Najlepsze albumy muzyczne

wg użytkowników RateYourMusic.com

Bartosz Łuksza, Rafał Głodek

Wprowadzenie

Muzyka towarzyszy człowiekowi od tysięcy lat. Zawsze stanowiła nieodłączną część naszej kultury. Niestety ograniczenia technologiczne przez długi czas nie pozwalały artystom utrwalić swoich dzieł. Fonografia narodziła się w XIX wieku, a jej największy rozkwit przypada na drugą połowę wieku XX. Z tego względu najstarsze oryginalne dzieła muzyczne, do których mamy obecnie dostęp, pochodzą poprzedniego stulecia. W ostanich latach rynek muzyczny przeżywa niebywały rozwkit. Każdego roku miliardy słuchaczy na całym świecie, przesłuchuje miliony nowych albumów, generując przychody rzędu dziesiątek miliardów dolarów ze sprzedaży nagrań. Rynek muzyczny jest jednak ściśle powiązany z wieloma innymi gałęziami biznesu, takimi jak: film, moda, czy technologie cyfrowe. Szacuje się, że każdego dnia na serwisy streamingowe trafia nawet 120 000 utworów! W tej sytuacji można pokusić się o stwierdzenie, że obecny przemysł muzyczny jest wręcz "przeładowany" muzyką. Warto zadać sobie pytanie, czy za ilością idzie również jakość?

W naszej pracy zajrzymy wgłąb współczesnej historii muzyki i przeanalizujemy bazę pięciu tysięcy najlepiej ocenianych albumów muzycznych przez użytkowników RateYourMusic.com - największego portalu do oceniania muzyki w internecie. Dane pochodzą z 12 grudnia 2021 r. i zostały pobrane z serwisu kaggle.com. Wyodrębniliśmy z nich następujące zmienne:

1. ***Album*** - nazwa albumu
   - Zawiera 4928 unikalne wartości
2. ***Artist Name*** - artysta (imię i nazwisko lub pseudonim artystyczny)
   - Zawiera 2787 unikalne wartości
   - 25 rekordów to *Various Artists*, czyli różni artyści, których jednak nie możemy wyodrębnić, więc pomijamy te wartości
3. ***Release Date*** - dokładna data wydania albumu (dzień/miesiąc/rok), wyodrębniliśmy z niej dwie zmienne:

a) ***Year*** - rok wydania albumu

- najmniejsza wartość: 1947
- największa wartość: 2021
- średnia arytmetyczna: 1987,46
- mediana: 1988
- wariancja: 253,23

b) ***Month*** - miesiąc wydania albumu

4. ***Genres*** - gatunki (lub gatunek), do których należy album

- Niekiedy trudno jest ustalić, do jakiego gatunku należy album. Wówczas określany jest mianem międzygatunkowego i klasyfikuje się go jako przynależnego do każdego z wymienionych gatunków.
- Możliwe wartości to: "Rock", "Hip Hop", "Pop" , "Jazz", "Soul", "Dance", "Techno", "Punk", "Metal", "Folk"

5. ***Descriptors*** - krótki opis albumu

- Opis zawiera kilka przymiotników najlepiej oddających charakter albumu, np. "melancholic, anxious, futuristic, alienation, existential, male vocals, atmospheric, lonely, cold, introspective"
- Opisy będą potrzebne, by sprawdzić jak oceniane są albumy w zależności od nastroju, jaki wywołują w słuchaczu

6. ***Average Rating*** - średnia ocen użytkowników

- Na stronie RateYourMusic.com użytkownicy mogą wystawiać albumom oceny w skali od 0 do 5, z uwzględnieniem "połówek"
- Średnie oceny, które są brane pod uwagę w tym spisie uwzględniają także wagi ocen - oceny użytkowników wykazujących się dużą aktywnością i doświadczeniem mają wyższą wagę niż tych, którzy oceniają muzykę sporadycznie
- najmniejsza wartość: 3,52
- największa wartość: 4,34
- średnia arytmetyczna: 3,771
- mediana: 3,75
- wariancja: 0,0098

7. ***Number of Ratings*** - liczba ocen użytkowników

- najmniejsza wartość: 260

- największa wartość: 70 400
- średnia arytmetyczna: 4084.511
- mediana: 1820
- wariancja: 36016085

8. ***Number of Reviews*** - liczba recenzji użytkowników
   - najmniejsza wartość: 0
   - największa wartość: 1 549
   - średnia arytmetyczna: 71.4492
   - mediana: 34
   - wariancja: 11766.56

Dogłębna analiza pozwoli nam znaleźć korelacje między różnymi zmiennymi ujętymi w zestawieniu i wyciągnąć nieoczywiste wnioski. W ten sposób nie tylko dowiemy się, jak muzyka rozwijała się w ubiegłych dekadach, ale także nakreślimy scieżkę jej dalszego rozwoju.

W jakich latach powstawało najwięcej "dobrych" albumów? Czy istnieje korelacja między średnią oceną użytkowników a datą wydania dzieła? Jakie są średnie ocen dla różnych gatunków muzycznych? Którzy artyści mogą się poszczycić najlepiej ocenianią dyskografią? Na te i wiele innych pytań odpowiemy w naszej pracy.

Analiza danych

Analiza rozkładów lat oraz średnich ocen

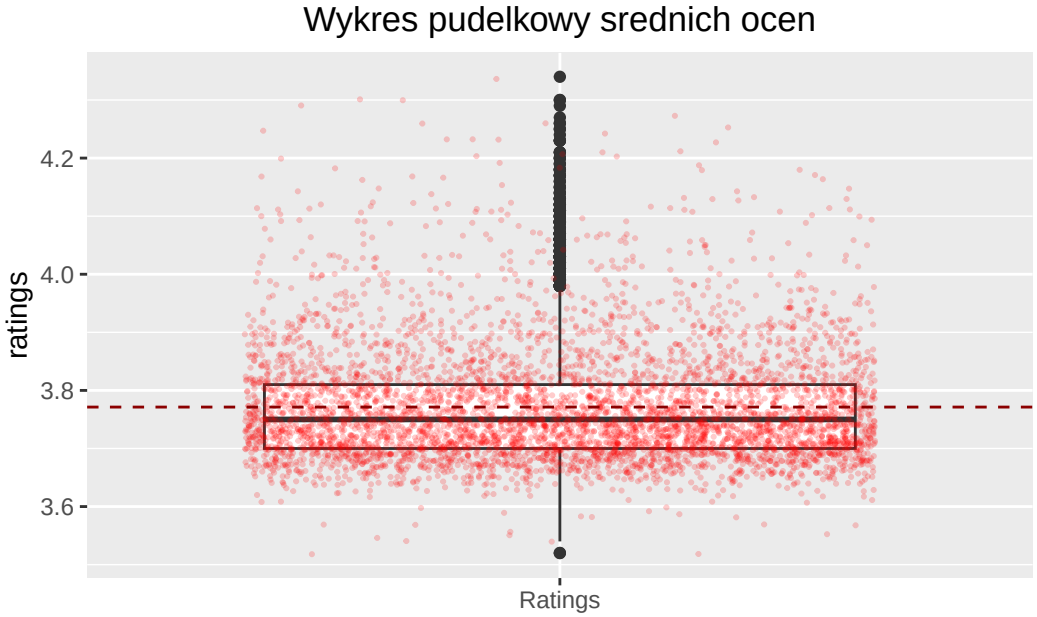
W pierwszej części naszej analizy zbadamy, jak rozkładają się średnie ocen wystawionych przez użytkowników oraz lata wydania albumów. Zaczniemy od wygenerowania wykresów pudełkowych dla każdej z nich oraz wyliczenia jego parametrów - mediany, pierwszego i trzeciego kwartyla, rozstępu międzykwartylowego oraz górnego i dolnego wąsa. Ponadto na wykres pudełkowy nałożymy także realizację naszej zmiennej w postaci punktów oraz jej średnią arytmetyczną.

Wykres pudełkowy dla średnich ocen prezentuje się następująco

```
Warning: package 'hrbrthemes' was built under R version 4.4.2
```

```
Warning: package 'viridis' was built under R version 4.4.2
```

```
Loading required package: viridisLite
```

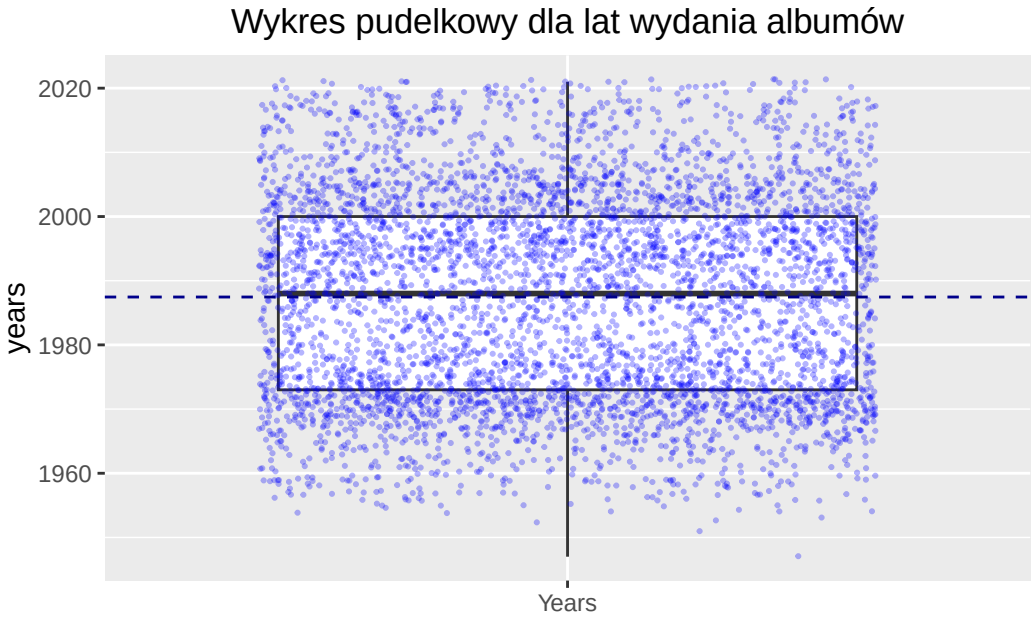


Korzystając z funkcji *boxplot.stats* wydobędziemy z wykresu najważniejsze dane. Przedstawimy je w formie tabeli

Wąs dolny	3.54
Pierwszy kwartyl	3.70
Mediana	3.75
Trzeci kwartyl	3.81
Wąs górny	3.97

Na bazie wykresu oraz tabeli możemy wyciągnąć kilka istotnych wniosków. Zgodnie z ideą wykresu pudełkoweg, zdecydowana większość punktów znajduje się w przedziale między wąsem dolnym a wąsem górnym, czyli $(3.54, 3.97)$. Obserwacje wypadające z niego możemy uznać za odstające. Jeden punkt znajduje się pod wąsem dolnym, natomiast możemy odnaleźć dużo więcej punktów osadzonych ponad wąsem górnym. W kontekście tematyki naszej pracy, możemy interpretować je jako ścisłą czołówkę albumów. Średnia arytmetyczna, będąca nieobciążonym estymatorem wartości oczekiwanej, jest większa niż mediana, a zatem mamy w tym przypadku do czynienia z rozkładem prawoskośnym. Oznacza to dla nas, że wyniki poniżej średniej są w naszej próbce przeważające. Oceny znacznie odbiegające od średniej są zatem dużą rzadkością i tym samym czołówka rankingu zarysowuje się nam coraz mocniej.

Teraz przeprowadzimy analogiczną analizę dla lat wydania albumów. Wygenerujmy dla danych wykres pudełkowy



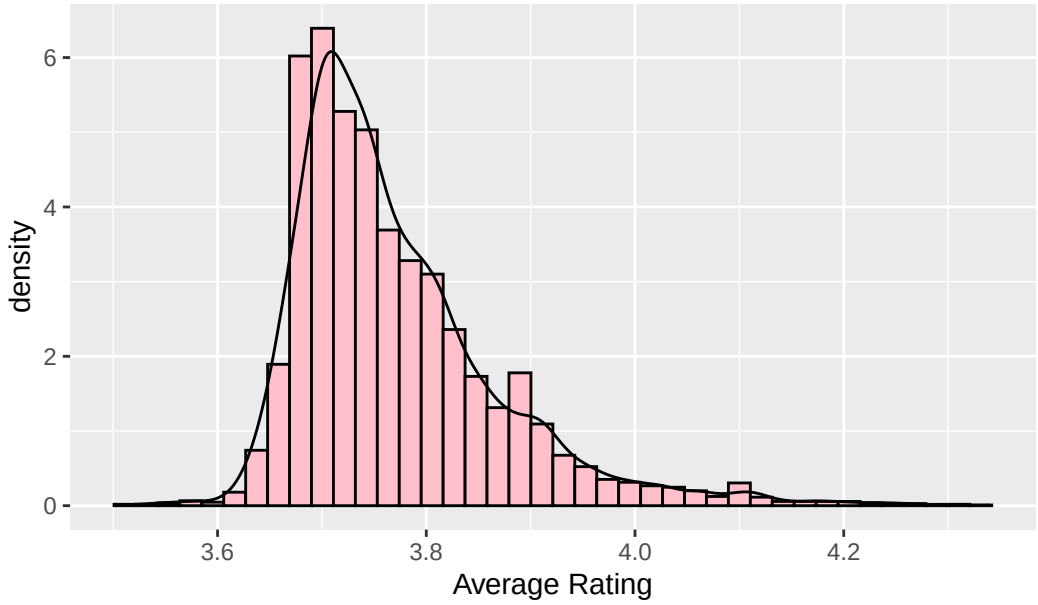
Znów wykorzystamy funkcję *boxplot.stats* i wyliczymy parametry tego wykresu pudełkowego. Zawiera je poniższa tabela.

Wąs dolny	1947
Pierwszy kwartyl	1973
Mediana	1988
Trzeci kwartyl	2000
Wąs górny	2021

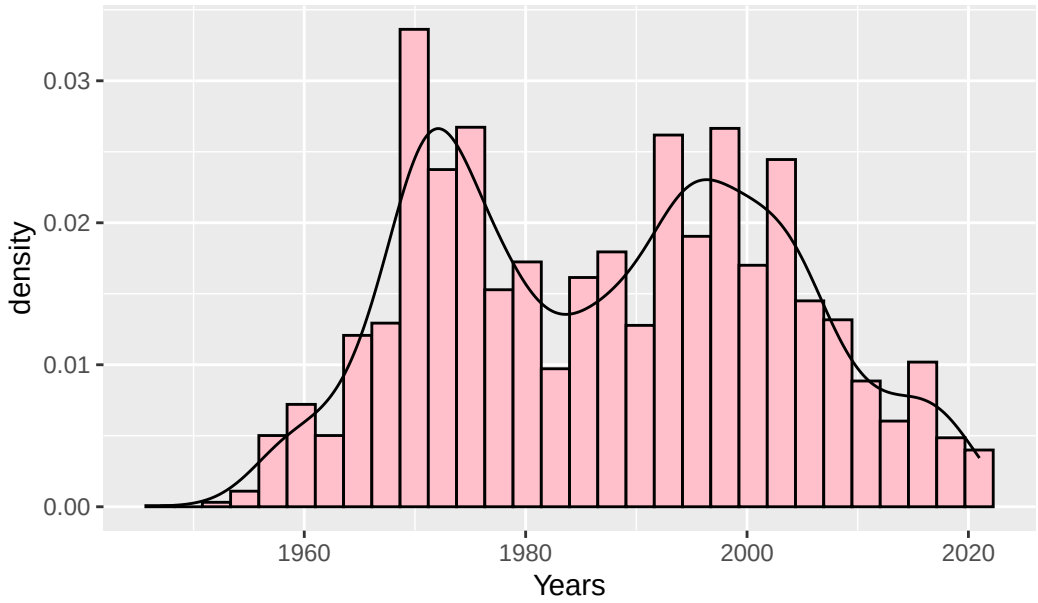
Wyciągnijmy teraz wnioski z tabeli i wykresu.

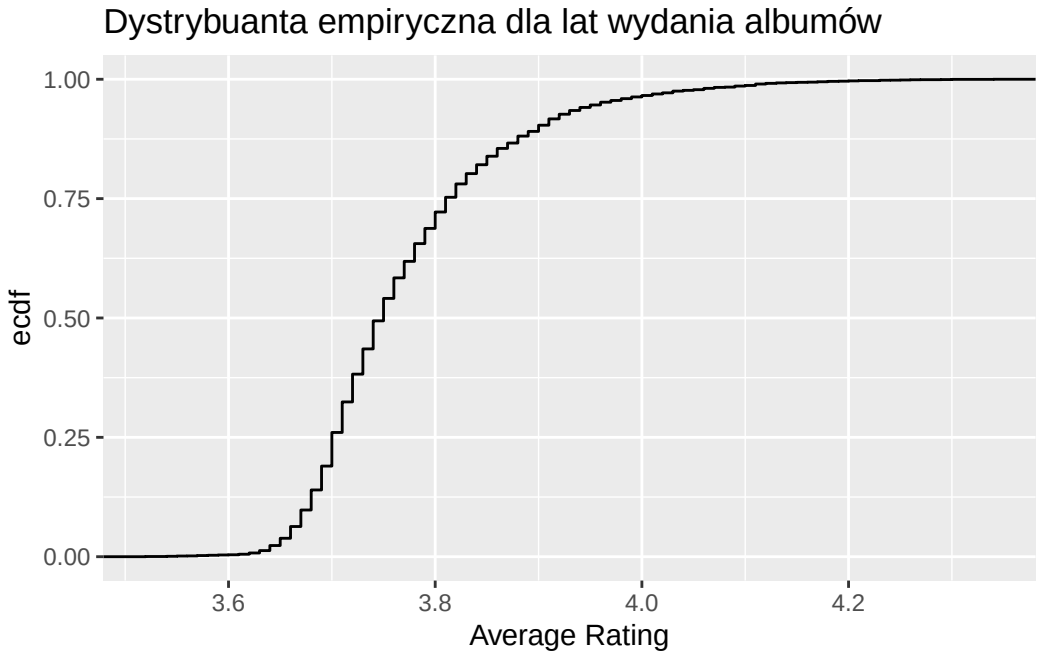
Dla każdej z tych zmiennych stworzymy histogram prawdopodobieństwa za pomocą funkcji $geom_histogram$ i dopasujemy do niego jądrowy estymator gęstości używając $geom_density$. Wynik dla średnich ocen prezentuje się następująco

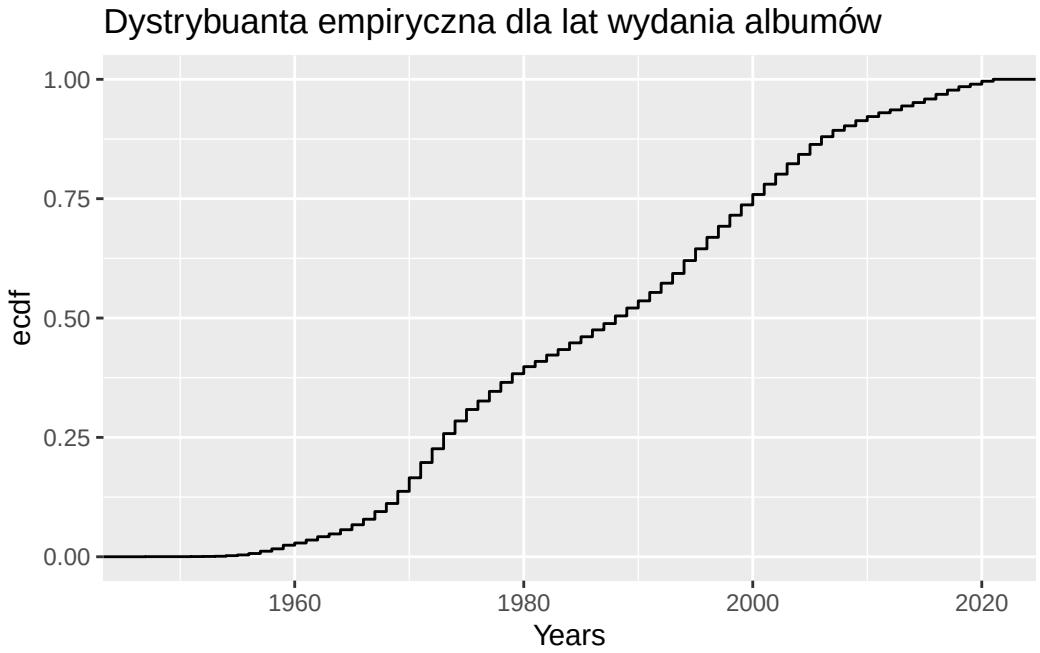
Histogram gestosci dla srednich ocen



Histogram gestosci dla lat wydania albumów





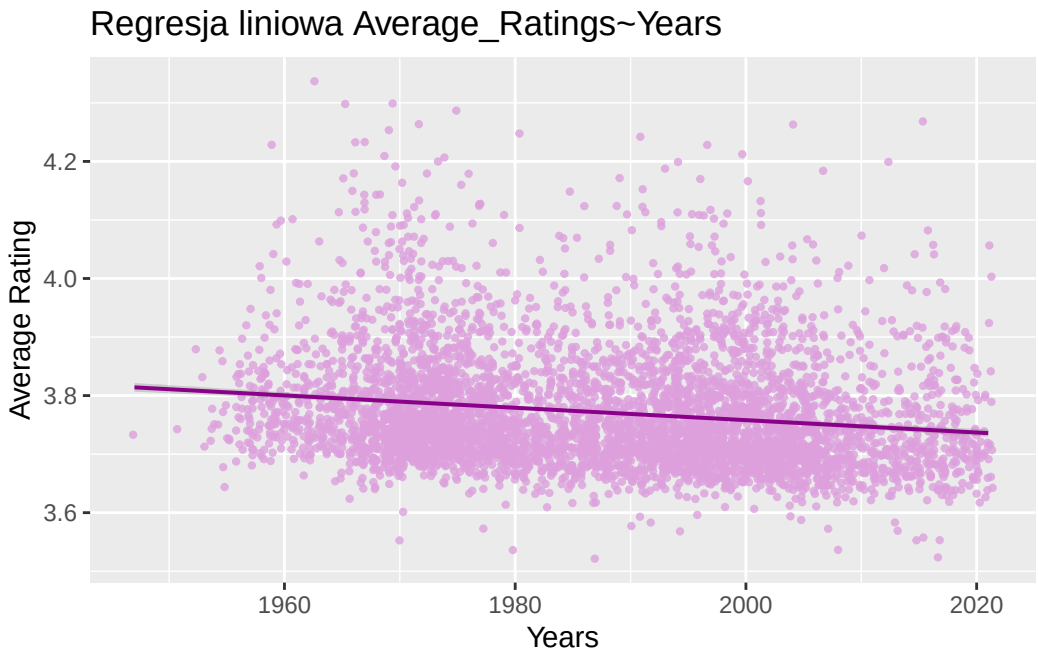


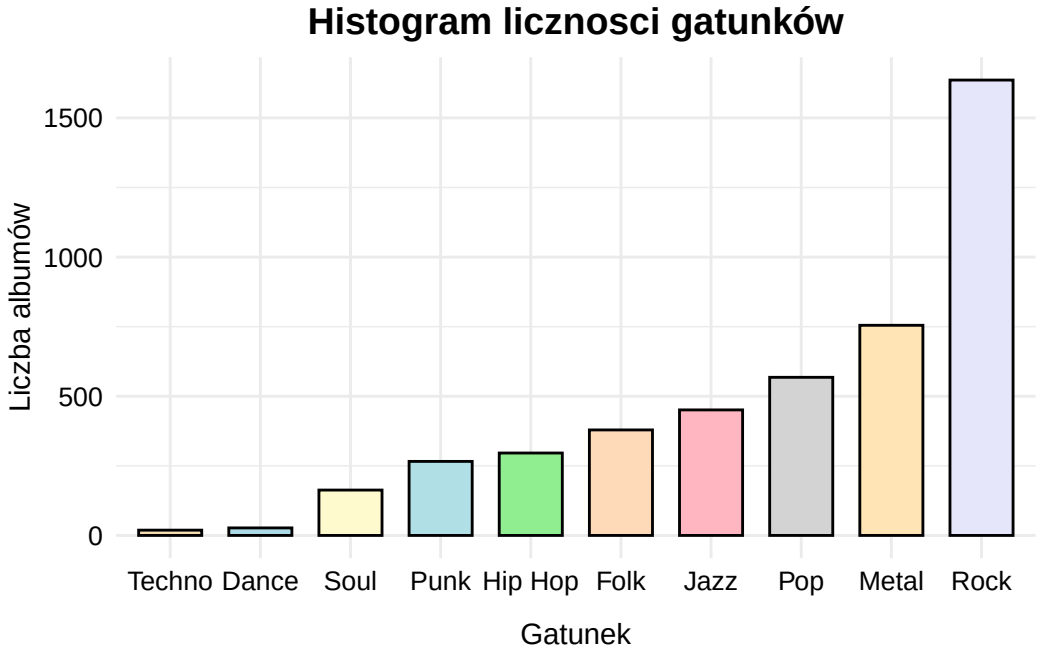
	Ratings	Years	Number of Ratings	Number of Reviews
Min	3.54	1947	260.0	0.0
First Quartile	3.70	1973	752.5	14.0
Median	3.75	1988	1820.0	34.0
Third Quartile	3.81	2000	4720.5	82.5

```
Maximum           3.97  2021           10649.0             185.0
```

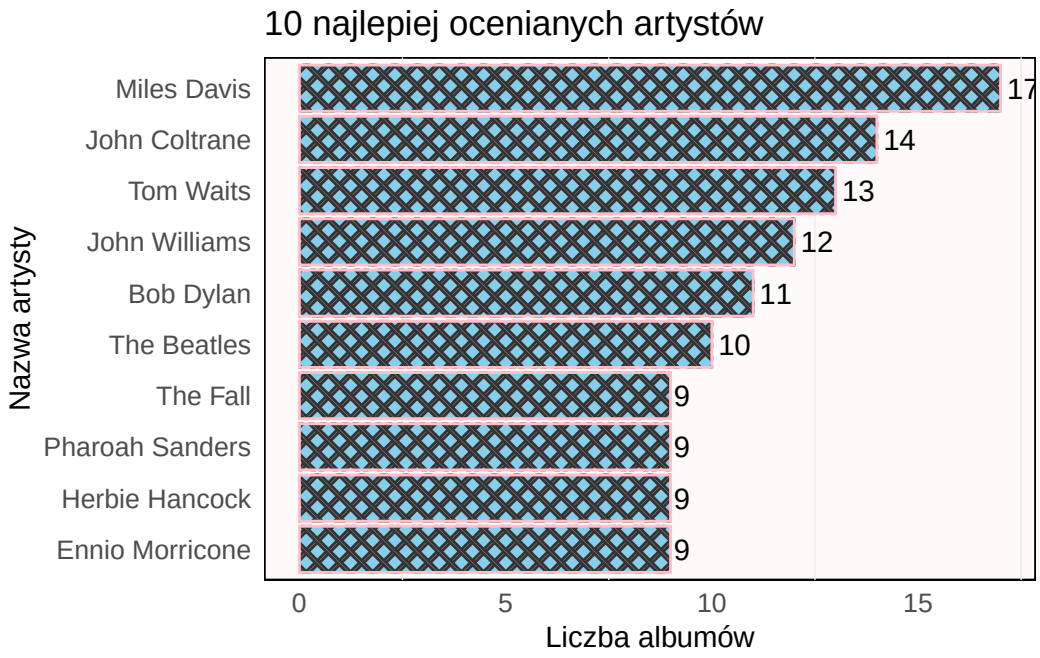
```r
lm_ratings_and_years = lm(ratings~years)
summary(lm_ratings_and_years)
```

```

Call:
lm(formula = ratings ~ years)

Residuals:
     Min       1Q   Median       3Q      Max
-0.25179 -0.06707 -0.02552  0.04131  0.54285

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  5.871e+00  1.724e-01   34.05   <2e-16 ***
years       -1.057e-03  8.676e-05  -12.18   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.09762 on 4998 degrees of freedom
Multiple R-squared:  0.02882,   Adjusted R-squared:  0.02863
F-statistic: 148.3 on 1 and 4998 DF,  p-value: < 2.2e-16


`geom_smooth()` using formula = 'y ~ x'
```



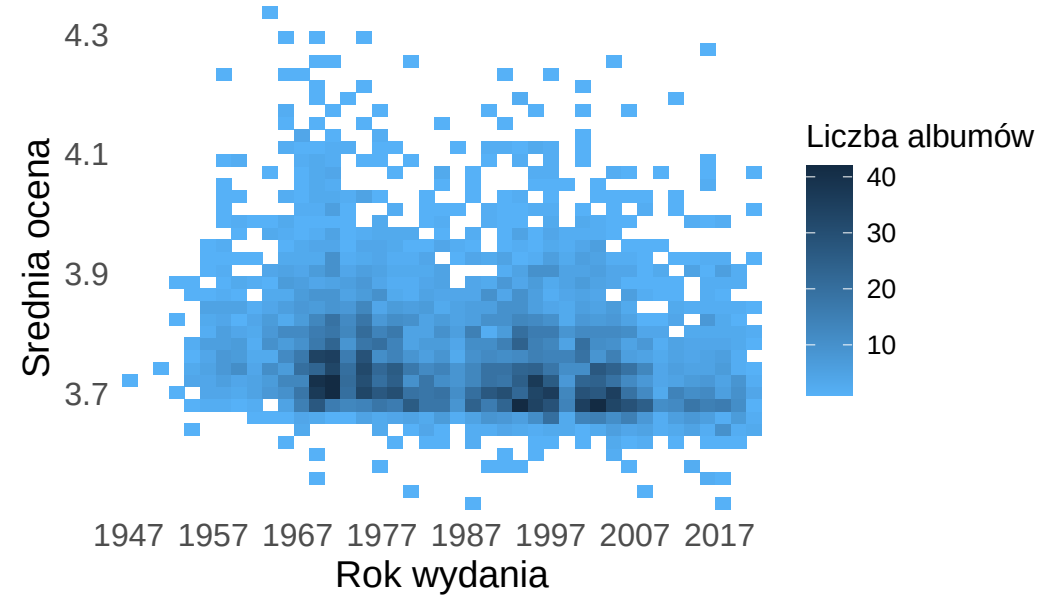



```
Warning: package 'ggpattern' was built under R version 4.4.2
```



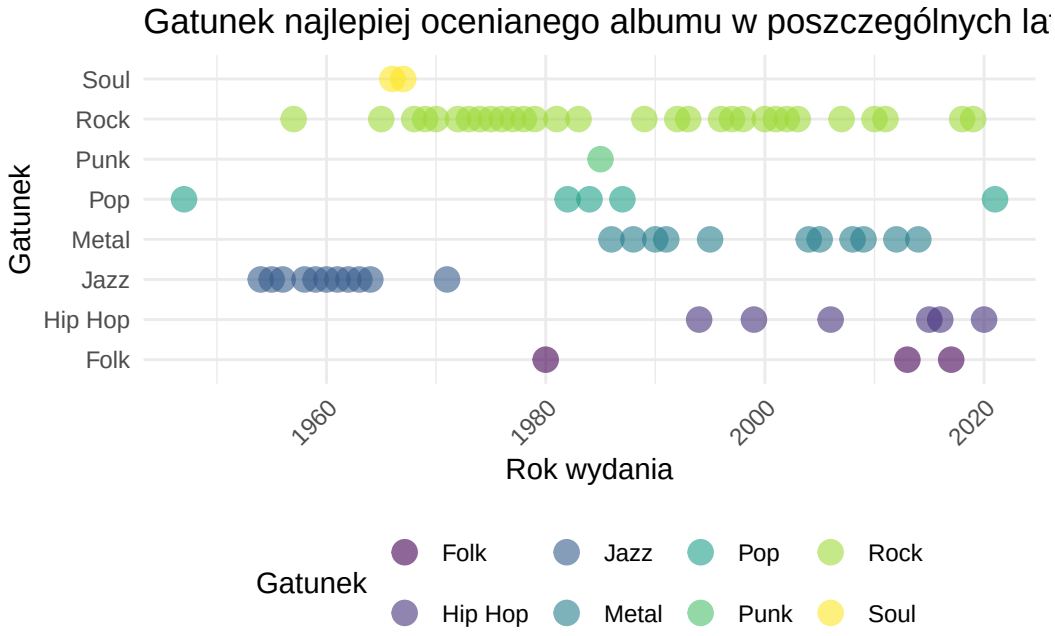

```
filtered_data <- rym %>%
  filter(!is.na(Years) & !is.na(`Average Rating`))  # Użycie backticków dla nazw kolumn ze sp

ggplot(filtered_data, aes(x = Years, y = as.numeric(`Average Rating`))) +
  geom_bin2d(bins = 40) +  # Więcej binów dla większej szczegółowości
  scale_fill_gradient(low = "#56B1F7", high = "#132B43", name = "Liczba albumów") +
  labs(
    title = "Dystrybucja albumów: Rok wydania vs Średnia ocena",
    x = "Rok wydania",
    y = "Średnia ocena",
    fill = "Liczba albumów"
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0.5, size = 16, face = "bold"),
    axis.title = element_text(size = 14),
    axis.text = element_text(size = 12),
    legend.title = element_text(size = 12),
    legend.text = element_text(size = 10),
    panel.grid.major = element_blank(),  # Usunięcie siatki dla czystszej wizualizacji
    panel.grid.minor = element_blank()
  ) +
  scale_x_continuous(
    breaks = seq(min(filtered_data$Years, na.rm = TRUE), max(filtered_data$Years, na.rm = TRU
```

```
    expand = c(0, 0)
  ) +
  scale_y_continuous(expand = c(0, 0))
```




```
rym <- rym %>%
  rowwise() %>%
  mutate(
    Basic_Genres_List = list(basic_genres[sapply(basic_genres, function(g) grepl(g, Genres, i
  ) %>%
  ungroup()

# Wybieramy tylko jeden gatunek (pierwszy z listy, jeśli istnieje)
rym <- rym %>%
  mutate(
    Basic_Genres = sapply(Basic_Genres_List, function(x) if (length(x) > 0) x[1] else NA)
  ) %>%
  filter(!is.na(Basic_Genres))

# Wybór najlepszego albumu dla każdego roku
best_album_per_year <- rym %>%
  group_by(Years) %>%
  filter("Average Rating" == max("Average Rating", na.rm = TRUE)) %>%
  slice_sample(n = 1) %>% # Losowy wybór w przypadku remisu
```

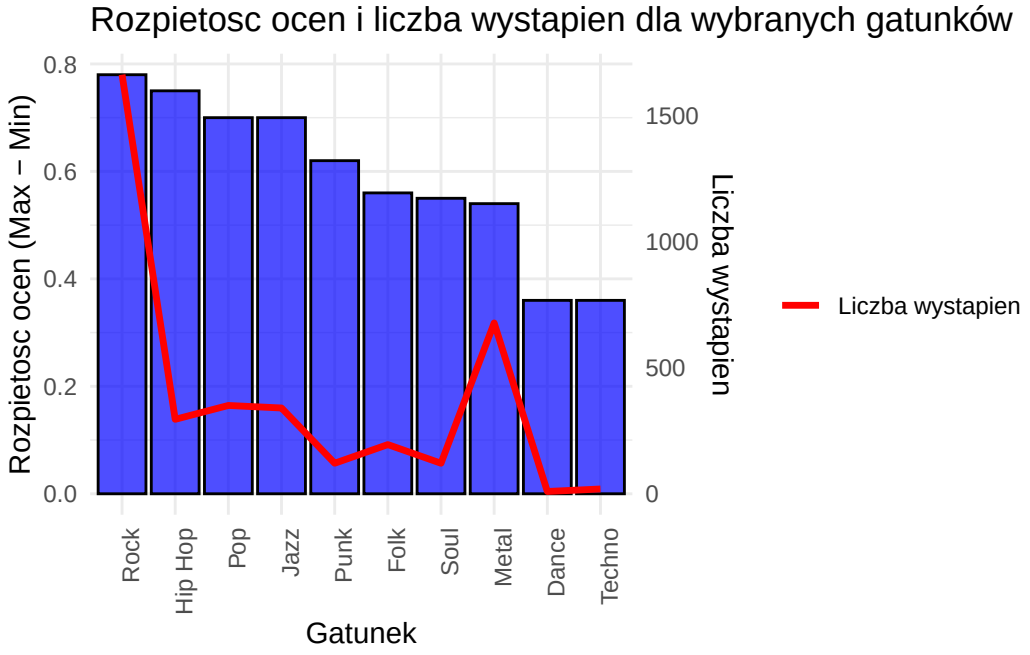
```
  ungroup()

# Wizualizacja najlepszego gatunku w każdym roku
ggplot(best_album_per_year, aes(x = Years, y = Basic_Genres, color = Basic_Genres)) +
  geom_point(size = 4, alpha = 0.6) +
  scale_color_viridis_d() +
  labs(
    title = "Gatunek najlepiej ocenianego albumu w poszczególnych latach",
    x = "Rok wydania",
    y = "Gatunek",
    color = "Gatunek"
  ) +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1),
    legend.position = "bottom"
  )
```




```
library(dplyr)
library(ggplot2)

# Lista podstawowych gatunków
basic_genres <- c("Rock", "Hip Hop", "Pop", "Jazz", "Soul", "Dance", "Techno", "Punk", "Metal
```

```r
# Liczenie wystąpień każdego gatunku w kolumnie Basic_genres
genre_counts <- sapply(basic_genres, function(genre) {
  sum(grepl(genre, rym$Basic_Genres, ignore.case = TRUE))
})

# Tworzenie ramki danych z wynikami
genre_counts_df <- data.frame(
  Genre = names(genre_counts),
  Count = genre_counts
) %>%
  arrange(desc(Count))

# Filtrowanie danych pod kątem wybranych gatunków
filtered_data <- rym %>%
  filter(sapply(Basic_Genres, function(genres) {
    any(grepl(paste(basic_genres, collapse = "|"), genres, ignore.case = TRUE))
  }))

# Obliczenie zakresu ocen dla każdego gatunku
rating_range <- sapply(basic_genres, function(genre) {
  genre_data <- filtered_data[grepl(genre, filtered_data$Basic_Genres, ignore.case = TRUE),
  if (nrow(genre_data) > 0) {
    min_rating <- min(as.numeric(genre_data$"Average Rating"), na.rm = TRUE)
    max_rating <- max(as.numeric(genre_data$"Average Rating"), na.rm = TRUE)
    range <- max_rating - min_rating
    return(c(min_rating, max_rating, range))
  } else {
    return(c(NA, NA, NA))
  }
})

# Tworzenie ramki danych z wynikami dla zakresu ocen
rating_range_df <- data.frame(
  Genre = basic_genres,
  Min.Rating = rating_range[1, ],
  Max.Rating = rating_range[2, ],
  Range = rating_range[3, ]
) %>%
  arrange(desc(Range))

# Połączenie liczby wystąpień z zakresem ocen
final_data <- merge(rating_range_df, genre_counts_df, by = "Genre")
```

```r
# Wykres
ggplot(final_data, aes(x = reorder(Genre, -Range))) +
  geom_bar(aes(y = Range), stat = "identity", fill = "blue", color = "black", alpha = 0.7) +
  geom_line(aes(y = Count / max(Count) * max(Range), group = 1, color = "Liczba wystąpień"),
  scale_y_continuous(
    name = "Rozpiętość ocen (Max - Min)",
    sec.axis = sec_axis(~ . * max(final_data$Count) / max(final_data$Range), name = "Liczba
  ) +
  labs(
    title = "Rozpiętość ocen i liczba wystąpień dla wybranych gatunków",
    x = "Gatunek",
    y = "Rozpiętość ocen (Max - Min)"
  ) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  scale_color_manual(values = "red", name = "")
```




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

# Lista podstawowych gatunków
basic_genres <- c("Rock", "Hip Hop", "Pop", "Jazz", "Soul", "Dance", "Techno", "Punk", "Meta
```

```r
# Filtrowanie danych i przypisanie gatunków z listy basic_genres
rym_filtered <- rym %>%
  filter(sapply(Basic_Genres, function(genres) {
    any(grepl(paste(basic_genres, collapse = "|"), genres, ignore.case = TRUE))
  }))

# Podsumowanie średnich ocen i liczby wystąpień dla każdego gatunku
genre_summary <- sapply(basic_genres, function(genre) {
  genre_data <- rym_filtered[grepl(genre, rym_filtered$Basic_Genres, ignore.case = TRUE), ]
  if (nrow(genre_data) > 0) {
    avg_rating <- mean(as.numeric(genre_data$`Average Rating`), na.rm = TRUE)
    count <- nrow(genre_data)
    return(c(avg_rating, count))
  } else {
    return(c(NA, 0))
  }
})

# Konwersja wyników do ramki danych
genre_summary_df <- data.frame(
  Genre = basic_genres,
  Average_Rating = genre_summary[1, ],
  Count = genre_summary[2, ]
) %>%
  filter(!is.na(Average_Rating)) %>%
  arrange(desc(Count)) %>%
  slice_max(order_by = Count, n = 20)  # Wybór 20 najczęściej występujących gatunków

# Tworzenie wykresu
ggplot(genre_summary_df, aes(x = reorder(Genre, -Count))) +
  geom_bar(aes(y = Average_Rating), stat = "identity", fill = "blue", color = "black", alpha
  geom_line(aes(y = Count / max(Count) * max(Average_Rating), group = 1), color = "red", size
  scale_y_continuous(
    name = "Średnia ocena",
    sec.axis = sec_axis(~ . * max(genre_summary_df$Count) / max(genre_summary_df$Average_Rati
  ) +
  labs(
    title = "Średnie oceny i liczba wystąpień dla gatunków",
    x = "Gatunek",
    y = "Średnia ocena"
  ) +
  theme_minimal() +
```

```
  theme(
    plot.title = element_text(hjust = 0.5, size = 16, face = "bold"),
    axis.title = element_text(size = 14),
    axis.text.x = element_text(angle = 90, hjust = 1, size = 10),  # Obrót etykiet na osi X
    legend.position = "top"
  )
```

```
Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
i Please use `linewidth` instead.
```