Podstawowa analiza danych klientów banku oraz danych dotyczących albumów muzycznych

Dane:

```
wiek <- readRDS('age.rds')
df_age = as.data.frame(wiek, stringsAsFactors = FALSE)

braki = sum(is.na(wiek))  # Liczba brakujących wartości w danych
braki
```

```
## [1] 53
```

```
DANE_WIEK <- na.omit(wiek)
```

1. Jaki wiek ma najmłodszy i najstarzy klient?

```
min_wiek = min(DANE_WIEK)
print(paste("Najmłodszy klient ma", min_wiek, "lat."))
```

```
## [1] "Najmłodszy klient ma 16 lat."
```

```
max_wiek = max(DANE_WIEK)
print(paste("Najstarszy klient ma", max_wiek, "lat."))
```

```
## [1] "Najstarszy klient ma 86 lat."
```

2. Jaki jest przeciętny wiek klientów banku?

```
średni_wiek = mean(DANE_WIEK)
print(paste("Przeciętny wiek klientów to", round(średni_wiek, 2), "lat."))
```

```
## [1] "Przeciętny wiek klientów to 44.55 lat."
```

3. Jak bardzo zróżnicowani są klienci banku pod względem wieku?

```
odchylenie_std <- sd(DANE_WIEK)
print(paste("Odchylenie standardowe wieku klientów:", round(odchylenie_std,2)))
```

```
## [1] "Odchylenie standardowe wieku klientów: 10"
```

4. Ilu klientów banku jest niepełnoletnich? Jaki procent całości?

```
liczba_niepelnoletnich_klientów <- sum(DANE_WIEK < 18)
procent_niepelnoletnich_klientów <- (liczba_niepelnoletnich_klientów / length(wiek)) * 100

print(paste("Liczba klientów niepełnoletnich:", liczba_niepelnoletnich_klientów))
```

```
## [1] "Liczba klientów niepełnoletnich: 33"
```

```
print(paste("Procent klientów niepełnoletnich:", round(procent_niepelnoletnich_klientów, 2), "%"))
```

```
## [1] "Procent klientów niepełnoletnich: 0.33 %"
```

5. Ilu klientów banku jest w wieku 30-50 lat? Jaki to procent całości?

```
klienci_30_50 <- sum(DANE_WIEK >= 30 & DANE_WIEK <= 50)
procent_30_50 <- (klienci_30_50 / length(wiek)) * 100

print(paste("Liczba klientów w wieku 30-50:", klienci_30_50))
```

```
## [1] "Liczba klientów w wieku 30-50: 6536"
```

```
print(paste("Procent klientów w wieku 30-50:", round(procent_30_50, 2), "%"))
```

```
## [1] "Procent klientów w wieku 30-50: 65.36 %"
```

6. Ilu klientów nie podało swojego wieku? Jaki to procent całości?

```
procent_NA <- (braki / length(wiek)) * 100
print(paste("Liczba klientów, która nie podała wieku:", braki))
```

```
## [1] "Liczba klientów, która nie podała wieku: 53"
```

```
print(paste("Procent klientów, którzy nie podali wieku:", round(procent_NA, 2), "%"))
```

```
## [1] "Procent klientów, którzy nie podali wieku: 0.53 %"
```

7. Ile klientów bank posiada w segmentach wiekowych [16,17], [18,24], [25,34],[35,44], [65, inf]? Jaki to procent całości?

```
segment_wiekowy <- cut(DANE_WIEK, breaks = c(16, 17, 24, 34, 44, Inf), right = TRUE, labels = c("[16,

licznosc <- table(segment_wiekowy)
procenty <- prop.table(licznosc) * 100

print("Liczba klientów w poszczególnych przedziałach wiekowych:")
```

```
## [1] "Liczba klientów w poszczególnych przedziałach wiekowych:"
```

```
print(licznosc)
```

```
## segment_wiekowy
##  [16,17]  [18,24]  [25,34]  [35,44] [65,Inf]
##       11      192     1387     3336     4999
```

```
print("\nProcentowy udział klientów w poszczególnych przedziałach wiekowych:")
```

```
## [1] "\nProcentowy udział klientów w poszczególnych przedziałach wiekowych:"
```

```
print(procenty)
```

```
## segment_wiekowy
##    [16,17]    [18,24]    [25,34]    [35,44]   [65,Inf]
##  0.1108312  1.9345088 13.9748111 33.6120907 50.3677582
```

Dane:

```
df_albumy = read.csv('albums.csv')
str(df_albumy)
```

```
## 'data.frame':    100000 obs. of  10 variables:
##  $ id                  : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ artist_id           : int  1767 23548 17822 19565 24941 1988 43032 648 19441 13687 ...
##  $ album_title         : chr  "Call me Cat Moneyless That Doggies" "Down Mare" "Embarrassed Hungry" "
##  $ genre               : chr  "Folk" "Metal" "Latino" "Pop" ...
##  $ year_of_pub         : int  2006 2014 2000 2017 2010 2018 2006 2005 2008 2002 ...
##  $ num_of_tracks       : int  11 7 11 4 8 15 5 3 10 14 ...
##  $ num_of_sales        : int  905193 969122 522095 610116 151111 537615 254802 488696 239081 531099
##  $ rolling_stone_critic: num  4 3 2.5 1.5 4.5 4.5 5 0.5 1.5 3.5 ...
##  $ mtv_critic          : num  1.5 4 1 2 2.5 1.5 2 5 2 1 ...
##  $ music_maniac_critic : num  3 5 2 4 1 2 2.5 4.5 2.5 3.5 ...
```

```
head(df_albumy, 10)
```

```
##    id artist_id                                album_title       genre
```

```
## 1   1      1767            Call me Cat Moneyless That Doggies        Folk
## 2   2     23548                                     Down Mare       Metal
## 3   3     17822                              Embarrassed Hungry      Latino
## 4   4     19565          Standard Immediate Engineer Slovakia         Pop
## 5   5     24941                       Decent Distance Georgian Black Metal
## 6   6      1988                                          Tall Progressive
## 7   7     43032 Steel Patrol Officer The Queen Panda Pleasant    Pop-Rock
## 8   8       648           kneel Llama Hanger Uzbekistan Relax       Retro
## 9   9     19441                         Lovers Totally Manager      Latino
## 10 10     13687                                          Rock     Western
##    year_of_pub num_of_tracks num_of_sales rolling_stone_critic mtv_critic
## 1         2006            11       905193                 4.0        1.5
## 2         2014             7       969122                 3.0        4.0
## 3         2000            11       522095                 2.5        1.0
## 4         2017             4       610116                 1.5        2.0
## 5         2010             8       151111                 4.5        2.5
## 6         2018            15       537615                 4.5        1.5
## 7         2006             5       254802                 5.0        2.0
## 8         2005             3       488696                 0.5        5.0
## 9         2008            10       239081                 1.5        2.0
## 10        2002            14       531099                 3.5        1.0
##    music_maniac_critic
## 1                  3.0
## 2                  5.0
## 3                  2.0
## 4                  4.0
## 5                  1.0
## 6                  2.0
## 7                  2.5
## 8                  4.5
## 9                  2.5
## 10                 3.5
```

```
summary(df_albumy)
```

```
##        id           artist_id     album_title           genre
##  Min.   :     1   Min.   :    1   Length:100000      Length:100000
##  1st Qu.: 25001   1st Qu.:12388   Class :character   Class :character
##  Median : 50000   Median :24940   Mode  :character   Mode  :character
##  Mean   : 50000   Mean   :24982
##  3rd Qu.: 75000   3rd Qu.:37498
##  Max.   :100000   Max.   :50000
##   year_of_pub   num_of_tracks     num_of_sales    rolling_stone_critic
##  Min.   :2000   Min.   : 2.000   Min.   :  1009   Min.   :0.500
##  1st Qu.:2004   1st Qu.: 5.000   1st Qu.:251604   1st Qu.:1.500
##  Median :2010   Median : 8.000   Median :499532   Median :2.500
##  Mean   :2010   Mean   : 8.489   Mean   :500045   Mean   :2.749
##  3rd Qu.:2015   3rd Qu.:12.000   3rd Qu.:749354   3rd Qu.:4.000
##  Max.   :2019   Max.   :15.000   Max.   :999994   Max.   :5.000
##    mtv_critic    music_maniac_critic
##  Min.   :0.500   Min.   :0.500
##  1st Qu.:1.500   1st Qu.:1.500
##  Median :3.000   Median :3.000
##  Mean   :2.752   Mean   :2.748
##  3rd Qu.:4.000   3rd Qu.:4.000
```

```
##  Max.   :5.000   Max.   :5.000
```

1. Średnia ocena dla każdej kolumny krytyków:

```
mean_rolling_stone <- mean(df_albumy$rolling_stone_critic, na.rm = TRUE)
mean_mtv <- mean(df_albumy$mtv_critic, na.rm = TRUE)
mean_music_maniac <- mean(df_albumy$music_maniac_critic, na.rm = TRUE)
cat("Średnia ocena Rolling Stone:", mean_rolling_stone, "\n")
```

```
## Średnia ocena Rolling Stone: 2.748945
```

```
cat("Średnia ocena MTV:", mean_mtv, "\n")
```

```
## Średnia ocena MTV: 2.75178
```

```
cat("Średnia ocena Music Maniac:", mean_music_maniac, "\n")
```

```
## Średnia ocena Music Maniac: 2.748225
```

2. Korelacje między ocenami krytyków a liczbą sprzedanych płyt:

```
cor(df_albumy$rolling_stone_critic, df_albumy$num_of_sales, use = "complete.obs")
```

```
## [1] -0.002493506
```

```
cor(df_albumy$mtv_critic, df_albumy$num_of_sales, use = "complete.obs")
```

```
## [1] 0.000523106
```

```
cor(df_albumy$music_maniac_critic, df_albumy$num_of_sales, use = "complete.obs")
```

```
## [1] -0.001194907
```

Wynik korelacji są bardzo bliskie zeru, co oznacza, że nie ma praktycznie żadnej linowej zależności między ocenami krytyków a liczbą sprzedanych płyt.

3. Jaki zakres lat obejmuje zbiór?

```
min_rok = min(df_albumy$year_of_pub)
max_rok = max(df_albumy$year_of_pub)
print(paste("Zbiór obejmuje lata:", min_rok,"-", max_rok))
```

```
## [1] "Zbiór obejmuje lata: 2000 - 2019"
```

4. W jakim roku wyszło najwięcej albumów a w jakim najmniej?

```
liczba_albumow_na_rok <- aggregate(album_title ~ year_of_pub, data = df_albumy, FUN = length)

rok_najwiecej_albumow <- liczba_albumow_na_rok[which.max(liczba_albumow_na_rok$album_title), "year_of
rok_najmniej_albumow <- liczba_albumow_na_rok[which.min(liczba_albumow_na_rok$album_title), "year_of_

print(paste("Najwięcej albumów wyszło w roku:", rok_najwiecej_albumow))
```

```
## [1] "Najwięcej albumów wyszło w roku: 2019"
```

```
print(paste("Najmniej albumów wyszło w roku:", rok_najmniej_albumow))
```

```
## [1] "Najmniej albumów wyszło w roku: 2009"
```

5. Ile albumów średnio nagrywa artysta?

```
liczba_albumów <- aggregate(album_title ~ artist_id, data = df_albumy, FUN = length)
średnia_albumów <- mean(liczba_albumów$album_title)
średnia_albumów
```

```
## [1] 2.313476
```

6. Średnia liczba sprzedanych płyt przez artystę:

```
suma_sprzedazy <- aggregate(num_of_sales ~ artist_id, data = df_albumy, sum)
srednia_sprzedaz <- mean(suma_sprzedazy$num_of_sales)

print(paste("Średnia liczba sprzedanych płyt przez artystę:", round(srednia_sprzedaz,0)))
```

```
## [1] "Średnia liczba sprzedanych płyt przez artystę: 1156841"
```

7. Który artysta sprzedał najwięcej płyt?

```
liczba_sprzedanych_plyt <- aggregate(num_of_sales ~ artist_id, data = df_albumy, sum)
najlepszy_artysta <- liczba_sprzedanych_plyt[which.max(liczba_sprzedanych_plyt$num_of_sales), ]

print(paste("Najwięcej płyt sprzedał artysta o identyfikatorze:", najlepszy_artysta$artist_id))
```

```
## [1] "Najwięcej płyt sprzedał artysta o identyfikatorze: 11290"
```

8. Który album sprzedawał się najlepieja a który najgorzej?

```
album_najlepiej_sprzedajacy_sie <- df_albumy[which.max(df_albumy$num_of_sales), ]
album_najmniej_sprzedajacy_sie <- df_albumy[which.min(df_albumy$num_of_sales), ]
print(paste("Album sprzedający się najlepiej to:", album_najlepiej_sprzedajacy_sie$album_title))
```

```
## [1] "Album sprzedający się najlepiej to: Decent Waterbuck"
```

```
print(paste("Album sprzedający się najsłabiej to:", album_najmniej_sprzedajacy_sie$album_title))
```

```
## [1] "Album sprzedający się najsłabiej to: Nervous Aggressive Trick Arabic"
```

9. Który artysta ma najlepiej oceniane albumy?

```
najlepiej_oceniany_artysta_RSM <- df_albumy[which.max(df_albumy$rolling_stone_critic), ]
print(paste("Najlepiej oceniany artysta według Rollinf Stone Critic to:", najlepiej_oceniany_artysta_
```

```
## [1] "Najlepiej oceniany artysta według Rollinf Stone Critic to: 43032"
```

```
najlepiej_oceniany_artysta_MTV <- df_albumy[which.max(df_albumy$mtv_critic), ]
print(paste("Najlepiej oceniany artysta według MTV to:", najlepiej_oceniany_artysta_MTV$artist_id))
```

```
## [1] "Najlepiej oceniany artysta według MTV to: 648"
```

```
najlepiej_oceniany_artysta_MM <- df_albumy[which.max(df_albumy$music_maniac_critic), ]
print(paste("Najlepiej oceniany artysta według Music Maniac Critic to:", najlepiej_oceniany_artysta_M
```

```
## [1] "Najlepiej oceniany artysta według Music Maniac Critic to: 23548"
```

```
df_albumy$ocena_łączna <- df_albumy$rolling_stone_critic + df_albumy$mtv_critic + df_albumy$music_mani
najlepiej_oceniany_artysta_łącznie <- df_albumy[which.max(df_albumy$ocena_łączna), ]
print(paste("Najlepiej oceniany artysta łącznie to:", najlepiej_oceniany_artysta_łącznie$artist_id))
```

```
## [1] "Najlepiej oceniany artysta łącznie to: 22206"
```

10. Czy najlepiej oceniany album należy do najlepiej ocenionego artysty?

```
najlepszy_album_RSM <- df_albumy[which.max(df_albumy$rolling_stone_critic), ]
najlepszy_album_MTV <- df_albumy[which.max(df_albumy$mtv_critic), ]
najlepszy_album_MM <- df_albumy[which.max(df_albumy$music_maniac_critic), ]

if (najlepszy_album_RSM$artist_id == najlepszy_album_MTV$artist_id &&
    najlepszy_album_RSM$artist_id == najlepszy_album_MM$artist_id) {
```

```
  print("Tak, najlepiej oceniany album należy do najlepiej ocenionego artysty.")
} else {
  print("Nie, najlepiej oceniany album nie należy do najlepiej ocenionego artysty.")
}
```

```
## [1] "Nie, najlepiej oceniany album nie należy do najlepiej ocenionego artysty."
```

11. Ile albumów przypada na każdy gatunek muzyczny?

```
gatunek_l_albumów<- aggregate(id ~ genre, data = df_albumy, FUN = length)
gatunek_l_albumów <- setNames(gatunek_l_albumów, c("genre", "nr_of_albums"))
gatunek_l_albumów <- gatunek_l_albumów[order(gatunek_l_albumów$nr_of_albums, decreasing = TRUE), ]
print(gatunek_l_albumów)
```

```
##           genre nr_of_albums
## 18        Indie         9384
## 28          Pop         7755
## 32          Rap         5788
## 23       Latino         3898
## 29     Pop-Rock         3880
## 34         Rock         3804
## 31         Punk         3787
## 9         Dance         3775
## 14       Gospel         2008
## 7   Compilation         2003
## 8       Country         1993
## 22        K-Pop         1986
## 17   Holy Metal         1979
## 36         Trap         1977
## 21         Jazz         1975
## 10  Death Metal         1968
## 25       Lounge         1958
## 20       J-Rock         1957
## 27       Parody         1957
## 16  Heavy Metal         1953
## 1   Alternative         1947
## 26        Metal         1934
## 4         Blues         1928
## 33        Retro         1924
## 6      Brit-Pop         1921
## 38      Western         1920
## 15    Hard Rock         1919
## 30  Progressive         1916
## 13         Folk         1912
## 35       Techno         1910
## 11   Deep House         1909
## 24         Live         1905
## 5      Boy Band         1894
## 2       Ambient         1874
## 3   Black Metal         1860
## 19 Indietronica         1858
## 12  Electro-Pop         1855
## 37    Unplugged         1829
```

12. Jaki gatunek muzyczne najlepiej się sprzedaje?

```r
sprzedaz_gatunku <- aggregate(num_of_sales ~ genre, data = df_albumy, sum)
najlepiej_sprzedajacy_sie_gatunek <- sprzedaz_gatunku[which.max(sprzedaz_gatunku$num_of_sales), ]
print(paste("Najlepiej sprzedający się gatunek muzyczny to:", najlepiej_sprzedajacy_sie_gatunek$genre
```

```
## [1] "Najlepiej sprzedający się gatunek muzyczny to: Indie"
```

13. Jaki gatunek ma najlepsze oceny?

```r
df_albumy$srednia_ocen <- (df_albumy$rolling_stone_critic + df_albumy$mtv_critic + df_albumy$music_ma
najlepiej_oceniany_gatunek <- df_albumy[which.max(df_albumy$srednia_ocen), ]

print(paste("Najlepiej oceniany gatunek muzyczny to:", najlepiej_oceniany_gatunek$genre))
```

```
## [1] "Najlepiej oceniany gatunek muzyczny to: Boy Band"
```

14. Czy z biegiem lat płyty sprzedają się lepiej czy gorzej?

```r
sprzedaz_na_rok <- aggregate(num_of_sales ~ year_of_pub, data = df_albumy, sum)
plot(sprzedaz_na_rok$year_of_pub, sprzedaz_na_rok$num_of_sales, type = "l", col = 'blue',
     xlab = "Rok publikacji", ylab = "Liczba sprzedanych płyt",
     main = "Zmiana sprzedaży płyt na przestrzeni lat")
```