Statystyka

Rafał Szyiński 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, Febuary, 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, Trabel, 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                      Mean   : 2.359
##                     3rd Qu.:2008                      3rd Qu.: 4.000
##                     Max.   :2008                      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 kwartylu (25% wyników jest poniżej tej wartości)
- *Median* - Wartość mediany
- *Mean* - Wartość średnia
- *3rd Qu.* - Wartość trzeciego kwartylu
- *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 kwartyl jest równy zero dla obu przypadków co oznacza że więcej niż 25% wyników jest równa zero

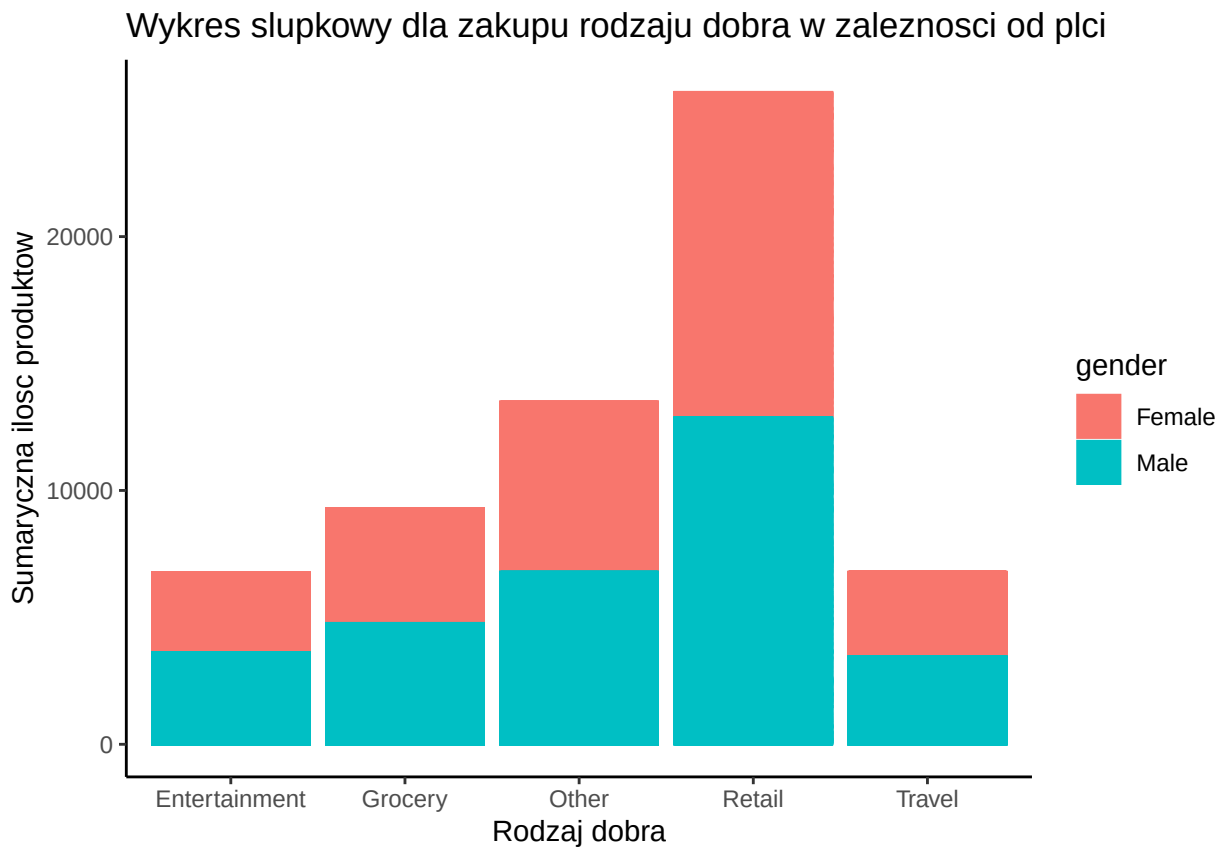
4 Wykresy

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 jake 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 slupkowy dla zakupu rodzaju dobra w zaleznosci od plci",
    x="Rodzaj dobra",
    y="Sumaryczna ilosc produktow"
    ) +
  theme_classic() # Ustawianie klasycznego wyglądu wykresu
```



Interpretacja wyników:

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

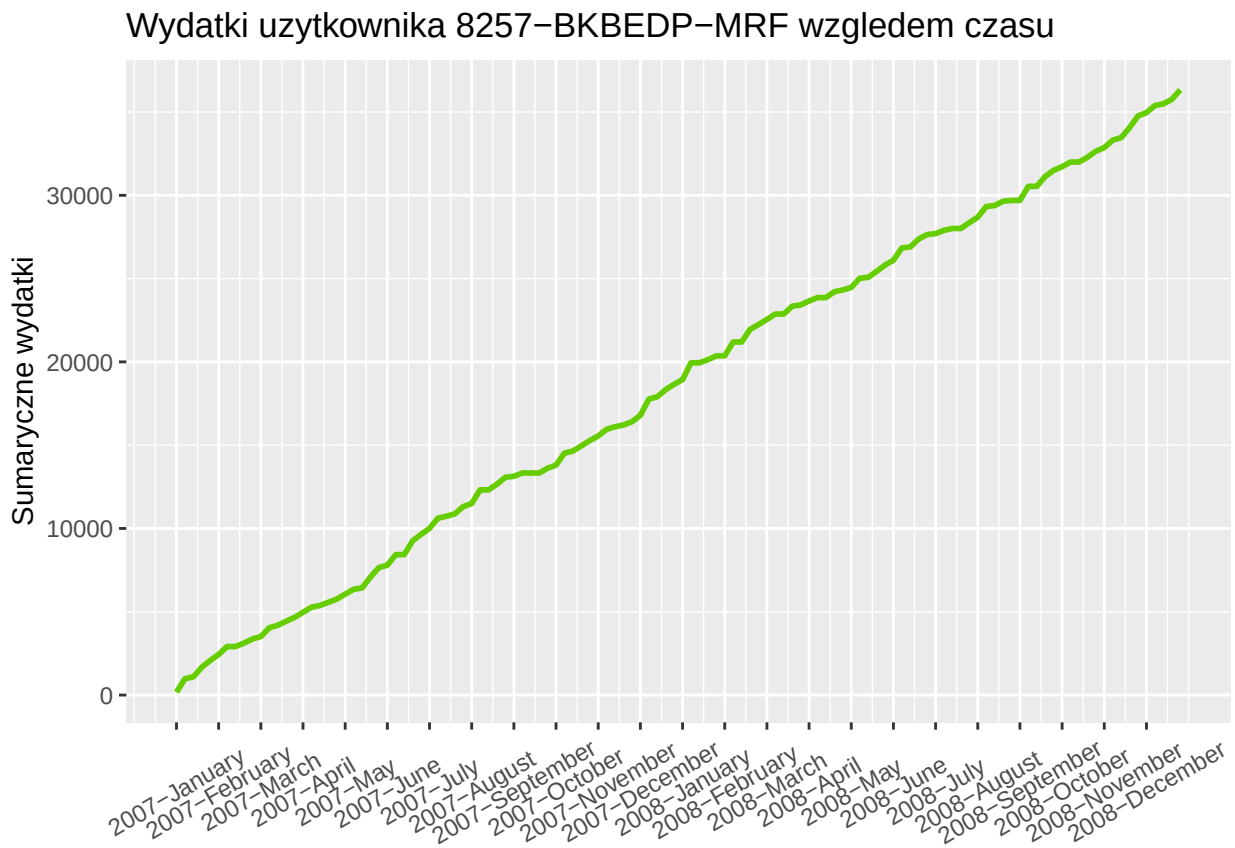
4.2 Wykres liniowy

Problem:

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

```
# Filtrujemy wysztkie 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")
```

```r
# Zamiana miesiąca z słowa na liczbe 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 uzytkownika 8257-BKBEDP-MRF wzgledem czasu",
    x = NULL,
    y = "Sumaryczne wydatki"
  ) +
  theme(axis.text.x = element_text(angle = 30, hjust = 0.5, vjust = 0.5))
```



Interpretacja wyników:

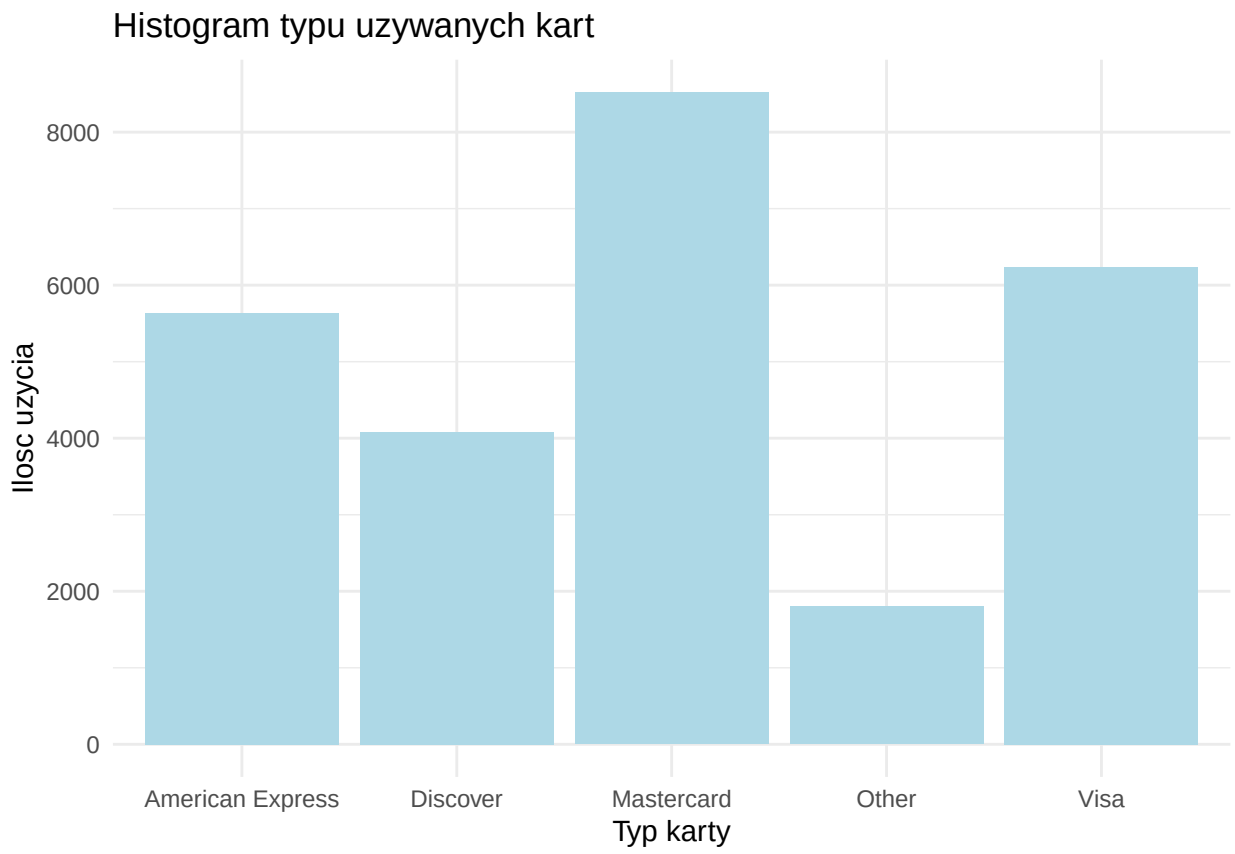
- Użytkownik sumarycznie wydał 36330.45
- Użytkownik używał karty tylko przez 2 lata
- Wydatki użytkownika są w miarę stałe (wykres ten jest używany do regresji linowej [ostatni podpunkt projektu], w którym możemy sprawdzić jak bardzo wydatki odstają od stałych)
- Robiąc pochodną wykresu, można określić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

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 uzywanych kart",
    x = "Typ karty",
    y = "Ilosc uzycia"
    ) +
  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.:

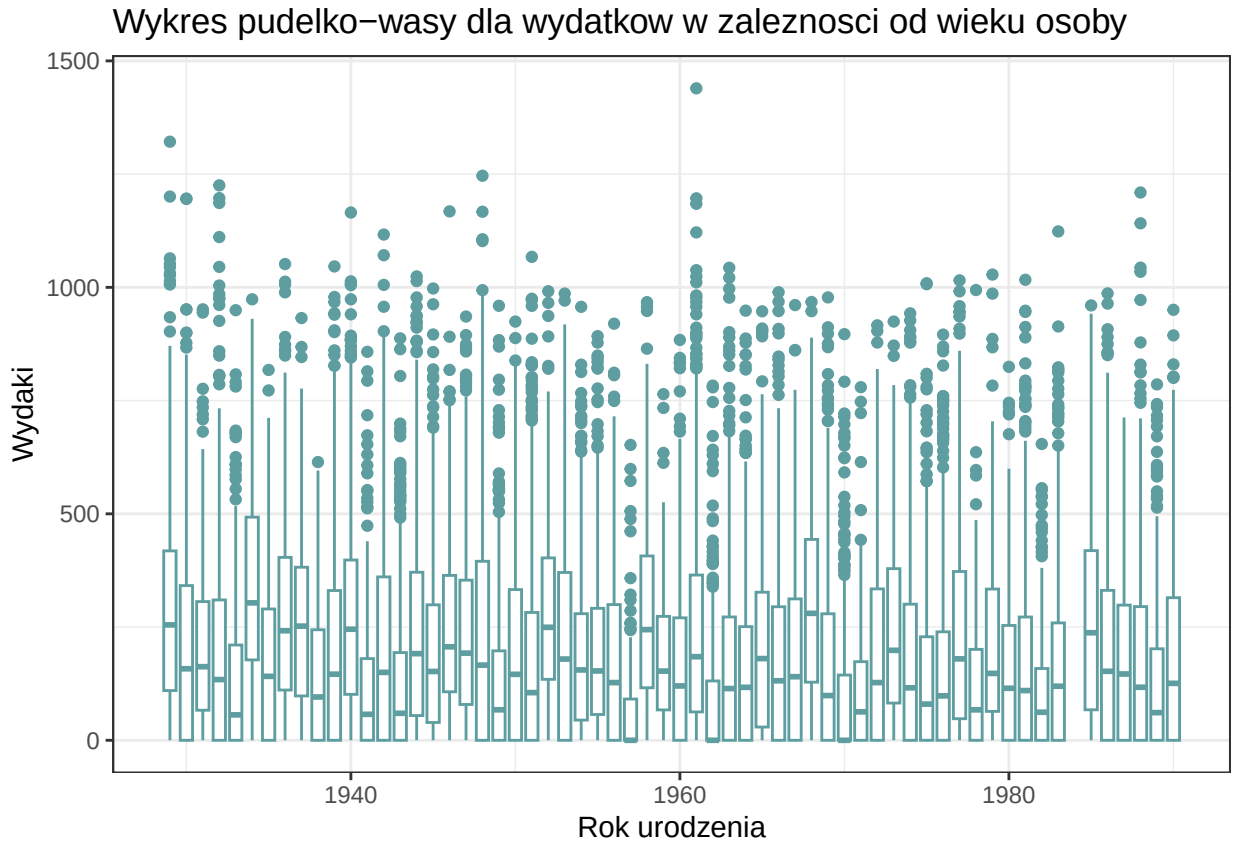
- *Mediana* - Linia wewnątrz pudełka, która przedstawia środkową wartość danych.
- *Pudełko* - Prostokąt, który rozciąga się od pierwszego kwartyla ($Q1$) do trzeciego kwartyla ($Q3$). Obejmuje środkowe 50% danych.
- *Wąsy* - Linie wychodzące z pudełka, które sięgają do najmniejszej i największej wartości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

- *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 kwartyl
q3 <- quantile(spent_1929, 0.75) # Ostatni kwartyl

# 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 pudelko-wasy dla wydatkow w zaleznosci od wieku osoby",
    x = "Rok urodzenia",
    y = "Wydaki",
    ) +
  theme_bw()
```



Interpretacja wyników:

- Dla osób urodzonych w 1929 wartość obserwacji dostających zaczynają się od 880.82 i jest ich 11
- Nie ma wyników odstających które są mniejsze niż 0.0

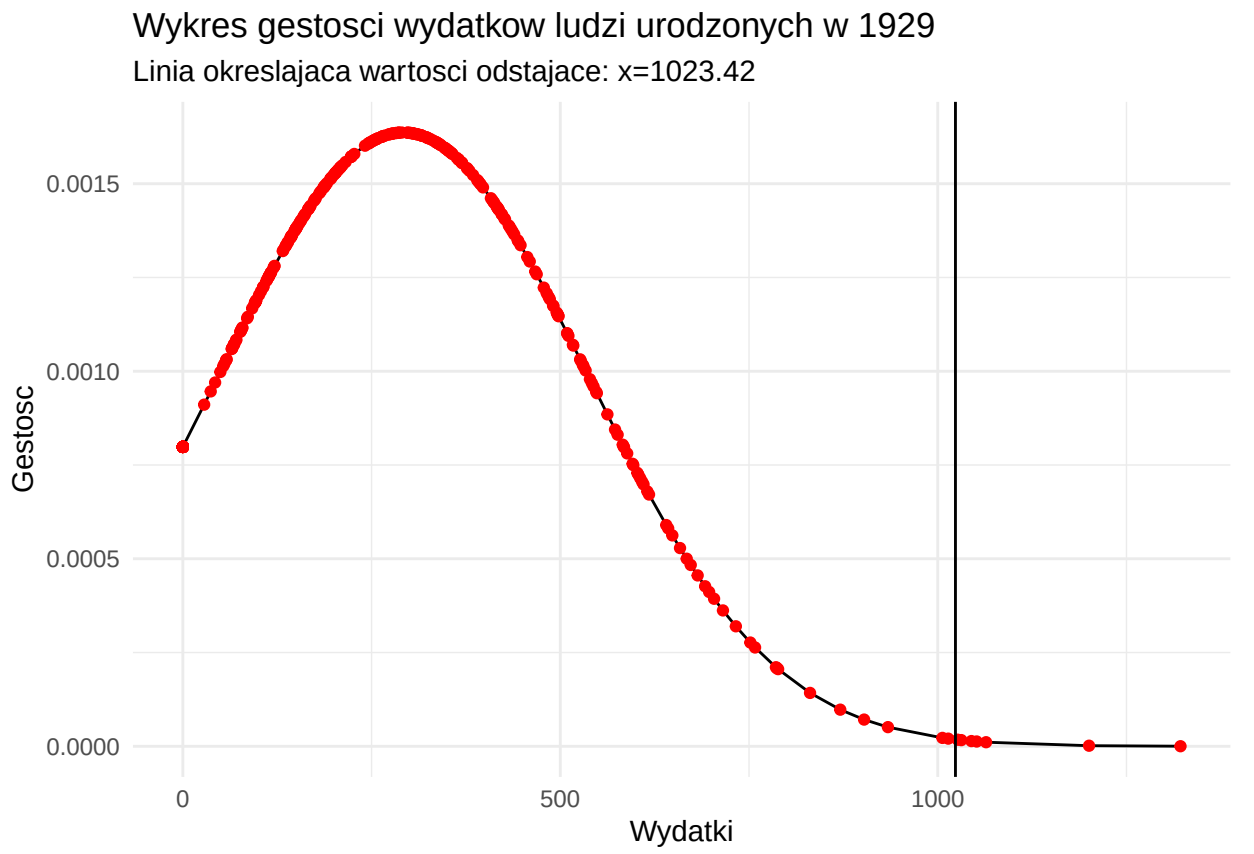
5.2 Odchylenie standardowe

```
# 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_threashold <- mean_spent_1929 + 3 * sd_spent_1929
lower_threashold <- mean_spent_1929 - 3 * sd_spent_1929
outliers_1929 <- spent_1929[spent_1929 > upper_threashold |
                        spent_1929 < lower_threashold]
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_threashold) +
  labs(
    title = "Wykres gestosci wydatkow ludzi urodzonych w 1929",
    subtitle = sprintf("Linia okreslajaca wartosci odstajace: x=%.02f", upper_threashold),
    x = "Wydatki",
    y = "Gestosc"
  ) +
  theme_minimal()
```



6 Wyliczenie prawdopodobieństwa dla zmiennej

6.1 Gerenowanie prób losowych

```
x <- sort(data$birth_year)
mean_x <- mean(x)
sd_x <- sd(x)

continuous_dnorm <- dnorm(x, mean_x, sd_x)
continuous_pnorm <- pnorm(x, mean_x, sd_x)

discreet_dbinom <- dbinom(x, length(x), mean_x / length(x))
discreet_pbinom <- pbinom(x, length(x), mean_x / length(x))
```

6.2 Obliczanie prawdopodobieństwa punktowego i przedziałowego

```
x_point <- 1969
n <- last(which(x == x_point))
point <- continuous_dnorm[n]
interval <- continuous_pnorm[n]
point
```

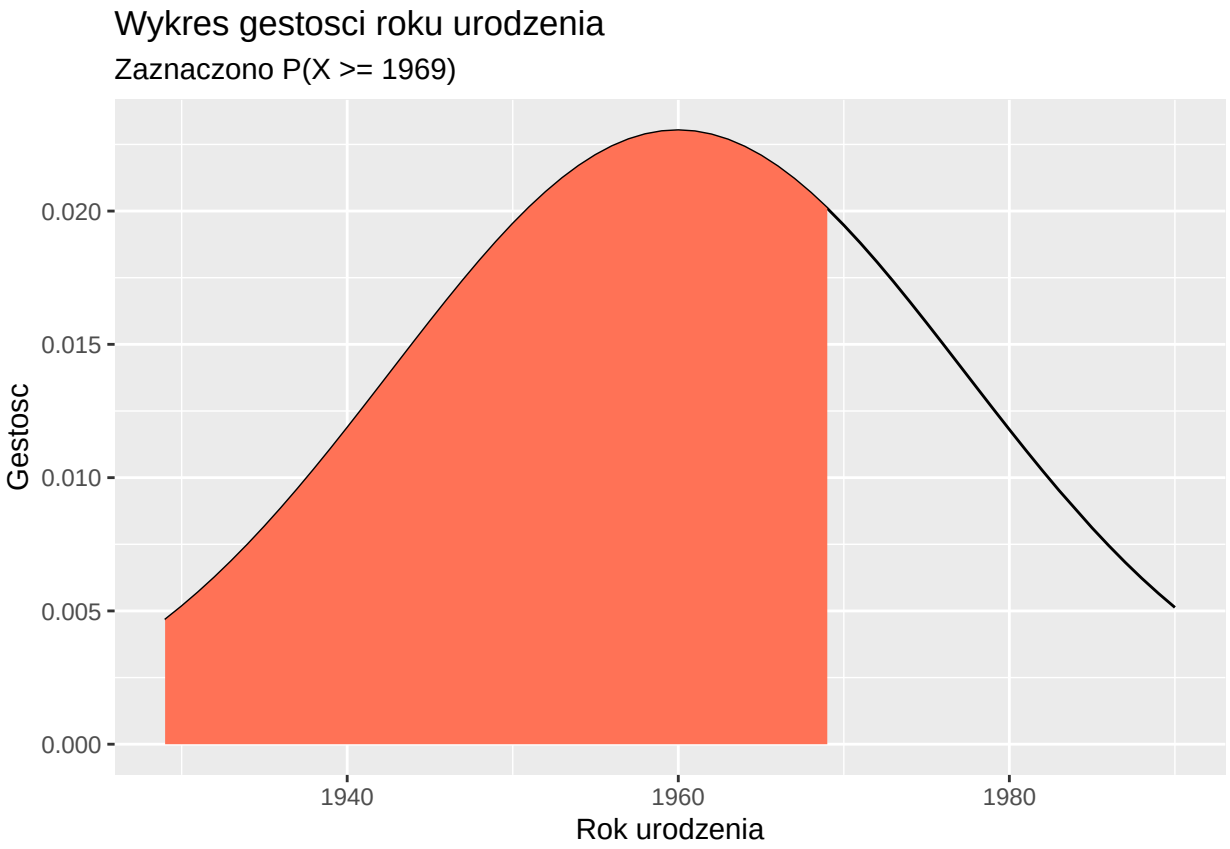
```
## [1] 0.02009985
```

```
interval
```

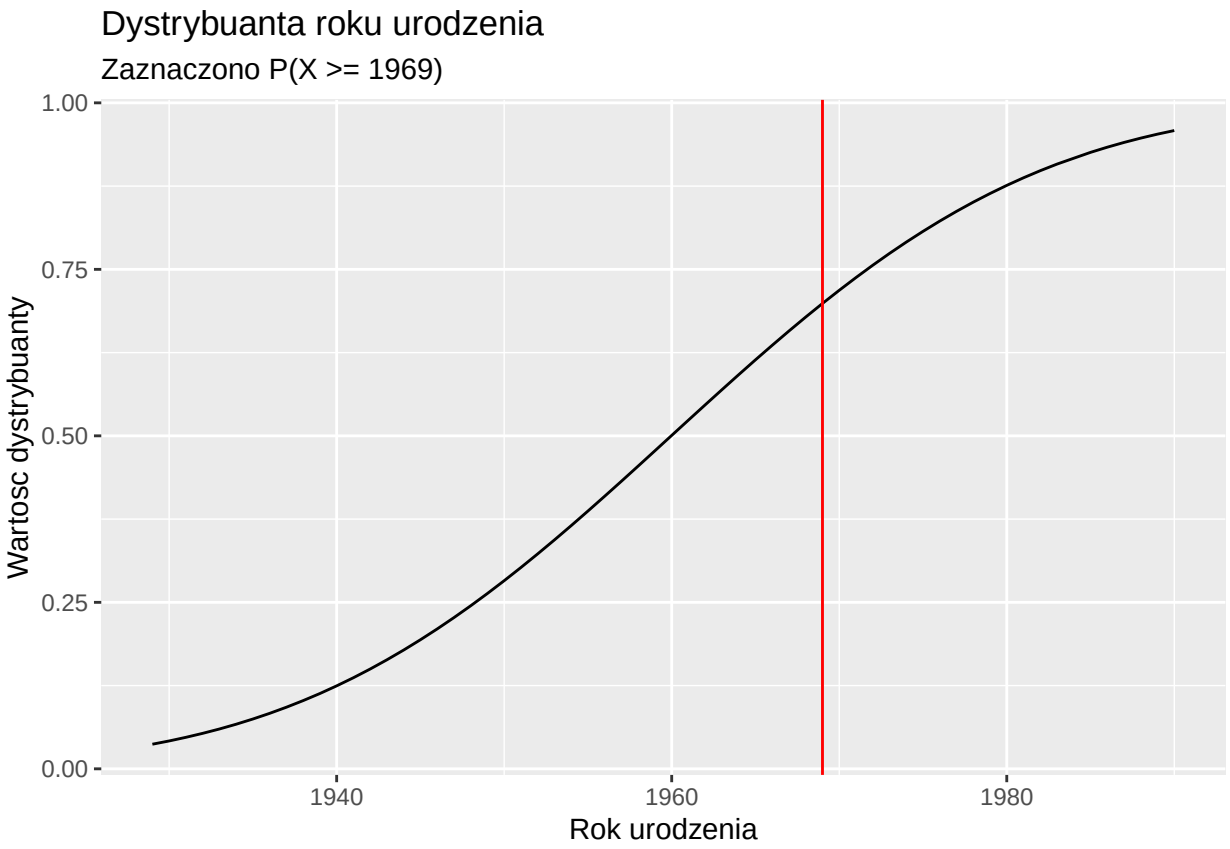
```
## [1] 0.6987881
```

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"
  )
```

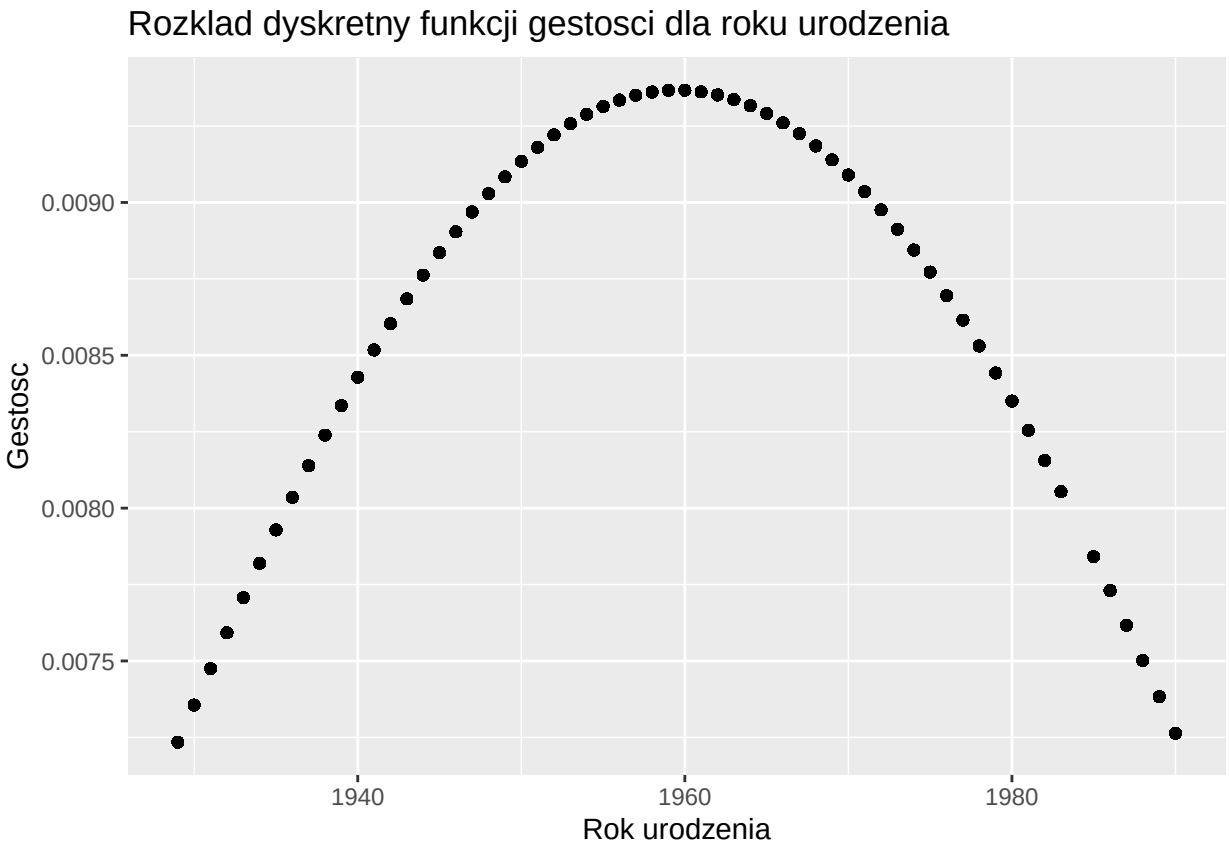



```
ggplot() +
  geom_line(data=data.frame(x=x, y=continuous_pnorm), aes(x, y)) +
  geom_vline(xintercept = x_point, color="red") +
  labs(
    title = "Dystrybuanta roku urodzenia",
    subtitle = sprintf("Zaznaczono P(X >= %i)", x_point),
    x = "Rok urodzenia",
    y = "Wartosc dystrybuanty"
  )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

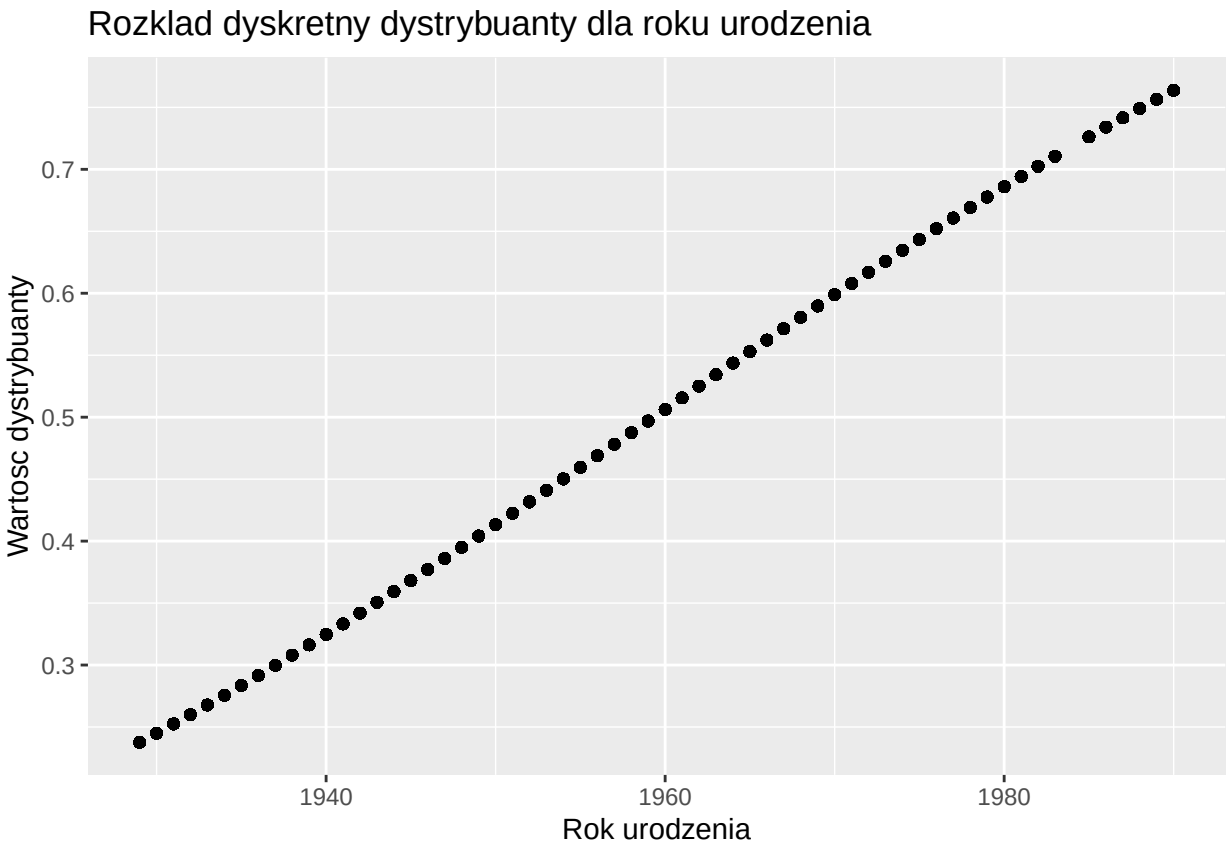


6.4 Wykres dyskretny

```
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"
  )
```




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



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_colum = ncol(matrix),
  sum_of_columns = colSums(matrix),
  sum_of_first_two_row = rowSums(matrix[1:2,]),
  sum_of_all_elemets = sum(matrix))

matrix_data
```

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

```
##
## $sum_of_first_two_row
## [1] 2172.81 2821.87
##
## $sum_of_all_elemets
## [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 nieparamteryczny 2

10 Regresja liniowa

10.1 Przygotowanie danych

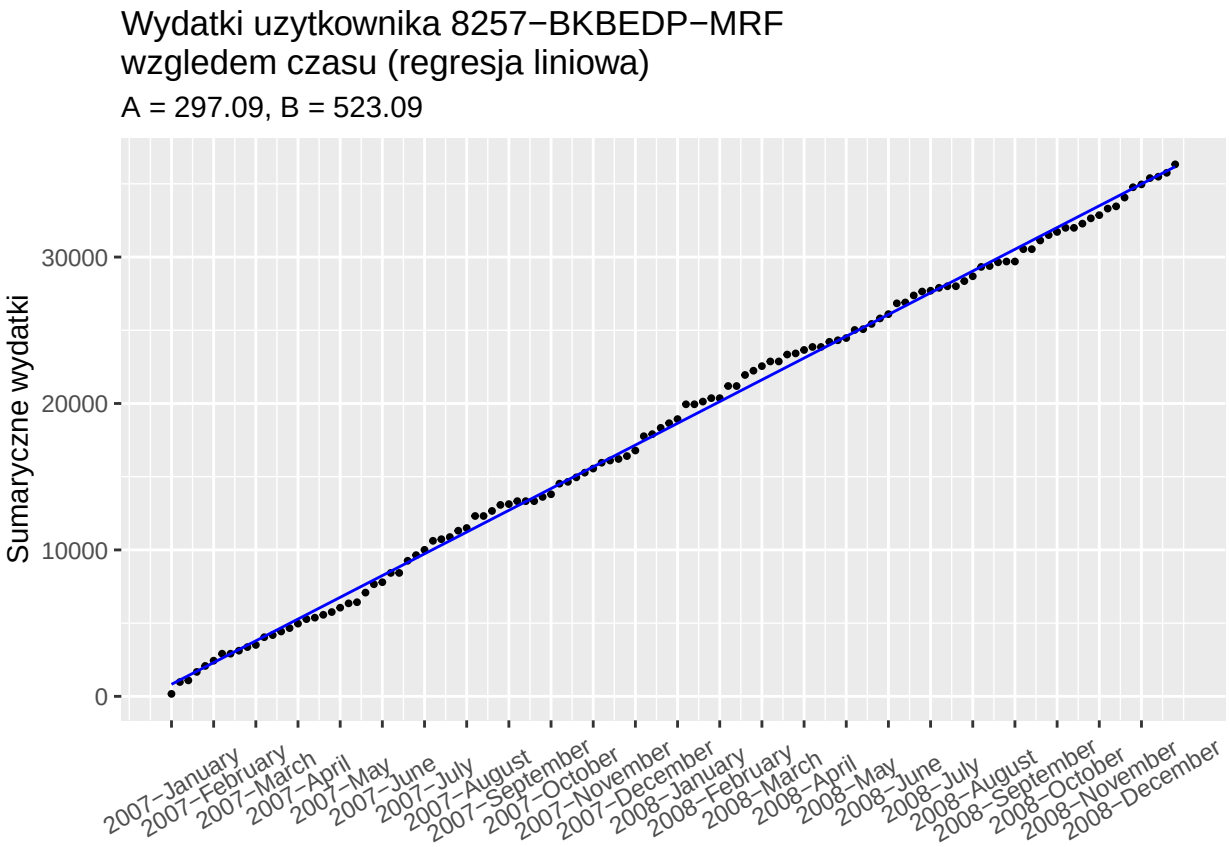
Powtórzony kod z wykresu liniowego

```
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\nwzględem czasu (regresja liniowa)",
    subtitle = sprintf("A = %.02f, B = %.02f", model$coefficients[2], model$coefficients[1]),
    x = NULL,
    y = "Sumaryczne wydatki"
  ) +
  theme(axis.text.x = element_text(angle = 30, hjust = 0.5, vjust = 0.5))
```


Wydatki uzytkownika 8257−BKBEDP−MRF
wzgledem czasu (regresja liniowa)
A = 297.09, B = 523.09
Sumaryczne wydatki
30000
20000
10000
0
2007−January
2007−February
2007−March
2007−April
2007−May
2007−June
2007−July
2007−August
2007−September
2007−October
2007−November
2007−December
2008−January
2008−February
2008−March
2008−April
2008−May
2008−June
2008−July
2008−August
2008−September
2008−October
2008−November
2008−December