Analiza statystyczna danych dialogowych z serialu Gra o tron

Mikołaj Wielgos

20 stycznia 2022 r.

Spis treści

Wstęp

Projekt na przedmiot Rachunek prawdopodobieństwa i statystyka 2021/22.

Obejmuje on analizę statystyczną dialogów w serialu *Gra o tron*

Źródło danych opracowywanych *www.kaggle.com/gopinath15/gameofthrones*

Czyszczenie danych

Importujemy pakiet DBI by łączyć się z bazą danych SQLite.

```
library(DBI)
```

```
## Warning: package 'DBI' was built under R version 4.1.2
```

```
library(dbplot)
```

```
## Warning: package 'dbplot' was built under R version 4.1.2
```

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.1.2
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.1.2
```

Łączymy się z bazą danych "got-dataset.db"

```
con <- dbConnect(drv=RSQLite::SQLite(), dbname="data/got-dataset.db")
```

Sprawdzamy jakie tabele zawiera baza danych.

```
dbListTables(con)
```

```
## [1] "got-dialogues"
```

Interesują nas dialogi wypowiedziane tylko przez postacie w serialu (wpis Speaker nie może być pusty). Przykładowy błędny wiersz

```
query <- "
SELECT Text, Speaker, Episode, Season
FROM [got-dialogues]
WHERE Speaker=''
LIMIT 1"
dbGetQuery(con, query)
```

```
## [1] Text    Speaker Episode Season
## <0 rows> (or 0-length row.names)
```

Kwerenda usuwająca wiersze (wraz z informacją o ilości usuniętych)

```
query <- "
DELETE
FROM [got-dialogues]
WHERE SPEAKER = ''"
res <- dbSendStatement(con, query)
dbGetRowsAffected(res)
```

```
## [1] 0
```

```
dbClearResult(res)
```

Usuwamy dialogi postaci w tle, przedstawianych jako np. Woman #4 Przykładowy błędny wiersz

```
query <- "
SELECT Text, Speaker, Episode, Season
FROM [got-dialogues]
WHERE Speaker LIKE '%#%'
LIMIT 1"
dbGetQuery(con, query)
```

```
## [1] Text    Speaker Episode Season
## <0 rows> (or 0-length row.names)
```

Kwerenda usuwająca wiersze (wraz z informacją o ilości usuniętych)

```
query <- "
DELETE
FROM [got-dialogues]
WHERE Speaker LIKE '%#%'"
res <- dbSendStatement(con, query)
dbGetRowsAffected(res)
```

```
## [1] 0
```

```
dbClearResult(res)
```

Usuwamy wspólne dialogi, dotyczy pozycji typu 'ALL TOGETHER', 'ALL THREE', 'ALL AT THE BACK'. Przykładowy błędny wiersz

```
query <- "
SELECT Text, Speaker, Episode, Season
FROM [got-dialogues]
WHERE Speaker LIKE '%ALL%'
LIMIT 1"
dbGetQuery(con, query)
```

```
## [1] Text    Speaker Episode Season
## <0 rows> (or 0-length row.names)
```

Kwerenda usuwająca wiersze (wraz z informacją o ilości usuniętych)

```
query <- "
DELETE
FROM [got-dialogues]
WHERE Speaker LIKE '%ALL%'"
res <- dbSendStatement(con, query)
dbGetRowsAffected(res)
```

```
## [1] 0
```

```
dbClearResult(res)
```

Usuwamy pojedyncze, mało znaczące wypowiedzi. Przykładowy błędny wiersz

```
query <- "
SELECT Text, Speaker, Episode, Season
FROM [got-dialogues]
GROUP BY Speaker
HAVING COUNT(*)=1
LIMIT 1"
dbGetQuery(con, query)
```

```
## [1] Text    Speaker Episode Season
## <0 rows> (or 0-length row.names)
```

Kwerenda usuwająca wiersze (wraz z informacją o ilości usuniętych)

```
query <- "
DELETE
FROM [got-dialogues]
where Speaker in (SELECT Speaker
FROM [got-dialogues]
GROUP BY Speaker
HAVING COUNT(*)=1)"
res <- dbSendStatement(con, query)
dbGetRowsAffected(res)
```

```
## [1] 0
```

```
dbClearResult(res)
```

W bazie danych pojawiają się wpisy typu Speaker=‘Roose’ oraz Speaker=‘ROOSE’, dlatego trzymamy się jednej wersji (WIELKIE LITERY)

```
query <- "
UPDATE [got-dialogues]
SET Speaker = UPPER(Speaker)"
res <- dbSendStatement(con, query)
dbGetRowsAffected(res)
```

```
## [1] 24096
```

```
dbClearResult(res)
```

Po przeczyszczeniu przykładowe wpisy w bazie wyglądają następująco (kolumna ‘Text’ ograniczona do 20 znaków, by zwiększyć czytelność).

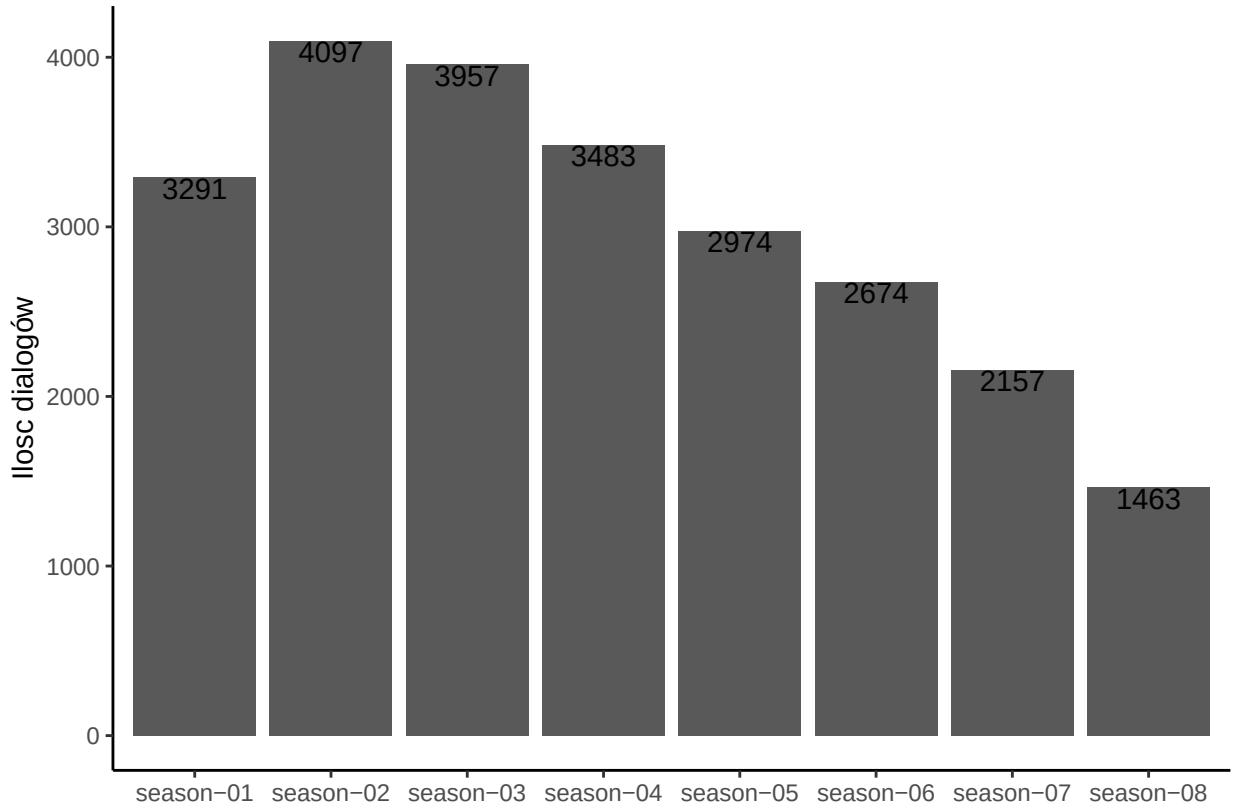
```
query <- "
SELECT substr(Text,1,20)||'...' as 'Text', Speaker, Episode, Season
FROM [got-dialogues]
LIMIT 5"
dbGetQuery(con, query)
```

```
##                       Text      Speaker             Episode    Season
## 1  What d'you expect? ... WAYMAR ROYCE e1-Winter is Coming season-01
## 2  I've never seen wil...         WILL e1-Winter is Coming season-01
## 3  How close did you g... WAYMAR ROYCE e1-Winter is Coming season-01
## 4  Close as any man wo...         WILL e1-Winter is Coming season-01
## 5  We should head back...        GARED e1-Winter is Coming season-01
```

Analiza eksploracyjna

Ilość dialogów w poszczególnych sezonach

```
df <- dbGetQuery(con,"
SELECT Season, COUNT(*)
FROM [got-dialogues]
GROUP BY Season")
ggplot(data=df, aes(x = df[,1], y=df[,2])) +
  geom_bar(stat = "identity") +
  geom_text(aes(label = df[,2]), vjust = 1) +
  labs(x="", y="Ilość dialogów") +
  theme_classic()
```



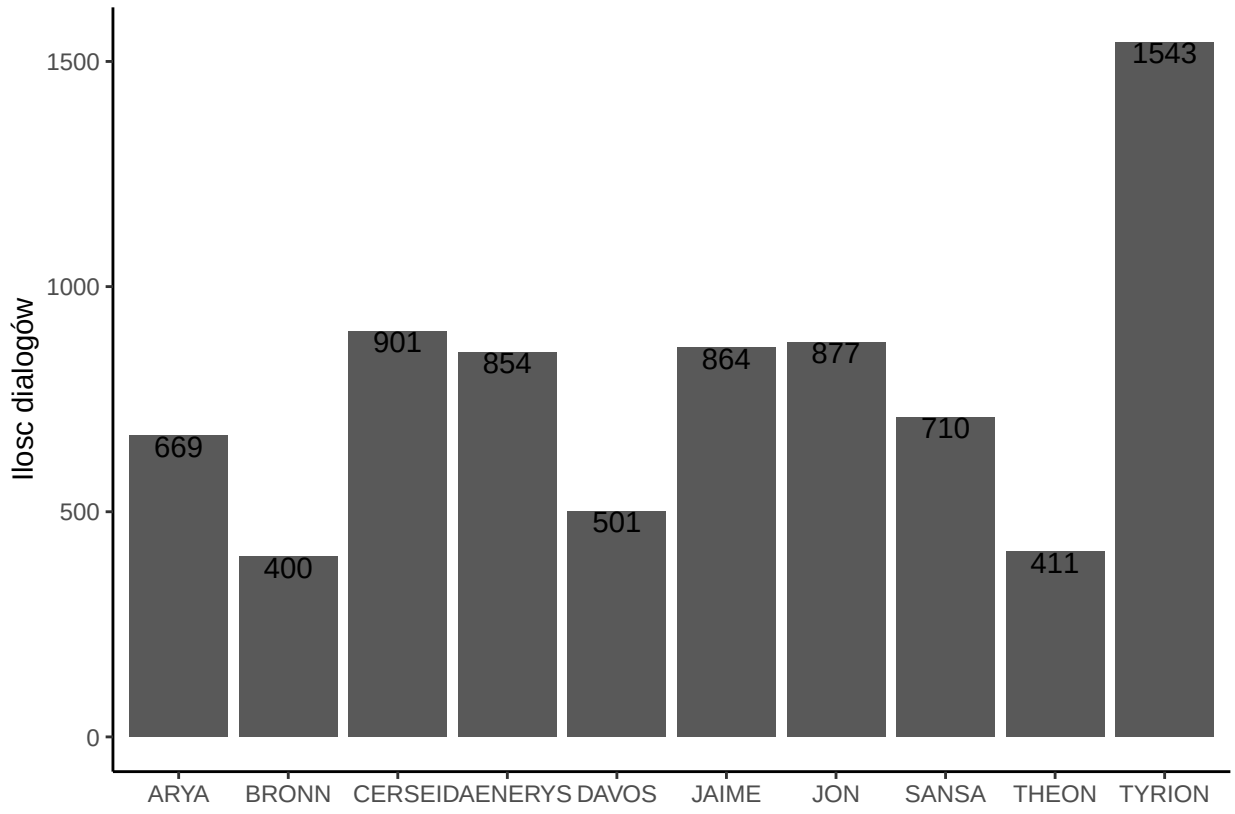
Wskaźniki:

- Średnia 3012

- Mediana 3132.5
- Wariancja 8.0229229×10^5
- Odchylenie standardowe 895.707701

Ilość dialogów poszczególnych bohaterów (wszystkie sezony, 10 największych)

```
df <- dbGetQuery(con,"
SELECT Speaker, COUNT(*)
FROM [got-dialogues]
GROUP BY Speaker
ORDER BY 2 DESC
LIMIT 10")
ggplot(data=df, aes(x = df[,1], y=df[,2])) +
  geom_bar(stat = "identity") +
  geom_text(aes(label = df[,2]), vjust = 1) +
  labs(x="", y="Ilość dialogów") +
  theme_classic()
```



By policzyć wskaźniki wszystkich bohaterów, ponownie wybieram dane (tym razem bez limitu)

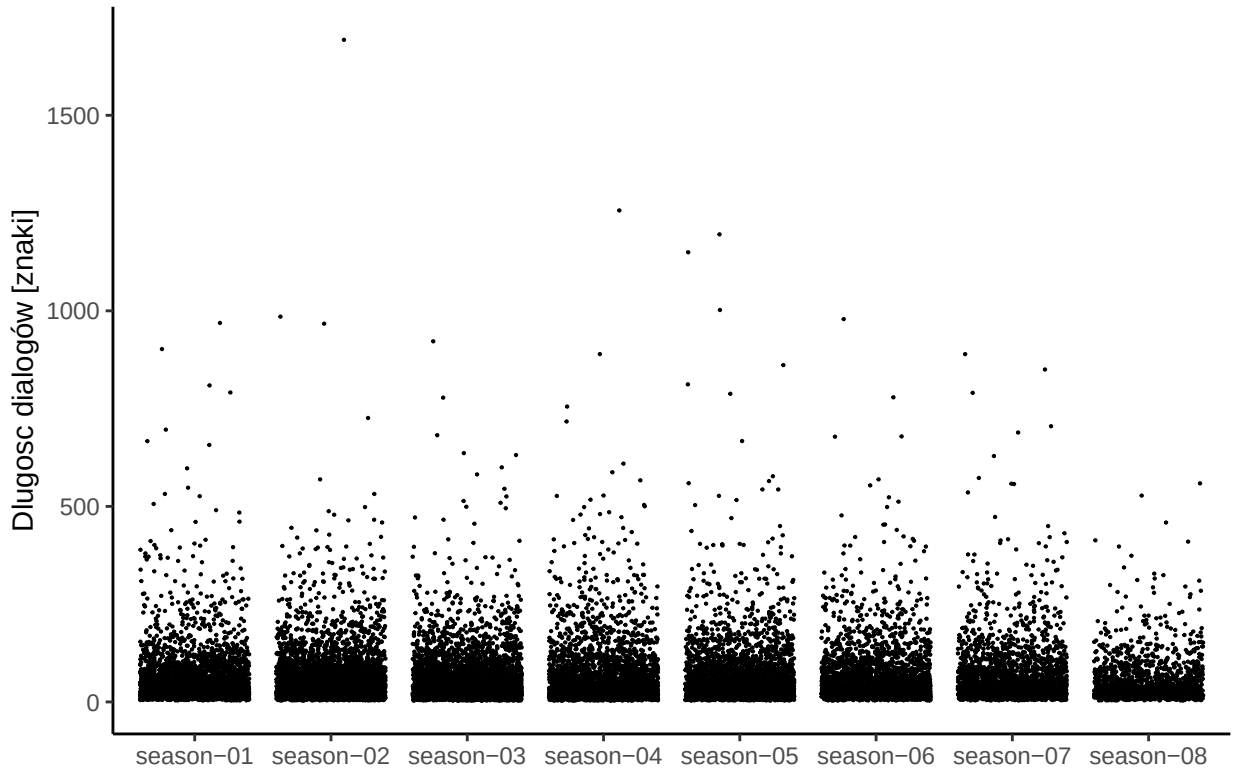
```
df <- dbGetQuery(con,"
SELECT Speaker, COUNT(*)
FROM [got-dialogues]
GROUP BY Speaker")
```

Wskaźniki (wszystkich bohaterów):

- Średnia 50.0956341
- Mediana 10
- Wariancja 1.7203058×10^4
- Odchylenie standardowe 131.1604266

Długości dialogów (znaki) w poszczególnych sezonach

```
df <- dbGetQuery(con,"
SELECT Season,length(Text)
FROM [got-dialogues]")
ggplot(data=df, aes(x = df[,1], y=df[,2])) +
  geom_jitter(size = 0.1) +
  labs(x="", y="Długość dialogów [znaki]", caption="(Kropka odpowiada pojedynczemu dialogowi)") +
  theme_classic()
```



(Kropka odpowiada pojedynczemu dialogowi)

Wskaźniki:

- Średnia 61.7116949
- Mediana 39
- Wariancja 5249.6609315
- Odchylenie standardowe 72.4545439

NA KONIEC /

```
dbDisconnect(con)
unlink("data/got-dataset.db")
```