

# Statystyka

Rafał Szyński 259380, Kajetan Leszak 259321

2024-06-05

## 1 Wstęp

Do wykonywania projektu będziemy używać dwóch bibliotek:

- *ggplot2* - biblioteka do rysowania wykresów.
- *dplyr* - biblioteka do manipulowania danymi (np. filtrowanie, grupowanie itp.).

```
library(ggplot2)
library(dplyr)
```

Uwaga: Jeśli komendy nie działają należy pobrać poszczególne biblioteki używając komendy `install.packages("packageName")` w konsoli.

## 2 Opis baza danych

Baza **credit\_card.xls** pochodzi z eportalu. Zawiera ona dane o użytkownikach kart kredytowych oraz wykonywanych przez nich transakcjach.

Baza posiada 26280 rekordów opisane przez 13 kolumn, które mówią nam o:

- *custid* - id indywidualnego klienta.
- *date\_birth* - data urodzenie danego klienta.
- *birth\_year* - rok urodzenia danego klienta.
- *gender* - płeć danego klienta (dostępne opcje: Female, Male).
- *card* - typ używanej karty kredytowej (dostępne opcje: Mastercard, Visa, American Express, Discover, Other).
- *card\_data* - data utworzenia karty kredytowej.
- *card\_year* - rok utworzenia karty kredytowej.
- *month* - miesiąc w którym karta została użyta (dostępne opcje: January, February, March, April, May, June, July, August, September, October, November, December).
- *quarter* - kwartał w którym karta została użyta (dostępne opcje: Q1, Q2, Q3, Q4).
- *year* - rok w którym karta została użyta.
- *type\_trans* - rodzaj dobra, które zostało zakupione (dostępne opcje: Entertainment, Grocery, Retail, Travel, Other).
- *items* - ilość kupionego dobra.
- *spent* - wartość kupionego dobra.

```
data <- read.csv2("credit_card.xls");
dim(data) # Rozmiary bazy danych [wiersze x kolumny]
```

```
## [1] 26280    13
```

```
colnames(data) # Wypisanie nazw kolumn
```

```
## [1] "custid"      "date_birth"  "birth_year"  "gender"      "card"
```

```
## [6] "card_date" "card_year" "month"      "quarter"    "year"
## [11] "type_trans" "items"      "spent"
```

```
summary(data) # Podstawowe statystyki z każdej kolumny
```

```
##      custid      date_birth      birth_year      gender
## Length:26280      Length:26280      Min.   :1929      Length:26280
## Class :character      Class :character      1st Qu.:1946      Class :character
## Mode  :character      Mode  :character      Median :1960      Mode  :character
##                                     Mean   :1960
##                                     3rd Qu.:1975
##                                     Max.   :1990
##      card      card_date      card_year      month
## Length:26280      Length:26280      Min.   :1991      Length:26280
## Class :character      Class :character      1st Qu.:1999      Class :character
## Mode  :character      Mode  :character      Median :2002      Mode  :character
##                                     Mean   :2002
##                                     3rd Qu.:2005
##                                     Max.   :2009
##      quarter      year      type_trans      items
## Length:26280      Min.   :2007      Length:26280      Min.   : 0.000
## Class :character      1st Qu.:2007      Class :character      1st Qu.: 0.000
## Mode  :character      Median :2008      Mode  :character      Median : 2.000
##                                     Mean   :2008
##                                     3rd Qu.:2008
##                                     Max.   :2008
##                                     Mean   : 2.359
##                                     3rd Qu.: 4.000
##                                     Max.   :13.000
##      spent
## Min.   : 0.0
## 1st Qu.: 0.0
## Median :141.8
## Mean   :196.3
## 3rd Qu.:311.3
## Max.   :1439.4
```

```
glimpse(data) # Przykładowe dane, które występują w każdej kolumnie
```

```
## Rows: 26,280
## Columns: 13
## $ custid      <chr> "8257-BKBEDP-MRF", "8257-BKBEDP-MRF", "8257-BKBEDP-MRF", "8~
## $ date_birth <chr> "12/15/1961", "12/15/1961", "12/15/1961", "12/15/1961", "12~
## $ birth_year <int> 1961, 1961, 1961, 1961, 1961, 1961, 1961, 1961, 1961, 1961,~
## $ gender      <chr> "Female", "Female", "Female", "Female", "Female", "Female",~
## $ card        <chr> "Mastercard", "Mastercard", "Mastercard", "Mastercard", "Ma~
## $ card_date   <chr> "8/9/2003", "8/9/2003", "8/9/2003", "8/9/2003", "8/9/2003",~
## $ card_year   <int> 2003, 2003, 2003, 2003, 2003, 2003, 2003, 2003, 2003, 2003,~
## $ month       <chr> "January", "January", "January", "January", "January", "Jan~
## $ quarter     <chr> "Q1", "Q1", "Q1", "Q1", "Q1", "Q1", "Q1", "Q1", "Q1", "Q1",~
## $ year        <int> 2007, 2007, 2007, 2007, 2007, 2008, 2008, 2008, 2008, 2008,~
## $ type_trans  <chr> "Grocery", "Retail", "Entertainment", "Travel", "Other", "G~
## $ items       <int> 2, 9, 1, 3, 8, 5, 10, 0, 1, 3, 5, 9, 0, 1, 3, 0, 9, 0, 4, 4~
## $ spent       <dbl> 167.81, 809.87, 111.09, 579.10, 409.63, 281.34, 1011.05, 0.~
```

### 3 Wyliczenie podstawowych statystyk

Do obliczenia podstawowych statystyk używa się funkcji `summary()`, która wylicza:

- *Min.* - Wartość minimalną.
- *1st Qu.* - Wartość pierwszego kwartyłu (25% wyników jest poniżej tej wartości).
- *Median* - Wartość mediany.
- *Mean* - Wartość średnia.
- *3rd Qu.* - Wartość trzeciego kwartyłu.
- *Max.* - Wartość maksymalną (75% wyników jest poniżej tej wartości).

```
summary(data$items) # Podstawowe statystyki dla kolumny items
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.000  0.000   2.000   2.359   4.000   13.000
```

```
summary(data$spent) # Podstawowe statystyki dla kolumny spent
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      0.0      0.0   141.8   196.3   311.3   1439.4
```

Interpretacja wyników:

- Pierwszy kwartył jest równy zero dla obu przypadków co oznacza że więcej niż 25% wyników jest równa zero.

## 4 Wykresy

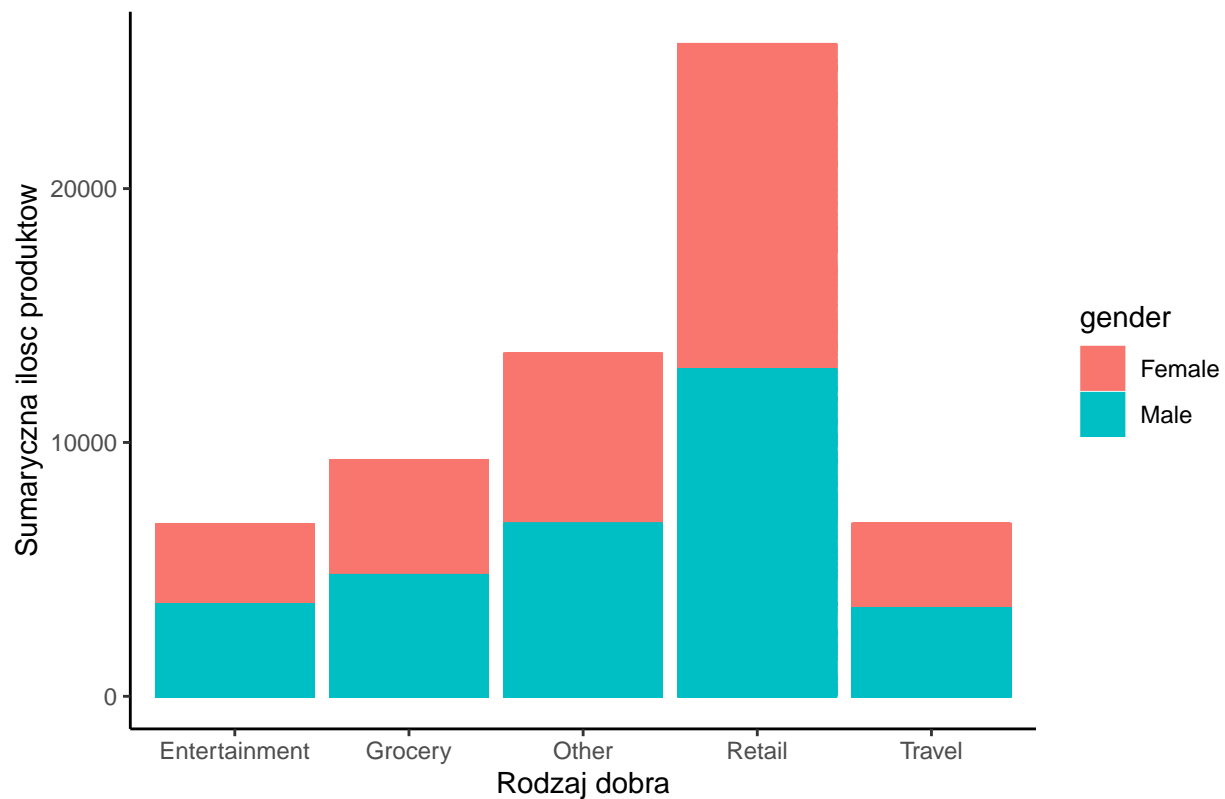
### 4.1 Wykres słupkowy

**Problem:**

Jak dużo konkretnego dobra (z kolumny *type\_trans*) jest kupowane w zależności od płci.

```
ggplot() + # Podstawa do rysowania wykresu
  geom_bar( # Wykres słupkowy
    data=data, # Używane dane do rysowania
    # Określanie jakie dane są na konkretnej osi
    # (x - typ dobra, y - sumaryczna ilość,
    # color i fill = podział względem płci)
    aes(x=type_trans, y=items, color=gender, fill=gender),
    stat="identity" # Zlicza sumaryczną ilość dobra
  ) +
  labs( # Podpisy na wykresie
    title="Wykres słupkowy dla zakupu rodzaju dobra w zależności od płci",
    x="Rodzaj dobra",
    y="Sumaryczna ilość produktów"
  ) +
  theme_classic() # Ustawianie klasycznego wyglądu wykresu
```

Wykres słupkowy dla zakupu rodzaju dobra w zależności od płci



#### Interpretacja wyników:

- Kobiety kupują więcej dóbr niż mężczyźni.
- Najwięcej transakcji występuje w sprzedaży detalicznej.
- Najmniej transakcji jest na podróże.

## 4.2 Wykres liniowy

### Problem:

Jaki jest sumaryczny wydatek danego użytkownika (8257-BKBEDP-MRF) względem czasu (podział na rok i miesiąc).

```
# Filtruujemy wszystkie dane pierwszego użytkownika
user_data <- data[data$custid == "8257-BKBEDP-MRF",]

month_numeric <- c("January",
                   "February",
                   "March",
                   "April",
                   "May",
                   "June",
                   "July",
                   "August",
                   "September",
                   "October",
                   "November",
                   "December")
```

```

# Zamiana miesiąca z słowa na liczbę np. January=1
month <- match(user_data$month, month_numeric)

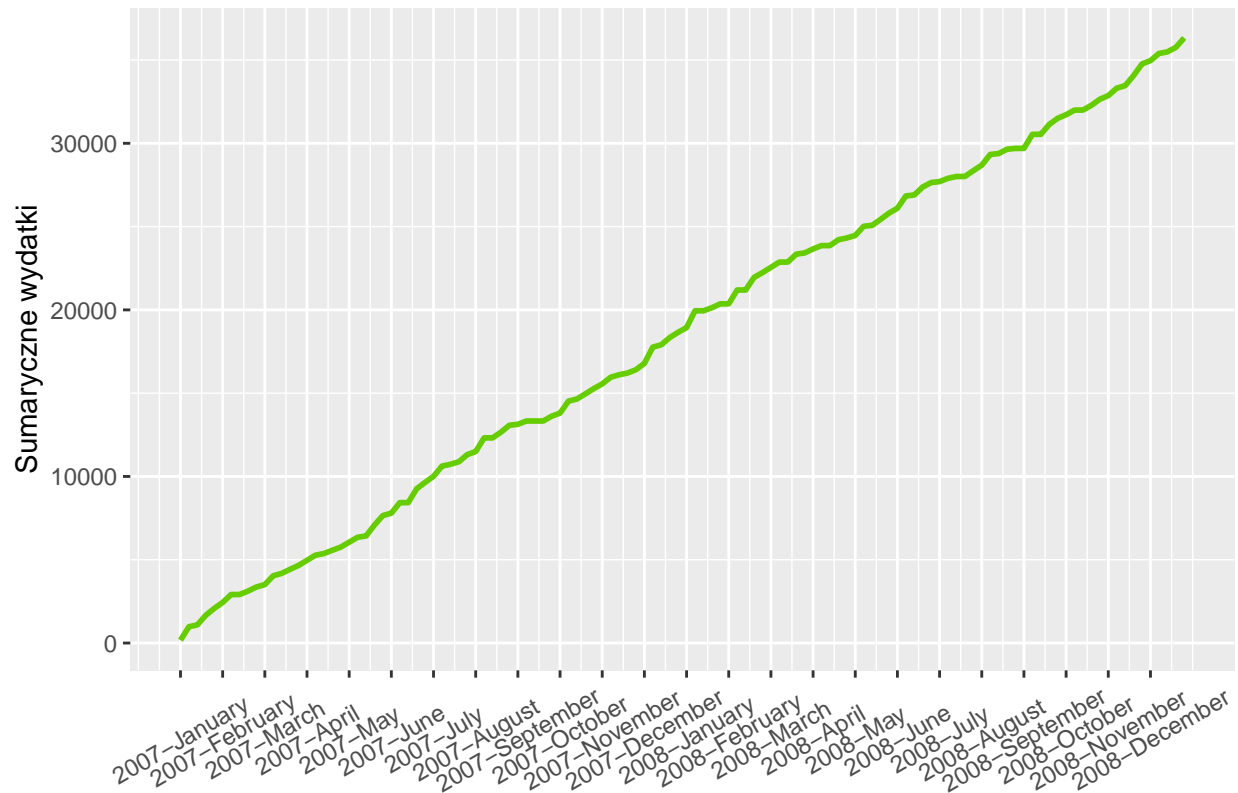
user_date_spent = data.frame(
  year = user_data$year,
  month = month,
  spent = user_data$spent
)

# Sortowanie po roku i miesiącu
sorted_user <- user_date_spent[order(user_date_spent$year, user_date_spent$month),]
# Sumaryczny wektor wydatków
sorted_user$spent <- cumsum(sorted_user$spent)
data_length <- length(sorted_user$spent)

ggplot() +
  geom_line( # Wykres liniowy
    data=sorted_user,
    aes(x = seq(from=1, to=data_length), y = spent),
    color = "chartreuse3",
    linewidth = 1
  ) +
  scale_x_continuous(
    breaks=seq(from=1, to=data_length, by=5),
    labels=c(
      paste(rep(2007, 12),
        month_numeric,
        sep="-"),
      paste(rep(2008, 12),
        month_numeric,
        sep="-"))
    ) +
  labs(
    title = "Wydatki użytkownika 8257-BKBEDP-MRF względem czasu",
    x = NULL,
    y = "Sumaryczne wydatki"
  ) +
  theme(axis.text.x = element_text(angle = 30, hjust = 0.5, vjust = 0.5))

```

### Wydatki użytkownika 8257–BKBEDP–MRF względem czasu



#### Interpretacja wyników:

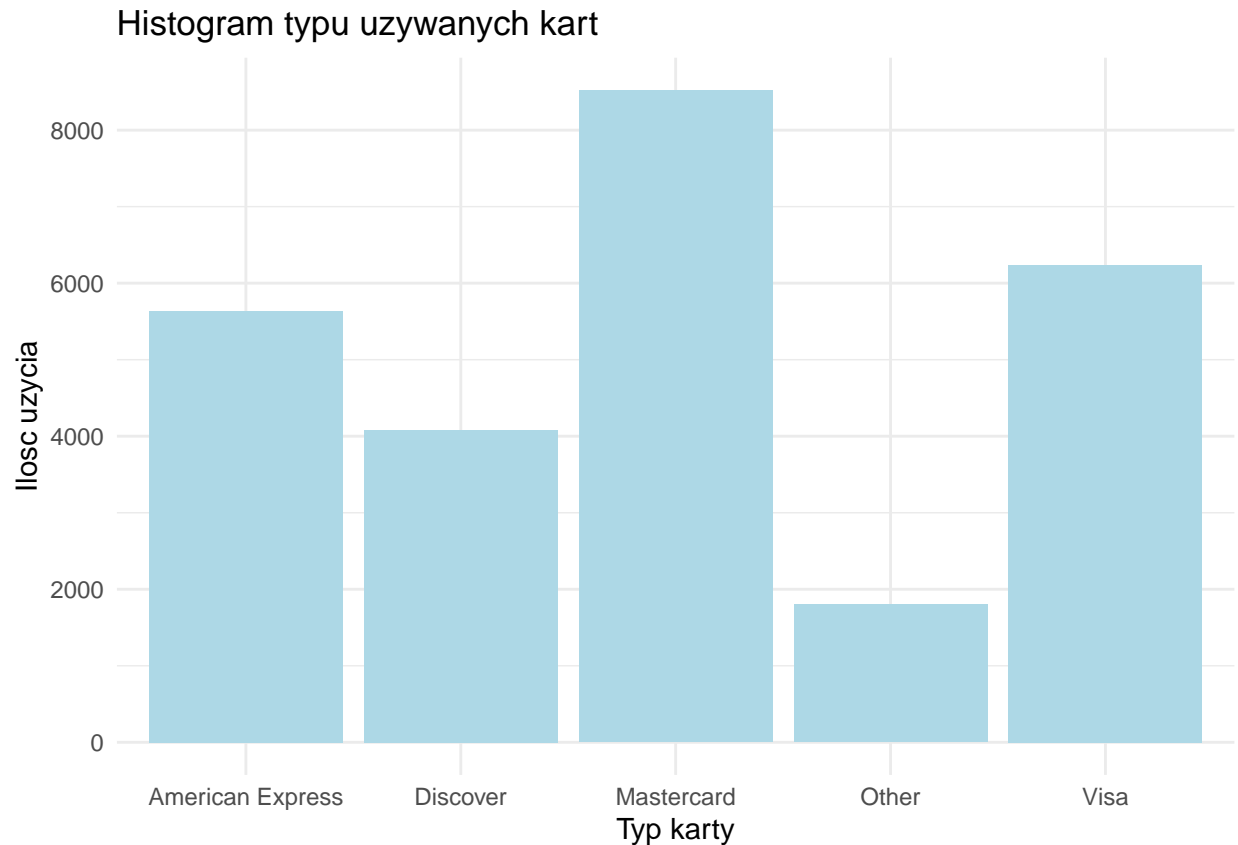
- Użytkownik sumarycznie wydał 36330.45.
- Użytkownik używał karty tylko przez 2 lata.
- Wydatki użytkownika są w miarę stałe (wykres ten jest używany do regresji linowej [ostatni podpunkt projektu], w którym możemy sprawdzić jak bardzo wydatki odstają od stałych).
- Robiąc pochodną wykresu, można określić miesiąc w którym użytkownik wydał najwięcej:  $\max(\frac{d}{dx}f(x))$ .

## 4.3 Histogram

### Problem:

Jaki typ karty jest najczęściej używany.

```
ggplot() +  
  geom_histogram( # Histogram  
    data=data,  
    aes(x=card),  
    stat="count", # Zliczanie wystąpień  
    fill="lightblue") +  
  labs(  
    title = "Histogram typu używanych kart",  
    x = "Typ karty",  
    y = "Ilość użycia"  
  ) +  
  theme_minimal() # Motyw minimalistyczny
```



#### Interpretacja wyników:

- Najczęściej używaną kartą jest Mastercard.

#### 4.4 Inne wykresy

Wykres gęstości i pudełko-wąsy są używane w dalszej części projektu.

### 5 Obserwacje odstające

Obserwacje odstające to punkty danych, które znacząco różnią się od innych obserwacji w zestawie danych.

#### Problem:

- Wyznacz dane odstające w wydatkach dla osób urodzonych w 1929.
- Pokaż dane odstające w wydatkach dla każdego wieku użytkownika.

#### 5.1 Wykres pudełko-wąsy

Wykres pudełko-wąsy składa się z kilku kluczowych elementów, które pomagają wizualizować różne aspekty zestawu danych, t.j.:

- *Mediana* - Linia wewnątrz pudełka, która przedstawia środkową wartość danych.
- *Pudełko* - Prostokąt, który rozciąga się od pierwszego kwartyła ( $Q1$ ) do trzeciego kwartyła ( $Q3$ ). Obejmuje środkowe 50% danych.
- *Wąsy* - Linie wychodzące z pudełka, które sięgają do najmniejszej i największej wartości w obrębie zasięgu  $Q1 - 1.5 \cdot IQR$  i  $Q3 + 1.5 \cdot IQR$ , gdzie  $IQR$  to rozstęp między kwartyłowy ( $Q3 - Q1$ ).

- *Obserwacje odstające* - Punkty znajdujące się poza wąsami, które są wartościami ekstremalnymi w zestawie danych (dane które będziemy wyznaczać w tym zadaniu).

```
# Wszystkie wydatki osób urodzonych w 1929
spent_1929 <- data[data$birth_year == 1929,]$spent

q1 <- quantile(spent_1929, 0.25) # Pierwszy kwartył
q3 <- quantile(spent_1929, 0.75) # Ostatni kwartył

# Wartość dolnego "wąsa"
lower_whisker <- max(min(spent_1929), q1 - 1.5 * (q3 - q1))
# Wartość górnego "wąsa"
upper_whisker <- min(max(spent_1929), q3 + 1.5 * (q3 - q1))

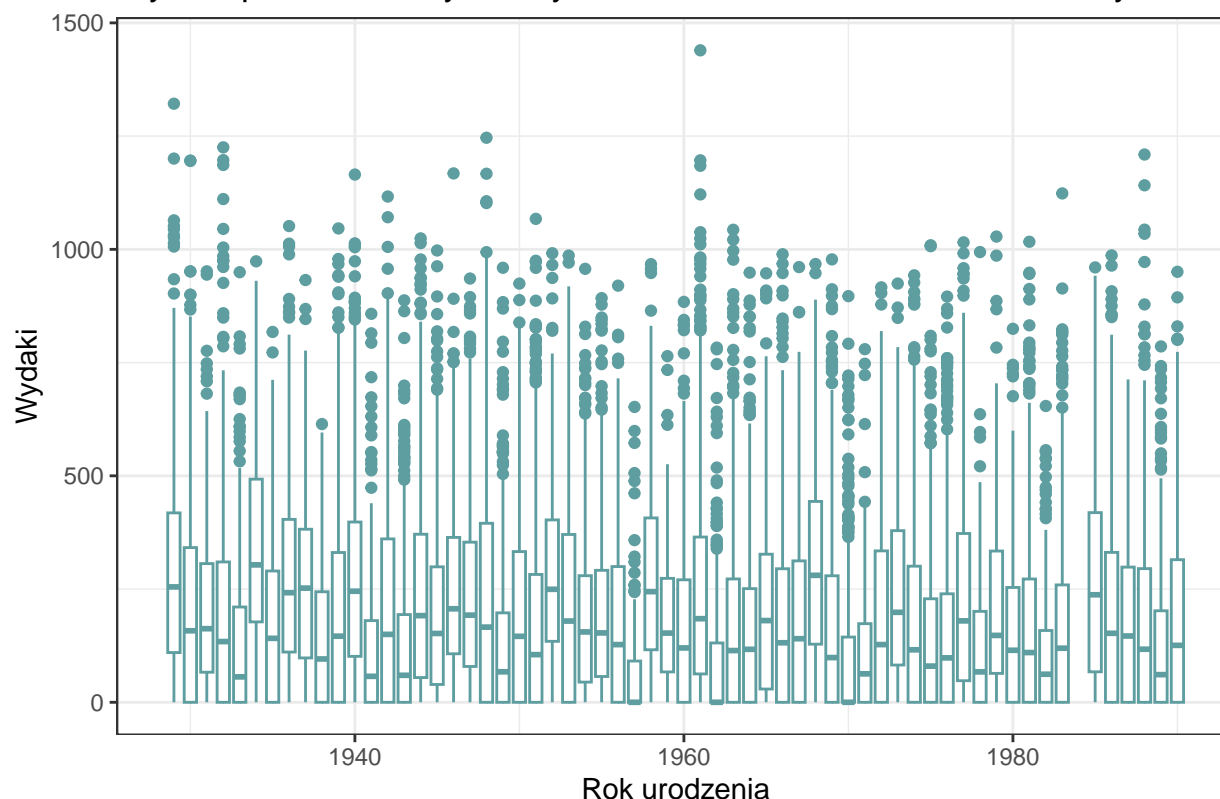
# Obserwacje odstające
outliers_1929 <- spent_1929[spent_1929 > upper_whisker |
                             spent_1929 < lower_whisker]
outliers_1929

## [1] 1044.57 1026.36 934.00 1031.21 1013.86 1006.19 1051.57 1321.55 902.44
## [10] 1064.00 1200.39

ggplot() +
  geom_boxplot( # Wykres pudełko-wąsy
    data=data,
    aes(x=birth_year, y=spent, group=birth_year),
    color="cadetblue"
  ) +
  labs(
    title = "Wykres pudełko-wasy dla wydatków w zależności od wieku osoby",
    x = "Rok urodzenia",
    y = "Wydatki",
  ) +
  theme_bw()
```



Wykres pudełko–wasy dla wydatków w zależności od wieku osoby



#### Interpretacja wyników:

- Dla osób urodzonych w 1929 wartość obserwacji dostających zaczyna się od 880.82 i jest ich 11.
- Nie ma wyników odstających które są mniejsze niż 0.0, ponieważ pierwszy kwartyl jest równy zero.
- Większość wydatków jest mniejsza niż 500.

## 5.2 Odchylenie standardowe

Wykres gęstości pokazuje, gdzie wartości są najbardziej skoncentrowane. Obszar pod krzywą jest równy 1, co oznacza, że wykres gęstości przedstawia rozkład prawdopodobieństwa danej zmiennej.

Obserwacje mogą być uznane za odstające, jeśli znajdują się poza określoną liczbą odchyłeń standardowych od średniej. Na przykład, dane poza granicami  $\mu \pm 3\sigma$  są często uznawane za odstające, ponieważ poza tą wartością znajdują się około 0.1 danych.

```
# Wszystkie wydatki osób urodzonych w 1929
spent_1929 <- data[data$birth_year == 1929,]$spent
mean_spent_1929 <- mean(spent_1929) # Średnia
sd_spent_1929 <- sd(spent_1929) # Odchylenie standardowe

# Funkcja gęstości na podstawie wydatków
norm_spent_1929 <- dnorm(spent_1929, mean_spent_1929, sd_spent_1929)

# Górna granica
upper_threshold <- mean_spent_1929 + 3 * sd_spent_1929
lower_threshold <- mean_spent_1929 - 3 * sd_spent_1929
outliers_1929 <- spent_1929[spent_1929 > upper_threshold |
                             spent_1929 < lower_threshold]
```

```
mean_spent_1929 + sd_spent_1929
```

```
## [1] 535.8646
```

```
mean_spent_1929 - sd_spent_1929
```

```
## [1] 48.30621
```

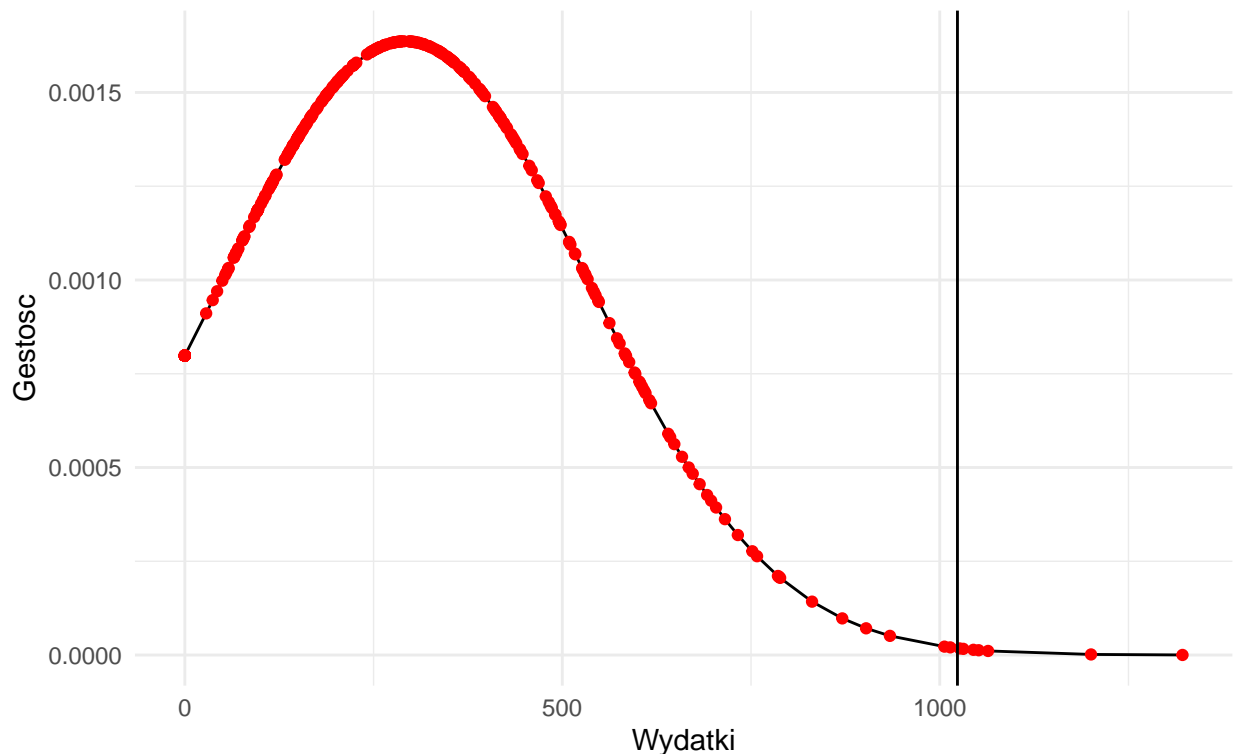
```
outliers_1929
```

```
## [1] 1044.57 1026.36 1031.21 1051.57 1321.55 1064.00 1200.39
```

```
ggplot() +  
  geom_line(data=data.frame(x=spent_1929, y=norm_spent_1929), aes(x, y)) +  
  geom_point(data=data.frame(x=spent_1929, y=norm_spent_1929), aes(x, y), color="red") +  
  geom_vline(xintercept=upper_threshold) +  
  labs(  
    title = "Wykres gestosci wydatkow ludzi urodzonych w 1929",  
    subtitle = sprintf("Linia okreslajaca wartosci odstajace: x=%.02f", upper_threshold),  
    x = "Wydatki",  
    y = "Gestosc"  
  ) +  
  theme_minimal()
```

## Wykres gestosci wydatkow ludzi urodzonych w 1929

Linia okreslajaca wartosci odstajace: x=1023.42



### Interpretacja wyników:

- Dla osób urodzonych w 1929 wartość obserwacji dostających zaczynają się od 1026.36 i jest ich 7.
- Nie ma wartości dostających mniejszych od 0 dla osób urodzonych w 1929.
- Średnia wydatków dla osób urodzonych w 1929 wynosi 292.0854.

- Około 66% wartości wydatków dla osób urodzonych w 1929 znajdują się pomiędzy wartościami 48.3 a 535.86 ( $\mu \pm \sigma$ ).

## 6 Wyliczenie prawdopodobieństwa dla zmiennej

### 6.1 Gerenowanie prób losowych

```
x <- sort(data$birth_year) # Posortowane dane roku urodzenia
mean_x <- mean(x)
sd_x <- sd(x)

# Wartości funkcji gęstości (ciągła)
continuous_dnorm <- dnorm(x, mean_x, sd_x)
# Wartości dystrybucyjności (ciągła)
continuous_pnorm <- pnorm(x, mean_x, sd_x)

# Wartości funkcji gęstości (dyskretna)
discreet_dbinom <- dbinom(x, length(x), mean_x / length(x))
# Wartości dystrybucyjności (dyskretna)
discreet_pbinom <- pbinom(x, length(x), mean_x / length(x))
```

### 6.2 Obliczanie prawdopodobieństwa punktowego i przedziałowego

TODO: do poprawy Prawdopodobieństwo punktowe powinno być równe 0 gdyż wykres pod wykresem w danych punkcie jest równe 0 dla wykresu ciągłego. Jeśli chcemy wyznaczyć prawdopodobieństwo punktowe należy skorzystać z wykresu dyskretnego, które będzie największym przybliżeniem wartości.

```
x_point <- 1969
n <- tail(which(x == x_point), 1)

# Prawdopodobieństwo punktowe P(X = 1929)
continuous_dnorm[n]

## [1] 0.02009985
discreet_dbinom[n]

## [1] 0.009139813
# Prawdopodobieństwo przedziałowe P(X <= 1929)
continuous_pnorm[n]

## [1] 0.6987881
discreet_pbinom[n]

## [1] 0.5897367
```

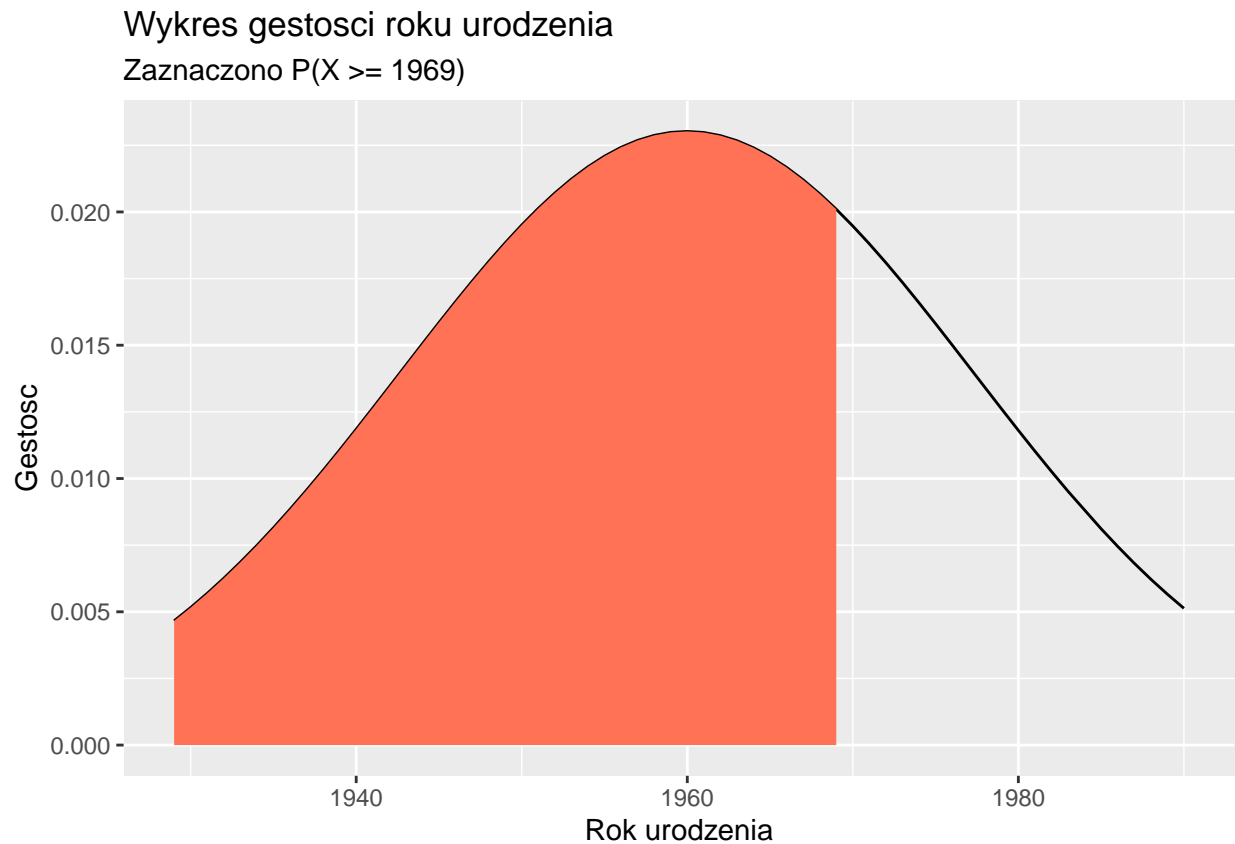
### 6.3 Wykres ciągły

```
ggplot() +
  geom_line(data=data.frame(x=x, y=continuous_dnorm), aes(x, y)) +
  geom_polygon(
    data=data.frame(
      x=c(min(x), head(x, n), x_point),
      y=c(0, head(continuous_dnorm, n), 0)),
```

```

aes(x, y),
fill = "coral1"
) +
labs(
  title = "Wykres gestosci roku urodzenia",
  subtitle = sprintf("Zaznaczono P(X >= %i)", x_point),
  x = "Rok urodzenia",
  y = "Gestosc"
)

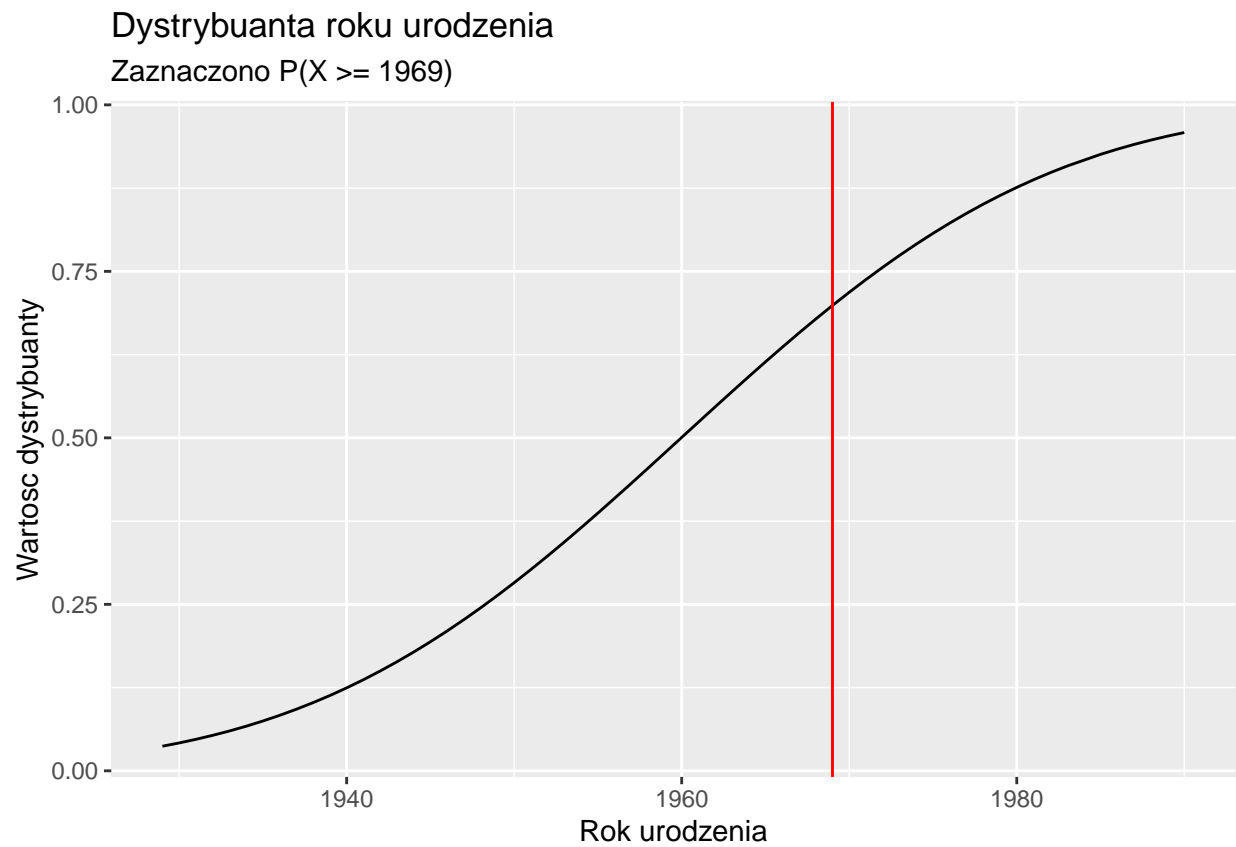
```



```

ggplot() +
  geom_line(data=data.frame(x=x, y=continuous_pnorm), aes(x, y)) +
  geom_vline(xintercept = x_point, color="red") +
  labs(
    title = "Dystrybuanta roku urodzenia",
    subtitle = sprintf("Zaznaczono P(X >= %i)", x_point),
    x = "Rok urodzenia",
    y = "Wartosc dystrybuanty"
  )

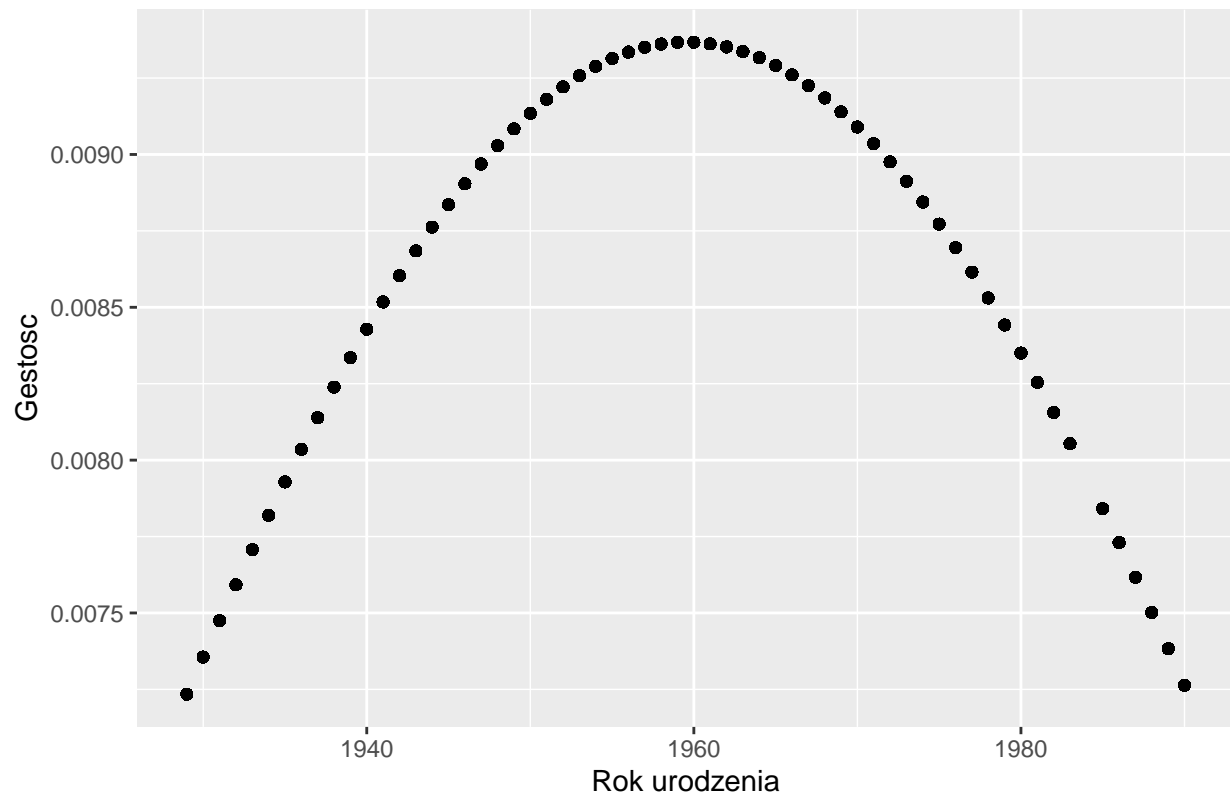
```



## 6.4 Wykres dyskretny

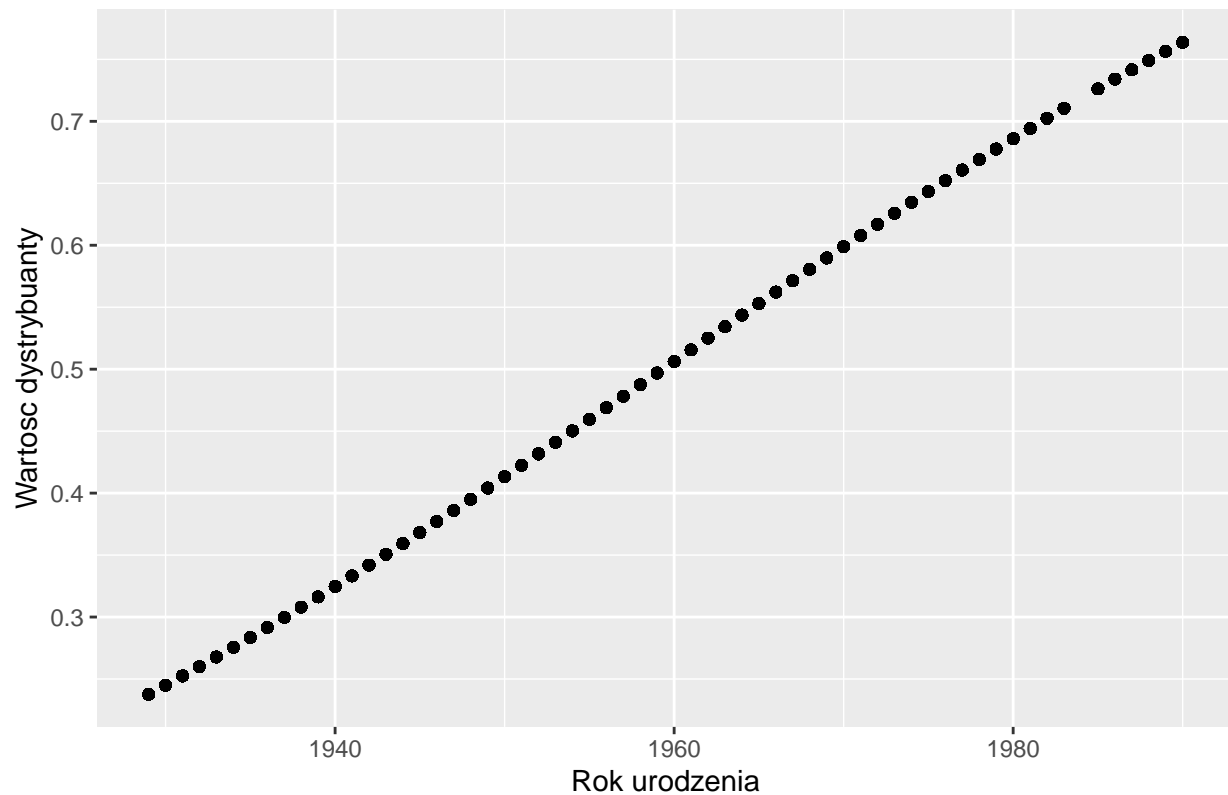
```
ggplot() +
  geom_point(data=data.frame(x=x, y=discreet_dbinom), aes(x, y)) +
  labs(
    title = "Rozkład dyskretny funkcji gęstości dla roku urodzenia",
    x = "Rok urodzenia",
    y = "Gęstość"
  )
```

Rozkład dyskretny funkcji gestosci dla roku urodzenia



```
ggplot() +  
  geom_point(data=data.frame(x=x, y=discreet_pbinom), aes(x, y)) +  
  labs(  
    title = "Rozkład dyskretny dystrybucyj dla roku urodzenia",  
    x = "Rok urodzenia",  
    y = "Wartość dystrybucyj"  
  )
```

## Rozkład dyskretny dystrybucyj dla roku urodzenia



## 7 Macierz

```
matrix <- matrix(data$card_year) %>% cbind(data$items) %>% cbind(data$spent)
```

```
matrix_data = list(  
  dimension = dim(matrix),  
  number_of_row = nrow(matrix),  
  number_of_column = ncol(matrix),  
  sum_of_columns = colSums(matrix),  
  sum_of_first_two_row = rowSums(matrix[1:2,]),  
  sum_of_all_elems = sum(matrix))
```

```
matrix_data
```

```
## $dimension  
## [1] 26280      3  
##  
## $number_of_row  
## [1] 26280  
##  
## $number_of_column  
## [1] 3  
##  
## $sum_of_columns  
## [1] 52603560    61990  5157512
```

```
##
## $sum_of_first_two_row
## [1] 2172.81 2821.87
##
## $sum_of_all_elems
## [1] 57823062
```

## 8 Przedziały ufności

### 8.1 Zmienna numeryczna

```
x <- data$items
n <- length(x)
alpha <- 0.01
z <- qnorm(1 - alpha / 2)

x_mean <- mean(x)
x_sd <- sd(x)
x_dnorm <- dnorm(x, x_mean, x_sd)

lower_bound <- x_mean - (z * x_sd / sqrt(n))
upper_bound <- x_mean + (z * x_sd / sqrt(n))

lower_bound

## [1] 2.317891
upper_bound

## [1] 2.399765
```

### 8.2 Zmienna jakościowa

Przedział ufności Walda

```
cards_data <- data %>% group_by(card) %>% summarise(count = n())
n <- length(data$card)
p <- cards_data[cards_data$card == "Mastercard",]$count / n

alpha <- 0.001
z <- qnorm(1 - alpha / 2)

lower_bound <- p - z * sqrt(p * (1 - p) / n)
upper_bound <- p + z * sqrt(p * (1 - p) / n)

lower_bound

## [1] 0.3146999
upper_bound

## [1] 0.3337019
```



## 9 Hipotezy

### 9.1 Test parametryczny - średnia urodzenia to 1960

```
birth_year_data <- data$birth_year

t.test(birth_year_data, mu = 1960)

##
## One Sample t-test
##
## data: birth_year_data
## t = -0.25629, df = 26279, p-value = 0.7977
## alternative hypothesis: true mean is not equal to 1960
## 95 percent confidence interval:
## 1959.763 1960.182
## sample estimates:
## mean of x
## 1959.973
```

### 9.2 Test parametryczny - ludzie urodzeni w 1960 wydają więcej niż ludzie urodzeni 1970

```
spent_1960 = data[data$birth_year == 1960,]$spent
spent_1970 = data[data$birth_year == 1970,]$spent

t.test(spent_1960, spent_1970)

##
## Welch Two Sample t-test
##
## data: spent_1960 and spent_1970
## t = 5.897, df = 797.21, p-value = 5.464e-09
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## 45.81122 91.52689
## sample estimates:
## mean of x mean of y
## 168.56717 99.89811
```

### 9.3 Test nieparametryczny 1

### 9.4 Test nieparametryczny 2

## 10 Regresja liniowa

### 10.1 Przygotowanie danych

Powtórzony kod z wykresu liniowego

```
user_data <- data[data$custid == "8257-BKBEDP-MRF",]

month_numeric <- c("January", "February", "March", "April", "May", "June", "July", "August", "September")

month <- match(user_data$month, month_numeric)
```

```

user_date_spent = data.frame(
  year = user_data$year,
  month = month,
  spent = user_data$spent
)

sorted_user <- user_date_spent[order(user_date_spent$year, user_date_spent$month),]

sorted_user$spent <- cumsum(sorted_user$spent)
data_length = length(sorted_user$spent)

# Obliczanie regresji liniowej
x <- seq(from=1, to=data_length)
model <- lm(sorted_user$spent ~ x)
y <- model$coefficients[2] * x + model$coefficients[1]

```

## 10.2 Wykres

```

ggplot() +
  geom_point(
    data=sorted_user,
    aes(x = seq(from=1, to=data_length), y = spent),
    size = 0.7
  ) +
  geom_line(
    data=data.frame(x=x, y=y),
    aes(x = x, y = y),
    color = "blue"
  ) +
  scale_x_continuous(
    breaks= seq(from=1, to=data_length, by=5),
    labels=c(paste(rep(2007, 12), month_numeric, sep="-"), paste(rep(2008, 12), month_numeric, sep="-"))
  ) +
  labs(
    title = "Wydatki użytkownika 8257-BKBEDP-MRF\newzględem czasu (regresja liniowa)",
    subtitle = sprintf("A = %.02f, B = %.02f", model$coefficients[2], model$coefficients[1]),
    x = NULL,
    y = "Summaryczne wydatki"
  ) +
  theme(axis.text.x = element_text(angle = 30, hjust = 0.5, vjust = 0.5))

```

Wydatki użytkownika 8257-BKBEDP-MRF  
względem czasu (regresja liniowa)

$A = 297.09$ ,  $B = 523.09$

