## (25, 40]	2378		3296
## (40, 60]	3429		5673
## (60, 110]	2629		5098

Proporcje

Używając funkcji `prop.table()` można wyznaczyć frakcje w wierszach lub kolumnach.

Zobaczmy jak wyglądają proporcje odpowiedzi na pytanie o poczucie sensu w różnych grupach edukacji i wieku.

```
prop.table(edykacjaTab, 1)
```

	PRZYJEMNOSC:
## podstawowe i niższe	0.3924528
## zasadnicze zawodowe/gimanzjum	0.4284544
## średnie	0.3847541
## wyższe i policealne	0.3362222

```
prop.table(wiekTab, 1)
```

	PRZYJEMNOSCI	POCZUCIE	SENSU
## (0, 25]	0.4904132		0.5095868
## (25, 40]	0.4191047		0.5808953
## (40, 60]	0.3767304		0.6232696
## (60, 110]	0.3402355		0.6597645

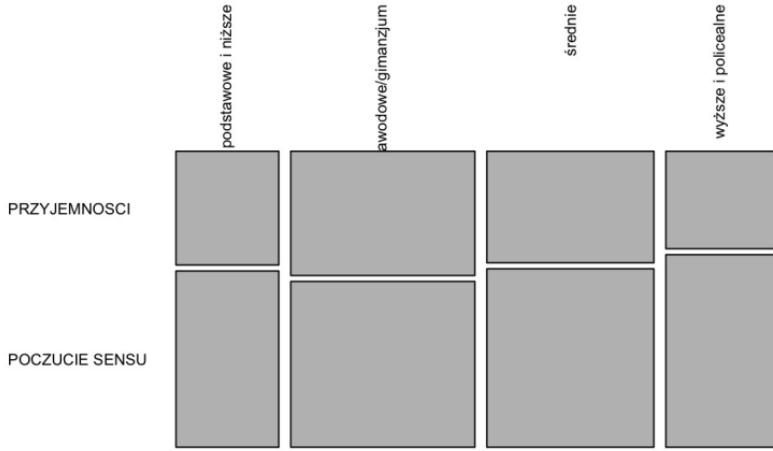
Wykresy mozaikowe

Tabele częstości można przedstawić graficznie.

Często wykorzystuje się do tego wykres mozaikowy. Pola kwadratów oznaczają liczbę odpowiedzi w tej kombinacji cech. Względne wysokości słupków odpowiadają względnym proporcjom w grupach określonych przez kolumny.

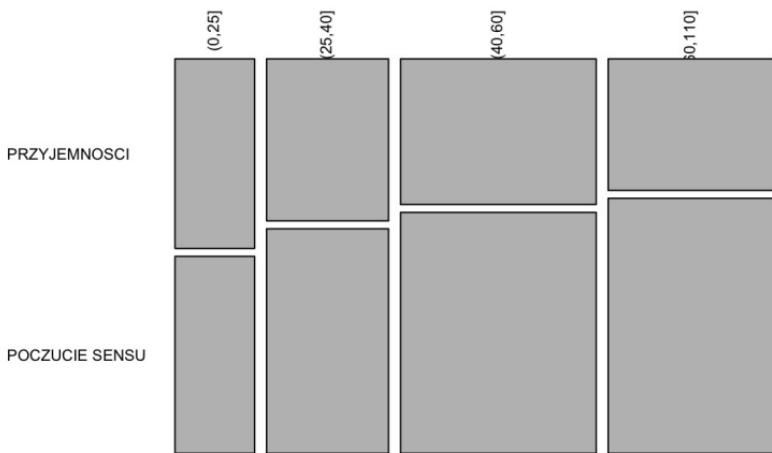
```
mosaicplot(edykacjaTab, las=2, main="Edukacja vs. co jest ważne w życiu")
```

Edukacja vs. co jest ważne w życiu



```
mosaicplot(wiekTab, las=2, main="Wiek vs. co jest ważne w życiu")
```

Wiek vs. co jest ważne w życiu



Więcej kolumn / wierszy

A teraz wykonajmy modelowanie dla innej zmiennej. Np. zobaczymy na ile identyfikacja z pytanie, *Podziwiam ludzi, którzy mają drogie domy, samochody i ubrania* koreluje z wiekiem (więcej informacji o pytaniach znaleźć można w kwestionariuszu kwestionariusza

http://diagnoza.com/pliki/kwestionariusze_instrukcje/kwes

Pytanie dotyczy odpowiedzi na pytanie *Niektórzy są więcej warci od innych*

```
wiekPodziwiam <- table(diagnoza$gp54_13, diagnoza$gp54_13)
```

```
##
```

		(0, 25]	(25, 40]	(40, 60]	(60, 80]
##	ZDECYDOWANIE TAK	333	398	612	661
##	TAK	665	1147	2156	2260
##	RACZEJ TAK	583	872	1498	1560
##	ANI TAK, ANI NIE	718	1093	1729	1780
##	RACZEJ NIE	364	588	951	980
##	NIE	651	1124	1688	1780
##	ZDECYDOWANIE NIE	406	464	477	480

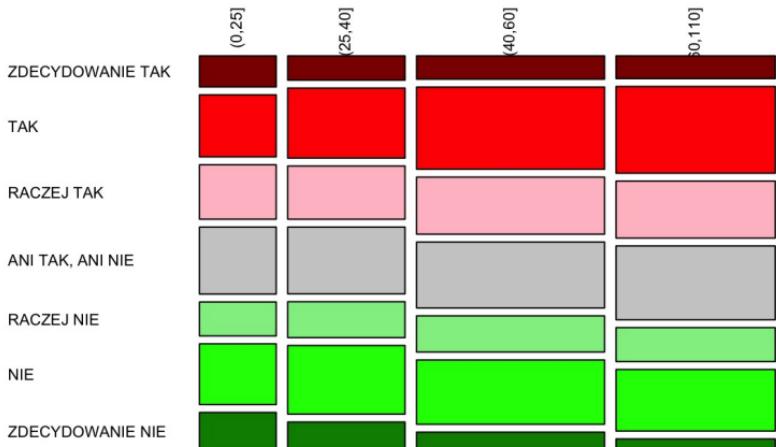
```
## zależność przedstawiona graficznie
```

```
mosaicplot(t(wiekPodziwiam), las=2,
```

```
    col=c("red4","red1","pink","grey80"),
```

```
    main="Podziwiam ludzi, którzy mają co najmniej 25 lat")
```

Podziwiam ludzi, którzy mają drogie domy, samochody i ubrania



Czy jest zależność?

Aby sprawdzić, czy pomiędzy parą zmiennych jakościowych jest zależność, można wykonać test χ^2 .

Sprawdza on na ile obserwowana tabela częstości jest daleka od takiej, którą byśmy obserwowali, gdyby zmienne nie były zależne.

Test χ^2 wykonuje się funkcją `chisq.test()`. Wynikiem jest p-wartość dla hipotezy o niezależności.

Małe p-wartości (zwyczajowo poniżej 0.05) interpretuje się jako przesłanki o zależności pomiędzy zmiennymi.

```
wiekPodziwiam <- table(diagnoza$gp54_13, diagnoza$zdecydowanie)
chisq.test(wiekPodziwiam)

## 
## Pearson's Chi-squared test
## 
## data: wiekPodziwiam
## X-squared = 418.81, df = 18, p-value < 2.2e-16
```

Czy jest zależność?

Z wyniku funkcji `chisq.test()` można też odczytać, jak wyglądają obie porównywane tabele.

W polu `observed` znajduje się tabela częstości obserwowana w danych.

```
chisq.test(wiekPodziwiam)$observed
```

		(0, 25]	(25, 40]	(40, 60]	(60, 80]
##	ZDECYDOWANIE TAK	333	398	612	600
##	TAK	665	1147	2156	2000
##	RACZEJ TAK	583	872	1498	1500
##	ANI TAK, ANI NIE	718	1093	1729	1700
##	RACZEJ NIE	364	588	951	900
##	NIE	651	1124	1688	1700
##	ZDECYDOWANIE NIE	406	464	477	400

A w polu `$expected` znajduje się tabela częstości oczekiwanych, gdyby nie było żadnej zależności.

```
chisq.test(wiekPodziwiam)$expected
```

		(0, 25]	(25, 40]	(40,
##	ZDECYDOWANIE TAK	262.0677	400.5691	641..
##	TAK	835.9790	1277.7893	2047..
##	RACZEJ TAK	598.3028	914.5027	1465..
##	ANI TAK, ANI NIE	734.5837	1122.8073	1799..
##	RACZEJ NIE	376.9350	576.1432	923..
##	NIE	685.0915	1047.1587	1677..
##	ZDECYDOWANIE NIE	227.0403	347.0298	556..

Jeżeli te tabele są daleko od siebie, to jest to przesłanka, że zmienne są zależne (=nie są niezależne).

Zadanie

W zbiorze danych `TCGA_BRCA` zbadaj czy jest i jaka jest zależność pomiędzy

- przeżyciami (kolumna `czy.zyje` a mutacjami `TP53`),
- płcią a mutacjami `TP53`,
- płcią a zmienną `czy.zyje`.

Zależność przedstaw za pomocą tabeli z liczebnosciami oraz częstosciami.

Sondaże bez tajemnic

Przemysław Biecek @ Uniwersytet Warszawski

sezon 2 / odcinek 15

pogRomcy danych

- [Film](#)
- [Wprowadzenie](#)
- [1. Wstęp](#)
- [Zadania 1:](#)
- [2. Metoda reprezentacyjna](#)
- [Zadania 2:](#)
- [3. Przykłady prób](#)
- [Zadania 3:](#)
- [4. Sondaż](#)
 - [Tabela 4.1. Rozkład preferencji wyborczych w \(fikcyjnej\) populacji dorosłych obywateli Polski](#)
- [Zadania 4:](#)
- [5. Problem badawczy -> populacja -> technika realizacji badania](#)
 - [Tabela 5.1. Techniki realizacji badań sondażowych](#)
- [Zadania 5:](#)

- 6. Kwestionariusz
- Zadania 6:
- 7. Dobór próby
 - Tabela 7.1. Absencja wyborcza w (fikcyjnej) populacji mieszkańców Polski
 - Tabela 7.2. Operat losowania
 - Tabela 7.3. Liczba wystąpień obywateli w próbach dwuosobowych
 - Wykres 7.1. Poziom absencji wyborczej w próbach dwuosobowych
 - Wykres 7.2. Symulacja - poziom absencji w 10 000 prób dwuosobowych
 - Tabela 7.4. Przeciętny poziom absencji w próbie - podsumowanie obliczeń
- Zadania 7:
- 8. Błąd oszacowania
 - Tabela 8.1. Rozkład poparcia dla partii X w (fikcyjnej) populacji dorosłych obywateli Polski
 - Tabela 8.2. Liczba wystąpień obywateli w próbach sześciuoosobowych
 - Tabela 8.3. Poparcie dla partii X w próbach sześciuoosobowych - obliczenia
 - Wykres 8.1. Poparcie dla partii X w próbach sześciuoosobowych
 - Tabela 8.4. Odchylenie standardowe - obliczenia

- Wykres 8.2. Poparcie dla partii X w próbach 6, 9, 12 oraz 15-osobowych
- Wykres 8.3. Odchylenie standardowe poparcia dla partii X dla schematów losowania od 6 do 20 respondentów
- Tabela 8.5. Skumulowany odsetek prób sześciuoosobowych
- Wykres 8.4. Zmiana szerokości przedziału ufności w zależności od liczby respondentów w próbie (od 6 do 20)

- Zadania 8:

- 9. Błędy systematyczne

- Tabela 9.1. Rozkład preferencji wyborczych w (fikcyjnej) populacji dorosłych obywateli Polski
- Tabela 9.2. Wyniki z prób sześciuoosobowych przy niepełnej realizacji (mieszkańcy miast są niedostępni)
- Tabela 9.3. Liczba dostępnych respondentów w próbach sześciuoosobowych
- Tabela 9.4. Poparcie dla partii X w próbach sześciuoosobowych przy niepełnej realizacji (niedostępni mieszkańcy miast)
- Wykres 9.1. Rozkład poparcia dla partii X w próbach 6 osobowych przy niepełnej realizacji (niedostępni mieszkańcy miast)

- Zadania 9:

Film

Odcinek o sondażach jest też dostępny w formacie video na kanale youtube <https://youtu.be/HeuEStD4noU>

W serii odcinków „Dane wokół nas” jest również odcinek „Robotyka i analiza danych” dostępny pod adresem <https://youtu.be/I4163EGXHCE> oraz odcinek „Inżynieria lingwistyczna” dostępny pod adresem <https://youtu.be/Xmoqiw1WDaI>

Wprowadzenie

Zespół Na Straży Sondaży przygotował dla PogRomców Danych materiały pomocnicze do odcinka dotyczącego badań sondażowych.

Dowiecie się z nich, jak przygotowuje się kwestionariusze, czym różni się sondaż telefoniczny od bezpośredniego, na czym polega dobór losowy próby, jak za jego pomocą przeprowadzić prawidłowe wnioskowanie o populacji, a także jakimi błędami mogą być obarczone badania sondażowe.

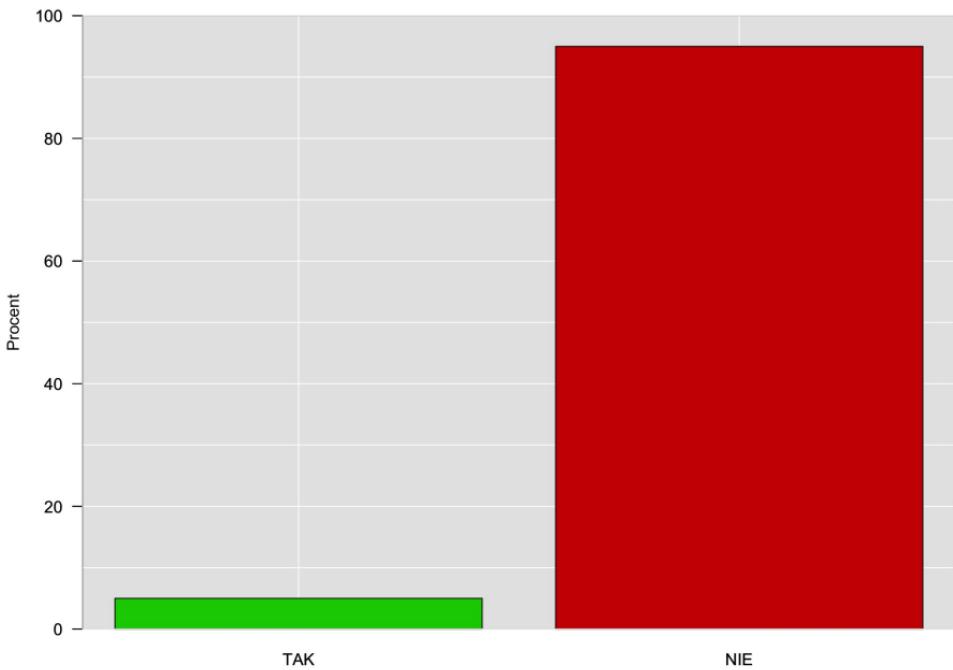
Mamy nadzieję, że dzięki naszym materiałom przekonacie się, że prawdziwe sondaże to nie wróżenie z fusów, lecz

rzetelna analiza danych uzyskanych przy pomocy metody reprezentacyjnej.

1. Wstęp

Z badań przeprowadzonych przez zespół Na Straży Sondaży w 2014 r. wynika, że 95% Polaków nie wie jak powstają sondaże!

Czy wiesz jak powstają sondaże?



Jak interpretować ten wynik? Co on oznacza? Odpowiedź

jest prosta. Praktycznie nic, bo ta informacja nie ma żadnej wartości. Dlaczego? Nie tylko dlatego, że jest zmyślona. Podstawowy problem stanowi brak jakiejkolwiek noty metodologicznej: nie powiedziałem dokładnie kiedy zorganizowano badanie, jaką techniką je przeprowadzono (telefonicznie czy bezpośrednio), jak zadano pytanie - co to znaczy, że ktoś wie jak powstają sondaże, a także nie zdefiniowałem kim są Polacy - czy są to obywatele, czy mieszkańcy Polski, w jakim byli wieku. Nie podałem również jakim błędem mogą być obarczone wyniki (tzw. błąd statystyczny), ani ile osób wzięło udział w badaniu. Codziennie w prasie i innych mediach pojawiają się podobne "dane sondażowe". Czy mają one jakąkolwiek wartość? Czy można im zaufać? Jak odróżnić "dobry" sondaż od "złego". Na te oraz inne pytania postaramy się udzielić odpowiedzi w trakcie naszego kursu. Pokażemy na czym polega sondaż, z jakich elementów się składa, jakie są jego ograniczenia - czego nie powie nam nawet najlepsze badanie. Zaczniemy od przykładów łatwych, a skończymy na bardziej zaawansowanych próbując wcielić się na chwilę w rolę "sondażysty". W imieniu zespołu Na Straży Sondaży zapraszamy do odkrywania niezwykłych możliwości, ale także pewnych ograniczeń metody zwanej reprezentacyjną, metody która stoi za wszystkimi sondażami.

Zadania 1:

1. Wymień trzy artykuły prasowe z 2014 r., w których powołano się na wyniki badań społecznych (sondaże). Napisz ile osób wzięło udział w badaniu oraz kiedy je zrealizowano, o ile w artykule podano tego rodzaju informacje.

L.p.	Tytuł artykułu	Link do strony z artykułem ją jeżeli została podana	Czy podano wielkość próby? Zapisz	Czy podano dokładną datę realizacji badania? Jaką?
1.				
2.				
3.				

2. Metoda reprezentacyjna

Na pierwszy rzut oka badanie sondażowe, badanie na próbie, może się wydawać zadaniem karkołomnym. Oto na podstawie niewielkiej liczby obserwacji np. 1000 respondentów, staramy się opisać dużo większą, czasami nawet o kilka rzędów, populację. Czy ma to jakikolwiek sens? Czy da się wyznaczyć na tej podstawie przeciętną wagę, wzrost, miesięczne wydatki, liczbę przeczytanych książek albo czas spędzany dziennie na Facebooku przez dorosłych mieszkańców Polski. Trzeba wiedzieć, że

oficjalnie mieszka około 31 mln ludzi w wieku 18 i więcej lat. W tej sytuacji 1000 osób stanowi w zaokrągleniu trzy dziesięciotysięczne PROCENTA populacji ($1\ 000 / 31\ 000\ 000 = 0,000032$)!!! To bardzo mało. Trudno uwierzyć, że taka garstka obserwacji może dostarczyć nam wiarygodnych informacji o całej badanej zbiorowości. Dla porównania zastanówmy się, czy na podstawie jednego kilometra drogi da się powiedzieć, jak będzie wyglądała cała podróż mierząca 300 000 km (Ziemia w obwodzie liczy tylko 40 000 km). Na pierwszy rzut oka nie. Okazuje się jednak, że nauka, pod postacią statystyki, daje nam pewne narzędzia, które pozwalają trafnie wnioskować o dużych “obiektach” nawet na podstawie ich niewielkiego wycinka. W przypadku podróży kluczem do sukcesu byłoby umiejętne wybranie takich małych odcinków z całej drogi, które ułożą się w próbny kilometr. Statystycy i badacze społeczni wiedzą doskonale, że dobrze dobrana próba stanowi świetny opis całej populacji. Jak to możliwe? Co trzeba zrobić, żeby przy użyciu małego kamyka dowiedzieć się czegoś o wielkiej “górze”?

Zadania 2:

1. Wymień trzy badania na próbach przeprowadzone przez instytucje państwowie i podaj link do ich wyników lub raportu. Mogą to być badania z roku

2014, ale także wcześniejszych lat.

L.p.	Nazwa instytucji publikującej wyniki	Tytuł badania	Link do strony z raportem
1.			
2.			
3.			

3. Przykłady prób

Zacznijmy od naszych codziennych doświadczeń. Wbrew pozorom większość naszej wiedzy o świecie czerpiemy z prób. I nie chodzi tu o metodę prób i błędów. Przyjrzyjmy się naszemu zdrowiu. Czasami zdarza się, że lekarz każe nam zrobić badanie krwi. Ale czy to oznacza, że trzeba zbadać całą krew w organizmie, wszystkie komórki? Na szczęście nie. Wystarczy mała próbka. Lekarz pobiera od nas zaledwie 10 ml krwi. W całym organizmie mamy jej aż 4,5l (4500ml). Tak więc próba stanowi 1/450. całej objętości krwi. To bardzo bardzo mało. Mimo to lekarz potrafi określić, co dzieje się w całym organizmie, a nie tylko w pobranej próbce. Weźmy inny, mniej dramatyczny przykład. Wyobraźmy sobie, że chcemy ugotować zupę i lubimy, gdy jest ona odpowiednio słona. Jak to sprawdzamy? Czy musimy wypić całą zupę z garnka? Absolutnie nie. Wystarczy jedna łyżeczka, która zawiera

15ml zupy i stanowi zaledwie 3/1000 pięciolitrowego garnka. Znowu dobraliśmy niewielką próbkę, żeby zbadać większą całość. Ale przypadek zupy jest szczególny. Żeby przekonać się, czy zupa jest odpowiednio słona, musimy ją najpierw dobrze WYMIESZAĆ. Tylko wtedy proporcja soli w łyżce zupy, będzie taka sama jak proporcja w całym garnku - łyżka zupy będzie dobrze reprezentować całą zupę. Kluczową kwestią jest więc REPREZENTATYWNOŚĆ PRÓBY. Na pewno niektórzy słyszeli już to pojęcie. Zapamiętajmy je na chwilę, chociaż później będziemy musieli z niego zrezygnować. Reprezentatywność można rozumieć na różne sposoby. Często mówi się, że tak jak w przypadku łyżki zupy, próba musi być "miniaturą" populacji. Innymi słowy powinna odtwarzać strukturę i zależności obserwowane w całej zbiorowości. W przypadku badania krwi lub zupy brzmi to sensownie. Zauważmy jednak, że badane substancje są "jednorodne". Każda porcja zupy czy krwi jest właściwie identyczna (dla uproszczenia, bo specjalisci z pewnością powiedzą, że to nie jest takie proste). A co jeśli badana zbiorowość nie jest i nie może być jednorodna? To problem, z którym bardzo często mierzą się nauki społeczne.

Zadania 3:

1. Podaj przykład badania na próbie, które można

przeprowadzić w naszym codziennym życiu (np. badanie ilości soli w zupie).

L.p. Czego dotyczy badanie? Co jest próbą?

- 1.
- 2.
- 3.

4. Sondaż

Przejdźmy do badań społecznych i tzw. sondaży politycznych. Ich wyniki często pojawiają się w prasie i mają duże znaczenie dla polityków oraz pewnie trochę mniejsze dla wyborców. Wyobraźmy sobie, że chcemy zmierzyć poziom poparcia dla wybranej partii X w wyborach do parlamentu. Dla naszych celów odsłońmy kilka faktów dotyczących badanej zbiorowości:

Tabela 4.1. Rozkład preferencji wyborczych w (fikcyjnej) populacji dorosłych obywateli Polski

Popieram partię X	Nie popieram partii X	Nie biorę udziału w wyborach	Razem
-------------------	-----------------------	------------------------------	-------

Miasto	25%	5%	30%	60%
Wies	10%	10%	20%	40%
Razem	35%	15%	50%	100%

W powyższej tabeli umieściliśmy procentowy rozkład dwóch cech w populacji pełnoletnich mieszkańców Polski (dane fikcyjne). Pierwszą cechą jest miejsce zamieszkania (w wierszach), a drugą poparcie dla partii X (w kolumnach). Widzimy, że w miastach mieszka 60% ludności, a na wsi 40%. Łącznie w całej zbiorowości (RAZEM) 35% obywateli zagłosowały na partię X, 15% na inną partię, a 50% w ogóle nie wzięłyby udziału w wyborach. Możemy również powiedzieć, że osoby mieszkające w mieście i popierające partię X stanowią 25% ogółu uprawnionych do głosowania, a także że osoby które mieszkają na wsi i nie biorą udziału w wyborach stanowią 20% populacji. Oczywiście, w normalnych warunkach tego typu informacje są niedostępne dla badacze. My je “odsłaniamy” na potrzeby kursu.

Zastanówmy się więc, jak z powyższej populacji dobrać próbę reprezentatywną, na podstawie której będziemy mogli trafnie oszacować wielkość odsetka mieszkańców Polski popierających partię X. Tu pojawia się pierwszy problem. Nie da się bowiem “wymieszać” obywateli tak jak zupy. Zbiorowość nie jest jednorodna, a preferencje wśród mieszkańców miast i wsi nie są identyczne. Nie ma

więc gwarancji, że jeśli idąc ulicą w mieście lub na wsi zapytamy dowolnych 10 osób o ich preferencje polityczne to będziemy mogli powiedzieć, jakie jest poparcie w całym kraju. W uproszczeniu w mieście 4 na 10 osób zagłosowałoby na partię X (bo $25\%/60\% = 0,4$), a na wsi 1 na 4 (bo $10\%/40\% = 0,25$). W pierwszym przypadku poparcie będzie zawyżone, a w drugim zniżone, w stosunku do ogólnokrajowych wyników (35%). Widać więc, że nie każda próba będzie

“REPREZENTATYWNA”. Zatrzymajmy się znowu przy tym pojęciu. Żeby prawidłowo przeprowadzić nasz polityczny sondaż musimy zrealizować badanie zarówno na wsi jak i w mieście. Każda osoba należąca do populacji musi mieć szansę znalezienia się w próbie. To bardzo ważne! Sposób dobierania próby, zwany **schematem doboru próby**, nie może uniemożliwić nikomu znalezienia się w próbie. W dalszej części kursu pokażemy, że “REPREZENTATYWNOŚĆ” wcale nie polega na tworzeniu miniatury populacji, lecz na tworzeniu warunków, w których za pomocą próby, z dużą dokładnością można opisać wybrane pojedyncze cechy populacji. Opowiem również, jak prawidłowo zdefiniować populację, jak błędy w kwestionariuszu mogą wpływać na zachowanie respondentów, a także jak dobrą, a dokładnie wylosować próbę reprezentatywną.

Zadania 4:

1. Czy w przypadku badania populacji mieszkańców Polski, wyniki sondy ulicznej przeprowadzonej w Warszawie, Krakowie lub Poznaniu można uznać za wiarygodne (dające się uogólnić na całą populację)?
 - Tak
 - Nie
2. Na podstawie danych z *Tabela 4.1. Rozkład preferencji...* odpowiedz na poniższe pytania:
 - a. Jaki procent mieszkańców wsi popiera partię X?
.....
 - b. Jaki procent mieszkańców miasta nie bierze udziału w wyborach?
.....
 - c. Jaki procent osób które NIE popierają partii X mieszka w mieście?
.....

5. Problem badawczy -> populacja -> technika realizacji badania

Co powinien zrobić każdy prawdziwy sondażysta przed rozpoczęciem badania? Powienien odpowiedzieć sobie na trzy podstawowe pytania.

1. Jak zdefiniować problem badawczy?

W przypadku badań sondażowych to pytanie możemy uściślić w następujący sposób: jaką cechę naszej zbiorowości chcemy zmierzyć. Co rozumieć przez cechę w tym przypadku? Opisując dużą zbiorowość musimy zdecydować się na jakiś kompromis i uogólnienia. Sondaż nie odpowie nam na pytanie, jak zachowują się poszczególni mieszkańcy Polski. Możemy natomiast za jego pomocą określić, jak zachowują się przeciętnie albo jak zachowuje się większość z nich. Sondaż pozwala więc badać parametry, cechy, populacji takie jak średnia (np. waga) lub odsetek osób w pewien sposób wyróżnionych (np. odsetek osób uczestniczących w wyborach).

Najczęściej sondaże realizuje się, aby poznać preferencje wyborcze obywateli. Problemem badawczym może być jednak także coś innego. Wiele sondaży służy ocenie przeciętnych dochody, wydatków, a także preferencji konsumenckich w danej grupie społecznej.

Określenie problemu badawczego stanowi punkt wyjścia do następnego pytania.

2. Jak zdefiniować badaną populację? O jakiej zbiorowości chcemy wnioskować?

Założmy, że interesuje nas populacja Polaków. Czy to znaczy, że będziemy badać wszystkich ludzi na świecie posiadających obywatelstwo polskie albo mówiących po polsku? A może tylko tych spośród nich, którzy mieszkają

w kraju? Albo ogólnie mieszkańców Polski niezależnie od tego, czy posiadają obywatelstwo czy też nie?

Musimy również zadecydować, czy interesują nas ludzie w każdym wieku, czy może tylko pełnoletni z prawem do głosowania (18+)? Lub też osoby w wieku produkcyjnym i poprodukcyjnym (15+)?

Najczęściej sondaże w Polsce obejmują populację pełnoletnich obywateli zamieszkałych na terenie kraju. Istnieją jednak badania społeczne dla których, ze względu na poruszaną problematykę, populacje definiuje się zupełnie inaczej. I tak istnieją badania zbiorowości osób w wieku 15 lat, zbiorowości pełnoletnich obywateli województwa małopolskiego itp.

Określenie problemu badawczego i zdefiniowanie populacji to nie wszystko. Trzeba bowiem jeszcze dostosować do nich sposób w jaki pozyskiwane będą dane od respondentów.

3. Jaką techniką zrealizować badanie?

Trudno sobie bowiem wyobrazić badanie dochodów mieszkańców Polski przeprowadzone przy pomocy ankiety internetowej. Dalece bowiem nie wszyscy członkowie tej zbiorowości są w łączności z “globalną siecią”. Bywają oczywiście sytuacje, w których stosowanie takiej metody jest uzasadnione i z pewnością w przyszłości, wraz z poprawą dostępu do sieci, zdominuje ona badania sondażowe. Jak na razie jednak

ankiety internetowe funkcjonują w cieniu dwóch innych częściej stosowanych metod zbierania danych - bezpośredniej i telefonicznej. Ogólnie możemy więc wyróżnić trzy następujące grupy technik realizowania badań sondażowych:

1. Wywiady bezpośrednie;
2. Wywiady telefoniczne;
3. Ankiety internetowe.

Wywiady "bezpośrednie" są realizowane w "terenie", najczęściej w miejscu zamieszkania losowo dobranych respondentów. Pozwalają on dotrzeć do najszerzej grupy ludności, nieosiągalnej poprzez połączenie telefoniczne, bądź przez Internet. Tego typu technikę często stosuje się w badaniach ogólnopolskich, dotyczących populacji mieszkańców całego kraju. Metoda bezpośrednią wiąże się z dużymi kosztami oraz długim okresem realizacji. Ankieterzy muszą bowiem dotrzeć osobiście do wszystkich respondentów. Czasami pokonują w tym celu kilkadziesiąt kilometrów. Zdarza się bowiem, że odległości między poszczególnymi respondentami są bardzo duże. Co więcej, respondenci bywają nieuchwytni i w związku z tym, aby się z nimi skontaktować, trzeba wielokrotnie ponawiać wizyty. Sondaże bezpośrednie wymagają więc dużych nakładów pracy. Mają one jednak jedną poważną zaletę. Są nią ankieterzy. Dzięki nim respondentci częściej biorą udział w badaniu. Odsetkiem

odmów wzięcia udziału w sondażach realizowanych metodą bezpośrednią jest stosunkowo niższy. Respondentom trudniej jest odmówić ankieterowi, którego widzi na własne oczy, niż takiemu który kontaktuje się z nimi telefonicznie.

Wywiady telefoniczne to zupełnie inna bajka. Obecnie jest to bardzo powszechnie stosowana metodą zbierania danych. Jej popularność wynika przede wszystkim z niskich kosztów oraz krótkiego czasu realizacji. Łatwiej jest bowiem przeprowadzić 1000 rozmów telefonicznych niż zorganizować 1000 spotkań z respondentami. Sami jednak wiemy, jak takie badania wyglądają w praktyce. „Przepraszam, nie mam czasu” albo „już brałem udział w tym badaniu” przychodzi nam przez telefon dużo łatwiej, niż podobne zachowanie w bezpośredniej konfrontacji z ankieterem. Nic więc dziwnego, że w tego rodzaju badaniach tylko co dziesiąty telefon kończy się zrealizowanym wywiadem. Dla porównania technika bezpośrednią pozwala zrealizować wywiad z co trzecią dobraną do próby osobą.

Poważnym ograniczeniem badań telefonicznych jest to, że mogą one być realizowane tylko w odniesieniu do populacji, której członkowie posiadają telefony. Co więcej wywiady telefoniczne muszą być w miarę krótkie ponieważ respondenci nie lubią długo “wisieć” na telefonie. W związku z tym konieczne jest również ograniczenie liczby pytań w ankiecie, a co za tym idzie

ograniczenie liczby możliwych do przeprowadzenia analiz.

Ostatnia grupa technik to wspomniane wcześniej ankiety internetowe. Nie chodzi tu o sondy umieszczane na stronach WWW, ale o badania, w których sondażysta wybiera respondentów np. wysyłając im maile z kluczem do ankiety umieszczonej w sieci. Tego typu techniki budzą wiele kontrowersji. W szczególności, gdy są wykorzystywane do wnioskowania o populacjach, takich jak mieszkańcy Polski. Wiadomo bowiem, że nie wszyscy w tej zbiorowości posiadają komputer i wiedzą jak korzystać z Internetu. Innymi słowy jest to technika wykluczająca duże grupy społeczne, prawdopodobnie większe niż w przypadku badania telefonicznego. W związku z tym trudno jest w tym przypadku mówić o reprezentatywności (będziemy jeszcze mówili co to znaczy) wyników dla populacji wszystkich mieszkańców Polskie. Powodem dla którego realizuje się tego rodzaju badania jest koszt, niewątpliwie niższy niż w przypadku wywiadów bezpośrednich czy telefonicznych.

Aby łatwiej rozróżnić poszczególne metody zbierania danych sondażyści nadali im specjalne oznaczania. Można się na nie natknąć w artykułach prasowych lub w notach metodologicznych różnych badań. Poniżej znajduje się tabela z objaśnieniami oznaczeń czterech najczęściej stosowanych technik zbierania danych wraz z ich

oznaczeniem.

Tabela 5.1. Techniki realizacji badań sondażowych

	PAPI (Paper Technika And Pencil Interviewing)	CAPI, (ComputerAssisted Personal Interviewing)	CATI (Compute Assisted Telephone Interviewi
Opis	Wywiad bezpośredni - respondent dostaje od ankietera kartkę z pytaniami, bierze ołówek lub długopis i zapisuje (zaznacza) swoje odpowiedzi;	Wywiad bezpośredni - ankieter przychodzi do respondenta z laptopem (lub innym urządzeniem mobilnym) i odczytuje ze specjalnego programu pytania, a następnie zapisuje w nim odpowiedzi.	Wywiad telefoniczny ankieter dz pod losow wybrany na telefonu i osobie, któ odbierze, z pytania z przygotowa kwestionar Odpowied zapisywane specjalnym programie komputero

	możliwość stosowania techniki do bardzo różnych populacji; stosunkowo niski odsetek odmów wzięcia udziału w badaniu; możliwość gromadzenia danych bezpośrednio w systemie komputerowym	bardzo szybka realizacja; stosunkowo niskie koszty możliwość gromadzenia danych bezpośredniem systemie komputerowym
--	--	---

Zalety	wysokie koszty; długi czas realizacji; konieczność przepisywania wyników z ankiety papierowej do systemu komputerowego - możliwe błędy
--------	--

Wady	wysoki koszt; długi czas realizacji;	duży odsetek odmów wzrostu udziału w badaniu; ograniczona zastosowanie techniki
------	--------------------------------------	---

Szczególnie istotną cechą różnicującą opisane powyżej techniki jest tak zwany poziom realizacji próby (nazywany

również z angielskiego *response-rate*), czyli odsetek osób, z którymi udało się przeprowadzić wywiad, wśród ogółu osób wytypowanych przez nas do udziału w badaniu. Warto wiedzieć, że praktycznie nigdy nie występuje sytuacja, w której sondażystom udaje się zapytać o opinię wszystkie osoby dobrane próby. Dzieje się tak z kilku powodów. Po pierwsze, nie do wszystkich osób udaje się dotrzeć/dodzwonić. Powodem mogą być choroby, wakacje, delegacje służbowe itp. Ponadto część osób, do których uda się dotrzeć, z różnych przyczyn odmawia wzięcia udziału w badaniu. Oba te czynniki są dodatkowym źródłem błędów w badaniu i celem każdego dobrego badacza jest zminimalizowanie ich wpływu. Tak jak wspominaliśmy wcześniej, duże znaczenie ma tutaj wybór techniki zbierania danych - jedne "odstraszają" respondentów bardziej a inne mniej.

Problem poprawnego zdefiniowania populacji oraz dobrania do niej odpowiedniej techniki badawczej dobrze ilustruje przykład historyczny. W 1936 roku „Literary Digest”, popularny magazyn informacyjny przeprowadził w USA badanie przedwyborcze. Do ludzi wybranych z książek telefonicznych i list rejestracyjnych samochodów wysłano DZIESIĘĆ MILIONÓW kart pocztowych, pytając, na kogo zamierzają oddać głos w wyborach prezydenckich – na republikanina Alfa Landona czy demokratę Franklina Roosevelta? Odpowiedziało ponad dwa miliony ludzi, wskazując że nowym prezydentem

wybrany zostanie Alf Landon (57%), a nie Franklin Roosevelt (43%). Mogłoby się wydawać, że przebadanie tylu osób jest dużo bardziej wiarygodne i miarodajne niż przeprowadzenie badania na niewielkim wycinku populacji. Nic bardziej mylnego. Realne wybory dość drastycznie zweryfikowały wnioski z tych badań – nowym prezydentem został Franklin Roosevelt, mając największą przewagę głosów w historii – otrzymał 61%.

Dla porównania, w tym samym czasie, przedwyborczy sondaż przeprowadził również George Gallup, który trafnie przewidział wyniki wyborów. W swoim badaniu posłużył się on jednak niewielką próbą kwotową (czyli opartą na znajomości określonych cech populacji, np. płeć, dochód, wiek, miejsce zamieszkania itp.) zrealizowaną technika bezpośrednią.

Na czym więc polegał problem „Literary Digest”? Na całkowitym braku kontroli nad próbą. Na pytanie zadane przez magazyn odpowiedziało zaledwie 22% wszystkich zapytanych osób. Jak się okazało karty w większości odsyłali republikanie. Drugi problem polegał na nieprawidłowym zdefiniowaniu populacji. Respondenci do badania zostali dobrani na podstawie spisu abonentów telefonicznych i właścicieli samochodów. Taka konstrukcja próby dawała nadreprezentację zamożnych wyborców, czyli pominięcie ludzi biednych, którzy w większości głosowali na „New deal” Roosevelta. Przykład ten ilustruje, jak realizacja nawet dużej próby

przy użyciu błędnej techniki i przy niepoprawnej definicji populacji może doprowadzić do zupełnie nietrafnych wniosków. Aby się przed tym ustrzec pamiętajmy, aby analizując wyniki badań sondażowych zwracać uwagę na to, czy metoda przeprowadzania wywiadów nie “odstraszała” respondentów oraz czy definicja zbiorowości generalnej zawierała informacje o:

- położeniu w przestrzeni zbiorowości (mieszkańcy Polski, mieszkańcy Wielkopolski, mieszkańcy Małopolski, mieszkańcy Łodzi itp.)
- wieku respondentów (osoby pełnoletnie, osoby w wieku 15 i więcej ukończonych lat, osoby w wieku 65+ itp.)
- inne dodatkowe cechy wyróżniające (osoby posiadające obywatelstwo polskie, osoby z wykształceniem wyższym itp.)

Zadania 5:

1. Wymień trzy firmy zajmujące się badaniem rynku i opinii społecznej, które w 2014 r. prowadziły badania sondażowe w Polsce. Podaj link do strony, na której firmy te publikują swoje raporty.

L.p. **Nazwa firmy badawczej**

Link do strony z raportem?

1.
2.
3.

2. Odpowiedz na pytania związane z następującym problemem badawczy: Chcesz zmierzyć poziom czytelnictwa tygodnika “*Na Straży Sondaży*”.
- Ukazuje się on wyłącznie w formie drukowanej w miastach wojewódzkich (siedzibach wojewodów lub/i w siedzibach sejmików wojewódzkicj). Pismo ma charakter popularnonaukowy i jest przeznaczone dla wszystkich, niezależnie od wykształcenia czy wieku.
- a. Jak zdefiniujesz czytelnictwo? (Pytania pomocnicze: Kiedy ktoś staje się czytelnikiem? Jak często trzeba czytać żeby stać się czytelnikiem? Ile trzeba przeczytać żeby stać się czytelnikiem?)
-
.....
.....
- b. Jak zdefiniujesz populację czytelników? (Pytania pomocnicze: jaki jest minimalny wiek czytelnika? czy czytelnicy mieszkają tylko w miastach wojewódzkich czy także w innych miejscowościach?)
-

.....

.....

.....

.....

.....

3. Odpowiedz na pytania związane z następującym badaniem: Dom Badań Marketingowych przygotowała na zlecenie Fundacji Wspieram jmy Potrzebujące Dzieci raport pt.: "Niedożywienie polskich dzieci". Został on przygotowany na podstawie badanie przeprowadzono telefonicznie (technika CATI) na ogólnopolskiej próbie 800 przedstawicieli instytucji zajmujących się dziećmi i ich sytuacją życiową. Badanie dotyczyło dzieci uczęszczających do klas 1-3. Z raportu wynika, że pracownicy szkół i pracownicy Ośrodków Pomocy Społecznej szacują, że co dziesiąte dziecko w tej grupie wiekowej dotyka problem niedożywienia.

a. Zdefiniuj badaną populację:

.....

.....

.....

.....

b. Podaj liczbę uczniów klas 1-3 w Polsce w 2015 r. (mogą to być dane zgodne ze stanem na 1 czerwca 2014 r. lub dla wcześniejszej daty przed końcem 2013 r.):

.....

c. Maksymalnie w trzech zdaniach napisz, co rozumiesz przez **niedożywienie**:

.....

-
-
-
- d. W raporcie z badania stwierdzono, że w Polsce z głodu cierpi około 800 000 dzieci. Czy dane zebrane na potrzeby badania dają podstawy dla takiego wniosku? Odpowiedź uzasadnij.
- Tak,

.....

 - Nie,

.....
4. Uniwersytecki Zespół Na Straży Sondaży zamówił badanie dotyczące popularności strony internetowej “www.nastrazysondazy.uw.edu.pl”. Chodziło oszacowanie odsetka osób w wieku 18-35 lat zamieszkałych w Polsce, które w ciągu ostatniego miesiąca zapoznały się z treścią (przeczytały cały lub prawie cały) przynajmniej jednego artykułu na stronie. Badanie zostało przeprowadzone metodą CAWI przez firmę “Polski Panel Internetowy” na próbie 917 osób w wieku 18-35 lat spośród 50 tys. osób które dobrowolnie zarejestrowały się do bazy internetowej firmy i za drobną opłatą zgadzają się odpowiadać na pytania w różnych ankietach. Wiadomo również, że osoby do badania zostały dobrane w ten sposób, aby rozkład płci wieku oraz wielkości miejscowości deklarowanego

zamieszkania był zgodny z danymi podawanymi przez GUS na temat mieszkańców Polski. Odpowiedzi na pytania związane z tym badaniem:

- a. Czy populacja osób posiadających dostęp do Internetu (korzystających z Internetu do celów prywatnych w domu, bibliotece, pracy lub szkole/uczelnii) obejmuje wszystkich mieszkańców Polski?

■ Tak,

.....

■ Nie,

.....

- b. Czy próba badawcza 917 respondentów została dobrana z populacji osób posiadających dostęp do Internetu?

■ Tak,

.....

■ Nie,

.....

- c. Czy badanie zlecone przez Na Straży Sondaży obejmuje populację polskich internautów.

■ Tak,

.....

■ Nie,

.....

- d. Czy rozkład płci, wieku i wielkości miejscowości zamieszkania w próbie jest

zgodny z rozkładem tych cech w populacji osób posiadających dostęp do Internetu w Polsce?

- Tak,

.....

- Nie,

.....

6. Kwestionariusz

Mając wybrany problem badawczy i dobraną do niego odpowiednią technikę zbierania danych możemy zająć się kolejnym składnikiem sondażu - kwestionariuszem. Choć rzadko się o tym mówi, jego konstrukcja ma ogromne znaczenie dla wyników badania. Ilustruje to dobrze klasyczny już przykład eksperymentu opisanego przez Schumana, zrealizowanego w 1986 roku w Stanach Zjednoczonych. Badacze z Uniwersytetu Michigan zapytali o najważniejsze wydarzenia lub zmiany, jakie zaszły w ostatnich 50 latach i wydają się respondentom najbardziej istotne. Połowa ankietowanych miała do dyspozycji następującą listę odpowiedzi:

- II wojna światowa,
- podbój kosmosu,
- zabójstwo J. F. Kennedy'ego,
- wynalezienie komputera,
- wojna w Wietnamie,

- inne,
- nie wiem.

Pozostali otrzymali pytanie otwarte, a więc sami musieli zaproponować odpowiedzi. W pierwszej grupie aż 30% respondentów wybrało wynalezienie komputera jako najbardziej istotne wydarzenie lub zmianę ostatnich 50 lat. Wśród pozostałych osób podobnej odpowiedzi udzielił zaledwie 1% ankietowanych. Ta ogromna różnica najlepiej pokazuje, jak wiele zależy od formy zadawanych pytań.

To jednak nie wszystko. Na sposób udzielania odpowiedzi przez respondentów wpływ ma nie tylko konstrukcja pytań, ale także użyty w nich język. Dlatego kwestionariusz powinien odnosić się do rzeczywistości w sposób neutralny, bez sądów czy sugerowania, które odpowiedzi są “dobre”.

Czasami jednak kontrowersje są nie do uniknięcia. Dotyczy to tzw. kwestii drażliwych. Stanowią one poważny problem dla sondażystów. Respondenci raczej niechętnie udzielają odpowiedzi na pytania dotyczące ich intymnych spraw. Trudno im się przyznać do problemów z nałogami, do niewierności w związku czy popełnionych przestępstw. Zazwyczaj w takich sytuacjach nawet gwarancja anonimowości nie jest wystarczającą “zachętą”, aby udzielać szczerzych odpowiedzi. Czasami,

aby przełamać nieczęć respondentów do opowiadania o swoich intymnych sprawach, stosuje się techniki niebezpośrednie.

Problemy związane z budowaniem kwestionariusza i interpretacją otrzymanych na jego podstawie wyników przeanalizujemy na przykładzie zadań. Pokażę nam one, że za takie a nie inne wyniki badań często odpowiedzialny jest sposób formułowania pytań.

Zadania 6:

1. Odpowiedz na pytania związane z następującym badaniem: Firma EkstraSondaż przygotowała w 2015 r. raport „Młodzież kupująca substancje psychoaktywne przez Internet” dla Państwowego Zakładu Badań nad sieciami komputerowymi. Dane zostały zebrane od respondentów przez Internet. W badaniu wzięli udział uczniowie wybranych szkół w całej Polsce, którzy dostali adres internetowy oraz indywidualne hasła do strony z ankietą. Łącznie próba badawcza liczyła 1050 nastolatków - osób w wieku 13-16 lat. Spośród nich 560 zadeklarowało, że robi zakupy przez Internet. W tej grupie 15 osób zadeklarowało, że zdarzyło im się kupić substancje psychotropowe (Pytanie brzmiało: „Czy zdarzyło Ci się kupić przez internet substancje psychotropowe, w

tym leki wpływające na funkcjonowanie mózgu?”). Wiadomo również, że w badaniu wzięło udział 180 szesnastolatków, z czego 110 robi zakupy przez internet, a 10 deklaruje zakup substancji psychotropowych.

a. Czy uważasz, że respondenci generalnie (a wiec przytaczająca większość z nich) udzielali szczerzych odpowiedzi na pytanie o to, czy kupuję substancje psychotropowe - zarówno Ci którzy przyznali się do tego typu zachowań jak i ci którzy ich nie potwierdzili?

- Tak - generalnie odpowiadali szczerze;
- Nie - generalnie odpowiadali nieszczerze;
- Nikt tego nie wie. Może część tak, a część nie.
- Mam własne zdanie:
.....
.....

b. Czy uważasz, że badanie przez Internet daje większe poczucie anonimowości niż klasyczne badania prowadzone przez telefon lub twarzą w twarz z ankieterem? Uzasadnij maksymalnie w trzech zdaniach.

- Tak,

-
.....
■ Nie,

-
.....
- c. Czy w badaniu przeprowadzonym przez Internet byłabyś / byłbyś skłonny odpowiadać szczerze na pytania dotyczące seksualności, chorób intymnych lub łamania prawa? Dlaczego?
(Pytania pomocnicze: zastanów się, czy ważna w tych kwestiach jest anonimowość i poufność danych, a także, czy w ogóle mówienie na ten temat sprawia Ci jakieś problemy):
- Tak,
.....
.....
.....
 - Nie,
.....
.....
.....
- d. Jaki procent nastolatków, którzy wzięli udział w powyższym badaniu zadeklarował, że robi zakupy przez internet i kupuje w ten sposób środki psychotropowe?
-
- e. Jaki procent osób, które zadeklarowały w badaniu, że kupują środki psychotropowe to szesnastolatkowie?
-
- f. Jaki procent nastolatków, które kupują coś przez internet, kupuje substancje psychotropowe?

-
- g. Czy znając wyniki badania uznałabyć/uznałbyć następujące nagłówki prasowe za trafne:
- i. „Nowa plaga w sieci. Nastolatki kupują narkotyki przez internet”
 - Tak
 - Nie
 - ii. „Gimnazjalistki kupują psychotropy w sieci. Nowa plaga w internecie”
 - Tak
 - Nie
 - iii. “Nowe zjawisko w internecie. Gizmazjaliści kupują narkotyki w sieci”
 - Tak
 - Nie
 - iv. “Uwaga na zakupy przez internet. Niektóre nastolatki kupują w ten sposób narkotyki”
 - Tak
 - Nie
2. Poniżej znajdują się dwa sondaże. Wypełnił je i odpowiedz na pytania:
- a. Sondaż 1.
 - i. Czy gdyby wybory odbyły się w najbliższą niedzielę to wziąłby(łaby) Pan(i) w nich udział?
 - Tak
 - Nie

ii. Jeżeli tak, to na jaką jedną partię oddałby Pan(i) głos?

- oddam pusty/nieważny głos;
- na partię X;
- na partię Przyjaciół Demokracji;
- na partię Przyjaciół Otwartości;
- na partię Przyjaciół Społeczeństwa;
- na partię Przyjaciół Środowiska;
- na partię Przyjaciół Uczciwości;

b. Sondaż 2.

i. Proszę określić, czy zgadza się Pan/Pani z następującymi stwierdzeniami:

- Podatki w Polsce są za wysokie i należy je niezwłocznie obniżyć.
 - + Tak
 - + Nie
- ZUS jest organizacją drogą, nieefektywną i marnującą publiczne pieniądze.
 - + Tak
 - + Nie
- Obywatele lepiej będą zarządzać swymi pieniędzmi niż urzędnicy w ich imieniu.
 - + Tak
 - + Nie
- Każda rodzina powinna móc liczyć na

wsparcie ze strony państwa.

+ Tak

+ Nie

ii. Czy słyszał(a) Pan(i) o powstaniu nowej partii X, której programem jest m. in. obniżenie podatków, ograniczenie obciążień biurokratycznych, zmniejszenie liczby urzędników oraz wsparcie dla rodzin?

■ Tak

■ Nie

iii. Czy gdyby partia X brała udział w najbliższych wyborach do Sejmu to na jaką jedną partię oddałby Pan(i) głos?

■ oddam pusty/nieważny głos;

■ na partię X;

■ na partię Przyjaciół Demokracji;

■ na partię Przyjaciół Otwartości;

■ na partię Przyjaciół Społeczeństwa;

■ na partię Przyjaciół Środowiska;

■ na partię Przyjaciół Uczciwości;

c. Pytanie do sondaży 1. oraz 2.:

i. Czy w obu sondażach wybrałaś/wybrałeś taką samą partię?

■ Tak

■ Nie

ii. Czy twoim zdaniem kolejność pytań w sondażu 1. 2. może mieć wpływ na

odpowiedzi respondentów?

- Tak
- Nie

7. Dobór próby

Mając już określony problem badawczy, wybraną technikę zbierania danych oraz przygotowany, odporny na kwestie wrażliwe, kwestionariusz, możemy zająć się doborem respondentów. W branży sondażowej korzysta się w tym zakresie z różnych rozwiązań, ale tylko jedno jest dobrze opracowane od strony teoretycznej i, przy odpowiedniej staranności wykonania, pozwala rzetelnie wnioskować o populacji. Dlatego też zajmiemy się wyłącznie nim – mowa o doborze losowym.

Zacznijmy od tego, że aby dobrać próbę z populacji w sposób losowy potrzebujemy spisu wszystkich osób do niej należących. Musimy mieć z czego wybierać. Taka listę nazywamy operatem losowania. Aby był on użyteczny musi zawierać nie tylko informację o tym, ile osób znajduje się w populacji, ale także jak można się z nimi skontaktować, czyli np. gdzie mieszkają. W Polsce istnieją przynajmniej dwa takie operaty, które umożliwiają dobieranie prób sondażowych w sposób losowy. Pierwszy z nich to rejestr **PESEL** (Powszechny Elektroniczny System Ewidencji Ludności) zawierający

spis wszystkich obywateli Polski oraz osób posiadających prawo do pobytu na terenie kraju wraz z danymi o miejscu ich zameldowania. Rejestr ten jest zarządzany przez *Ministerstwo Spraw Wewnętrznych*. Drugim operatorem jest **TERYT** ([Krajowy Rejestr Urzędowy Podziału Terytorialnego Kraju](#)) zawierający informację o wszystkich mieszkaniach w Polsce. Za jego pomocą, dobierając mieszkania, można dobierać próby mieszkańców Polski. Rejestr ten prowadzi *Główny Urząd Statystyczny*.

Wiemy już czego potrzebujemy, żeby dobrać próbę więc możemy się zastanowić, jak to zrobić. Pomoże nam w tym uproszczony przykład. Założymy, że chcemy oszacować poziom absencji wyborczej w (fikcyjnej) populacji pełnoletnich obywateli Polski zamieszkujących w kraju, składającej się z 20 osób (korzystaliśmy już z niej wcześniej). Będziemy chcieli na podstawie próby oszacować, jaki odsetek obywateli nie pójdzie na wybory. Skorzystamy przy tym z techniki wywiadu bezpośredniego. Naszym respondentom zdamy pytanie: “**Gdyby w najbliższą niedzielę odbywały się wybory do Sejmu i Senatu, to czy wział(ęła)by Pan(i) w nich udział?**”. Zakładamy, że pytanie to nie jest drażliwe i wszyscy respondenci odpowiedzą na nie zgodnie ze swoimi przekonaniami.

Tabela 7.1. Absencja wyborcza w (fikcyjnej) populacji mieszkańców Polski

Miejsce zamieszkania	Nie, nie pójdę na wybory	Tak, pójdę na wybory	Razem
Miasto	6	6	12
Wieś	4	4	8
Razem	10	10	20

Powyższa tabela zawiera rozkład odpowiedzi na nasze pytanie wśród mieszkańców wsi oraz miast. Widzimy, że w miastach mieszka 12 osób, na wsi 8. Głosować nie zamierza łącznie 10 osób, co oznacza, że absencja wyborcza w populacji wynosi 50%. W normalnych warunkach rozkład ten byłby dla nas zupełną tajemnicą. Odkrywamy go zeby zobaczyć jaki ma on wpływ na to co będzie się działo w dobieranych przez nas próbach.

Teraz potrzebujemu operatu losowania. W naszym przypadku wygląda on następująco:

Tabela 7.2. Operat losowania

L.p. Miejsce zamieszkania Preferencje wyborcze

- | | |
|----------|---------------------|
| 1 MIASTO | Głosuję na partię X |
| 2 MIASTO | Głosuję na partię X |

3 MIASTO	Głosuję na partię X
4 MIASTO	Głosuję na partię X
5 MIASTO	Głosuję na partię X
6 MIASTO	Głosuję na inną partię niż X
7 MIASTO	Nie idę na wybory
8 MIASTO	Nie idę na wybory
9 MIASTO	Nie idę na wybory
10 MIASTO	Nie idę na wybory
11 MIASTO	Nie idę na wybory
12 MIASTO	Nie idę na wybory
13 WIEŚ	Głosuję na partię X
14 WIEŚ	Głosuję na partię X
15 WIEŚ	Głosuję na inną partię niż X
16 WIEŚ	Głosuję na inną partię niż X
17 WIEŚ	Nie idę na wybory
18 WIEŚ	Nie idę na wybory
19 WIEŚ	Nie idę na wybory
20 WIEŚ	Nie idę na wybory

Ustalmy, że kolejne kolumny w powyższym zbiorze oznaczaj:

- L.p.- liczbę porządkową obywatela - jego identyfikator;

- Miejsce zamieszkania - miasto lub wieś;
- Preferencje wyborcze - czy i jak dany obywatel zagłosowałby, gdyby wybory odbyły się w najbliższą niedzielę;

Operat można również pobrać ze strony:

<https://docs.google.com/spreadsheets/d/1iSt2ZD9F8DhEhGw/pubhtml?gid=1189066294&single=true>

Nasz problem badawczy dotyczy absencji wyborczej, co oznacza, że interesuje nas odsetek osób, które na pytanie o preferencje wyborcze (kolumna “Preferencje wyborcze”) odpowiadają: “Nie, nie pójdę na wybory”.

Określiliśmy problem badawczym, zdefiniowaliśmy populację, wybraliśmy technikę realizacji badania, stworzyliśmy kwestionariusz, mamy operat losowania więc możemy wreszcie zająć się losowaniem respondentów.

Skorzystamy z bardzo uproszczonego schematu doboru próby (w ten sposób sondażyści nazywają zasady wg których dobierają respondentów). Z naszej 20 osobowej populacji będziemy losowali 2 różne osoby w następujących krokach:

1. Losujemy pierwszego respondenta - jedną osobę

spośród 20. Każdy obywatel w populacji ma takie samo prawdopodobieństwo znalezienia się w próbie - wynosi ono 1/20;

2. Spośród pozostałych 19 osób losujemy drugiego respondenta. Ponownie wszystkie osoby pozostające w populacji mają takie samo prawdopodobieństwo znalezienia się w próbie - wynosi ono 1/19;
3. Otrzymujemy próbę, w której znajdują się dwie osoby. Poza próbą zostaje 18 obywateli.

Powyższy schemat losowania nazwiemy: **losowaniem prostym bez zwracania**. Jest ono proste ponieważ na każdym etapie wszyscy obywatele pozostający w populacji mają takie samo prawdopodobieństwo dostania się do próby. Bez zwracania ponieważ po wylosowaniu jednej osoby do próby nie zwracamy jej do populacji. Każdy może zostać wylosowany tylko raz. Ten uproszczony schemat doboru zaledwie dwóch respondentów ułatwi nam analizę podstawowych zagadnień związanych z metodą reprezentacyjną.

Zacznijmy od tego, jak będą wyglądały próby dobierane wg naszego schematu. Ustalmy, że respondentów będziemy identyfikować na podstawie ich liczby porządkowej, czyli liczby z kolumny L.p. w naszym operacie. Zapis (1,2) oznaczać będzie, że do próby wylosowaliśmy najpierw osobę o liczbie porządkowej 1, a następnie osobę o liczbie porządkowej 2. Wszystkie

możliwe próby możemy więc rozpisać korzystając z prostej reguły. Jeżeli w pierwszym kroku dobierzemy osobę o liczbie porządkowej 1 to w drugim kroku, do pary, możemy dobrać osoby z liczbą porządkową 2, 3, 4, 5, ...lub 20. W ten sposób otrzymamy próbę: (1,2), (1,3), (1,4), (1,5), ... lub (1,20). Jeżeli do próby w pierwszym kroku dobierzemy osobę o liczbie porządkowej 2 to w drugim kroku do pary możemy dobrać osoby z liczbą porządkową 1, 3, 4, 5, ...lub 20. W ten sposób otrzymamy próbę: (2,1), (2,3), (2,4), (2,5), ... lub (2,20). Widzimy więc, że dla każdej z 20 osób w populacji możemy dobrać 19 różnych par, czyli łącznie możemy stworzyć w ten sposób $20 \times 19 = 380$ różnych dwuosobowych prób.

Rozpiszmy je, żeby zobaczyć jak wyglądają.

(1,2); (1,3); (1,4); (1,5); (1,6); (1,7); (1,8); (1,9); (1,10);
(1,11); (1,12); (1,13); (1,14); (1,15); (1,16); (1,17);
(1,18); (1,19); (1,20); (2,1); (2,3); (2,4); (2,5); (2,6);
(2,7); (2,8); (2,9); (2,10); (2,11); (2,12); (2,13); (2,14);
(2,15); (2,16); (2,17); (2,18); (2,19); (2,20); (3,1); (3,2);
(3,4); (3,5); (3,6); (3,7); (3,8); (3,9); (3,10); (3,11);
(3,12); (3,13); (3,14); (3,15); (3,16); (3,17); (3,18);
(3,19); (3,20); (4,1); (4,2); (4,3); (4,5); (4,6); (4,7); (4,8);
(4,9); (4,10); (4,11); (4,12); (4,13); (4,14); (4,15); (4,16);
(4,17); (4,18); (4,19); (4,20); (5,1); (5,2); (5,3); (5,4);
(5,6); (5,7); (5,8); (5,9); (5,10); (5,11); (5,12); (5,13);
(5,14); (5,15); (5,16); (5,17); (5,18); (5,19); (5,20); (6,1);
(6,2); (6,3); (6,4); (6,5); (6,7); (6,8); (6,9); (6,10); (6,11);

(6,12); (6,13); (6,14); (6,15); (6,16); (6,17); (6,18);
(6,19); (6,20); (7,1); (7,2); (7,3); (7,4); (7,5); (7,6); (7,8);
(7,9); (7,10); (7,11); (7,12); (7,13); (7,14); (7,15); (7,16);
(7,17); (7,18); (7,19); (7,20); (8,1); (8,2); (8,3); (8,4);
(8,5); (8,6); (8,7); (8,9); (8,10); (8,11); (8,12); (8,13);
(8,14); (8,15); (8,16); (8,17); (8,18); (8,19); (8,20); (9,1);
(9,2); (9,3); (9,4); (9,5); (9,6); (9,7); (9,8); (9,10); (9,11);
(9,12); (9,13); (9,14); (9,15); (9,16); (9,17); (9,18);
(9,19); (9,20); (10,1); (10,2); (10,3); (10,4); (10,5);
(10,6); (10,7); (10,8); (10,9); (10,11); (10,12); (10,13);
(10,14); (10,15); (10,16); (10,17); (10,18); (10,19);
(10,20); (11,1); (11,2); (11,3); (11,4); (11,5); (11,6);
(11,7); (11,8); (11,9); (11,10); (11,12); (11,13); (11,14);
(11,15); (11,16); (11,17); (11,18); (11,19); (11,20); (12,1);
(12,2); (12,3); (12,4); (12,5); (12,6); (12,7); (12,8);
(12,9); (12,10); (12,11); (12,13); (12,14); (12,15);
(12,16); (12,17); (12,18); (12,19); (12,20); (13,1); (13,2);
(13,3); (13,4); (13,5); (13,6); (13,7); (13,8); (13,9);
(13,10); (13,11); (13,12); (13,14); (13,15); (13,16);
(13,17); (13,18); (13,19); (13,20); (14,1); (14,2); (14,3);
(14,4); (14,5); (14,6); (14,7); (14,8); (14,9); (14,10);
(14,11); (14,12); (14,13); (14,15); (14,16); (14,17);
(14,18); (14,19); (14,20); (15,1); (15,2); (15,3); (15,4);
(15,5); (15,6); (15,7); (15,8); (15,9); (15,10); (15,11);
(15,12); (15,13); (15,14); (15,16); (15,17); (15,18);
(15,19); (15,20); (16,1); (16,2); (16,3); (16,4); (16,5);
(16,6); (16,7); (16,8); (16,9); (16,10); (16,11); (16,12);

(16,13); (16,14); (16,15); (16,17); (16,18); (16,19);
(16,20); (17,1); (17,2); (17,3); (17,4); (17,5); (17,6);
(17,7); (17,8); (17,9); (17,10); (17,11); (17,12); (17,13);
(17,14); (17,15); (17,16); (17,18); (17,19); (17,20);
(18,1); (18,2); (18,3); (18,4); (18,5); (18,6); (18,7);
(18,8); (18,9); (18,10); (18,11); (18,12); (18,13); (18,14);
(18,15); (18,16); (18,17); (18,19); (18,20); (19,1); (19,2);
(19,3); (19,4); (19,5); (19,6); (19,7); (19,8); (19,9);
(19,10); (19,11); (19,12); (19,13); (19,14); (19,15);
(19,16); (19,17); (19,18); (19,20); (20,1); (20,2); (20,3);
(20,4); (20,5); (20,6); (20,7); (20,8); (20,9); (20,10);
(20,11); (20,12); (20,13); (20,14); (20,15); (20,16);
(20,17); (20,18); (20,19);

Widzimy, że niektóre pary się powtarzają. Możemy bowiem wylosować najpierw osobę o numerze 1, a potem numerze 2. Otrzymujemy wtedy próbę (1,2). Ale może być też na odwrót. Najpierw wylosujemy osobę o numerze 2. a potem osobę o numerze 1. Otrzymujemy wtedy próbę (2,1). W taki razie nasuwa się pytanie, w ilu próbach występuje każdy obywatele.

Tabela 7.3. Liczba wystąpień obywateli w próbach dwuosobowych

L.p. Liczba wystąpień

2	38
3	38
4	38
5	38
6	38
7	38
8	38
9	38
10	38
11	38
12	38
13	38
14	38
15	38
16	38
17	38
18	38
19	38
20	38

Sprawdźmy teraz jakie wyniki generuje nasz schemat losowania respondentów. Policzmy dla wszystkich wymienionych wcześniej prób odsetek osób, które nie chcą iść na wybory. Ponieważ za każdym razem mamy

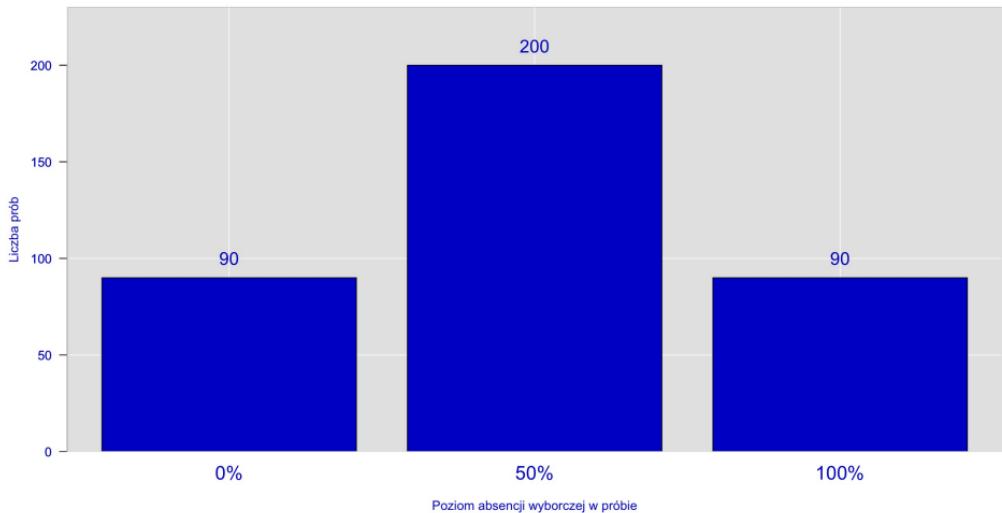
tylko dwóch respondentów więc możliwe są tylko trzy rodzaje wyników:

- (ABSENCA, ABSENCA) => (100%) - żadna z dwóch osób NIE ZAMIERZA pójść na wybory;
 - (ABSENCA, GŁOSOWANIE) lub (GŁOSOWANIE, ABSENCA) => (50%) - jedna z dwóch osób NIE ZAMIERZA pójść na wybory.
 - (GŁOSOWANIE, GŁOSOWANIE) => (0%) - obie osoby zamierzają pójść na wybory;

Skoro wiemy już czego możemy się spodziewać, to rozpiszmy poziomy absencji we wszystkich próbach (z zachowaniem wcześniejszej kolejności prób):

Na pierwszy rzut oka widać, że nie wszystkie wyniki są zgodne z tym, co obserwujemy w całej populacji. W wielu próbach szacowana ABSENCJA jest zaniżona (0%) lub zawyżona (100%). Ale to nas nie dziwi, bo wynika to z wybranego przez nas schematu losowania próby. Sprawdźmy ile dokładnie wyników każdego rodzaju

Wykres 7.1. Poziom absencji wyborczej w próbach dwuosobowych

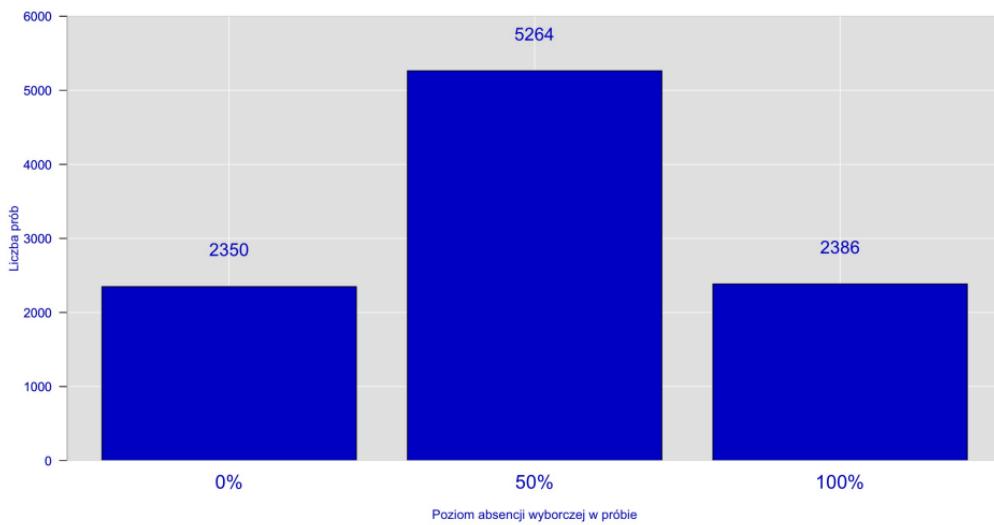


Na powyższym wykresie widzimy, że w 90 próbach absencja wynosi 0%, w 200 próba 50%, a w 90 próbach 100%. To całkiem logiczny rezultat. Jeżeli w populacji połowa obywateli będzie głosować w wyborach, a połowa nie, to możemy się spodziewać, że najczęściej jedna z dwóch osób w próbie będzie zwiększać absencję, a druga zmniejszać.

Wiemy teraz dokładnie czego możemy się spodziewać po naszym schemacie losowania próby - które wynika generuje najczęściej, a które najrzadziej. Ale jakie to ma dla nas znaczenie skoro badanie sondażowe polega na wylosowaniu tylko jednej próby? Żeby się przekonać założymy na chwilę, że jednak możemy nasz eksperyment powtarzać wielokrotnie. Wyobraźmy sobie, że dobieramy naszą dwuosobową próbę nie raz ale 10000 razy. Dla

każdej z nich będziemy obliczać poziom absencji wyborczej, a potem “zwracać” respondentów do populacji. W ten sposób otrzymamy 10000 niezależnych wyników naszego badania. Czy da się przewidzieć, jaki będzie ich rozkład? Okazuje się, że tak. Na 10000 przypadków w około 2368 ($90/380*10000$) absencja wyniesie 0%, podobnie w 5263 ($200/380*10000$) 50%, a w 2368 ($90/380*10000$) 100%. Czyli rozkład wyników powinny być podobne do tego na wykresie słupkowym powyżej, ilustrującym rozkład wyników wśród zbiorowości prób losowanych naszym schematem doboru. Zobaczmy więc teraz jak będą się kształtowały wyniki z naszej symulacji

Wykres 7.2. Symulacja - poziom absencji w 10 000 prób dwuosobowych



Jak widać wyniki symulacji nie odbiegają od tego, co przewidywaliśmy. W 2350 próbach absencja wyborcza wyniosła 0%, w 5264 wyniosła 50%, a w 2386 wyniosła 100%. Różnice są niewielkie. Widzimy więc teraz, że schemat losowania próby determinuje szansę uzyskania poprawnego oszacowania. W przypadku naszego badania wynoszą one 200/380, a więc około 53%.

Dla ponad połowy wszystkich możliwych prób oszacowanie poziomu absencji jest prawidłowe. Wciąż jednak istnieje spore ryzyko, że się pomylimy. Czy to znaczy, że nasz schemat losowania jest “zły”? Jak w ogóle sprawdzić, czy jest “dobry” i nasze wnioski z badania mogą być trafne? Spójrzmy na ten problem w następujący sposób: wiadomo, że poszczególne próby dobierane do badania sondażowego mogą dawać trochę inne wyniki.

Najlepiej gdyby oszacowania te, jeżeli nie trafiały w punkt, to przynajmniej oscylowały wokół prawidłowego wyniki. Schemat doboru próby powinien być tak zaprojektowany, aby wyniki z generowanych przy jego użyciu prób “ciążyły” w kierunku wartości obserwowanej w populacji. Co to znaczy? Zastanówmy się jaki wynik przeciętnie dają próby generowane przez nasz schemat losowania. Zastosujemy przy tym zasadę często stosowaną w szkole, gdy chcemy się czegoś dowiedzieć o wynikach dane ucznia w skali całego roku. W takim przypadku liczymy średnią arytmetyczną jego ocen. Podobnie zrobimy z naszymi próbami. Policzymy czego możemy się po nich przeciętnie spodziewać. W tym celu dodajemy do siebie wszystkie 380 możliwych oszacowań, a następnie dzielimy je przez liczbę wszystkich prób, czyli 380 - zupełnie jak średnią ocen w szkole. Możemy sobie uprościć to zadanie i zsumować wyniki w następujący sposób: $0\% * 90 + 100\% * 90 + 50\% * 200 = 19000\%$. Chwilowo rezultat jest absurdalny, ale to minie, gdy podzielimy go przez łączną liczbę prób: $19000\% / (90 + 90 + 200) = 19000\% / 380 = 50\% !!!$ I tu docieramy do sedna sprawy. Oto okazuje się, że “przeciętnie” na próbę przypada 50% absencja wyborcza!!! To jest dokładnie tyle, ile wynosi ono w całej populacji!!! Odkryliśmy w ten sposób podstawową “prawo” dotyczące metody reprezentacyjnej - przeciętny wynik z próby powinien być równy wynikowi dla całej populacji. Jest to najważniejsza

zasada badań sondażowych, ale także ogólnie wszystkich badań prowadzonych na próbach. Dzięki temu wiemy, że przeciętnie trafiamy w punkt, czyli próba nie jest “obciążona” i wyniki z próby nie “odbiegają” od rzeczywistej wartości w populacji. Osiągnęliśmy to dzięki nadaniu wszystkim obywatelom równych szans dostania się do próby lub inaczej równe prawo do swobodnego wyrażenia swojej opinii.

Tabela 7.4. Przeciętny poziom absencji w próbie - podsumowanie obliczeń

Absencja w próbie (A)	Liczba prób z daną absencją (B)	Iloczyn absencji i liczby prób z daną absencją (A*B)
0%	90	0%*90=0%
50%	200	50%*200=10000%
100%	90	100%*90=9000%
Razem	380	19000%

Oczywiście to, że nasz schemat doboru respondentów generuje próby, które przeciętnie się nie mylą, nie rozwiązuje problemu błędów, czyli przeszacowań i niedoszacowań. Zauważmy, że w przypadku losowania prób dwuosobowych i badania problemu absencji wyborczej prawie co druga próba daje wynik poważnie

“obciążone”, czyli odbiegające od rzeczywistych. Dlatego w następnej części naszego odcinka przyjrzymy się sposobom wnioskowania na podstawie próby i sposobom radzenia sobie z błędami oszacowań generowanymi dla danego schematu losowania próby.

Zadania 7:

1. Odpowiedz na pytania związane ze schematem losowania **4** respondentów w sposób prosty bezzwrotny z omawianej populacji 20 obywateli (załączonej do rozdziału):
 - a. Jakie są możliwe poziomy absencji wyborczej w próbie:
.....
.....
.....
.....
 - b. Ile różnych prób można wylosować na podstawie podanego schematu losowania:
.....
 - c. Napisz w ilu różnych próbach może się pojawić każdy obywatel:
.....
 - d. Jaki będzie przeciętny poziom absencji w próbie wylosowanej wg podanego schematu:
.....

-
- e. czy próba otrzymana z podanego schematu jest obciążona:
 - Tak
 - Nie
 2. Wylosuj w sposób prosty bez zwracania próbę 15 osób z populacji załączonej do rozdziału, a następnie:
 - a. Zapisz liczby porządkowe (L.p.) osób wybranych do próby;
.....
 - b. Policz poziom absencji wyborczej w otrzymanej wylosowanej próbie;
.....
 - c. Napisz jaki jest przeciętnych poziom absencji w próbach losowanych wg zastosowanego schematu;
.....
 - d. Napisz o ile punktów procentowych różni się poziom absencji wyborczej w twojej próbie od poziomu absencji w całej populacji (50%);
.....

8. Błąd oszacowania

Omówiliśmy już wcześniej dobór losowy respondentów do badania sondażowego i wiemy, że wyniki z prób

przeciętnie powinny “trafiąć w punkt”. Pozostaje jednak problem błędów. Jak zauważylismy na przykładzie sondażu dotyczącego absencji wyborczej, duża część prób może dawać nieprawidłowe oszacowania. Cóż z tego więc, że nasz schemat losowania średnio rzecz biorąc daje dobre wyniki skoro my dobierzemy jedną próbę i ona właśnie chybi?!

Przeciętnie dobra celność to za mało. Schemat losowania powinien dodatkowo gwarantować, że dla przeważającej większości prób oszacowania badanego parametru populacji będą bardzo bliskie rzeczywistym wartościom. Innymi słowy ryzyko popełnienia dużego błędu powinno być jak najmniejsze. Próby uzyskane przy pomocy schematu spełniającego powyższe warunki nazwiemy reprezentatywnymi (?).

Zagadnienie błędu oszacowań uzyskiwanych z prób losowych omówimy na nowym przykładzie badania sondażowego. Będzie ono dotyczyło poziomu poparcia dla partii X.

W stosunku do badania absencji wyborczej zmienimy niewiele. Po prostu w kolumnie “Preferencje wyborcze” zamiast odpowiedzi “Nie idę na wybory” teraz będziemy analizowali występowanie wartości “Głosuję na partię X”. Zwiększymy również liczbę respondentów - z 2 do 6. Tak samo jak wcześniej zastosujemy jednak losowanie proste bez zwracania i technikę CAPI. Pytanie w kwestionariuszu będzie natomiast brzmiało: **“Gdyby**

wybory do sejmu odbywały się w najbliższą niedzielę to czy zagłosowałby/ałaby Pan/i na partię X?”. Odpowiedzi “nie chodzę na wybory” oraz “zagłosuję na inną partię” liczymy razem, jako brak poparcia dla X.

Tabela 8.1. Rozkład poparcia dla partii X w (fikcyjnej) populacji dorosłych obywateli Polski

Miejsce zamieszkania	Nie biorę udziału w wyborach	NIE	Tak	Razem
Miasto	6	1	5	12
Wies	4	2	2	8
Razem	10	3	7	20

Zacznijmy od sprawdzenia, jakie poparcie dla partii X mogą generować próby dobrane przy użyciu naszego nowego schematu losowania. Ponieważ będziemy dobierać 6 respondentów więc możliwych jest 7 wyników:

1. (X, X, X, X, X, X) => 100% (6/6)
2. (X, X, X, X, X, nX) => 83% (5/6)
3. (X, X, X, X, nX, nX) => 67% (4/6)
4. (X, X, X, nX, nX, nX) => 50% (3/5)
5. (X, X, nX, nX, nX, nX) => 33% (2/6)
6. (X, nX, xX, nX, nX, nX) => 17% (1/6)

7. (nX, nX, nX, nX, nX, nX) => 0% (0/6)

gdzie X oznacza “tak, oddałabym/łbym głos na partię X”, a nX oznacza “nie, oddałabym/łbym głos na inną partię lub w ogóle nie posza/edł na wybory”.

Zwróćmy uwagę, że nasze oszacowania dotyczą poparcia wśród wszystkich obywateli, a nie tylko wyborców. Takie rozwiązanie zastosujemy dla uproszczenia naszych rozważań. Zazwyczaj jednak podstawę procentowania w polskich sondażach politycznych stanowi zbiorowość wyborów, więc tylko osób deklarujących udział w wyborach. My nie zastosujemy się jednak do tej reguły ponieważ utrudniłaby ona nam prowadzenie niektórych analiz.

Wróćmy do naszego nowego schematu losowania respondentów. Ogólnie rzecz biorąc generuje on aż $20*19*18*17*16*15=27907200$ różnych prób. To jest znacznie, znacznie więcej niż w przypadku badania absencji wyborczej, gdzie dobieraliśmy 2 respondentów. Niestety nie mamy miejsca, żeby rozpisać wszystkie próby. Możemy natomiast opisać ich zbiorowość wykonując pewne obliczenia. Zaczniemy od tego, w ilu próbach pojawi się każdy obywatel. W przypadku schematu doboru dwóch respondentów każdy obywatel mógł utworzyć 19 par z innymi osobami z populacji i dodatkowo zająć pierwsze lub drugie miejsce w próbie.

W konsekwencji występował w $19*2=38$ próbach. A co by się stało, gdybyśmy dobierali trzyosobowe próby? W tym przypadku dla każdego obywatela można dobrać najpierw jedną osobę spośród 19, a później drugą spośród pozostałych 18. Obywatel mógłby przy tym zajmować pierwsze, drugie lub trzecie miejsce w próbie. W efekcie każdy występowałby w $19*18*3=1026$ trzyosobowych próbach. Analogicznie w przypadku sześciuosobowej próby najpierw do każdego obywatela można dobrać jedną z 19 osób, potem jedną z 18 pozostałych osób, potem jeszcze jedną z pozostałych 17 osób itd., aż wreszcie ostatnią z pozostałych 15. Dodatkowo nasz obywatel może zająć 1,2,3,4,5 lub 6 miejsce w próbie. Ostatecznie więc występuje w $19*18*17*16*15*6=8372160$ sześciuosobowych próbach.

Tabela 8.2. Liczba wystąpień obywateli w próbach sześciuosobowych

Obywatel Liczba wystąpień

1	8372160
2	8372160
3	8372160
4	8372160
5	8372160
6	8372160

7	8372160
8	8372160
9	8372160
10	8372160
11	8372160
12	8372160
13	8372160
14	8372160
15	8372160
16	8372160
17	8372160
18	8372160
19	8372160
20	8372160

Wiemy już w ilu próbach wystąpi każdy obywatel. Teraz zastanówmy się, z jaką częstotliwością występują poszczególne wyniki, czyli ile jest takich prób, w których poparcie dla partii X wyniesie 100% (6/6), ile takich, w których poparcie dla partii X wyniesie 83% (5/6) itd. Odpowiedź na to pytanie można uzyskać na dwa sposoby. Albo stosując proste, ale wymagające dużego skupienia, obliczenia na papierze albo stosując mniej obciążające i szybsze obliczenia na komputerze. Zaczniemy od obliczeń na papierze. Zastanówmy się, ile

może być takich prób, w których poparcie dla partii X wyniesie 83,3% (5/6). Przykładowo preferencje w nich mogą się ułożyć w następującej kolejności (X, X, X, X, X, nX). Losowanie tego rodzaju próby przeprowadzimy w następujący sposób. Pierwszego respondenta X dobieramy spośród wszystkich osób popierających partię X w populacji - łącznie jest ich 7. Potem drugiego respondenta dobieramy spośród pozostałych 6 osób popierających X, potem trzeciego respondenta dobieramy spośród pozostałych 5 osób popierających X, itd aż dochodzimy do szóstego respondenta. Dobierzemy go z innej grupy, osób nie popierających X (nX). W populacji jest ich 13. Ostatecznie więc liczba prób, dla których preferencje układają się w kolejności (X, X, X, X, X, nX), jest równa $7*6*5*4*3*13=$. No tak, ale my chcemy wiedzieć, ile jest dokładnie wszystkich prób, w których poparcie wynosi 83,3%, nie tylko takich, w których osoba nX została wylosowana jako ostatnia. Żeby się tego dowiedzieć musimy zauważyc, że obywatel deklarujący nX może zostać wylosowany, jako szósty, piąty, czwarty, ... lub jako pierwszy. Czyli może on zostać ustawiony na sześć sposobów w próbie. Innymi słowy każda próba typu (X, X, X, X, X, nX) ma swoich pięć odpowiedników, w których występują ci sami obywatele X w tej samej kolejności i tylko obywatel nX "przeskakuje" między nimi. Ostatecznie więc liczba prób dających 83,3% (5/6) poparcia dla partii X wyniesie

$$(7*6*5*4*3*13)*6=32760*6=.$$

A ile będzie prób, w których poparcie dla partii X wynosi 66,7% (4/6)? Podobnie jak poprzednio, zacznijmy od przykładowej kombinacji odpowiedzi (X, X, X, X , nX, nX). Dokładnie w tej kolejności preferencje możemy mieć w $7*6*5*4*13*12=131040$ próbach. Teraz zastanówmy się, na ile sposobów możemy “ustawić” dwie preferencje nX na sześciu miejscach (alternatywnie można się zastanawiać, na ile sposobów ustawić 4 preferencje X, ale wynik będzie dokładnie taki sam).

Założymy, że preferencje nX otrzymaliśmy od obywatela 7 i 8 - dokładnie w tej kolejności. Mamy więc próbę (X,X,X,X,7,8). Na ile sposobów obywatele 7 i 8 mogą “przeskoczyć” innych respondentów, jednocześnie pozostając w takim samym układzie względem siebie? Jeżeli 7 zajmie pierwsze miejsce to 8 może zająć drugie, trzecie, czwarte, ..., szóste. Czyli jeżeli 7 zajmie pierwsze miejsce to 8 może zająć 5 pozostałych. Idąc dalej tym tokiem myślenia możemy napisać, że:

- jeżeli 7 zajmuje pierwsze miejsce to 8 może zająć 5 pozostałych;
- jeżeli 7 zajmuje drugie miejsce to 8 może zająć 4 pozostałych;
- jeżeli 7 zajmuje trzecie miejsce to 8 może zająć 3 pozostałych;
- jeżeli 7 zajmuje czwarte miejsce to 8 może zająć 2 pozostałych;

- jeżeli 7 zajmuje piąte miejsce to 8 może zająć 1 pozostałe;

Ostatecznie otrzymujemy więc $5+4+3+2+1=15$ różnych wersji początkowego układu preferencji (X,X,X,X,7,8) Pamiętajmy, że 7,8 to tylko jedna z $13*12=156$ par dających dwie preferencje nX w próbie (inna może być para 8,7). Dodatkowo na każdą z nich przypada $7*6*5*4=840$ różnych układów respondentów tworzących preferencje (X,X,X,X). Ostatecznie więc prób, w których poparcie dla partii X wyniesie 66,7% (4/7) jest $(7*6*5*4)*(13*12)*15=1965600$

Na powyższy problem można też popatrzeć inaczej. Wystarczy zauważyć, że dla dwóch preferencji X mamy do przydzielenia łącznie 6 miejsc. Pierwsza z preferencji może zająć jedno miejsce z 6, a druga jedno z 5 pozostałych. Łącznie możemy więc wyróżnić $6*5=30$ różnych ich ustawień. Ale uwaga! Pamiętajmy, że kolejność respondentów musi pozostać niezmieniona. Jeżeli najpierw dobieramy obywatela 7, a potem 8 to znaczy, że akceptujemy próbę (7,...,...,...,8) a odrzucamy (8,...,...,...,7). Pozbywamy się więc wszystkich "duplikatów". W ten sposób otrzymujemy $(6*5)/2=15$ różnych ustawień dla pary obywateli 7 i 8, przy zachowaniu ich kolejności. To jest dokładnie tyle ile otrzymaliśmy dla wcześniejszych obliczeń.
A co z próbami, w których poparcie dla partii X wynosi 50% (3/6). Na podobnej zasadzie jak wcześniej możemy

policzyć, że dla prób, w których preferencje występują w kolejności (X, X, X, nX, nX) jest $(7*6*5)*(13*12*11)=360360$. Następnie liczymy na ile sposobów możemy "wymieszać" każdą taką próbę, nie burząc jednocześnie porządku wśród respondentów o preferencjach X oraz nX oddzielnie. Pierwsza z preferencji nX może zająć jedno z 6 miejsc, druga jedno z pozostałych 5, trzecia jedno z pozostałych 4 - łącznie $6*5*4=120$. Musimy już tylko pozbyć się "duplikatów". Trzy preferencje nX można potencjalnie ustawić na $3*2*1=6$ sposobów. Nas interesuje tylko jedna z nich, bo nie chcemy zmieniać kolejności preferencji. W związku z tym respondentów nX można "ustawić" nie na $6*5*4=120$ tylko na $(6*5*4)/6=20$ sposobów. Szacowana poparcie dla partii X na poziomie 50% (3/6) wystąpi więc w $(7*6*5)*(13*12*11)*(6*5*4)/6=7207200$ próbach.

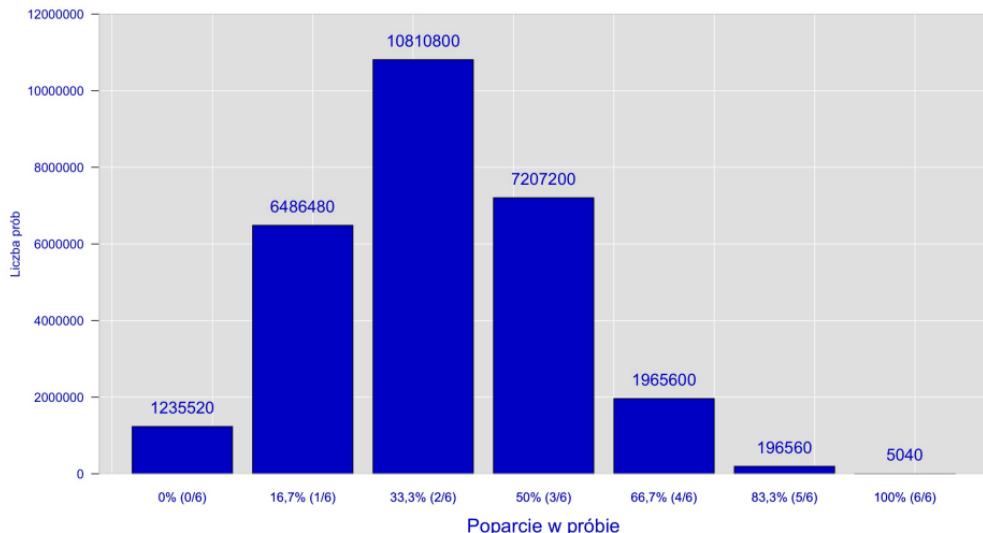
Obliczenia dla pozostałych wyników otrzymywanych przy użyciu schematu losowania 6 respondentów zostały umieszczone w poniższej tabeli oraz na wykresie.

Tabela 8.3. Poparcie dla partii X w próbach sześciuosobowych - obliczenia

Poparcie w próbie	Liczba prób - obliczenia
1	$(7*6*5*4*3*2)=5040$

5/6	$(7*6*5*4*3)*(13)*(6)=196560$
4/6	$(7*6*5*4)*(13*12)*(6*5)/2=1965600$
3/6	$(7*6*5)*(13*12*11)*$ $(6*5*4)/6=7207200$
2/6	$(7*6)*(13*12*11*10)*$ $(6*5)/2=10810800$
1/6	$(7)*(13*12*11*10*9)*(6)=6486480$
0	$(13*12*11*10*9*8)=1235520$
Razem	$20*19*18*17*16*15=27907200$

Wykres 8.1. Poparcie dla partii X w próbach sześciuosobowych



Uporaliśmy się z rozkładem wyników z próby 6-

osobowej. Wymagało to od nas trochę wysiłku, ale przynajmniej wiemy już, że nasz schemat losowania nie jest nieprzewidywalny. Widzimy teraz jak często generuje on poszczególne wyniki. Przyjrzyjmy się im dokładnie. Najwięcej prób zawiera poparcie dla partii X na poziomie 2/6. Dużo mniej 1/6 i 3/6. Wyniki 0 oraz 4/6 można nazwać rzadkimi, a 5/6 i 1 bardzo rzadkimi. Zauważmy również, że poparcia dla partii X nigdy nie jest równe 35% (7/20). Czy to oznacza, że nasz schemat jest wadliwy? Jak wiemy “celność” próby określa się na podstawie przeciętnej wartości oszacowania uzyskiwanego z w zbiorowości wszystkich prób. Gdy mierzyliśmy ten parametr dla sondażu dotyczącego absencji w wyborach, wiedzieliśmy, że przynajmniej część prób “trafia w punkt”. Tym razem tak nie jest. Żadna próba nie daje wyniku 35% (7/20) poparcia dla partii X. Jeżeli jednak sprawdzimy przeciętny wynik z próby to okaże się, że wynosi on dokładnie tyle ile w populacji. Zgodnie z tym, co mówiliśmy wcześniej, oznacza to, że nasz schemat losowania 6-osobowych prób jest poprawny. Ale co z tego skoro nie potrafi dokładnie “trafić” w wynik! Najwyraźniej potrzebujemy dodatkowej miary “jakości” schematu. Musi ona określać skalę błędu popełnianego przy korzystaniu z konkretnych schematów. Dzięki temu będziemy mogli porównywać różne schematy, a przede wszystkim określić, jak duże jest ryzyko, że się mylimy korzystając z jednej próby. Nasza miarą błędu

będzie wielkość wymyślonym przez statystyków. Dla każdej próby policzmy, jak duży jest rozstęp między wynikiem z próby, a prawdziwym wynikiem w populacji (35%).

Tabela 8.4. Odchylenie standardowe - obliczenia

Poparcie prób z dla partii X (A)	Liczba poparcia dla partii X (B)	Różnica między wynikiem w próbce, a poparciem w populacji (A -35%)	Kwadrat różnicy $(A - 35\%)^2$	Iloczyn kwadratu różnicy błędu i liczby prób $((A - 35\%)^2) * B$
1	5040	0,6500	0,4225	2129,4
5/6	196560	0,4833	0,2336	45918,6
4/6	1965600	0,3167	0,1003	197106
3/6	7207200	0,1500	0,0225	162162
2/6	10810800	-0,0167	0,0003	3003
1/6	6486480	-0,1833	0,0336	218017,8
0	1235520	-0,3500	0,1225	151351,2
Razem	27907200	-	-	779688

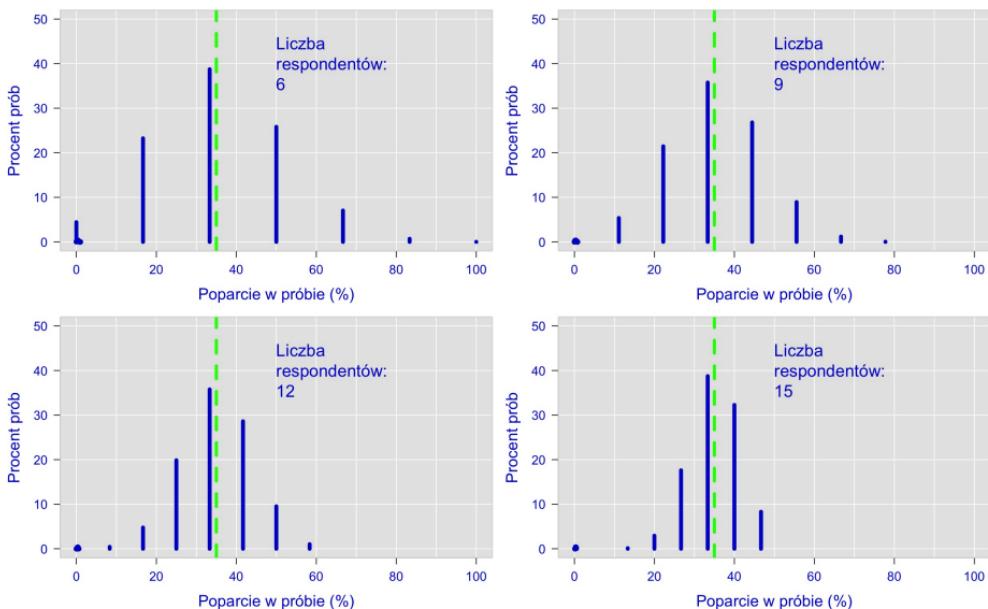
Zaczynamy od rozpisania wszystkich możliwych wyników

z próby (kolumna A). Następnie dopisujemy, ile prób w naszym schemacie daje konkretny wynik (kolumna B). Obliczenia błędy zaczniemy od wyznaczenia różnicy między wynikiem w próbie i wynikiem w populacji (kolumna A - 35%). Następnie wyznaczymy kwadrat tego błędu $((A-35\%)^2)$. W ten sposób otrzymaliśmy kwadrat błędu popełnianego w danym rodzaju próby. Nas interesuje nie błąd dla konkretnej próby (choć to też jest interesujące), ale błąd przeciętnie popełniany dla danego schematu losowania prób. Czyli chcemy po prostu policzyć średnią z naszego błędu - średnią arytmetyczną. Potrzebujemy do tego iloczynu kwadratów błędów w poszczególnych rodzajach prób i liczby prób każdego rodzaju $((A-35\%)^2 * B)$. Następnie sumujemy otrzymane wartości (779688) i dzielimy przez ogólną liczbę prób. W efekcie otrzymujemy $779688 / 27907200 = 0,02793859649$. To jest wartość przeciętnego błędu kwadratowego (statystycy nazywają go wariancją) popełnianego przy wnioskowaniu o poparcia dla partii X na podstawie prób generowanych przez nasz schemat losowania. Czy to dużo? Trudno powiedzieć. Zobaczmy, co się stanie, gdy policzymy pierwiastek naszego błędu kwadratowego. W ten sposób otrzymamy jak gdyby przeciętny (nie kwadratowy) błąd jaki popełniamy stosując nasz schemat losowania próby. Wynosi on $0,02793859649^{(1/2)} = 0,1671484265$. W ten sposób otrzymujemy miarę, którą nazywamy **odchyleniem**

standardowym. Trzeba podkreślić, że nie jest to przecienny błąd popełniany przez próby tylko pierwiastek przeciennego kwadratu błędu. Dla uproszczenia traktuje się go jednak jako zwykły przecienny błąd. Wiemy więc, że po naszej próbie możemy się spodziewać odchylenia standardowego na poziomie prawie 17% (0,167). To bardzo dużo biorąc pod uwagę, że rzeczywiste poparcie w próbie wynosi 35%!!!

Czy istnieje jakiś sposób żeby temu zaradzić? Na szczęście tak. Polega on na zwiększeniu liczebności próby. Ale czemu większa liczba respondentów pozwala zredukować błąd? W skrócie można powiedzieć, że zmniejsza ona ryzyko otrzymania skrajnych wyników w próbie. Jeżeli dobieramy tylko dwóch respondentów (tak jak w sondażu dotyczącym absencji wyborczej) to poparcie dla partii X możemy oszacować tylko na poziomie 0%, 50% oraz 100%. Wprowadzając trzeciego respondenta zwiększamy zakres możliwości do 0%, 30%, 60%, 90% i 100%. Widzimy więc, że zwiększając próbę zwiększamy różnorodność wyników i jednocześnie zmniejszamy ryzyko trafienia na wynik skrajny, najdalszy od 35%. Mniej skrajnych wyników oznacza mniejszy błąd. Możemy to zobaczyć na wykresie ilustrującym rokład wyników z próby dla schematów losowania od 6, 9, 12 oraz 15 respondentów w sposób prosty z naszej (fikcyjnej) populacji.

Wykres 8.2. Poparcie dla partii X w próbach 6, 9, 12 oraz 15-osobowych



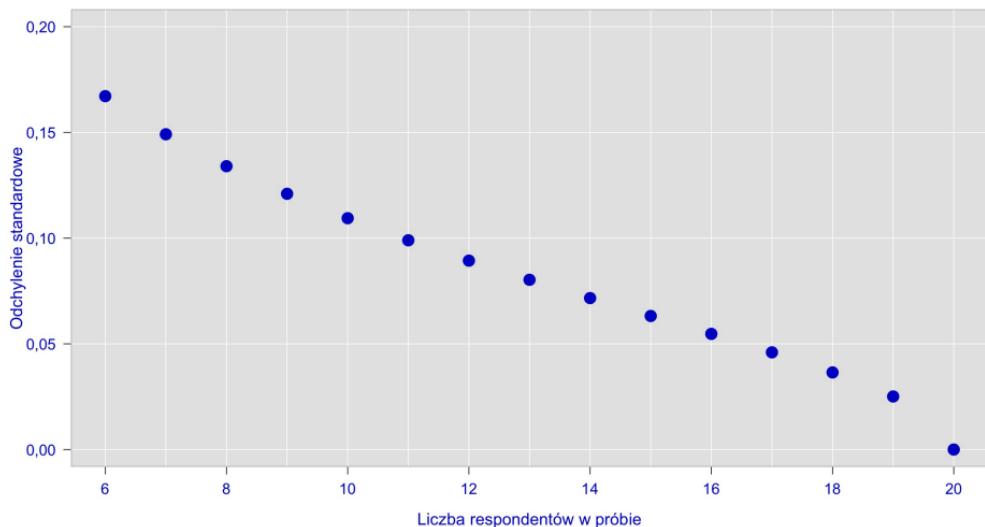
Na osi poziomej znajduje się poziom poparcia dla partii X w próbach wygenerowanych przy użyciu danego schematu losowania. Na osi pionowej mamy oznaczony odsetek prób, w których poparcie dla partii X osiągnęło konkretny poziom. Przerywaną linią zieloną zaznaczono rzeczywisty poziom poparcia dla partii X w populacji (35%). Widzimy więc, że dla naszego schematu losowania 6 respondentów:

- w 4% wszystkich prób poparcie dla X wynosi 0 (0/6);

- w 23% wszystkich prób poparcie dla X wynosi 17% (1/6);
- w 38% wszystkich prób poparcie dla X wynosi 33% (2/6);
- w 26% wszystkich prób poparcie dla X wynosi 50% (3/6);
- w 7% wszystkich prób poparcie dla X wynosi 67% (4/6);
- w 0,7% wszystkich prób poparcie dla X wynosi 83% (5/6);
- w 0,02% wszystkich prób poparcie dla X wynosi 100% (6/6);

Wiemy już teraz, że zwiększanie próby “przybliża” wyniki do prawidłowej wartości. Im więcej respondentów, tym większy odsetek prób daje oszacowania bliskie rzeczywistemu poparciu dla partii X. Jeżeli policzmy teraz błąd (odchylenie standardowe) poparcia dla partii X dla tych czterech schematów losowania również zauważymy, że maleje on wraz ze zwiększeniem liczby respondentów. Poniżej znajduje się wykres, który ilustruje jak wielkość błędu (odchylenia standardowego) zmniejsza się dla schematów losowania od 6 do 20 respondentów. Ten ostatni przypadek dotyczy sytuacji, w której dobierani są wszyscy obywatele z populacji i w związku z tym błąd oszacowania wynosi 0.

Wykres 8.3. Odchylenie standardowe poparcia dla partii X dla schematów losowania od 6 do 20 respondentów



Zwiększanie próby rzeczywiście działa! Na wykresie widzimy wyraźnie, jak błąd standardowy zmniejsza się wraz z przyrostem liczby respondentów. W ten sposób odkryliśmy sposób radzenia sobie z dużymi błędami generowanymi przez dobór losowy - duże próby. Dzięki nim nasze wyniki będą nie tylko przeciętnie zgodne z rzeczywistością, ale przede wszystkim będą przeciętnie obarczone niewielkim błędem.

Pozostaje więc już tylko jedna kwestia do wyjaśnienia: błąd statystyczny. Wyniki sondaży często są opatrzone

komentarzem: “błąd statystyczny wynosi +/-3%”. Co to oznacza? Żeby to zrozumieć musimy zacząć od małej dygresji dotyczącej sposobu postrzegania świata przez statystyków oraz sondażystów. Musimy wiedzieć, że przypisują oni bardzo duże znaczenie liczbie 95%. Często zakładają, że jeśli prawdopodobieństwo jakiegoś zdarzenia wynosi 95% to właściwie jest ono pewne. Skorzystajmy z tej zasady. Zauważmy, że spośród wszystkich sześciuosobowych prób w 95,6% z nich poparcie dla partii X jest nie większe niż 83%(5/6) i nie mniejsze niż 17%(1/5).

Tabela 8.5. Skumulowany odsetek prób sześciuosobowych

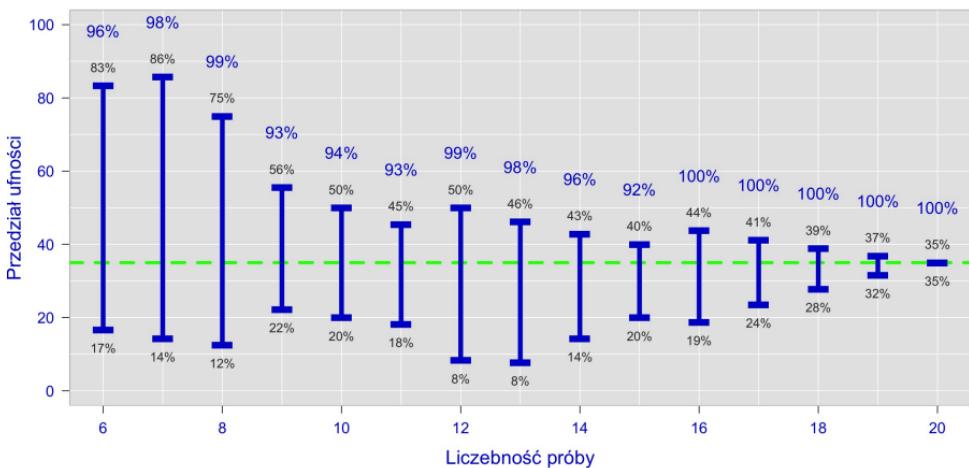
Poparcie w próbie	Procent prób	Skumulowany odsetek prób (przedział ufności)
100% (6/6)	0,02%	0,02%
83% (5/6)	0,70%	
66% (4/6)	7,04%	
50% (3/6)	25,83%	95,6%
33% (2/6)	38,74%	
17% (1/6)	23,24%	
0% (0/6)	4,43%	4,43%
Razem	100%	100%

Innymi słowy poparcie dla partii X w próbie dobieranej przy użyciu naszego schematu na 95,6% będzie nie mniejsze niż $35\%-18\% = 17\%$ i nie większe niż $35\% + 48\% = 83\%$. Statystycy oraz sondażyści powiedzieliby, że oszacowanie z próby „na pewno” będzie równe 35% (-18%, +48%). Odchylenia od rzeczywistego wyniku, umieszczone w nawiasie, nazywamy właśnie błędem statystycznym. Niestety nie jest on ani mały ani symetryczny, tak jak to bywa zazwyczaj w dużych badaniach sondażowych, gdy wynosi (-3%, +3%). Dzieje się tak ponieważ rozkład wyników z prób dla naszego schematu losowania jest asymetryczny względem rzeczywistego wyniku w populacji (czyli 35% - zielona przerywana linia na **Wykresie 8.2.**) i dodatkowo obarczony dużym odchyleniem standardowym. Przyczyną takiego stanu rzeczy jest oczywiście niewielka liczebność próby.

Skoro wiemy już czym jest błąd statystyczny to warto żebyśmy poznali również pojęcie przedziału ufności. W ten sposób określa się zakres 95% wartości otrzymywanych przy pomocy danego schematu losowania, które mieszczą się w granicach błędu statystycznego. Innymi słowy są to wyniki, które padną „na pewno”. Dla naszego schematu doboru sześciu respondentów przedział ufności ciągnie się od poparcia na poziomie 17% do poparcia na poziomie 83%, a więc jest okropnie szeroki. A co by się stało gdybyśmy zwiększyli liczbę

respondentów w próbie?

Wykres 8.4. Zmiana szerokości przedziału ufności w zależności od liczby respondentów w próbie (od 6 do 20)



Powyższy wykres ilustruje wpływ wielkości próby na szerokość przedziału ufności. Na osi poziomej zaznaczono liczbę respondentów. Oś pionowa to poziom poparcia dla partii X w próbach generowanych przez kolejne schematy losowania. Przedziały ufności zaznaczono pionowymi czarnymi liniami. Ich granice są zaznaczone poziomymi kreskami poniżej i powyżej których podano skrajne wartości przedziałów. Nad każdym z nich dodatkowo opisano jaki procent prób obejmują. Zielona przerywana

linia wyznacza rzeczywisty poziom poparcia dla partii X w populacji. Przykładowo przedział ufności dla pierwszego schematu losowania sześciu respondentów ciągnie się od 17% do 83% poparcia dla partii X i obejmuje 95,6% prób. Przedział ufności dla schematu losowania siedmiu respondentów ciągnie się od 14% do 86% poparcia dla partii X i obejmuje 98% prób itd. Od razu widać, że żaden przedział ufności nie obejmuje równo 95% prób. To dlatego, że wyników otrzymywanych dla poszczególnych schematów nie dało się skumulować dokładnie do takiego odsetka. Stąd pewne niewielkie odchylenia. Nie mają one jednak większego znaczenia, bo nawet najmniejsza wartość 91,6% oznacza, że prawie wszystkie wyniki z prób mieszają się w wyznaczonym zakresie.

Najważniejsze jest dla nas jednak to, że przedziały ufności mają tendencję do zwężania się wraz ze zwiększeniem liczby respondentów. Możemy obserwować pewne fluktuacje, ale ogólny trend jest jednoznaczny. Do tego stopnia, że dla schematów losowania 16 i więcej respondentów generowane przez nie poziomy poparcia dla partii X są już tak bliskie rzeczywistej wartości, że przedziały ufności obejmują wszystkie wyniki. W ostatnim przypadku, losowania 20 respondentów, przedział ufności ma zerową szerokość ponieważ w istocie jest to badanie na całej populacji, a więc nie jest one obarczone żadną niepewnością.

Dowiedzieliśmy się już czym jest odchylenie standardowe, błąd statystyczny, przedział ufności, co zrobić żeby zwiększyć precyzję pomiaru. Nie wyjaśniliśmy tylko jednej kwestii. W jaki sposób obliczyć te wartości na podstawie jednej próby? Zauważmy, że do tej pory wszystkie wskaźniki obliczaliśmy znając dokładnie populację oraz generowane z niej próby. W prawdziwym badaniu nie mamy tego typu danych. Nie znamy rzeczywistej wartości poszukiwanego wskaźnika w populacji, ani wyników ze wszystkich prób generowanych przez dany schemat losowania.

Niestety nie mamy tyle miejsca i czasu, żeby wyjaśnić, jak na podstawie jednej próby określa się wielkość błędu statystycznego dla danego schematu losowania. Warto jednak, żebyśmy wiedzieli, że to przede wszystkim zasługa procedur statystycznych. Z pomocą kilku twierdzeń można wnioskować nie tylko o populacji, ale również o rozkładzie wyników w próbach generowanych przez dany schemat losowania. Warto jeszcze na koniec dodać, że aparat statystyczny oraz wnioskowanie statystyczne najlepiej działają na dużych próbach. Dlatego większa próba przekłada się na większą dokładność wyników oraz większy zakres analiz, które można przeprowadzić.

W ten sposób zamknęliśmy rozważania dotyczące błędów oszacowania. Najważniejsze wnioski, które powinniśmy zapamiętać są następujące:

1. Przeciętny wynik z próby to nie wszystko - liczy się

także przecienny błąd;

2. Przecienny błąd można zmniejszyć zwiększając liczebność próby;

3. Określenie “błąd statystyczny” dotyczy granic przedziału, który zdaniem statystyków “na pewno” (czyli na 95%) obejmuje swym zasięgiem poszukiwany parametr.

Zadania 8:

1. Napisz, jak rozumiesz określenie “reprezentatywność próby”:

.....
.....
.....

2. Ile wynosi przecienny poziom poparcia dla partii X w próbach składających się z 15 osób losowanych z naszej populacji w sposób prosty bez zwracania (ten sam schemat losowania co w *Zadaniach 7.*):

.....

3. Dla wylosowanej wcześniej próby (*Zadania 7.*) oblicz błąd standardowy oszacowania poziomu poparcia dla partii X:

.....
.....
.....

4. Czy szacowany poziom poparcia dla partii X w

twojej próbie jest nie mniejszy niż 20% i nie większe niż 40%?

- Tak
- Nie

5. Jak rozumiesz stwierdzenie: “poparcie dla partii X wynosi 35% (-15%, +5%)”?

.....
.....
.....

9. Błędy systematyczne

Do tej pory zajmowaliśmy się głównie teorią badań ankietowych. Czas przejść do praktyki, czyli realizacji wywiadów w “terenie”. Zbieranie danych od respondentów, bo oni mowa, jest najważniejszą i zarazem najbardziej kosztowną częścią każdego badania sondażowego.

Wcześniej (5. *Problem badawczy -> populacja -> technika realizacji badania*) mówiliśmy o tym, że nie zawsze udaje się nawiązać kontakt ze wszystkimi osobami dobranymi do próby. Nawet gdy to się uda to wciąż nie ma gwarancji, że wywiady rzeczywiście zostaną zrealizowane. Niestety nieuchwytność oraz odmowy respondentów towarzyszą wszystkim technikom zbierania danych. Oczywiście czasem zjawisko to jest mniejsze (badania CAPI), a czasem większe (badania CAWI), ale

zawsze występuje. Dlatego też wymyślono wskaźnik do jego mierzenia , czyli **poziomem realizacji próby**. Oblicza się go wyznaczając stosunek liczby respondentów, którzy wzięli udział w badaniu, do ogólnej liczby wszystkich wylosowanych osób. Niższy poziom realizacji próby zwiększa ryzyko, że nasze wnioski o populacji będą obarczone dużym błędem. Dzieje się tak nie tylko dlatego, że niedostępność respondentów zmniejsza liczbę osób w próbie, ale także dlatego, że przyczynia się ona do pojawienia się **błędów systematycznych i obciążenia wyników z próby**. Co to oznacza? Jak zwykle wyjaśnimy to na przykładzie.

Tabela 9.1. Rozkład preferencji wyborczych w (fikcyjnej) populacji dorosłych obywateli Polski

Popieram partię X	Nie popieram partii X	Nie biorę udziały w wyborach	Razem
Miasto 5	1	6	12
Wies 2	2	4	8
Razem 7	3	10	20

Ponownie skorzystamy z naszej (fikcyjnej) populacji dorosłych obywateli Polski. Tak jak poprzednio (część 8.

Błęd oszacowania) przeanalizujemy przykład badania sondażowego dotyczącego poparcia dla partii X.

Wykorzystamy tą samą technikę zbierania danych (CAPI), ten sam kwestionariusz i znowu będziemy losować 6 respondentów w sposób prosty bezzwrotny. Cała metodologia badania pozostanie bez zmian. Inaczej natomiast będzie przebiegała realizacja. Tym razem będziemy musieli się zmierzyć z problemem niedostępnością niektórych respondentów. Okaże się, że osoby zamieszkałe w mieście konsekwentnie, pomimo próśb i gróźb naszych ankieterów, będą odmawiać udzielania odpowiedzi na pytania z naszego kwestionariusza. To oczywiście skrajny przypadek. W rzeczywistości nigdy nie jest aż tak źle. Prawdą jest jednak, że mieszkańcy miast, z różnych powodów, są trudniej uchwytni i sondażyści mają dużo problemów z dotarciem do tej grupy społecznej.

Jakie konsekwencje dla naszego schematu losowania będzie miała niedostępność respondentów zamieszkałych w mieście? Niestety bardzo poważne. Po pierwsze zmienią się możliwe do uzyskania wyniki otrzymywane z 6-osobowych prób. Będą one bowiem zależały w znacznym stopniu od tego, którzy respondenci zgodzą się wziąć udział w badaniu.

Tabela 9.2. Wyniki z prób sześciuoosobowych przy niepełnej realizacji (mieszkańcy miast są

niedostępni)

6 dostępnych 5 dostępnych 4 dostępnych 3 dostępnych
respondentów respondentów respondentów respondentów

-	-	-	-
-	-	-	2/3
-	-	2/4	-
-	2/5	-	-
2/6	-	-	1/3
-	-	1/4	-
-	1/5	-	-
1/6	-	-	-
0	0	0	0

Zauważmy, że w naszej sytuacji, gdy mieszkańcy miast są niedostępni, maksymalna liczba zwolenników partii X w próbie jest równa 2. Tylu dokładnie jest mieszkańców wsi, którzy zamierzają głosować na to ugrupowanie. Co więcej, przy niepełnej realizacji próby może się okazać, że mimo, iż początkowo dobierzemy 6 respondentów, to praktycznie wywiady uda się przeprowadzić tylko z 5, 4, 3, 2 z nich lub z żadnym. W efekcie nasz schemat losowania, mimo iż nic w nim nie zmieniliśmy, zacznie generować zupełnie inne wyniki. Łącznie jest ich 9 (tyle ile wierszy w powyższej tabeli) - od 0% poparcia do

100% poparcia. W znacznej mierze są one kreowane przez liczbę dostępnych respondentów. W związku z tym rozkład wyników jest trochę trudniejszy do wyznaczenia niż wcześniej. Tym razem pominiemy jednak obliczenie (niewiel różnią się od tych przy pełnej realizacji - trzeba tylko uwzględnić fakt, że w próbie może być różna liczba respondentów, od 0 do 6, i że wszyscy są mieszkańcami wsi) i skupimy się na rozwiązaniach.

Tabela 9.3. Liczba dostępnych respondentów w próbach sześciuosobowych

Liczba dostępnych respondentów	Liczba prób	Procent prób
0	665280	2,38
1	4561920	16,35
2	9979200	35,76
3	8870400	31,79
4	3326400	11,92
5	483840	1,73
6	20160	0,07
Razem	27907200	100,00

Jak widać powyżej tylko dla znikomej części prób liczba dostępnych respondentów wyniesie 6. Jednocześnie

łączna liczba prób, w których dostępnych jest przynajmniej jeden respondent wynosi 27241920, a więc wyraźnie mniej niż w przykładzie z pełną realizacją. Aż 665280 prób składa się wyłącznie z mieszkańców miast, a więc nie da się w nich przeprowadzić nawet jednego wywiadu. Zauważmy również, że w ponad połowie prób dostępnych respondentów jest nie więcej niż 3. W tej sytuacji oczywiste jest, że rozkład wyników z próby będzie inny niż gdy dostępni byli wszyscy respondenci. Zanim jednak przystąpimy do jego wyznaczania musimy ustalić kilka zasad. Po pierwsze, próby w których wszyscy respondenci odmówili wzięcia udziału w badaniu muszą zostać odrzucone. Nie da się bowiem w nich wyznaczyć poziomu poparcia dla partii X. Po drugie do analiz będziemy włączyć próby, w których dostępnych jest tylko jeden respondent. W takich przypadkach poparcie dla partii X może wynosić 0% lub 100%. Przekonajmy się teraz, jak wygląda rozkład uzyskiwanych wyników.

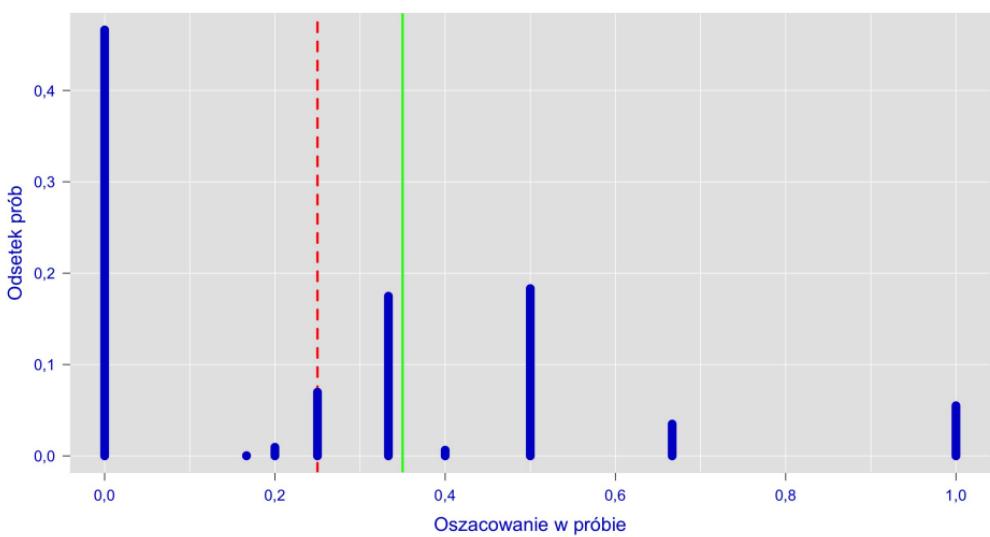
Tabela 9.4. Poparcie dla partii X w próbach sześciuosobowych przy niepełnej realizacji (niedostępni mieszkańcy miast)

Poparcie w próbie	Liczba prób	Procent prób
0	12700800	46,62
16,67	8640	0,03

20	259200	0,95
25	1900800	6,98
33,33	4762800	17,48
40	172800	0,63
50	4989600	18,32
66,67	950400	3,49
100	1496880	5,49
Razem	27241920	100,00

W powyższej tabeli widzimy, że w prawie połowie wszystkich analizowanych prób poparcie dla partii X wynosi 0%! Do tej grupy zaliczają się wszystkie próby 6, 5, 4, 3, 2 i 1 osobowe, w których żadeb respondent nie opowiada się za interesującym nas ugrupowaniem. Następną liczne grupy stanowią próby, w których poparcie dla partii X wyniosło 50% oraz 33%. Pozostałe wyniki pojawiają się sporadycznie. Może to zobaczyć wyraźnie na wykresie.

Wykres 9.1. Rozkład poparcia dla partii X w próbach 6 osobowych przy niepełnej realizacji (niedostępni mieszkańcy miast)



Inaczej niż przy pełnej realizacji, tym razem wyniki z prób są bardzo rozproszone. Oczywiście jest to spowodowane odmowami mieszkańców miast. Powstaje pytanie, czy przy niepełnej realizacji nasz schemat losowania respondentów wciąż jest tak dobry jak wcześniej? Możemy się spodziewać, że wyniki są obarczone większym błędem (z powodu rozproszenia). Ale co z przeciętnym wynikiem z próby? Łatwo się domyślić, że nie jest on równy wartości obserwowanej w populacji (pionowa ciągła zielona linia). Przeciętny poziom poparcia dla partii X w 6-osobowych przy niepełnej realizacji wynosi 25% (pionowa przerywana czerwona linia). Oto ostateczny dowód na to, że niedostępność respondentów może się przyczynić do poważnego obciążenia wyników z próby, a więc do ich “odchylenia” względem rzeczywistej wartości. W tym przypadku

obciążenie wynosi aż -10 (25%-30%=-10%) punktów procentowych. Ale to nie wszystko. Zauważmy, że przeciętny poziom poparcia dla partii X w próbach o niepełnej realizacji jest równy dokładnie poziomowi poparcia w populacji mieszkańców wsi! Okazuje się więc, że nasza próba, która w założeniu miała opisywać całą populację, w rzeczywistości opisuje wyłącznie mieszkańców wsi!

Tak oto przekonaliśmy się, że wszystkie wysiłki włożone w przygotowanie koncepcji badania, wybranie techniki przeprowadzania wywiadów, napisanie kwestionariusza i wylosowanie próby mogą pójść na marne jeżeli etap realizacji nie zostanie przeprowadzony rzetelnie.

Niski poziom realizacji próby, może się przyczynić do obciążenia wyników. Będą one wtedy systematycznie dążyły do innej wartości niż ta w badanej zbiorowości.

Właściwie będą one dążyły do wartości występującej w populacji dostępnych respondentów. W konsekwencji im bardziej dostępni respondenci różnią się od respondentów niedostępnych, tym większe obciążenie wyników. Na koniec kursu zapamiętajmy więc, że prawdziwi sondażyści robią wszystko co w ich mocy, żeby poziom realizacji próby był jak najwyższy. Pamiętajcie o tym, gdy ktoś zaprosi Was do udziału w badaniu sondażowym.

Zadania 9:

1. Z wylosowanej wcześniej próby 15 osób (*Zadania 8.*) wyłącz wszyskie osoby zamieszkałe w mieście. Podaj liczby porządkowe osób, które zostały w próbie.
.....
2. Jaki jest teraz poziom poparcia dla partii X w twojej próbie?
.....
3. Oblicz błąd kwadratowy twojego wyniku względem poparcia dla partii X w populacji.
.....
4. Czy wynik z twojej próby mieści się w przedziale (17%, 83%)?
 - Tak
 - Nie
5. Czy wiesz już jak powstają sondaże?
 - Tak
 - Nie

Zbiory danych

Przemysław Biecek @ Uniwersytet Warszawski

sezon 1 / odcinek 17

pogRomcy danych

- O czym jest ten odcinek
- Dwie wersje językowe
- Najszybsze koty i ptaki na świecie
- Imiona noworodków w Warszawie
- Wyniki wyborów samorządowych w 2014
- Ceny ofertowa używanych aut w roku 2012
- Indeks WIG z Giełdy Papierów Wartościowych
- Wzrost rodziców i dzieci
- Wyniki skoków narciarskich
- Diagnoza społeczna
- Oceny odcinków seriali filmowych
- Rokowania dla raka piersi

O czym jest ten odcinek

Praca z danymi może być bardzo różnorodna. Podczas kursu „Pogromcy Danych” będziemy tę różnorodność

przybliżać. W tym celu będziemy pracować na zbiorach danych o różnej wielkości (od kilkunastu do kilkuset tysięcy wierszy), różnym stopniu złożoności (od dwóch do dwóch tysięcy kolumn), oraz o różnym formacie (dane tekstowe, ilościowe, jakościowe).

Aby ułatwić dostęp do tak różnych zbiorów danych zebraliśmy je wszystkie w jednym pakiecie o nazwie `PogromcyDanych`. Znaleźć można w nim najróżniejsze dane, w tym o cenach ofertowych aut, dane ankietowe dotyczące badań społecznych, informacje o zmianach indeksów giełdowych, imionach noworodków, wynikach skoków narciarskich czy o wynikach leczenia pacjentek z nowotworem piersi.

Aby ten pakiet zainstalować należy w programie R wykonać instrukcję (wystarczy wykonać ją raz). Pobierze ona z Internetu wszystkie te zbiory danych oraz zainstaluje na dysku.

```
install.packages ("PogromcyDanych")
```

Po zainstalowaniu, aby korzystać z tych zbiorów danych, należy pakiet `PogromcyDanych` włączyć. Pakiet włącza się poniższą instrukcją (należy ją wykonać po każdym uruchomieniu programu R).

```
library (PogromcyDanych)
```

Na kolejnych slajdach przedstawimy wszystkie znajdujące się w tym pakiecie zbiory danych.

Dwie wersje językowe

Kurs „Pogromcy Danych” jest prowadzony w dwóch wersjach językowych, polskiej i angielskiej. Aby nie kopiować dwukrotnie danych dla obu wersji językowych po wczytaniu pakietu `PogromcyDanych` dostępne są dane w języku polskim.

Aby przełączyć się na wersję angielską należy użyć polecenia `setLang()` tak jak na poniższym przykładzie. Ta funkcja przetłumaczy polskie nazwy danych (też nazwy kolumn i wartości) na ich angielskojęzyczne odpowiedniki.

```
setLang("eng")
```

W wyniku jej działania do przestrzeni nazw wkopiowane będą następujące zbiory danych:

Nazwa polska	Nazwa angielska	Co to za dane?
koty_ptaki	cats_birds	Charakterystyki 13 gatunków Imiona

imiona_warszawa warsaw_names

noworodków w Warszawie

mandatySejmik2014 votes2014

Wyniki wyborów samorządowych 2014

auta2012

auta2012

Ceny ofertowe aut w roku 2012

WIG

WIG

Indeks WIG w roku 2014

pearson

pearson

Dane o wzroście Pearsona

galton

galton

Dane o wzroście Galtona

skiJumps2013

skiJumps2013

Wyniki skoków narciarskich w sezonie 2013/2014

skiJumps2013labels

skiJumps2013labels

Wyjaśnienia nazw kolumn

diagnoza

diagnosis

Dane z projektu Diagnoza Społeczna Wyjaśnienia

diagnozaDict	diagnosisDict	nazw kolumn
serialeIMDB	seriesIMDB	Dane o serialach na bazie IMDB
TCGA_BRCA	TCGA_BRCA	Dane o pacjentkach z nowotworem piersi

Instrukcja `setLang ("pol")` tłumaczy z powrotem na polskie nazwy.

Najszybsze koty i ptaki na świecie

Zbiór danych `koty_ptaki` powstał jako tzw. „toy-example”. Jest to niewielki (13 wierszy 7 kolumn) zbiór, który można w całości wyświetlić na ekranie i na którym można przećwiczyć podstawowe operacje na danych.

W danych zestawiono charakterystyki dla 13 wybranych gatunków przedstawionych w kolejnych wierszach. Wybrane gatunki to najszybsi przedstawiciele kotów i ptaków. Każdy wiersz to informacje o innym gatunku.

Kolejne kolumny w tym zbiorze danych przedstawiają:

- nazwę gatunku,
- maksymalną osobniczą wagę w kilogramach i maksymalną długość ciała w metrach,
- maksymalną prędkość (dla ptaków w locie poziomym, nie w nurkowaniu) w kilometrach na godzinę,
- obszar zamieszkania oraz żywotność w latach,
- ostatnia kolumna określa czy gatunek jest dużym kotem czy ptakiem.

Funkcją `head()` możemy wyświetlić pierwsze 6 wierszy z tego zbioru danych.

```
head(koty_ptaki)
```

```
##      gatunek waga dlugosc predkosc habitat zywo
## 1    Tygrys   300     2,5       60     Azja
## 2      Lew    200     2,0       80   Afryka
## 3   Jaguar   100     1,7       90 Ameryka
## 4     Puma    80     1,7       70 Ameryka
## 5 Leopard    70     1,4       85     Azja
## 6   Gepard    60     1,4      115   Afryka
```

Imiona noworodków w Warszawie

Zbiór danych `imiona_warszawa` przygotowaliśmy po to, by przedstawić metody analizy trendów oraz metody analizy napisów.

W tym zbiorze danych zawarto informację o liczbie urodzonych noworodków o danym imieniu w Warszawie w kolejnych miesiącach w okresie od roku 2004 do 2014. Dane te są pobrane ze strony <http://gorny.edu.pl/imiona/index.php>, na której można znaleźć również informacje o aktualnej liczbie urodzin.

W tym zbiorze danych jest 84816 wierszy, dane dotyczą 696 różnych imion od tych najpopularniejszych do bardzo rzadkich.

Kolejne kolumny w tym zbiorze danych przedstawiają:

- imię i płeć dziecka,
- rok i miesiąc którego dotyczy ta statystyka,
- liczbę noworodków o danym imieniu urodzonych w Warszawie w danym roku i miesiącu.

Funkcją `head()` wyświetlamy pierwsze 6 wierszy.

```
head(imiona_warszawa)
```

##	imie	plec	rok	miesiac	liczba
## 1	Aaron	M	2004	7	0
## 2	Aaron	M	2004	8	0
## 3	Aaron	M	2004	9	0
## 4	Aaron	M	2004	10	0
## 5	Aaron	M	2004	11	0
## 6	Aaron	M	2004	12	0

Funkcją `levels()` wyświetlamy listę różnych imion, które są opisane w tym zbiorze danych.

```
levels(imiona_warszawa$imie)
```

```
## [1] "Aaron"                 "Abigail"                "I  
## [4] "Adam"                  "Adela"                  "I  
## [7] "Adrian"                "Adriana"                "I  
## [10] "Agata"                 "Agnieszka"              "I  
## [13] "Albert"                 "Aldona"                 "I  
## [16] "Aleksander"             "Aleksandra"             "I  
## [19] "Alex"                   "Alicja"                 "I  
## [22] "Alma"                  "Alwin"                  "I  
## [25] "Amanda"                "Amelia"                 "I  
## [28] "Anastazja"             "Anatol"                 "I  
## [31] "Andrzej"                "Andżelika"               "I  
## [34] "Angela"                 "Angelika"               "I  
## [37] "Aniela"                 "Anika"                  "I  
## [40] "Anna"                   "AnnaMaria"              "I  
## [43] "Anton"                  "Antoni"                 "I  
## [46] "Antonina"               "Antoniusz"              "I  
## [49] "Apolonia"              "Apoloniusz"              "I  
## [52] "Ariana"                 "Arkadiusz"              "I  
## [55] "Armand"                 "Arnold"                 "I  
## [58] "Artur"                  "August"                 "I  
## [61] "Aura"                   "Aurelia"                "I  
## [64] "Bakary"                 "Balbina"                "I  
## [67] "Barbara"                "Barta"                  "I  
## [70] "Bastian"                "Bazyli"                 "I  
## [73] "Belzebub"               "Benedykt"               "I  
## [76] "Benita"                 "Berenika"               "I  
## [79] "Bernadetta"              "Bernard"                "I  
## [82] "Bibianna"               "Blanka"                 "I  
## [85] "Bogdan"                 "Bogna"                  "I  
## [88] "Bogumiła"               "Bogusław"               "I
```

##	[91]	"Bogusz"	"Bolesław"	"I
##	[94]	"Bożena"	"Brajan"	"I
##	[97]	"Bronisław"	"Bronisława"	"I
##	[100]	"Brygida"	"Calineczka"	"(
##	[103]	"Caspian"	"Cecylia"	"(
##	[106]	"Cezary"	"Chaim"	"(
##	[109]	"Chiara"	"Chloe"	"(
##	[112]	"Cypinia"	"Cyprian"	"(
##	[115]	"Czesław"	"Dąbrowka"	"I
##	[118]	"Dagmara"	"Dagna"	"I
##	[121]	"Dalia"	"Damian"	"I
##	[124]	"Daniela"	"Danuta"	"I
##	[127]	"Dariusz"	"Dawid"	"I
##	[130]	"Delfina"	"Denis"	"I
##	[133]	"Dionizy"	"Dobrawa"	"I
##	[136]	"Dobromir"	"Dobrosława"	"I
##	[139]	"Dominika"	"Donald"	"I
##	[142]	"Dorota"	"Dymitr"	"I
##	[145]	"Edmund"	"Edward"	"I
##	[148]	"Edyta"	"Elena"	"I
##	[151]	"Eliasz"	"Eliot"	"I
##	[154]	"Eljasz"	"Elmira"	"I
##	[157]	"Elwira"	"Elżbieta"	"I
##	[160]	"Emanuela"	"Emil"	"I
##	[163]	"Emilian"	"Emma"	"I
##	[166]	"Erwin"	"Eryk"	"I
##	[169]	"Esmee"	"Esterा"	"I
##	[172]	"Eunika"	"Ewa"	"I
##	[175]	"Fabian"	"Faustyna"	"I
##	[178]	"Felicjan"	"Felicyta"	"I
##	[181]	"Ferdynand"	"Filip"	"I
##	[184]	"Filomena"	"Florentyna"	"I
##	[187]	"Fontanna"	"Franciszek"	"I
##	[190]	"Franciszka"	"Franczeska"	"I
##	[193]	"Fryderyk"	"Gabor"	"(

## [196]	"Gabriela"	"Gaja"	"
## [199]	"Gaweł"	"Gerard"	"
## [202]	"Gniewko"	"Gniewosz"	"
## [205]	"Gracja"	"Gracjan"	"
## [208]	"Grażyna"	"Greta"	"
## [211]	"Grzegorz"	"Guantanamera"	"
## [214]	"Gwen"	"Gwidon"	"I
## [217]	"Halina"	"Hana"	"I
## [220]	"Hektor"	"Helena"	"I
## [223]	"Herbert"	"Herman"	"I
## [226]	"Hieronim"	"Hipolit"	"I
## [229]	"Horacy"	"Hubert"	"I
## [232]	"Ida"	"Idalia"	"I
## [235]	"Iga"	"Ignacy"	"I
## [238]	"Ilia"	"Ilian"	"I
## [241]	"Imre"	"Ina"	"I
## [244]	"Inez"	"Inga"	"I
## [247]	"Irena"	"Ireneusz"	"I
## [250]	"Irmina"	"Ivar"	"I
## [253]	"Iwo"	"Iwona"	"I
## [256]	"Izabela"	"Jacek"	"I
## [259]	"Jaga"	"Jagna"	"I
## [262]	"Jakub"	"Jan"	"I
## [265]	"Janina"	"JanJakub"	"I
## [268]	"Janko"	"Jano"	"I
## [271]	"Janusz"	"Jaromir"	"I
## [274]	"Jarowit"	"Jarzyna"	"I
## [277]	"Jaśmina"	"Jędrzej"	"I
## [280]	"Jeremiasz"	"Jerzy"	"I
## [283]	"Jesica"	"Jeżyna"	"I
## [286]	"Joanna"	"Jolanta"	"I
## [289]	"Jonata"	"Jonatan"	"I
## [292]	"Jowita"	"Józef"	"I
## [295]	"Judyta"	"Julia"	"I
## [298]	"Julianna"	"Julita"	"I

## [301]	"Jurand"	"Justyn"	"
## [304]	"Kacper"	"Kaj"	"I
## [307]	"Kajetan"	"Kalina"	"I
## [310]	"Kamil"	"Kamila"	"I
## [313]	"Karena"	"Karin"	"I
## [316]	"Karla"	"Karol"	"I
## [319]	"Kassandra"	"Kasjan"	"I
## [322]	"Kasper"	"Katarzyna"	"I
## [325]	"Kazimiera"	"Kazimierz"	"I
## [328]	"Kiara"	"Kinga"	"I
## [331]	"Klara"	"Klaudia"	"I
## [334]	"Klaudyna"	"Klemens"	"I
## [337]	"Kleofas"	"Kolin"	"I
## [340]	"Konstancja"	"Konstanty"	"I
## [343]	"Kordian"	"Korina"	"I
## [346]	"Kornelia"	"Korneliusz"	"I
## [349]	"Kryspin"	"Krystian"	"I
## [352]	"Krzesimir"	"Krzysztof"	"I
## [355]	"Ksawery"	"Ksawier"	"I
## [358]	"Laila"	"Lambert"	"I
## [361]	"Lara"	"Lars"	"I
## [364]	"Laura"	"Lea"	"I
## [367]	"Leila"	"Leja"	"I
## [370]	"Leo"	"Leokadia"	"I
## [373]	"Leonard"	"Leonardo"	"I
## [376]	"Leopold"	"Letycja"	"I
## [379]	"Lew"	"Lewin"	"I
## [382]	"Lili"	"Lilia"	"I
## [385]	"Lilla"	"Lilli"	"I
## [388]	"Liwia"	"Lorena"	"I
## [391]	"Lubomir"	"Łucja"	"I
## [394]	"Łucjan"	"Lucjusz"	"I
## [397]	"Ludmiła"	"Ludomił"	"I
## [400]	"Ludwika"	"Luiza"	"I
## [403]	"Lukrecja"	"Luna"	"I

## [406]	"Magdalena"	"Magnolia"	"I
## [409]	"Makary"	"Maks"	"I
## [412]	"Maksymilian"	"Maksymilianna"	"I
## [415]	"Malina"	"Malwina"	"I
## [418]	"Marcel"	"Marcela"	"I
## [421]	"Marcelina"	"Marcella"	"I
## [424]	"Marcin"	"Marcjanna"	"I
## [427]	"Marek"	"Margarita"	"I
## [430]	"MariaAntonina"	"Marian"	"I
## [433]	"Marietta"	"Marika"	"I
## [436]	"Mariola"	"Marisa"	"I
## [439]	"Marlena"	"Marletta"	"I
## [442]	"Martin"	"Martyna"	"I
## [445]	"Maryna"	"Marzena"	"I
## [448]	"Mateusz"	"Matylda"	"I
## [451]	"Maurycy"	"Max"	"I
## [454]	"Melchior"	"Melisa"	"I
## [457]	"Mia"	"Michał"	"I
## [460]	"Mieczysław"	"Mieszko"	"I
## [463]	"Mikaela"	"Mikołaj"	"I
## [466]	"Mila"	"Miła"	"I
## [469]	"Milan"	"Milena"	"I
## [472]	"Milo"	"Miłosz"	"I
## [475]	"Miranda"	"Mirella"	"I
## [478]	"Miron"	"Mirosław"	"I
## [481]	"Monika"	"Morfeusz"	"I
## [484]	"Myszon"	"Nadia"	"I
## [487]	"Naomi"	"Napoleon"	"I
## [490]	"Natalia"	"Natan"	"I
## [493]	"Natasza"	"Nawojka"	"I
## [496]	"Nel"	"Nela"	"I
## [499]	"Nestor"	"Nicola"	"I
## [502]	"Nika"	"Nike"	"I
## [505]	"Nikodem"	"Nikol"	"I
## [508]	"Nikolas"	"Nikole"	"I

## [511]	"Nikolina"	"Nina"	"I
## [514]	"Noam"	"Noe"	"I
## [517]	"Norbert"	"Norman"	"C
## [520]	"Odolan"	"Oksana"	"C
## [523]	"Oktawian"	"Olaf"	"C
## [526]	"Olena"	"Olga"	"C
## [529]	"Olimpia"	"Oliwia"	"C
## [532]	"Orest"	"Oriana"	"C
## [535]	"Oswald"	"Otylia"	"J
## [538]	"Pamela"	"Pascal"	"J"]
## [541]	"Patrycjusz"	"Patryk"	"J"]
## [544]	"Paula"	"Paulina"	"J"]
## [547]	"Petronela"	"Piotr"	"J"]
## [550]	"Przemysław"	"Rachel"	"I"]
## [553]	"Radomił"	"Radosław"	"I"]
## [556]	"Rafał"	"Rajmund"	"I"]
## [559]	"Raul"	"Rebeka"	"I"]
## [562]	"Remigiusz"	"Renata"	"I"]
## [565]	"Robert"	"Roch"	"I"]
## [568]	"Roger"	"Roksana"	"I"]
## [571]	"Roman"	"Romeo"	"I"]
## [574]	"Romualda"	"Ronald"	"I"]
## [577]	"Róża"	"Rozalia"	"I"]
## [580]	"Rufus"	"Rupert"	"I"]
## [583]	"Ryszard"	"Sabina"	"S"]
## [586]	"Salomea"	"Samanta"	"S"]
## [589]	"Samuel"	"Sandra"	"S"]
## [592]	"Sasha"	"Saturnin"	"S"]
## [595]	"Sebastian"	"Selena"	"S"]
## [598]	"Selma"	"Serafin"	"S"]
## [601]	"Seweryn"	"Sindi"	"S"]
## [604]	"Sofia"	"Sonia"	"S"]
## [607]	"Stefan"	"Stefania"	"S"]
## [610]	"Sylas"	"Sylwester"	"S"]
## [613]	"Symeon"	"Szarlota"	"S"]

## [616]	"Szymon"	"Tadeusz"
## [619]	"Tatiana"	"Telimena"
## [622]	"Teodor"	"Teofil"
## [625]	"Theo"	"Thorgal"
## [628]	"Tobiasz"	"Tola"
## [631]	"Torkil"	"Toro"
## [634]	"Tristan"	"Tycjan"
## [637]	"Tymoteusz"	"Tytus"
## [640]	"Urszula"	"Vanessa"
## [643]	"Viktoria"	"Viorika"
## [646]	"Waldemar"	"Waleria"
## [649]	"Walter"	"Wanda"
## [652]	"Wawrzyniec"	"Wera"
## [655]	"Wida"	"Wiera"
## [658]	"Wiktor"	"Wiktoria"
## [661]	"Wiliam"	"Wincenty"
## [664]	"Wisenna"	"Wit"
## [667]	"Witomir"	"Władysław"
## [670]	"Wojciech"	"Wolfgang"
## [673]	"Xawery"	"Xawier"
## [676]	"Zachariasz"	"Zachary"
## [679]	"Żaneta"	"Zara"
## [682]	"Zeira"	"Zenon"
## [685]	"Ziemosław"	"Ziemowit"
## [688]	"Zofia"	"Zoja"
## [691]	"Zoya"	"Zuzanna"
## [694]	"Żyraf"	"Zyta"

Wyniki wyborów samorządowych w 2014

Zbiór danych `mandatySejmik2014` przygotowaliśmy po to, by pokazać jak można analizować lub wizualizować dane przestrzenne. Dane pobrano ze stron Państwowej Komisji Wyborczej (<http://wybory2014.pkw.gov.pl/pl/>).

Dane przedstawiają informacje o liczbie wygranych mandatów w sejmikach w wyborach samorządowych w Polsce w roku 2014. Zbiór danych zawiera 16 wierszy i 9 kolumn. Każdy wiersz przedstawia wyniki dla innego województwa.

Kolejne kolumny opisują:

- nazwę województwa,
- liczbę zdobytych mandatów przez PSL, PiS, PO, SLD i inne partie,
- jaki procent z uprawnionych do głosowania stanowi procent ważnych głosów,
- długość i szerokość geograficzną środka geograficznego województwa.

Poniżej prezentujemy pierwsze 6 wierszy z tego zbioru danych.

```
head(mandatySejmik2014)
```

##	Województwo	PSL	PiS	PO	SLD	Inne	Pro
## 1	Dolnośląskie	5	9	16	2	4	
## 2	Kujawsko-Pomorskie	10	7	14	2	0	

```
## 3           Lodzkie   10 12 10  1  0
## 4           Lubelskie 12 13  7  1  0
## 5           Lubuskie  8  5 10  5  2
## 6           Malopolskie 8 17 14  0  0
##          lat
## 1 51,07988
## 2 53,02223
## 3 51,55958
## 4 51,24725
## 5 52,20549
## 6 49,84203
```

Ceny ofertowa używanych aut w roku 2012

Zbiór danych `auta2012` przygotowaliśmy po to, by dać możliwość pracy z dużym i ciekawym zbiorem danych o transakcjach. Dane transakcyjne są zazwyczaj bardzo długie, a sposób ich przetwarzania jest specyficzny.

Dane zostały pobrane w roku 2012 z serwisu ogłoszeń otomoto.pl. Zbiór danych zawiera informacje o ofertach dla przeszło 207 tysięcy ogłoszeń sprzedawy auta.

Każda oferta (każdy wiersz) opisana jest przez 21 zmiennych, w tym:

- cenę, walutę i informację czy jest to cena brutto czy

netto,

- informację o mocy silnika w koniach mechanicznych i kW, oraz jego pojemności,
- informację o modelu, marce, wersji auta,
- informację o rodzaju napędu, roku produkcji oraz przebiegu (w km),
- informację o kolorze, liczbie drzwi, kraju pochodzenia, kraju rejestracji oraz wyposażeniu.

Informacja o wyposażeniu jest opisana przez napis z wartościami rozdzielanymi przecinkiem, przez co sam napis może być bardzo długi.

Poniżej przedstawiamy 6 pierwszych wierszy z tego zbioru danych.

```
head(auta2012)
```

##	Cena	Waluta	Cena.w.PLN	Brutto.netto	KM
## 1	49900	PLN	49900	brutto	140
## 2	88000	PLN	88000	brutto	156
## 3	86000	PLN	86000	brutto	150
## 4	25900	PLN	25900	brutto	163
## 5	55900	PLN	55900	netto	NA
## 6	45900	PLN	45900	netto	150
##	Wersja	Liczba.drzwi	Pojemnosc.skokowa	Prze	
## 1		4/5		1991	
## 2		4/5		2179	
## 3		4/5		1996	
## 4		4/5		2400	
## 5		4/5		2200	

6 4 / 5 2200
Rodzaj.paliwa Rok.produkcji
1 olej napędowy (diesel) 2008
2 olej napędowy (diesel) 2008
3 olej napędowy (diesel) 2009
4 olej napędowy (diesel) 2003 sreb:
5 olej napędowy (diesel) 2007
6 olej napędowy (diesel) 2004 bord:
Kraj.aktualnej.rejestracji Kraj.pochodzen:
1 Polska
2 Polska
3 Polska
4 Polska Wlocl
5
6
Skrzynia.biegow Status.pojazdu.sprowadzone
1 manualna
2 manualna
3 manualna
4 manualna sprowadzony / zarejestrowan
5 manualna
6 manualna

1
2
3
4 ABS, hak, el. szyby, el. lusterka, klimat:
5
6

Indeks WIG z Giełdy Papierów Wartościowych

Zbiór danych WIG przygotowaliśmy po to, by przyjrzeć się analizie i wizualizacji danych o dłuższych szeregach czasowych, w tym przypadku o dziennych notowaniach na giełdzie.

Ze strony Giełdy Papierów Wartościowych

<http://www.gpwinfostrafa.pl/GPWIS2/pl/index/> pobrano dzienne notowania dla indeksu WIG (Warszawski Indeks Giełdowy) z okresu grudzień 2013 - listopad 2014.

W kolejnych kolumnach przedstawiono:

- datę, której dotyczą notowania,
- kurs otwarcia, zamknięcia, kurs minimalny i maksymalny,
- zmianę kursu oraz wartość obrotów w tysiącach złotych.

W zbiorze danych jest 248 wierszy, każdy wiersz odpowiada notowaniom z jednego dnia. W soboty, niedziele i święta giełda jest zamknięta, stąd ta liczba dni z notowaniami.

Poniżej przedstawiamy pierwszych 6 wierszy z tego zbioru danych.

```
head(WIG)
```

##	Data	Nazwa	Kurs.otwarcia	Kurs.maksymalny
----	------	-------	---------------	-----------------

```

## 1 2013-12-02 WIG      54627,26 5479
## 2 2013-12-03 WIG      54025,72 5402
## 3 2013-12-04 WIG      53222,49 5328
## 4 2013-12-05 WIG      52837,25 5290
## 5 2013-12-06 WIG      52837,58 5289
## 6 2013-12-09 WIG      53113,49 5318
##   Kurs.zamkniecia Zmiana Wartosc.obrotu.w.t]
## 1           53934,52 -1,41 640
## 2           53276,83 -1,22 912
## 3           52867,03 -0,77 968
## 4           52597,13 -0,51 808
## 5           52727,52  0,25 1012
## 6           52881,29  0,29 599

```

Wzrost rodziców i dzieci

Słowo regresja wywodzi się z pionierskich badań Francisza Galtona i Karla Pearsona nad zależnością wzrostu dzieci i rodziców.

Oryginalne zbiory danych obu tych badaczy są dostępne w zmiennych `galton` i `pearson`. Na bazie tych zbiorów danych można opisywać zależność pomiędzy wzrostem syna a ojca (`pearson`) oraz zależność pomiędzy ważoną średnią z wzrostu rodziców (`galton`) z wzrostem syna.

Poniżej przedstawiamy pierwszych 6 wierszy z każdego ze zbiorów danych. W zbiorze danych `galton` znajduje się 928 wierszy a w zbiorze danych `pearson` znajduje się

1078 wierszy.

```
head(galton)
```

```
##      syn rodzic
## 1 156,7 179,1
## 2 156,7 174,0
## 3 156,7 166,4
## 4 156,7 163,8
## 5 156,7 162,6
## 6 158,0 171,4
```

```
head(pearson)
```

```
##      syn ojciec
## 1 151,8 165,2
## 2 160,6 160,7
## 3 160,9 165,0
## 4 159,5 167,0
## 5 163,3 155,3
## 6 163,2 160,1
```

Wyniki skoków narciarskich

W zbiorze danych `skiJumps2013` zebrane są wyniki skoków narciarskich z sezonu 2013/2014. Ten zbiór danych został przygotowany na potrzebę konkursu na wizualizacje danych podczas konferencji [PAZUR](#).

W zbiorze danych znajduje się 2130 wierszy, każdy wiersz opisuje jeden oddany skok w zawodach w skokach

narciarskich z sezonu 2013/2014.

Każdy skok opisuje 16 kolumn, znaczenie poszczególnych kolumn jest opisane w zbiorze danych
skiJumps2013labels.

W zbiorze danych `skiJumps2013`, dla każdego oddanego skoku znaleźć można informacje o:

- konkursach, takie jak: miejscowości, kraj, parametry skoczni,
- skoczkach, takie jak: imię, nazwisko, narodowość, data urodzin,
- skokach (dwóch, jeżeli skoczek skakał dwa razy lub jednego jeżeli nie zakwalifikował się do drugiej serii): prędkość, odległość, punkty do klasyfikacji.

Poniżej przedstawimy 6 wierszy z tego zbioru danych.

```
head(skiJumps2013)
```

```
##      jumperSurname jumperName compName compCou
## 1          AHONEN       JANNE   Kuusamo    Fin
## 2          AMMANN        SIMON   Kuusamo    Fin
## 3          AMMANN        SIMON   Kuusamo    Fin
## 4      ASIKAINEN       LAURI   Kuusamo    Fin
## 5          BIEGUN     KRZYSZTOF   Kuusamo    Fin
## 6          BIEGUN     KRZYSZTOF   Kuusamo    Fin
##      jumpSeries jumperCountry jumpSpeed jumpDis
## 1            1             FIN      90,8
## 2            1             SUI      90,3
```

```
## 3      2      SUI      90,6
## 4      1      FIN      90,1
## 5      2      POL      90,1
## 6      1      POL      90,1
##   jumpTotalPoints compTotalPoints classPoint
## 1           114,8          114,8     1
## 2           122,6          122,6     1
## 3           128,3          250,9     1
## 4           104,9          104,9     1
## 5           119,2          242,3     1
## 6           123,1          123,1     1
```

Diagnoza społeczna

Ciekawym zbiorem danych jest wynik panelowego badania *Diagnoza Społeczna*. W ramach tego projektu co dwa lata ankietuje się osoby z wybranego zbioru gospodarstw domowych, za każdym razem tych samych gospodarstw. Podczas wywiadu członkowie gospodarstw są pytani o rozmaite zagadnienia, co pozwala na budowę obrazu przemian dzierżących się w Polsce. Więcej o tym badaniu, wynikach jak i zbiorze danych można przeczytać na stronie internetowej projektu <http://diagnoza.com>.

Zbiór danych w postaci gotowej do przetwarzania w programie R, znajduje się na stronie <https://github.com/pbiecek/Diagnoza>. Można go zainstalować polecением

```
install_github("pbiecek/Diagnoza")
```

wcześniejszym włączeniu pakietu `library(devtools)`.

Cały zbiór danych jest bardzo duży i mógłby sprawiać trudności na mniejszych komputerach. Dlatego na potrzeby tego kursu przygotowaliśmy podzbiór zbioru danych z badania Diagnoza Społeczna.

Podzbiór danych nazywa się `diagnoza` i zawiera 38461 wierszy. Każdy wiersz to odpowiedzi innej osoby.

Odpowiedzi uzyskane w badaniu ankietowym zapisane są w 36 kolumnach / zmiennych. Nazwy tych zmiennych odpowiadają numerom pytań z kwestionariusza

http://diagnoza.com/pliki/kwestionariusze_instrukcje/kwes

Opisy co znaczy które pytanie znajdują się w zbiorze danych `diagnozaDict`.

Wybrane zmienne opisują:

- imiona respondentów,
- wagi analityczne, wynikające ze sposobu losowania,
- liczbę lat nauki, płeć, wykształcenie, wzrost, wagę, dochody,
- odpowiedzi na wybrane pytania dotyczące światopoglądu.

Zbiór danych `diagnozaDict` opisuje nazwy kolumn ze zbioru danych `diagnoza`. Przedstawiamy opis pierwszych sześciu kolumn.

```
head(diagnozaDict[,2,drop=FALSE])
```

```
##  
## imie_2011           IMIE CZŁONKA Z POMIARÓW 200  
## waga_2013_osoby      Waga dla członków 9  
## lata_nauki_2013      Liczba lat nauki w 2013 roku  
## wiek2013  
## plec  
## wojewodztwo
```

Poniżej przedstawiamy dwa wybrane wiersze z tego zbioru danych.

```
head(diagnoza, 2)
```

```
##    imie_2011 waga_2013_osoby lata_nauki_2013  
## 1  WERONIKA          0,277653                      NA  
## 2     ERNEST          0,277653                     11  
##             wojewodztwo                         eduk4_2013  
## 1 Świętokrzyskie                  <NA>  
## 2 Świętokrzyskie zasadnicze zawodowe/gimnazjum  
##       gp3            gp29        gp54_01  
## 1   <NA>           <NA>           <NA>  
## 2 UDANE POCZUCIE SENSU RACZEJ TAK ZDECYDOWAĆ  
##                 gp54_04  gp54_05    gp54_06  gp54_07  
## 1           <NA>    <NA>           <NA>  
## 2 ANI TAK, ANI NIE      NIE RACZEJ NIE RACZEĆ  
##       gp54_10  gp54_11        gp54_12  
## 1   <NA>    <NA>           <NA>  
## 2     TAK      TAK ZDECYDOWANIE TAK ZDECYDOWAĆ  
##                 gp54_15        gp54_16  
## 1           <NA>           <NA>  
## 2 ANI TAK, ANI NIE ZDECYDOWANIE TAK ANI TAK  
##       gp54_19        gp54_20  gp54_21  
## 1           <NA>           <NA>           <NA>
```

```
## 2 ZDECYDOWANIE NIE ZDECYDOWANIE TAK      TAK
##   gp113 wiek2013_4g
## 1    NA      (0,25]
## 2    NA      (0,25]
```

Oceny odcinków seriali filmowych

W zbiorze danych `serialsIMDB` zebraliśmy informacje o popularności odcinków seriali.

Z serwisu <http://www.imdb.com> pobraliśmy dane o ocenach oraz liczbie głosów oddanych na dany odcinek dla 200 najpopularniejszych seriali telewizyjnych. W zbiorze danych są również umieszczone nazwy seriali oraz nazwy poszczególnych odcinków.

Każdy wiersz opisuje jeden odcinek, wierszy w sumie jest 20122. Kolejne zmienne/kolumny opisują:

- nazwę serialu, nazwę odcinka,
- numer sezonu, numer odcinka w sezonie,
- średnia ocena danego odcinka,
- liczba oddanych głosów,
- identyfikator serialu używany w bazie IMDB.

Poniżej przedstawimy pierwszych 6 wierszy z tego zbioru danych.

```
head(serialeIMDB)
```

```
##      id      serial
## 1 1 Breaking Bad
## 2 2 Breaking Bad
## 3 3 Breaking Bad
## 4 4 Breaking Bad
## 5 5 Breaking Bad
## 6 6 Breaking Bad
##          imdbId
## 1 tt0903747
## 2 tt0903747
## 3 tt0903747
## 4 tt0903747
## 5 tt0903747
## 6 tt0903747
```

Rokowania dla raka piersi

Z bazy danych *The Cancer Genome Atlas (TCGA)* <http://cancergenome.nih.gov/> pobrano podzbiór danych klinicznych i genetycznych pacjentów (głównie pacjentek) z nowotworem piersi.

W zbiorze danych `TCGA_BRCA` zebrano wyniki dla 999 pacjentów. Każdy wiersz to jeden pacjent. Dla każdego pacjenta podane jest 5 cech:

- informacja o mutacji genu TP53,
- płeć pacjenta,

- informacja czy pacjent żyje pięć lat po zabiegu,
- liczba dni od operacji do zgonu (jeżeli wystąpił),
- informacja o tym czy doszło do wznowy i czy pojawił się nowy guz.

Poniżej przedstawiamy 6 pierwszych wierszy z tego zbioru danych.

```
head(TCGA_BRCA)
```

```
##                               TP53     plec  czy.zyje dni.do..  
## 1           Wild type female    live  
## 2           Other   female    live  
## 3           Wild type female    live  
## 4           Wild type female    live  
## 5 Missense_Mutation female  dead  
## 6           Wild type female    live
```

Poniżej przedstawiamy liczbę pacjentów z podziałem na płci oraz z podziałem na rodzaj mutacji genu TP53 (wild type oznacza brak mutacji).

```
table(TCGA_BRCA$plec)
```

```
##  
## female     male  
##      989      10
```

```
table(TCGA_BRCA$TP53)
```

```
##  
## Missense_Mutation          Other  
##                  168            125
```


Zadania i odpowiedzi

Przemysław Biecek @ Uniwersytet Warszawski

sezon 2 / odcinek 16

pogRomcy danych

- [Zadanie, sezon 2, odcinek 2](#)
- [Odpowiedź, sezon 2, odcinek 2](#)
- [Zadanie, sezon 2, odcinek 3](#)
- [Odpowiedź, sezon 2, odcinek 3](#)
- [Zadanie, sezon 2, odcinek 4](#)
- [Odpowiedzi, sezon 2, odcinek 4](#)
- [Zadanie, sezon 2, odcinek 7](#)
- [Odpowiedź, sezon 2, odcinek 7](#)
- [Zadania, sezon 2, odcinek 8](#)
- [Odpowiedzi, sezon 2, odcinek 8](#)
- [Zadanie, sezon 2, odcinek 9](#)
- [Odpowiedź, sezon 2, odcinek 9](#)
- [Zadanie, sezon 2, odcinek 10](#)
- [Odpowiedź, sezon 2, odcinek 10](#)
- [Zadania, sezon 2, odcinek 11](#)
- [Odpowiedzi, sezon 2, odcinek 11](#)
- [Zadania, sezon 2, odcinek 13](#)
- [Zadania, sezon 2, odcinek 14](#)

Zadanie, sezon 2, odcinek 2

- Przedstaw graficznie za pomocą wykresu punktowego zależność pomiędzy żywotnością (kolumna `zywotnosc`) a wagą zwierzęcia (kolumna `waga`).
- Zaznacz kolorem lub kształtem punktu informację czy przedstawiany jest ptak czy kot. Czy są różnice pomiędzy żywotnością a wagą dla kotów i ptaków?
- Użyj etykiet by odczytać który ptak i który kot żyją najdłużej
- Używając geometrii wstęga (`geom_ribbon`) przedstaw kurs minimalny i maksymalny każdego dnia na podstawie danych ze zbioru `WIG`.

Odpowiedź, sezon 2, odcinek 2

```
library(PogromcyDanych)
library(ggplot2)
```

```
## Przedstaw graficznie za pomocą wykresu punktowego
## żywotnością (kolumna `zywotnosc`) a wagą zwierzęcia
ggplot(koty_ptaki, aes(x = waga, y=zywotnosc))
  geom_point()
```



```
## Zaznacz kolorem lub kształtem punktu informacji
```

```
## jest ptak czy kot. Czy są różnice pomiędzy :  
## kotów i ptaków?  
ggplot(koty_ptaki, aes(x = waga, y=zywotnosc, c  
geom_point())  
  
## Użyj etykiet by odczytać który ptak i który  
ggplot(koty_ptaki, aes(x = waga, y=zywotnosc, c  
geom_text())  
  
## Używając geometrii wstęga (`geom_ribbon`) p:  
## i maksymalny każdego dnia na podstawie danych  
ggplot(WIG, aes(x = Data, ymin=Kurs.minimalny,  
geom_ribbon())
```

Zadanie, sezon 2, odcinek 3

- Wybierz samochody marki Volkswagen model Passat a następnie narysuj jak średnia cena zależy od roku produkcji za pomocą geometrii `geom_smooth()`.
- Wybierz samochody marki Volkswagen, narysuj jak średnia cena zależy od roku produkcji, różnymi kolorami przedstaw różne modele Volkswagena.
- Wybierz pięcioletnie auta marki Volkswagen i za pomocą wykresu ramka - wąsy przedstaw jak cena auta zależy od modelu.
- Dla wybranych pięcioletnich aut marki Volkswagen

przedstaw w podziale na modele jaka część aut ma silnik diesla.

Odpowiedź, sezon 2, odcinek 3

```
## Wybierz samochody marki Volkswagen model Passat
## jak średnia cena zależy od roku produkcji za pomocą funkcji group_by()
## i funkcji summarise()

passat <- auta2012 %>%
  filter(Marka == 'Volkswagen', Model == 'Passat')
  select(Marka, Model, Rok.produkcji, Cena.w.PLN)

ggplot(passat, aes(x=Rok.produkcji, y=Cena.w.PLN))
  geom_smooth()

## Wybierz samochody marki Volkswagen, narysuj
## dla każdego modelu wykresy, różnymi kolorami przedstaw różnicę cenową
## dla każdego modelu

volkswagen <- auta2012 %>%
  filter(Marka == 'Volkswagen', Model != 'Passat')
  select(Marka, Model, Rok.produkcji, Cena.w.PLN)

ggplot(volkswagen, aes(x=Rok.produkcji, y=Cena.w.PLN))
  geom_smooth()

## Wybierz pięcioletnie auta marki Volkswagen z ramką - wąsy
## ramka - wąsy przedstaw jak cena auta zależy od modelu

volkswagen <- auta2012 %>%
  filter(Marka == 'Volkswagen', Model %in% c('Golf', 'Polo', 'Golf Plus', 'Polo Plus'))
  select(Marka, Model, Rok.produkcji, Cena.w.PLN)

ggplot(volkswagen, aes(x=Model, y=Cena.w.PLN))
```

```
geom_boxplot()
```

```
## Dla wybranych pięcioletnich aut marki Volkswagena
## na modele jaka część aut ma silnik diesla.
```

```
ggplot(volkswagen, aes(x=Model, fill=Rodzaj.paliwa))
  geom_bar(position = 'fill')
```

Zadanie, sezon 2, odcinek 4

- Podobnie jak w poprzednim odcinku, wybierz samochody marki Volkswagen model Passat a następnie narysuj jak średnia cena zależy od roku produkcji za pomocą geometrii `geom_smooth()`. Następnie zobacz jak ten wykres będzie wyglądał z motywami `theme_bw()`, `theme_excel()` i `theme_economist()`.
- Zmień poniższy wykres, zamieniając skalę kolorów na od zielonego do czerwonego, kropki zamień na kwadraty a do wykresu dodaj tytuł (i odpowiednie etykiety osi).

```
ggplot(koty_ptaki, aes(x=waga, y=predkosc, size=ilosc))
  geom_point()
```

Odpowiedzi, sezon 2, odcinek 4

```
## Podobnie jak w poprzednim odcinku, wybierz samochody marki Volkswagen model Passat a następnie narysuj jak średnia cena zależy od roku produkcji za pomocą geometrii geom_smooth(). Następnie zobacz jak ten wykres będzie wyglądał z motywami theme_bw(), theme_excel() i theme_economist().
```

```

## Passat a następnie narysuj jak średnia cena
## geometrii `geom_smooth()`. Następnie zobacz
## z motywami `theme_bw()`, `theme_excel()` i
## `theme_minimal()`.

passat <- auta2012 %>%
  filter(Marka == 'Volkswagen', Model == 'Passat')
  select(Marka, Model, Rok.produkcji, Cena)

pl <- ggplot(passat, aes(x=Rok.produkcji, y=Cena))
  geom_smooth()

library(ggthemes)
pl + theme_bw() + ggtitle("theme_bw")
pl + theme_excel() + ggtitle("theme_excel")
pl + theme_minimal() + ggtitle("theme_minimal")
pl + theme_economist() + ggtitle("theme_economist")

## Zmień poniższy wykres, zamieniając skalę koła
## kropki zameń na kwadraty a do wykresu dodać
## ggplot(koty_ptaki, aes(x=waga, y=predkosc, size=10))
##     geom_point()

ggplot(koty_ptaki, aes(x=waga, y=predkosc, size=10))
  geom_point(shape=18) +
  scale_color_gradient(high = "green3", low = "red")
  ggtitle("Zielone latające kwadraty") + xlab('Waga') + ylab('Prędkość')

```

Zadanie, sezon 2, odcinek 7

- Wybierz serial *Friday Night Lights* zobacz jak wygląda popularność tego serialu. Czy jest bardziej popularny niż *Breaking Bad*?

- Zobacz jak popularność tego serialu zmieniała się w czasie. Czy zaobserwowałeś coś interesującego?
- Zamiast prezentować informacje o ocenach odcinków przedstaw informacje o liczbie oddanych głosów (popularności) odcinków. Czy te same statystyki nadają się równie dobrze do prezentacji informacji o ocenie i o liczbie głosów?

Odpowiedź, sezon 2, odcinek 7

```
## Wybierz serial *Friday Night Lights* zobacz
## Czy jest bardziej popularny niż *Breaking Ba

FNL <- filter(serialeIMDB, serial == "Friday N:
summary(FNL$glosow)
BB <- filter(serialeIMDB, serial == "Breaking I
summary(BB$glosow)

## Zobacz jak popularność tego serialu zmienia:
## Czy zaobserwowałeś coś interesującego?

plot(FNL$glosow)
```

Zadania, sezon 2, odcinek 8

- Dla każdego wzrostu rodzica ze zbioru danych galton wyznacz o ile średnio niższe są dzieci od rodziców.

- Mając wyznaczone parametry $b0$ i $b1$ policz oceny średniego wzrostu dziecka dla każdego średniego wzrostu rodzica.
- Wyznacz różnice pomiędzy ocenami z modelu liniowego a średnimi licznymi osobno dla każdej grupy rodziców.
- W pakiecie `PogromcyDanych` udostępniony jest również zbiór danych `pearson` zebrany przez Pearsona. W tym zbiorze danych zebrane są wzrosty ojców i synów.

Dla tego zbioru danych wyznacz model regresji liniowej oraz narysuj zbiór danych z zaznaczoną krzywą regresji liniowej.

Czy parametry regresji liniowej w zbiorze `pearson` (z wysokością ojca) różnią się od parametrów ze zbioru `galtona` (ze średnią wysokością rodziców)?

Odpowiedzi, sezon 2, odcinek 8

```
## Dla każdego wzrostu rodzica ze zbioru danych
## o ile średnio niższe są dzieci od rodziców
## wykorzystując funkcje z pakietu dplyr
srednie %>%
  mutate(roznica = srednia - rodzic)
```

```
## Mając wyznaczone parametry _b0_ i _b1_ polic
## dziecka dla każdego średniego wzrostu rodzic
wspolczynniki = coef(lm(syn~rodzic, data=galton)
wspolczynniki

## Wyznacz różnice pomiędzy ocenami z modelu l:
## a średnimi liczymi osobno dla każdej grupy
srednie %>%
  mutate(m.liniowy = wspolczynniki[1] + rodzic,
        roznica = srednia - m.liniowy)
```

Zadanie, sezon 2, odcinek 9

- Znajdź serial o najsilniej rosnącym trendzie dotyczącym ocen (ponieważ oceny są ograniczone z góry przez 10 to może nie być serial o najwyższej średniej ocenie).
- Znajdź serial o najsilniejszym trendzie spadkowym.

Odpowiedź, sezon 2, odcinek 9

```
## Znajdź serial o najsilniej rosnącym trendzie
## (ponieważ oceny są ograniczone z góry przez
## 10 to może nie być serial o najwyższej
## średniej ocenie)

## Znajdź serial o najsilniejszym trendzie spadkowym
## (ponieważ oceny są ograniczone z góry przez
## 10 to może nie być serial o najwyższej
## średniej ocenie)

nazwy_seriali <- levels(serialeIMDB$serial)
wspolczynnikTrendu <- c()

for (ser in nazwy_seriali) {
```

```
jedenSerial <- serialsIMDB %>%
  filter(serial == ser) %>%
  select(id, ocena)
wspolczynnikTrendu[ser] <- lm(ocena~id, jedenSerial)
}
## na początku najsilniejsze spadki, na koniec
sort(wspolczynnikTrendu)
```

Zadanie, sezon 2, odcinek 10

- Znajdź serial o istotnym ujemnym trendem, to znaczy takim, którego oceny maleją.
- Znajdź serial (inny niż *Breaking Bad*) o istotnie dodatnim trendzie.
- [Trudne] Policz dla ilu serialu ich oceny rosną a dla ilu maleją i te wzrosty lub spadki są istotnie większe niż wynikające z przypadku.

Odpowiedź, sezon 2, odcinek 10

```
## Znajdź serial o istotnym ujemnym trendzie, taki jak
## Taki serial można znaleźć używając aplikacji
## https://smarterpoland.shinyapps.io/serialsIMDB/
## lub pisząc pętlę iterującą po serialach
## Znajdź serial (inny niż _Breaking Bad_) o istotnie dodatnim trendzie, taki jak
```

```
## Taki serial można znaleźć używając aplikacji:  
## https://smarterpoland.shinyapps.io/serialiII  
## lub pisząc pętlę iterującą po serialach
```

Zadania, sezon 2, odcinek 11

- Wykonaj modelowanie z użyciem modelu multiplikatywnego dla innej marki.
- Jeżeli model multiplikatywny wygląda na uzasadniony, policz ile procent corocznie traci na cenie auto danej marki.
- Zobacz jak wygląda zależność pomiędzy ceną a wiekiem dla innych marek.
- Wykonaj modelowanie z użyciem łamanej regresji dla innych marek.

Odpowiedzi, sezon 2, odcinek 11

```
toyota <- auta2012 %>%  
  filter(Marka == "Toyota") %>%  
  mutate(Wiek = 2012 - Rok.produkcji) %>%  
  filter(Wiek < 20)  
  
## Trend liniowy addytywny dla Toyota  
lm(Cena.w.PLN ~ Wiek, data=toyota)$coef
```

```
ggplot(toyota, aes(y=Cena.w.PLN, x = Wiek)) +
  geom_point() +
  geom_smooth(method="lm", col="red", size=2) +
  ylim(0,400000)

## Trend mnożnikowy dla Toyota
lm(log(Cena.w.PLN) ~ Wiek, data=toyota)$coef

ggplot(toyota, aes(y=Cena.w.PLN, x = Wiek)) +
  geom_point() +
  geom_smooth(method="lm", col="red", size=2) +
  ylim(0,400000) + scale_y_log10()

#
## Krzywe łamanej regresji dla wszystkich marek

marki <- names(head(rev(sort(table(auta2012$Marka))
invisibl(sapply(marki, function(marka) {
  wybranamarka <- auta2012 %>%
    filter(Marka == marka) %>%
    mutate(Wiek = 2012 - Rok.produkcji) %>%
    filter(Wiek < 20,
           Przebieg.w.km < 400000) %>%
    mutate(Wiek0 = ifelse(Wiek >= 0, Wiek, 0),
           Wiek5 = ifelse(Wiek >= 5, Wiek - 5, 0))

M1 = lm(Przebieg.w.km ~ Wiek0 + Wiek5, data=topred)
M2 = lm(Przebieg.w.km ~ Wiek0 + Wiek5 - 1, data=topred)

topred = data.frame(Wiek0 = c(0,5,20), Wiek5 = c(0,5,20))
topred$M1 = predict(M1, newdata = topred)
topred$M2 = predict(M2, newdata = topred)

plot(ggplot(wybranamarka, aes(y=Przebieg.w.km, x=Wiek)) +
  geom_point() +
```

```

    geom_smooth(method="lm", col="red", size=2)
    geom_line(data= topred, aes(y=M1, x = Wiek))
    geom_line(data= topred, aes(y=M2, x = Wiek))
    ggttitle(marka))
}

marki <- names(head(rev(sort(table(auta2012$Marka))))
sapply(marki, function(marka) {
  wybranamarka <- auta2012 %>%
    filter(Marka == marka) %>%
    mutate(Wiek = 2012 - Rok.produkcji) %>%
    filter(Wiek < 20) %>%
    mutate(logCena.w.PLN = log10(Cena.w.PLN))

  plot(ggplot(wybranamarka, aes(y=Cena.w.PLN, x=Wiek),
    geom_point() +
    geom_smooth(method="lm", col="red", size=2)
    ggttitle(marka) + scale_y_continuous(trans='log'))
wsp <- lm(logCena.w.PLN ~ Wiek, data=wybranamarka)
procent <- 1- 10^(wsp[2])
round(100*procent,1)
}
)

```

Zadania, sezon 2, odcinek 13

- W zbiorze danych `diagnoza` wiek respondenta jest w zmiennej o nazwie `wiek2013`. Wyznacz zależność pomiędzy wiekiem podzielonym na dwie grupy, poniżej i powyżej 30 roku życia, a odpowiedziami na

pytanie o to co w życiu ważne.

- W zbiorze danych TCGA_BRCA w drugiej kolumnie jest płeć pacjenta. Wyznacz zależność pomiędzy zmiennymi plec a czy.zyje. Przetestuj tą zależność testem Fishera.

W zbiorze danych TCGA_BRCA zbadaj czy jest i jaka jest zależność pomiędzy

- przeżyciami (kolumna czy.zyje a mutacjami TP53),
- płcią a mutacjami TP53,
- płcią a zmienną czy.zyje.

Zależność przedstaw za pomocą tabeli z liczebnościami oraz częstotliwościami.

Zadania, sezon 2, odcinek 14

```
## W zbiorze danych `diagnoza` wiek respondentów
## Wyznacz zależność pomiędzy wiekiem podzielony
## na trzy grupy (0, 30, 100)
tab <- table(diagnoza$gp29,
             cut(diagnoza$wiek2013, c(0, 30, 100)))
tab
prop.table(tab, 2)
chisq.test(tab)

## W zbiorze danych `TCGA_BRCA` w drugiej kolumnie
## Wyznacz zależność pomiędzy zmiennymi `plec`
## Przetestuj tą zależność testem Fishera.
```

```
table(TCGA_BRCA$czy.zyje,  
      TCGA_BRCA$TP53)  
fisher.test(table(TCGA_BRCA$czy.zyje,  
                 TCGA_BRCA$TP53), 2)
```