Statystyka

Rafał Szyiński 259380, Kajetan Leszak

2024-06-02

Statystyka

Wstęp

Do wykonywania projektu będziemy używać dwóch bibliotek:

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

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

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

Opis baza danych

Baza **credit_card.xls** pochodzi z eportalu. Zawiera ona dane o użytkownikach kart kredytowych oraz wykonywanych przez nie transakcji.

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.~
```

Wyliczenie podstawowych statystyk

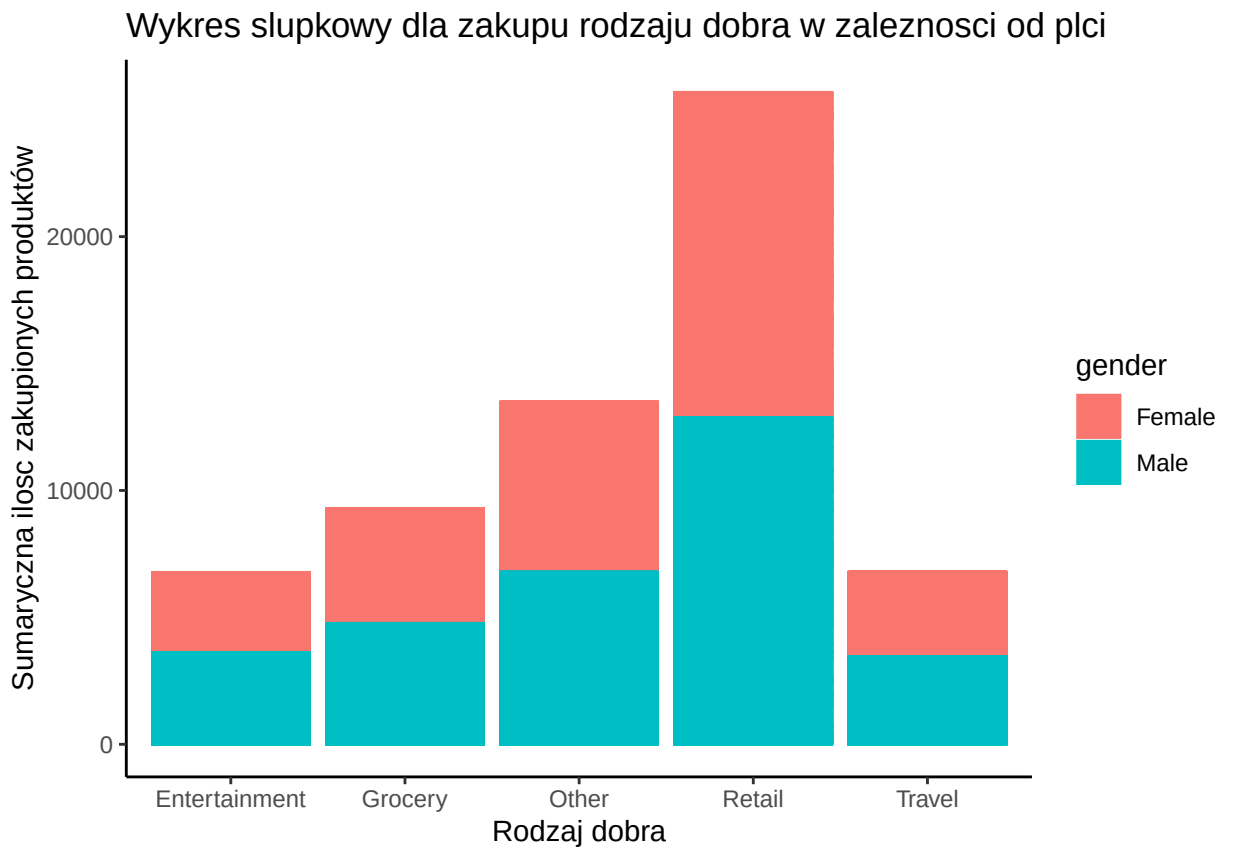
Statysyka będzie wyliczona z columny items i spent

```r
frame <- data.frame(
  items_summary=unclass(summary(data$items)),
  spent_summary=unclass(summary(data$spent))
  )
frame
```

```
##         items_summary spent_summary
## Min.         0.000000        0.0000
## 1st Qu.      0.000000        0.0000
## Median       2.000000      141.7750
## Mean         2.358828      196.2524
## 3rd Qu.      4.000000      311.2975
## Max.        13.000000     1439.3700
```

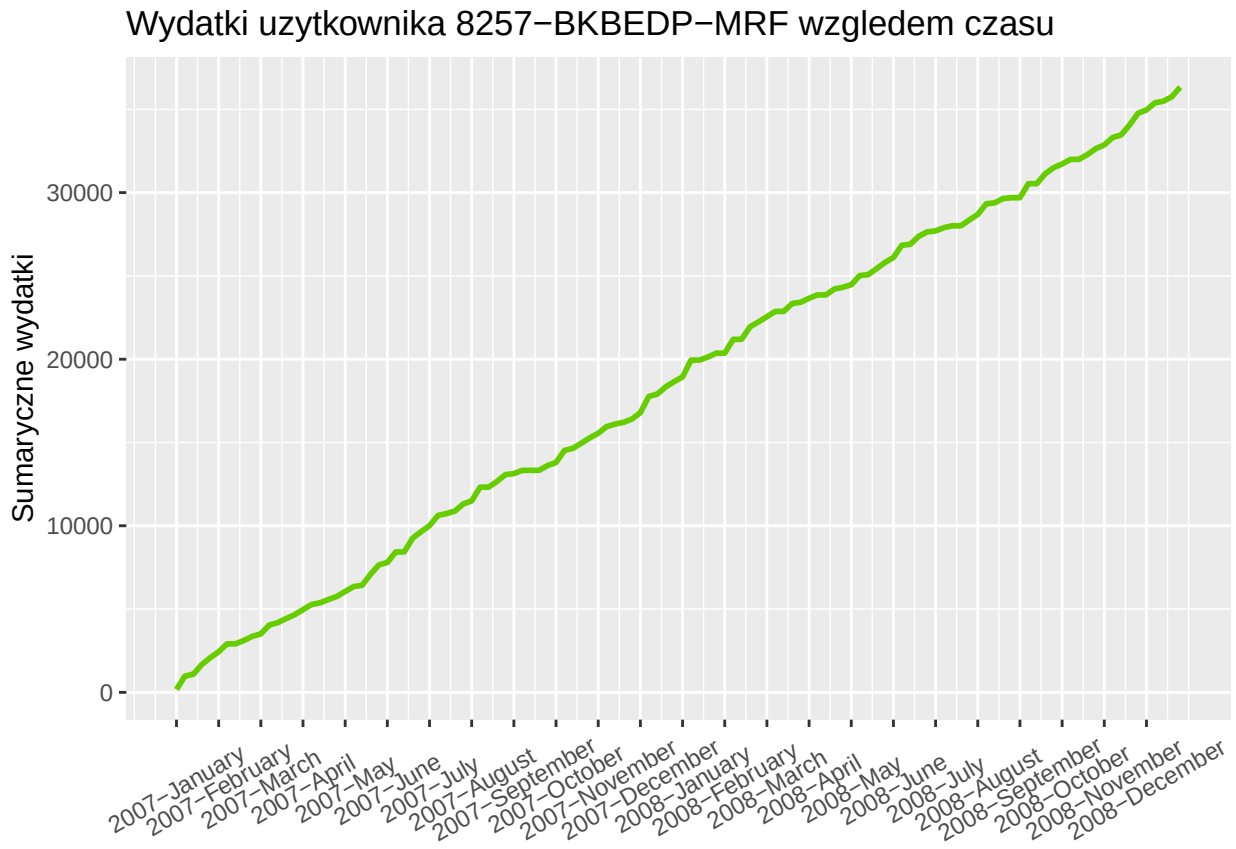
Wykresy

```r
ggplot() +
  geom_bar(
    data=data,
    aes(x=type_trans, y=items, color=gender, fill=gender),
    stat="identity"
    ) +
  labs(
    title="Wykres słupkowy dla zakupu rodzaju dobra w zależności od płci",
    x="Rodzaj dobra",
    y="Sumaryczna ilość zakupionych produktów"
    ) +
  theme_classic()
```

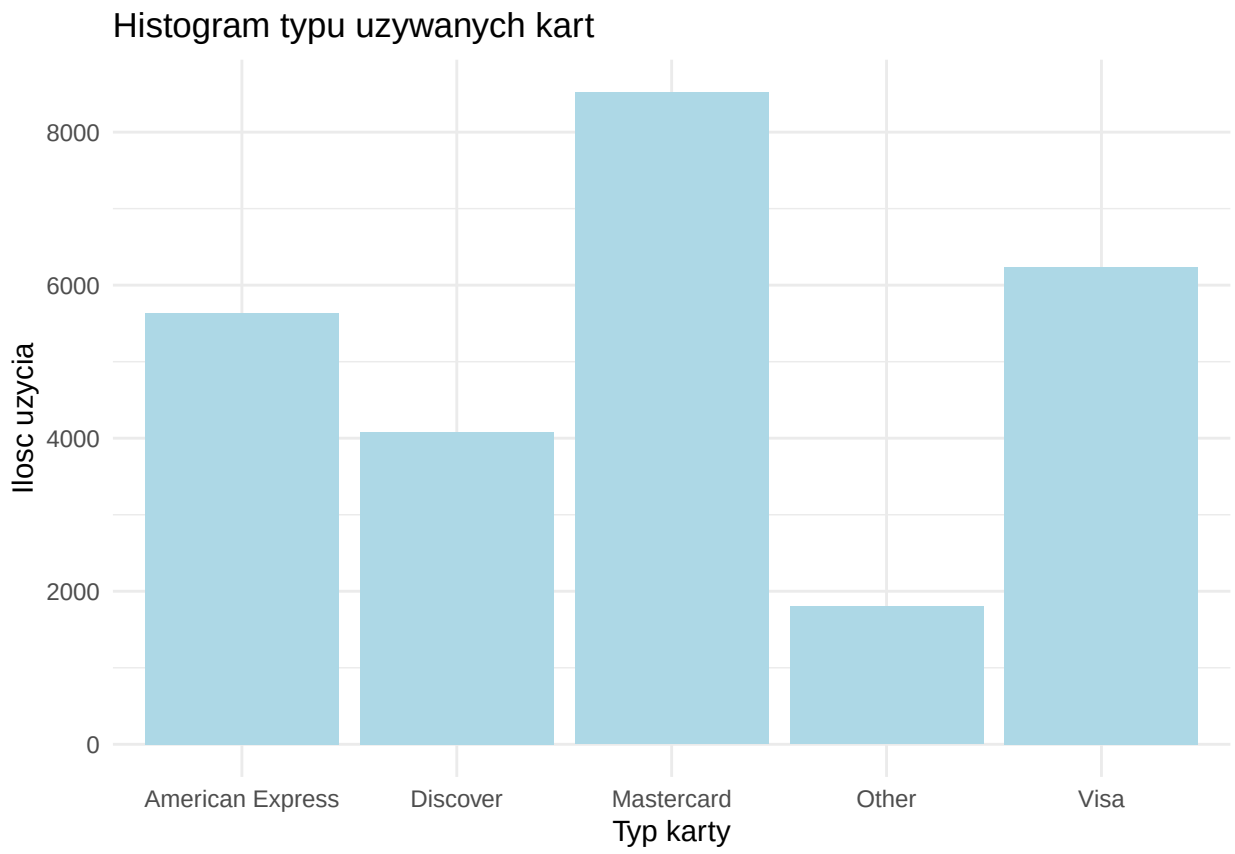



```r
# Filtrujemy wysztkie dane pierwszego użytkownika
user_data <- data[data$custid == "8257-BKBEDP-MRF",]

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

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

ggplot() +
  geom_line(
    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))
```




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



Obserwacje odstające

Będziemy obserwować wydatki osób w zależności od ich wieku

Przygotowanie danych

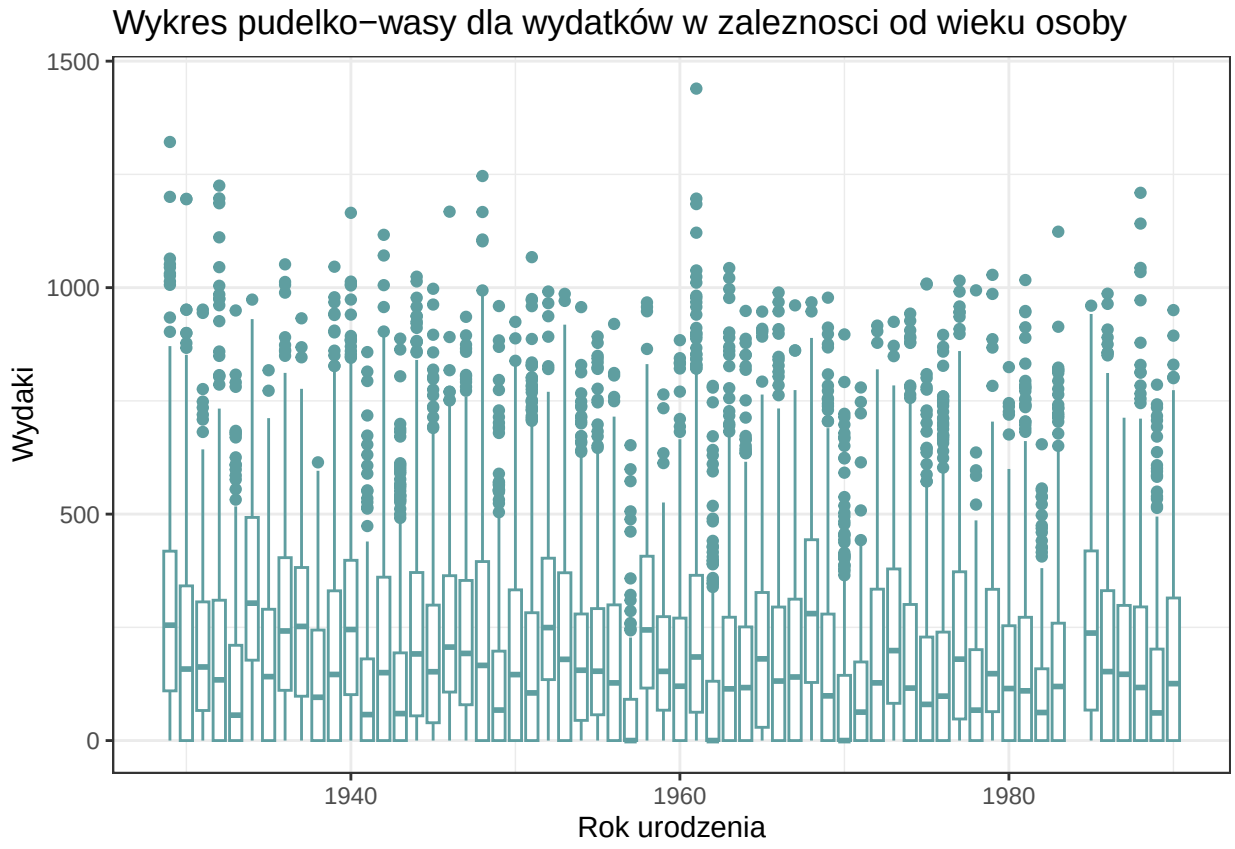
```
user_data =
  data.frame(year=data$birth_year, spent=data$spent) %>%
  group_by(year) %>%
  summarize(
    total_spent=sum(spent),
    q1 = quantile(spent, 0.25),
    median = median(spent),
    q3 = quantile(spent, 0.75),
    lower_whisker = max(min(spent), q1 - 1.5 * (q3 - q1)),
    upper_whisker = min(max(spent), q3 + 1.5 * (q3 - q1)),
    mean = mean(spent),
    sd = sd(spent)
  )
```

Wykres pudełko-wąsy

```
outliers_1929 <- data[data$birth_year == 1929,]$spent
outliers_1929 <- outliers_1929[outliers_1929 > user_data[user_data$year == 1929,]$upper_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(
    data=data,
    aes(x=birth_year, y=spent, group=birth_year),
    color="cadetblue"
    ) +
  labs(
    title = "Wykres pudełko-wąsy dla wydatków w zależności od wieku osoby",
    x = "Rok urodzenia",
    y = "Wydaki",
    ) +
  theme_bw()
```

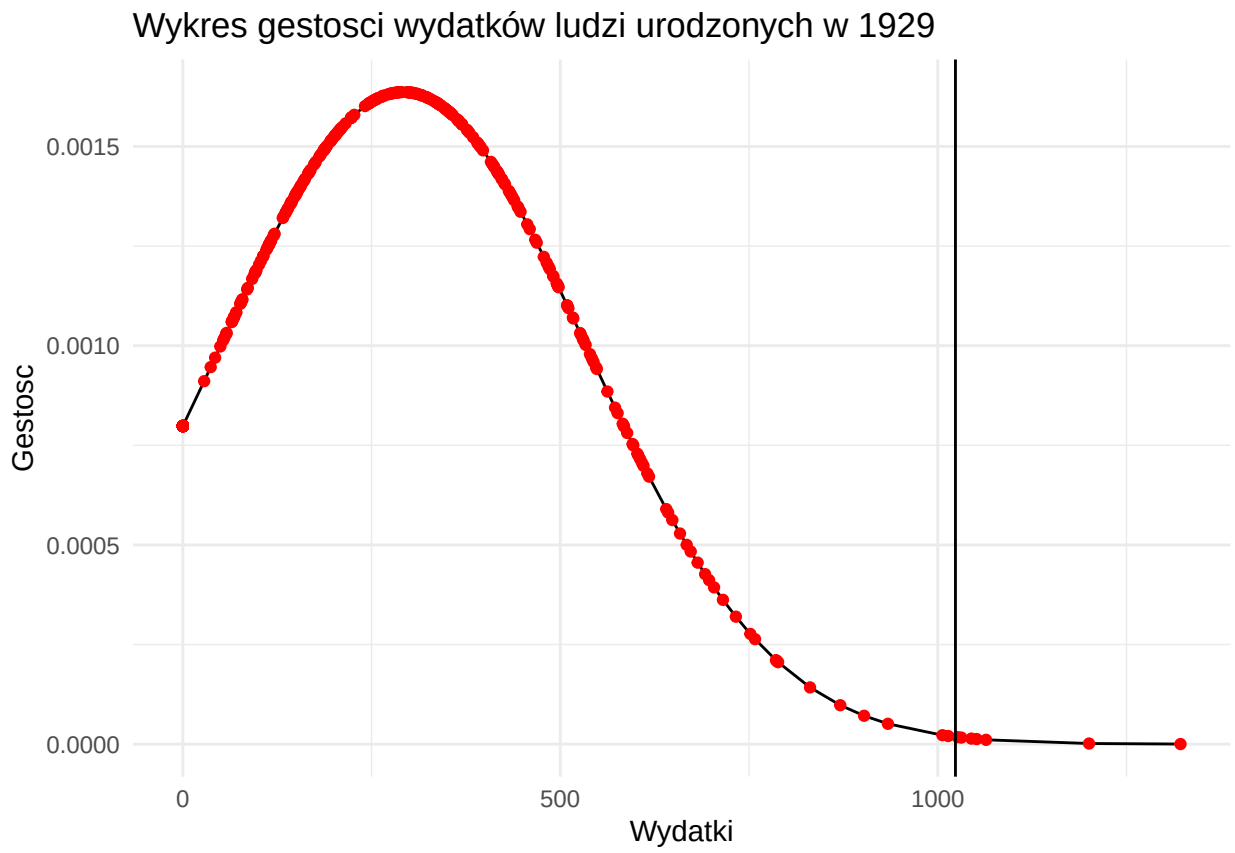


Odchylenie standardowe

```
spent = data[data$birth_year == 1929,]$spent
data_1929 = user_data[user_data$year == 1929,]
norm <- dnorm(spent, data_1929$mean, data_1929$sd)
threashold <- data_1929$mean + 3 * data_1929$sd
outliers_1929 <- spent[spent > 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, y=norm), aes(x, y)) +
  geom_point(data=data.frame(x=spent, y=norm), aes(x, y), color="red") +
  geom_vline(xintercept=threashold) +
  labs(
    title = "Wykres gęstości wydatków ludzi urodzonych w 1929",
    x = "Wydatki",
    y = "Gęstość"
  ) +
  theme_minimal()
```



Wyliczenie prawdopodobieństwa dla zmiennej

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))
```

Obliczanie prawdopodobieństwa punktowego i przedziałowego

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

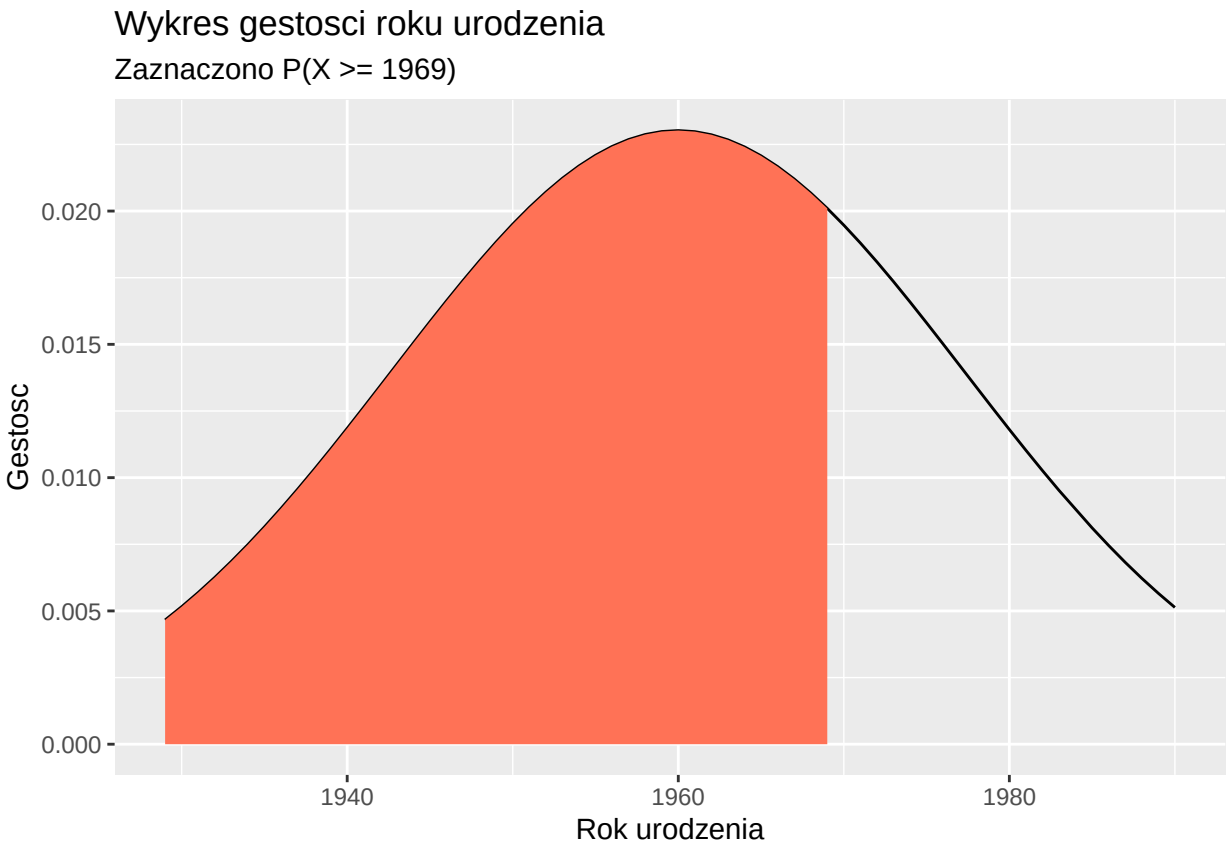
```
## [1] 0.02009985
```

```
interval
```

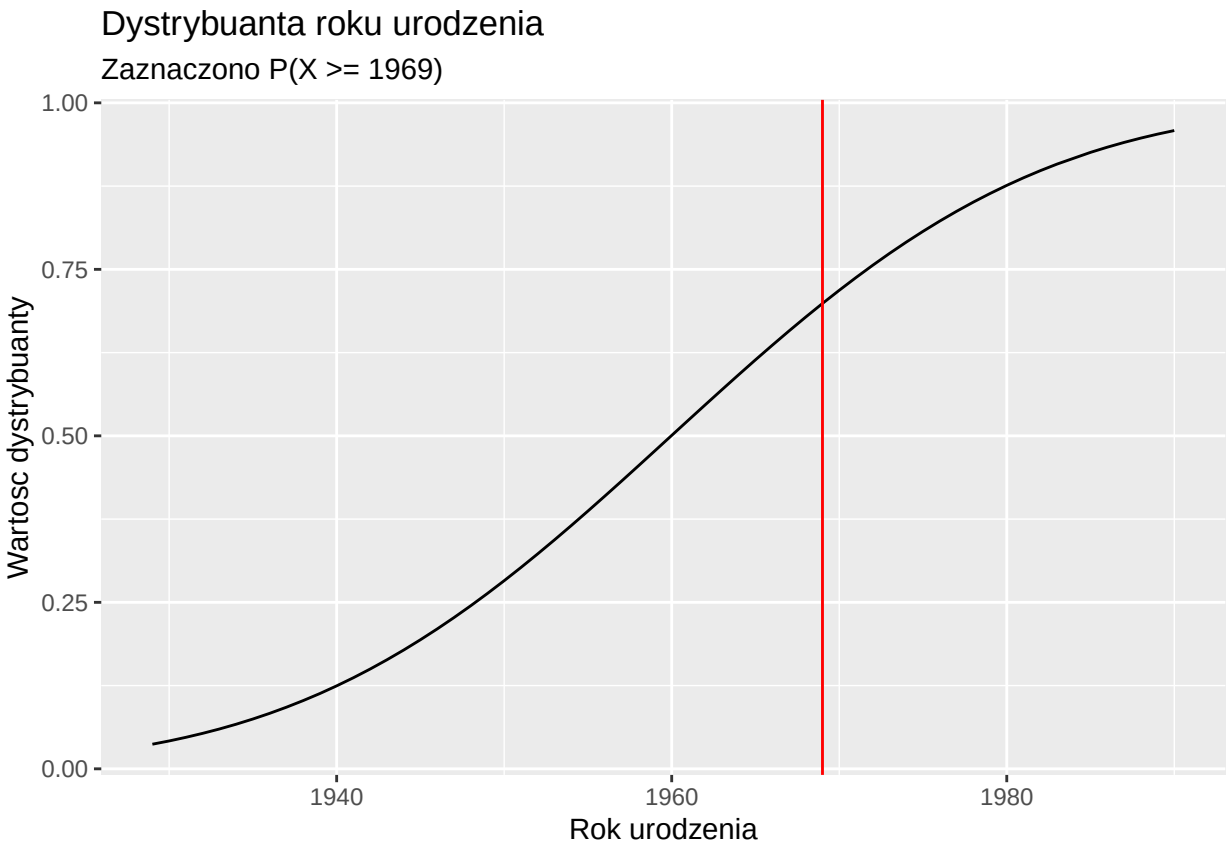
```
## [1] 0.6987881
```

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 gęstości roku urodzenia",
    subtitle = sprintf("Zaznaczono P(X >= %i)", x_point),
    x = "Rok urodzenia",
    y = "Gęstość"
  )
```

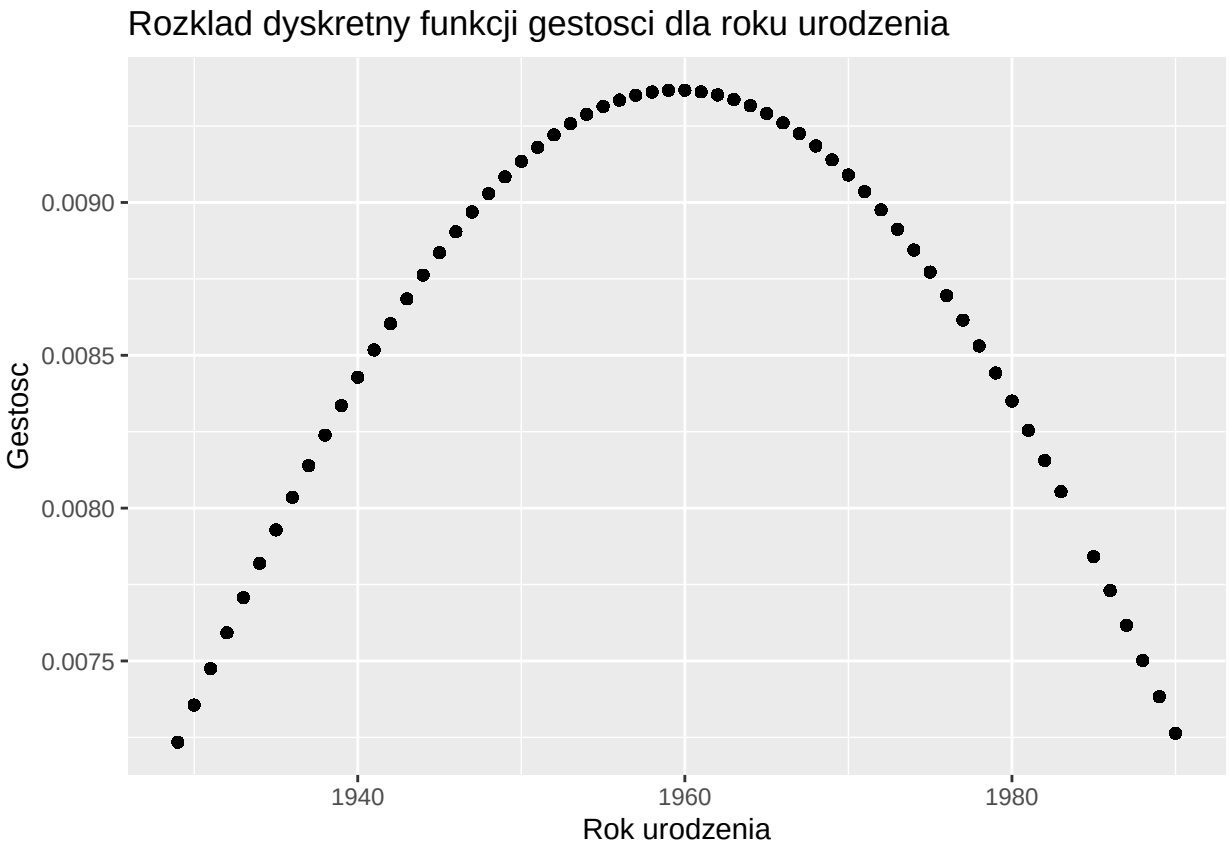



```
ggplot() +
  geom_line(data=data.frame(x=x, y=continuous_pnorm), aes(x, y)) +
  geom_vline(xintercept = x_point, color="red") +
  labs(
    title = "Dystrybuanta roku urodzenia",
    subtitle = sprintf("Zaznaczono P(X >= %i)", x_point),
    x = "Rok urodzenia",
    y = "Wartość dystrybuanty"
  )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

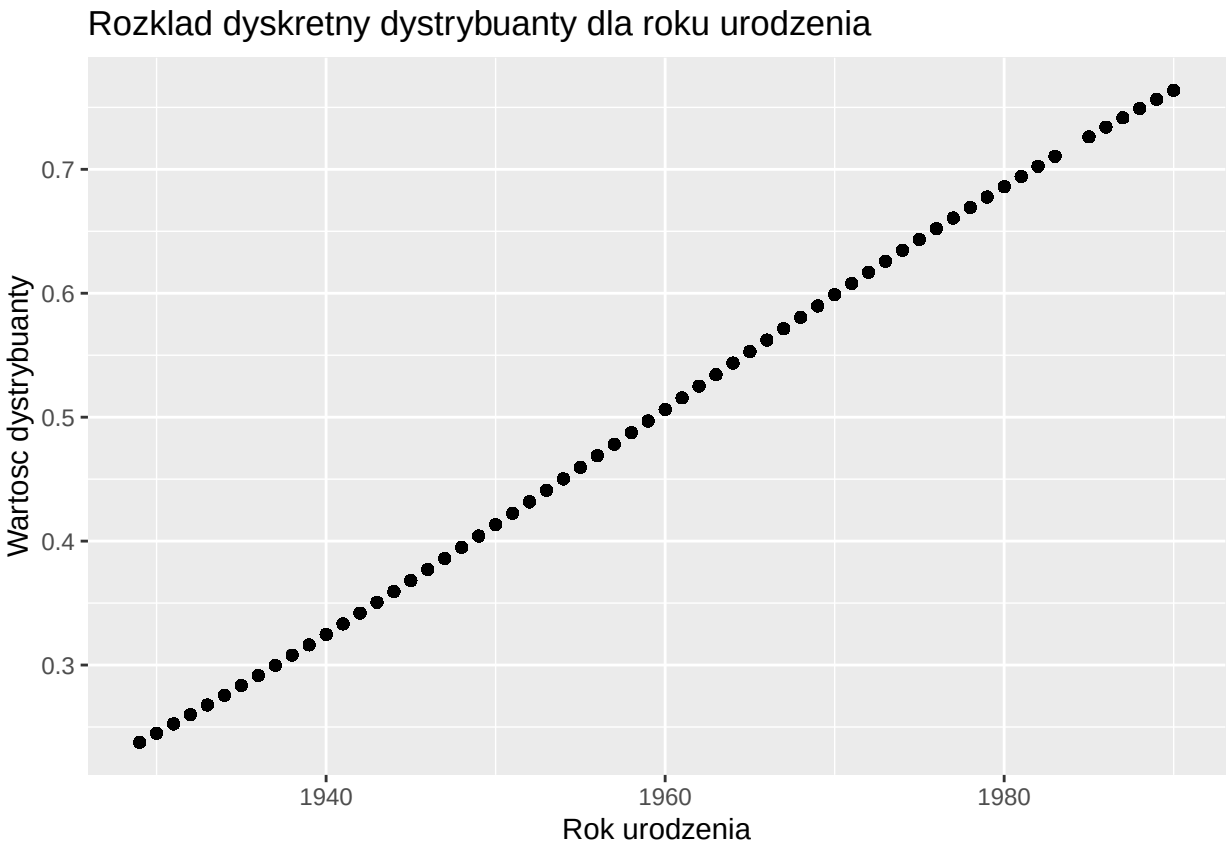


Wykres dyskretny

```
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"
  )
```



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
```

Przedziały ufności

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
```

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
```

Hipotezy

Test parametryczny - średnia urodzenia to 1960

```r
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
```

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

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

t.test(spent_1960, spent)
```

```
##
##  Welch Two Sample t-test
##
## data:  spent_1960 and spent
## t = -8.3203, df = 599.52, p-value = 5.931e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -152.67347  -94.36297
## sample estimates:
## mean of x mean of y
##  168.5672  292.0854
```

Test nieparametryczny 1

Test nieparamteryczny 2

Regresja liniowa

Przygotowanie danych

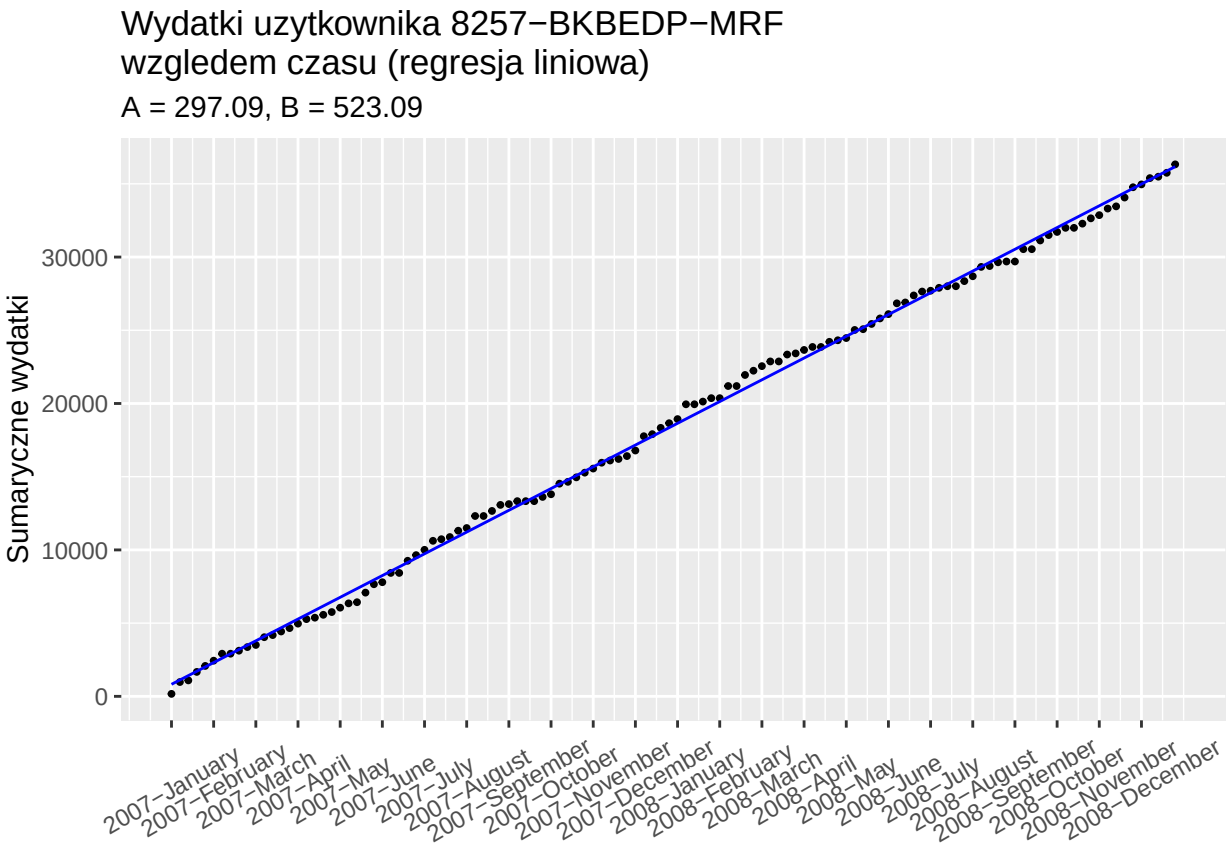
Powtórzony kod z wykresu liniowego

```r
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
  )
```

```r
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]
```

Wykres

```r
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