

# Analiza statystyczna danych dotyczących systemu wypożyczalni rowerów w rejonie zatoki San Francisco

Marceli Mietła

## Spis treści

<b>Wstęp</b>	<b>1</b>
<b>Czyszczenie danych</b>	<b>2</b>
Nieprawidłowy format daty . . . . .	2
Temperatura podana w skali Fahrenheita . . . . .	3
Prędkość wiatru w milach na godzinę . . . . .	3
Przejazdy trwające dłużej niż 24 godziny . . . . .	3
<b>Analiza eksploracyjna</b>	<b>4</b>
Liczba wypożyczeń z podziałem na poszczególne dni tygodnia . . . . .	4
Liczba wypożyczeń z podziałem na poszczególne godziny . . . . .	6
Rozkład długości wypożyczeń według dnia tygodnia . . . . .	9
Histogram długości wypożyczenia . . . . .	11
Zależność temperatury od ilości wypożyczeń . . . . .	13
<b>Model statystyczny</b>	<b>16</b>
Regresja liniowa . . . . .	16
Regresja liniowa z regulacją L1 . . . . .	24
<b>Podsumowanie</b>	<b>25</b>

## Wstęp

Analiza statystyczna systemu wypożyczalni rowerów w rejonie zatoki San Francisco w latach 2013-2015.  
Źródło danych: *Kaggle - SF Bay Area Bike Share*

# Czyszczenie danych

## Nieprawidłowy format daty

Nieprawidłowy format daty w plikach trip.csv, station.csv, weather.csv.

W SQLite daty muszą być zapisane w formacie *YYYY-MM-DD HH:MM*. Skrypt napisany w pythonie zamienia wszystkie daty z formatu *MM/DD/YYYY HH:MM* na przystosowany do SQLite.

```
import re
import os
import csv

def convert_date_format(date_str):
    # Regex pattern to match the date format
    pattern = r"(\d{1,2})/(\d{1,2})/(\d{4})(?: (\d{1,2}):(\d{1,2}))?"
    match = re.match(pattern, date_str)

    if not match:
        print(date_str)
        raise ValueError(f"Invalid date format: {date_str}")

    month, day, year, hours, minutes = match.groups()

    if hours is None or minutes is None:
        hours = "00"
        minutes = "00"

    # Ensure day and month are two digits
    hours = hours.zfill(2)
    minutes = minutes.zfill(2)
    day = day.zfill(2)
    month = month.zfill(2)

    # Return formatted date
    return f"{year}-{month}-{day} {hours}:{minutes}"

def process_csv(file_path, columns_name):
    # Read the CSV file
    with open(file_path, mode='r', newline='') as csvfile:
        reader = csv.DictReader(csvfile)
        fieldnames = reader.fieldnames
        rows = list(reader)

    # Convert the specified column to the correct format
    for row in rows:
        for column_name in columns_name:
            row[column_name] = convert_date_format(row[column_name])

    # Write the updated rows back to the same CSV file
    with open(file_path, mode='w', newline='') as csvfile:
        writer = csv.DictWriter(csvfile, fieldnames=fieldnames)
        writer.writeheader()
```

```

        writer.writerows(rows)

def process_multiple_csv(files_columns):
    for file_path, columns_name in files_columns.items():
        process_csv(file_path, columns_name)

files_columns = {
    os.path.join '..', 'data', 'trip.csv': ['start_date', 'end_date'],
    os.path.join '..', 'data', 'station.csv': ['installation_date'],
    os.path.join '..', 'data', 'weather.csv': ['date']
}

process_multiple_csv(files_columns)

```

## Temperatura podana w skali Fahrenheita

Dla łatwiejszej analizy wyników zamieniamy temperaturę na skalę Celsjusza

```

update weather
    SET
        max_temperature_f = round((max_temperature_f - 32) * 5/9, 2),
        mean_temperature_f = round((mean_temperature_f - 32) * 5/9, 2),
        min_temperature_f = round((min_temperature_f - 32) * 5/9, 2)

```

## Prędkość wiatru w milach na godzinę

```

update weather
    set
        mean_wind_speed_mph = mean_wind_speed_mph*1.609344

```

## Przejazdy trwające dłużej niż 24 godziny

Usuujemy przejazdy które trwały dłużej niż 24 godziny

```

delete from trip
where duration > 24*60*60;

```

## Analiza eksploracyjna

### Liczba wypożyczeń z podziałem na poszczególne dni tygodnia

```
query <- "
select case cast (strftime('%w', trip.start_date) as integer)
  when 0 then 'Niedziela'
  when 1 then 'Poniedzialek'
  when 2 then 'Wtorek'
  when 3 then 'Sroda'
  when 4 then 'Czwartek'
  when 5 then 'Piatek'
  else 'Sobota' end as weekday,
  count(*) as trip_count
from trip
group by strftime('%u', start_date)"

trip_day_week <- dbGetQuery(con, query)

trip_day_week[,1] <- factor(trip_day_week[,1], levels = trip_day_week[,1])

ggplot(data=trip_day_week, aes(x = weekday, y= trip_count)) +
  geom_bar(stat = "identity", fill="darkblue") +
  geom_text(aes(label = trip_count), vjust = 2, color="white") +
  labs(title = "Liczba wypożyczeń z podziałem na poszczególne dni tygodnia", x="Dzień tygodnia", y="Ilość wypożyczeń") +
  theme_minimal()
```



Wskaźniki:

```
summary(trip_day_week[,2])
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  38345   77008   115849   95666  119609  122235
```

```
var(trip_day_week[,2])
```

```
## [1] 1387898555
```

```
sd(trip_day_week[,2])
```

```
## [1] 37254.51
```

Wnioski:

- Najwięcej wypożyczeń odbywa się od poniedziałku do piątku, co sugeruje, że system wypożyczalni jest intensywnie wykorzystywany w dni robocze.
- Wtorek jest dniem o najwyższej liczbie wypożyczeń (122 235), natomiast niedziela to dzień o najmniejszej liczbie wypożyczeń (38 345).
- Wykres wskazuje na użytkowanie głównie przez osoby dojeżdżające do pracy lub szkoły.

## Liczba wypożyczeń z podziałem na poszczególne godziny

```
query <- "
select
  strftime('%H', trip.start_date) as hour,
  count(*) as trip_count
from trip
group by hour"

trip_hours <- dbGetQuery(con,query)

ggplot(data=trip_hours, aes(x = hour, y=trip_count)) +
  geom_bar(stat = "identity", fill="darkblue") +
  geom_text(aes(label = trip_count), vjust = -0.5, color = "black", size = 2) +
  labs(
    title = "Liczba wypożyczeń z podziałem na poszczególne godziny",
    x="Godzina", y="Ilość wypożyczeń") +
  theme_minimal()
```



```
summary(trip_hours[,2])
```

##	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
##	334	4194	28129	27903	36759	85846

```
var(trip_hours[,2])
```

```
## [1] 675907042
```

```
sd(trip_hours[,2])
```

```
## [1] 25998.21
```

### Wnioski:

- Szczytowe godziny (poranna i popołudniowa) odpowiadają typowym godzinom dojazdów do pracy i z pracy.
- W nocy i bardzo wczesnym rankiem liczba wypożyczeń jest znikoma.

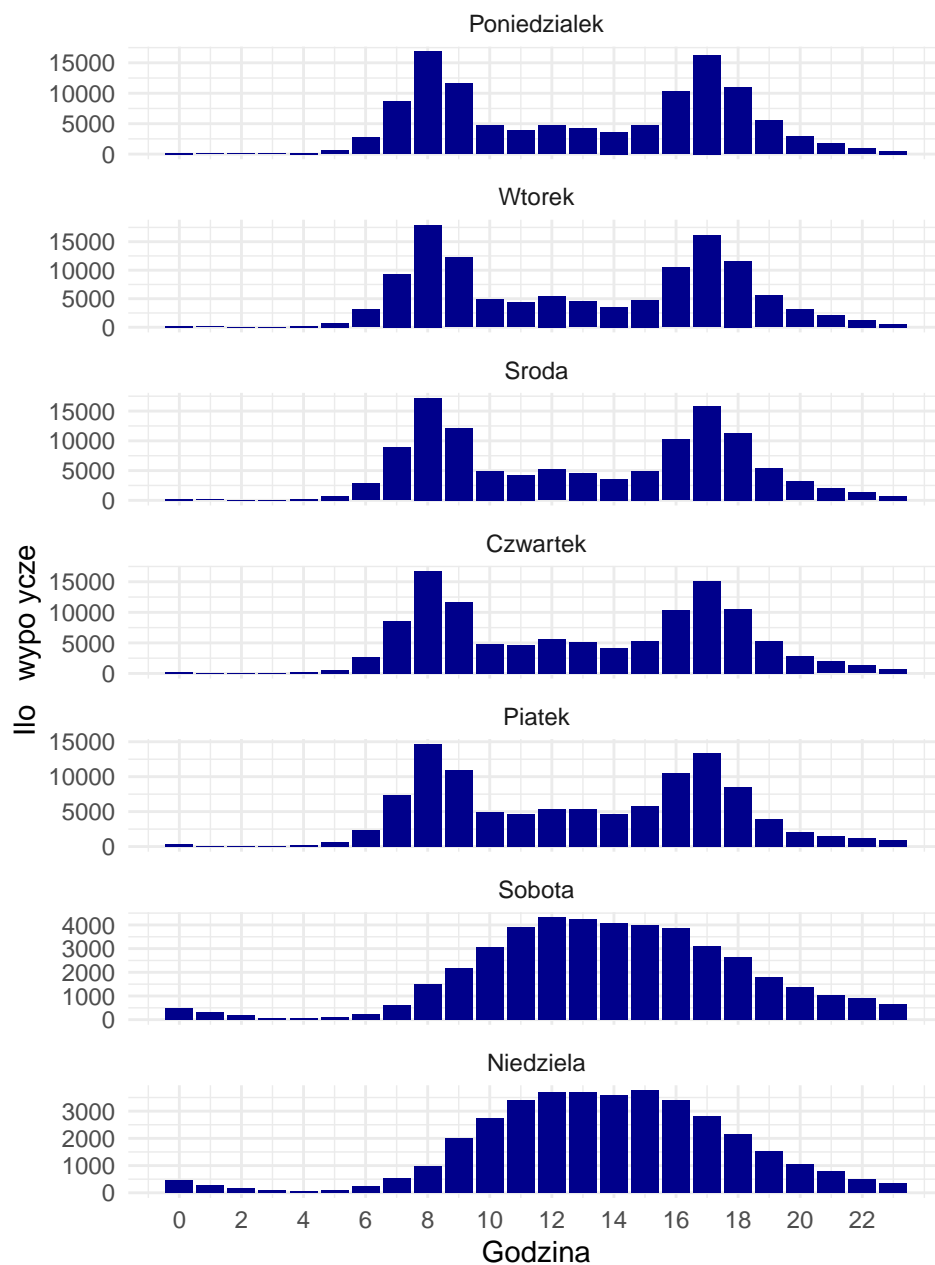
```
df <- dbGetQuery(con,"
select
  strftime('%H', start_date) AS hour,
  strftime('%u', trip.start_date) as day_of_week_number,
  case cast (strftime('%u', trip.start_date) as integer)
  when 1 then 'Poniedzialek'
  when 2 then 'Wtorek'
  when 3 then 'Sroda'
  when 4 then 'Czwartek'
  when 5 then 'Piatek'
  when 6 then 'Sobota'
  else 'Niedziela' end as day_of_week,
  COUNT(*) AS trip_count
from trip
group by hour, day_of_week"
)

df$day_of_week_number <- as.numeric(df$day_of_week_number)

df$day_of_week <- reorder(df$day_of_week, df$day_of_week_number)

ggplot(data=df, aes(x = as.numeric(hour), y = trip_count)) +
  geom_bar(stat = "identity", fill="darkblue") +
  facet_wrap(~ day_of_week, ncol=1, scales="free_y") +
  labs(title = "Ilość wypożyczeń w zależności od godziny",
       x = "Godzina",
       y = "Ilość wypożyczeń") +
  scale_x_continuous(breaks = seq(0, 23, by = 2), labels = seq(0, 23, by = 2)) +
  theme_minimal()
```

## Ilość wypożyczeń w zależności od godziny



### Wnioski:

- W dni powszednie występują dwa wyraźne szczyty: o godzinie 8:00 i 17:00.
- W weekendy liczba wypożyczeń jest równomiernie rozłożona w ciągu dnia, bez dużych szczytów.
- W weekendy rowery są wykorzystywane bardziej równomiernie przez cały dzień, co może sugerować ich używanie rekreacyjne lub do celów turystycznych.



## Rozkład długości wypożyczeń według dnia tygodnia

```
query <- "
select
  duration,
  strftime('%u', trip.start_date) as day_of_week_number,
  case cast (strftime('%u', trip.start_date) as integer)
    when 1 then 'Poniedzialek'
    when 2 then 'Wtorek'
    when 3 then 'Sroda'
    when 4 then 'Czwartek'
    when 5 then 'Piatek'
    when 6 then 'Sobota'
    else 'Niedziela' end as day_of_week
from trip
where duration < 2*60*60"

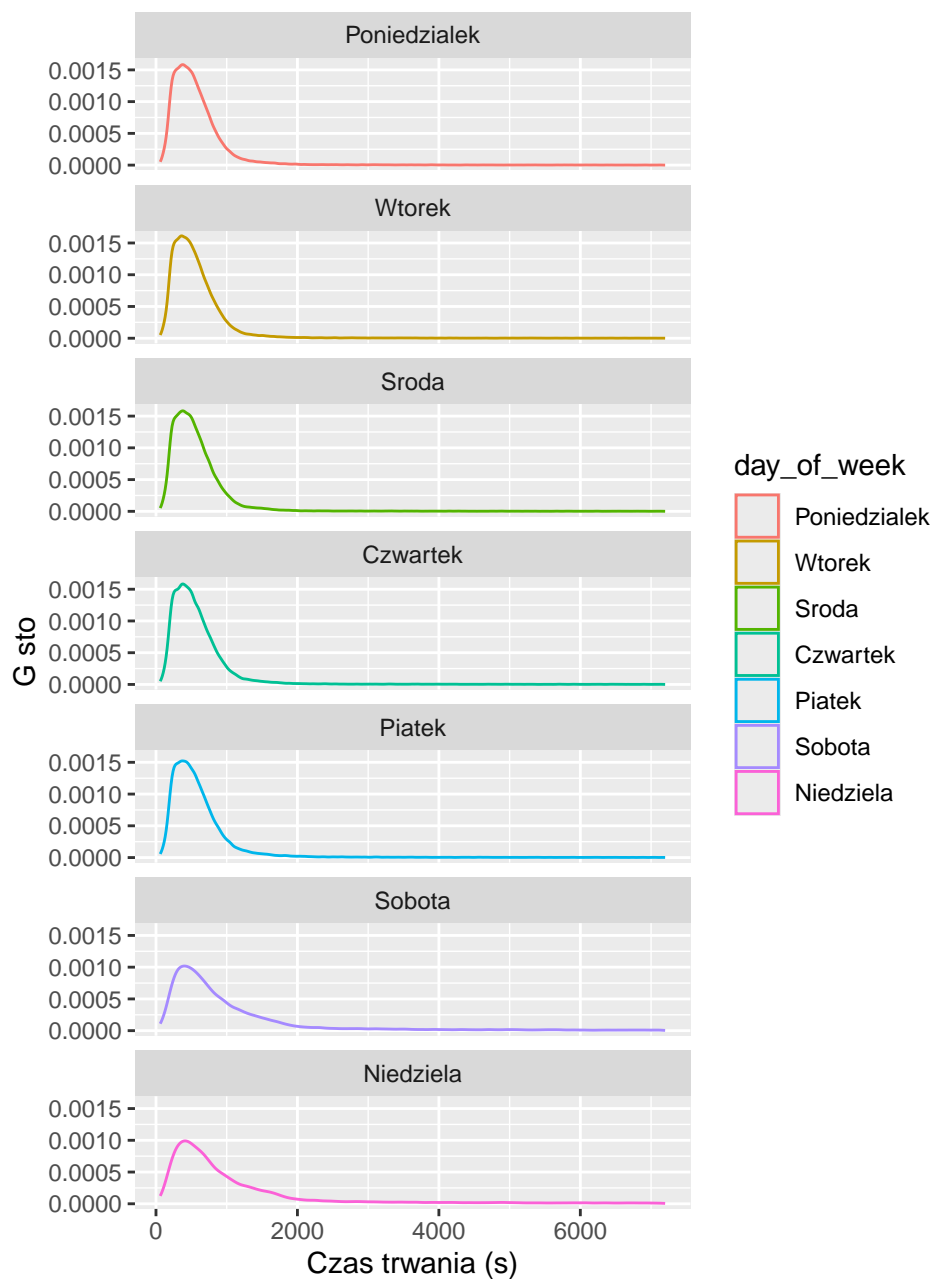
trip_data <- dbGetQuery(con, query)

trip_data$day_of_week_number <- as.numeric(trip_data$day_of_week_number)

trip_data$day_of_week <- reorder(trip_data$day_of_week, trip_data$day_of_week_number)

ggplot(trip_data, aes(x = duration, color = day_of_week)) +
  geom_density() +
  facet_wrap(~ day_of_week, ncol=1) +
  labs(title = "Rozkład długości wypożyczeń według dnia tygodnia", x = "Czas trwania (s)", y = "Gęstość")
```

## Rozkład długości wypożyczeń według dnia tygodnia



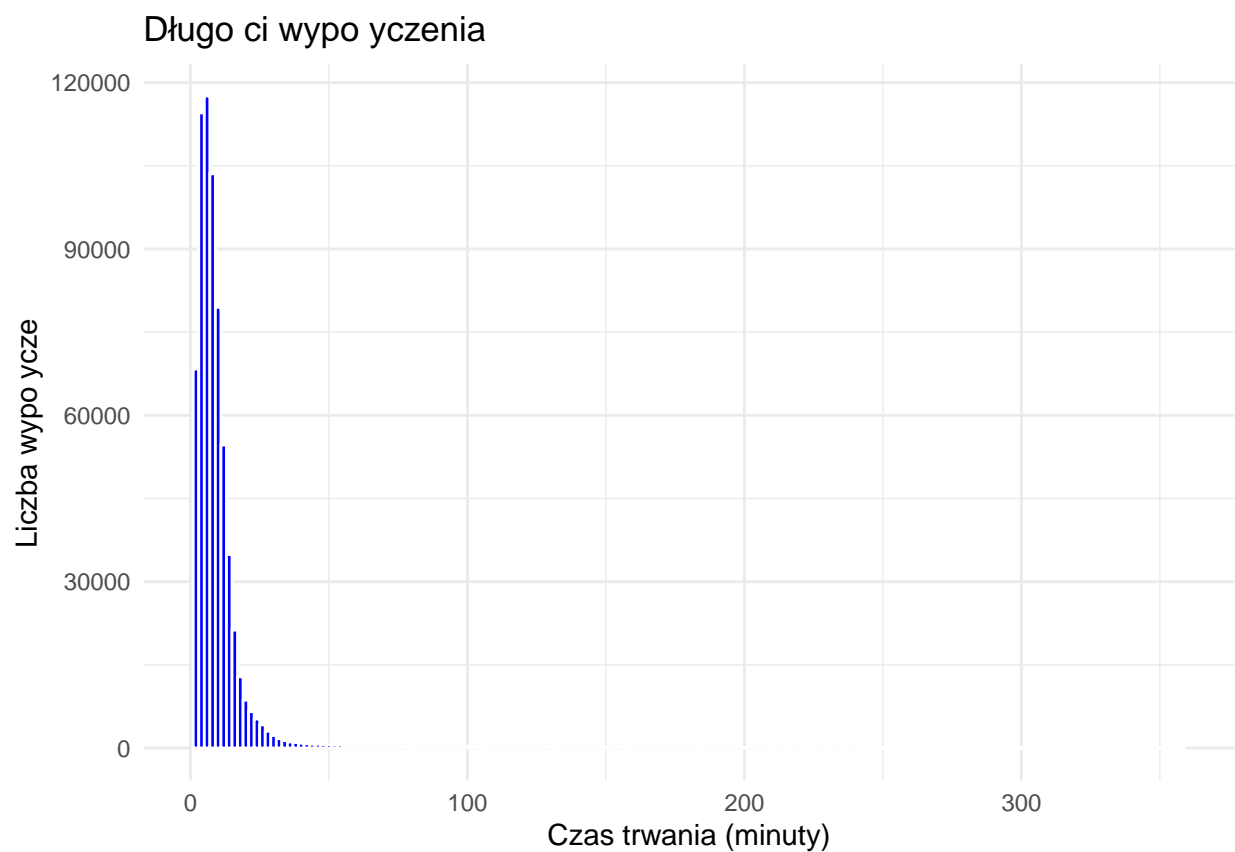
### Wnioski:

- Kształt rozkładu: Zarówno dla dni powszednich, jak i weekendów, większość wypożyczeń trwa krótko. Krzywa gęstości maleje szybko po osiągnięciu szczytu.
- Większość wypożyczeń jest bardzo krótka, niezależnie od dnia tygodnia.
- W dni robocze rozkład jest bardziej stromy, co oznacza, że wypożyczenia są krótsze i bardziej skoncentrowane w czasie.
- W weekendy (Sobota, Niedziela) rozkład jest nieco bardziej spłaszczony, co sugeruje większą różnorodność długości wypożyczenia, prawdopodobnie z powodu większej liczby przejazdów rekreacyjnych.

## Histogram długości wypożyczenia

Histogram długości wypożyczenia dla krótszych niż 6 godzin

```
query <- "  
select  
  duration / 60 as 'duration_min'  
from trip  
where duration < 6*60*60"  
  
trips_duration <- dbGetQuery(con, query)  
  
ggplot(trips_duration, aes(x = duration_min)) +  
  geom_histogram(binwidth = 2, fill = "blue", color = "white") +  
  labs(  
    title = "Długości wypożyczenia",  
    x = "Czas trwania (minuty)", y = "Liczba wypożyczeń") +  
  theme_minimal()
```



Histogram długości wypożyczenia dla krótszych niż 16 godzin (skala logarytmiczna)

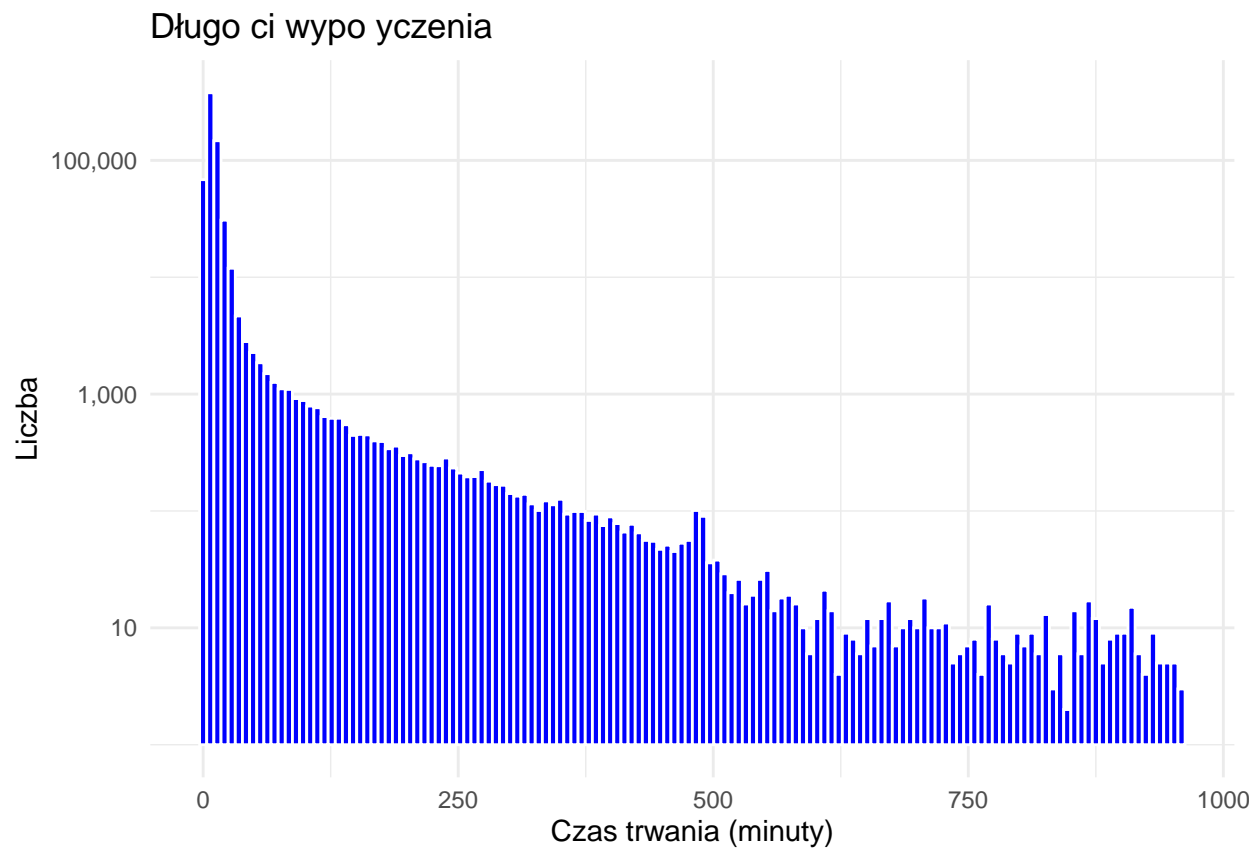
```

query <- "
select
  duration / 60 as 'duration_min'
from trip
where duration < 16*60*60
"

trips_duration <- dbGetQuery(con, query)

ggplot(trips_duration, aes(x = duration_min)) +
  geom_histogram(binwidth = 7, fill = "blue", color = "white") +
  labs(title = "Długości wypożyczenia", x = "Czas trwania (minuty)", y = "Liczba") +
  scale_y_log10(labels = label_number(scale = 1, big.mark = ",")) +
  theme_minimal()

```



#### Wnioski:

- Spadek liczby wypożyczeń jest stopniowy. Wypożyczenia trwające dłużej niż 60 minut nadal występują, choć rzadko.
- Długi ogon rozkładu jest wyraźniej widoczny.

## Zależność temperatury od ilości wypożyczeń

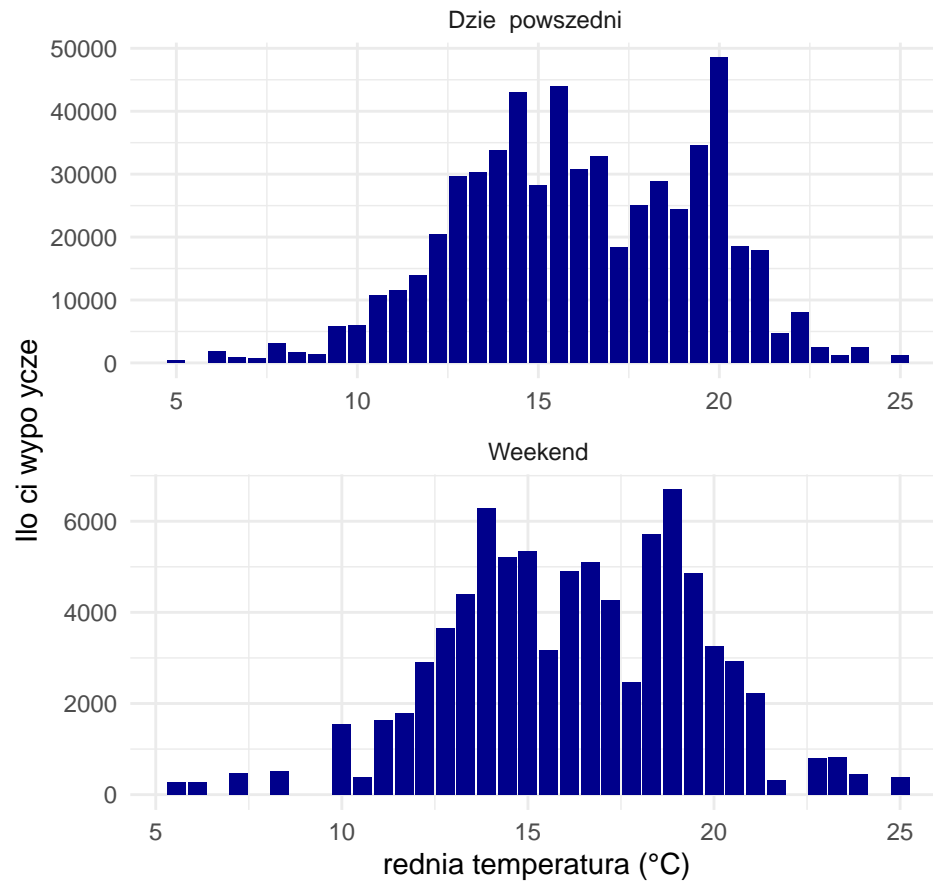
Liczba wypożyczeń dla danych temperatur

```
query <- "
select
  (
    SELECT w.mean_temperature_f
      from weather w
     where strftime('%Y-%m-%d', w.date) = strftime('%Y-%m-%d', t.start_date)
    ) as mean_temperature,
  t.id,
  case
    when cast(strftime('%u', t.start_date) as integer) in (6, 7) then 'Weekend'
    else 'Dzień powszedni'
  end as day_type
from trip t
"

temperature_trips <- dbGetQuery(con, query)

ggplot(temperature_trips, aes(x=mean_temperature)) +
  geom_bar(stat = "count", fill="darkblue") +
  labs(title = "Liczba wypożyczeń dla danych temperatur",
       y = "Ilości wypożyczeń",
       x = "Średnia temperatura (°C)",
       color = "Temperatura") +
  facet_wrap(~ day_type, ncol=1, scales="free") +
  theme_minimal()
```

## Liczba wypożyczeń dla danych temperatur



### Wnioski:

- Najwięcej wypożyczeń występuje w temperaturach (10-20°C)
- Ilość wypożyczeń w weekendy jest mniejsza niż w tygodniu co potwierdza poprzednie wnioski

### Wykres punktowy (scatter plot) – zależność temperatury od ilości wypożyczeń

```
query <- "
SELECT
  strftime('%Y-%m-%d', w.date) as trip_date,
  (SELECT
    count(*)
  from trip t
  join station s on s.id = t.start_station_id
  where strftime('%Y-%m-%d', w.date) = strftime('%Y-%m-%d', t.start_date)
  and s.city = 'San Francisco' and t.subscription_type = 'Subscriber'
  ) as trip_count,
  case
    when cast(strftime('%u', w.date) as integer) in (6, 7) then 'Weekend'
    else 'Dzień powszedni'
  end as day_type,
```

```

    w.mean_temperature_f as mean_temperature,
    w.mean_humidity,
    w.mean_wind_speed_mph,
    CASE
        WHEN LOWER(w.events) LIKE '%rain%' THEN 'rain'
        ELSE 'no rain'
    END AS rain
FROM weather w
WHERE w.mean_temperature_f IS NOT NULL;
"

```

```
trips_grouped <- dbGetQuery(con, query)
```

```
summary(trips_grouped['trip_count'])
```

```

##      trip_count
## Min.       : 41.0
## 1st Qu.: 209.0
## Median : 835.0
## Mean    : 696.5
## 3rd Qu.:1049.0
## Max.    :1245.0

```

```
var(trips_grouped['trip_count'])
```

```

##           trip_count
## trip_count 155753.8

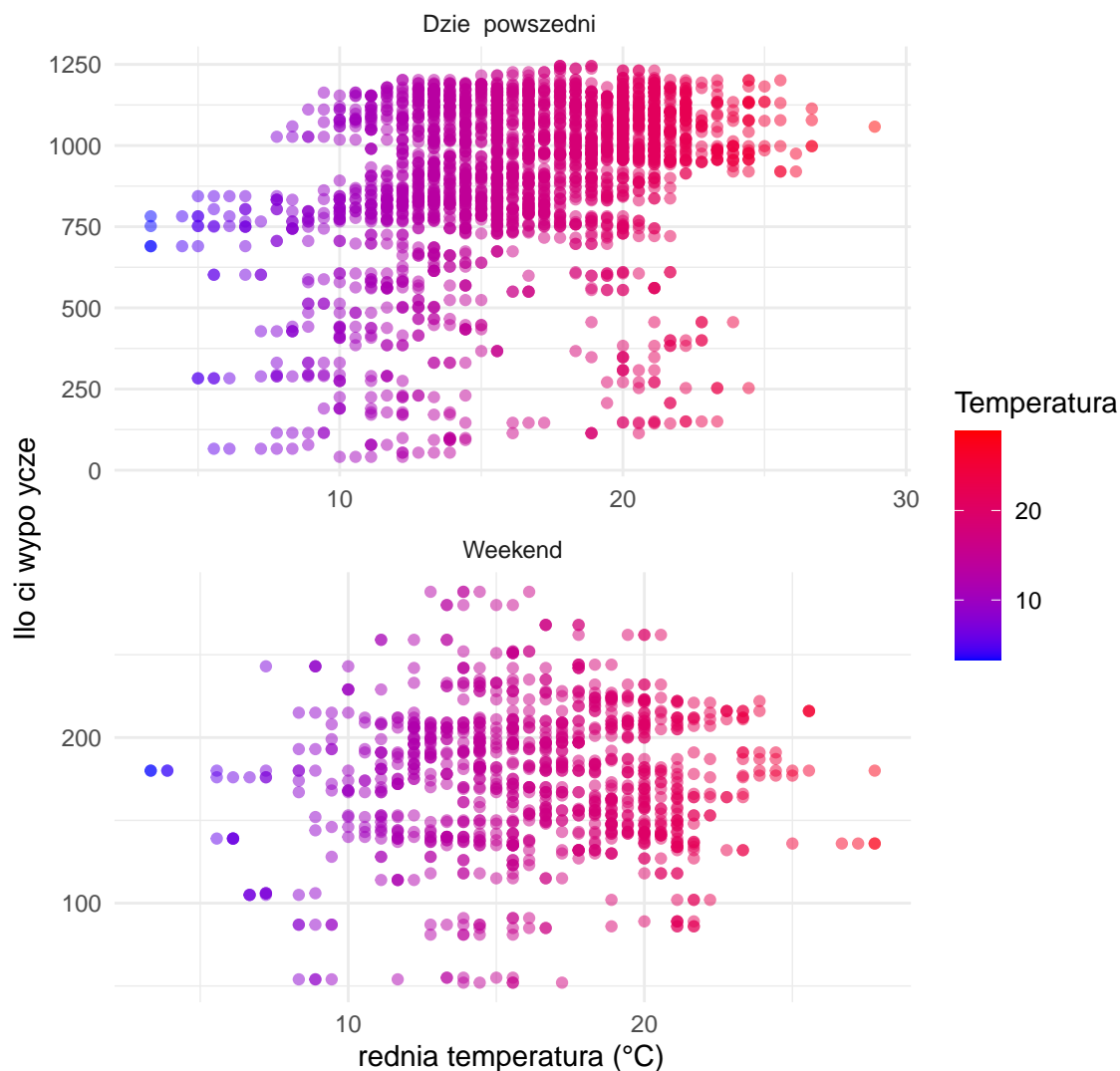
```

```

ggplot(trips_grouped, aes(y = trip_count, x = mean_temperature, color = mean_temperature)) +
  geom_point(alpha = 0.5) +
  scale_color_gradient(low = "blue", high = "red") +
  labs(title = "Zależność temperatury od ilości wypożyczeń",
       y = "Ilości wypożyczeń",
       x = "Średnia temperatura (°C)",
       color = "Temperatura") +
  facet_wrap(~ day_type, ncol=1, scales="free") +
  theme_minimal()

```

## Zależność temperatury od ilości wypożyczeń



### Wnioski:

- W niższych temperaturach (5-15 °C) wypożyczenia są mniej liczne, a wraz ze wzrostem temperatury do około 20-25 °C następuje znaczny wzrost wypożyczeń.
- W bardzo wysokich temperaturach (powyżej 25 °C) widać tendencję do spadku wypożyczeń, co może sugerować, że zbyt wysoka temperatura wpływa negatywnie na korzystanie z rowerów.
- W dni powszednie liczba wypożyczeń często przekracza 1000, podczas gdy w weekendy liczba wypożyczeń rzadko przekracza 250.

## Model statystyczny

### Regresja liniowa



```

query <- "
SELECT
  strftime('%Y-%m-%d', w.date) as trip_date,
  (SELECT
    count(*)
  from trip t
  join station s on s.id = t.start_station_id
  where strftime('%Y-%m-%d', w.date) = strftime('%Y-%m-%d', t.start_date)
  ) as trip_count,
  w.mean_temperature_f as mean_temperature,
  w.mean_humidity,
  w.mean_wind_speed_mph as mean_wind_speed,
  CASE
    WHEN LOWER(w.events) LIKE '%rain%' THEN 'rain'
    ELSE 'no rain'
  END AS rain
FROM weather w
WHERE strftime('%u', w.date) NOT IN ('6', '7')
and w.mean_temperature_f IS NOT NULL;
"

trips <- dbGetQuery(con, query)
trips <- na.omit(trips)

trips$rain <- as.factor(trips$rain)

lm <- lm(trip_count ~ rain + mean_temperature + mean_humidity +
         mean_wind_speed, data = trips)

summary(lm)

```

```

##
## Call:
## lm(formula = trip_count ~ rain + mean_temperature + mean_humidity +
##     mean_wind_speed, data = trips)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -954.64  -80.92   31.61  139.96  596.99
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    622.1888    35.6213   17.467 < 2e-16 ***
## rainrain       -254.5563    16.9499  -15.018 < 2e-16 ***
## mean_temperature  26.0040     1.2115   21.465 < 2e-16 ***
## mean_humidity     1.3362     0.4237    3.154 0.00163 **
## mean_wind_speed   1.3216     0.8967    1.474 0.14065
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 226.3 on 2453 degrees of freedom
## Multiple R-squared:  0.2694, Adjusted R-squared:  0.2682

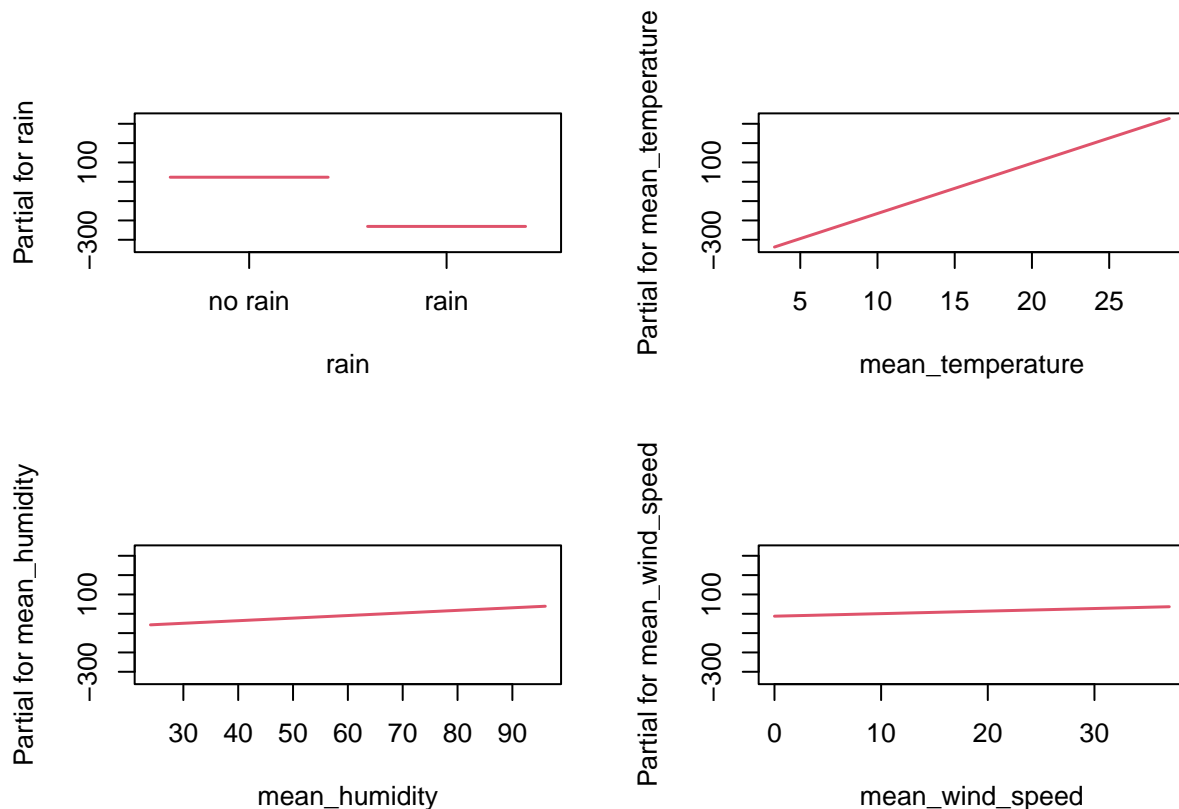
```

```
## F-statistic: 226.1 on 4 and 2453 DF, p-value: < 2.2e-16
```

### Interpretacja wyników `summary(lm)`

- Deszcz: współczynnik **-254.56** oznacza, że deszcz zmniejsza liczbę wypożyczeń średnio o 254 rowery dziennie, jest istotny statystycznie ( $p < 0.001$ )
- Średnia temperatura: współczynnik **26.00** wskazuje, że większa temperatura otoczenia zwiększa liczbę wypożyczeń o około 26, również istotny wpływ ( $p < 0.001$ )
- Średnia wilgotność: ma niewielki, ale istotny wpływ ( $p = 0.00163$ )
- Średnia prędkość wiatru: nie jest istotny ( $p = 0.14065$ ), co sugeruje, że prędkość wiatru nie ma dużego wpływu na liczbę wypożyczeń

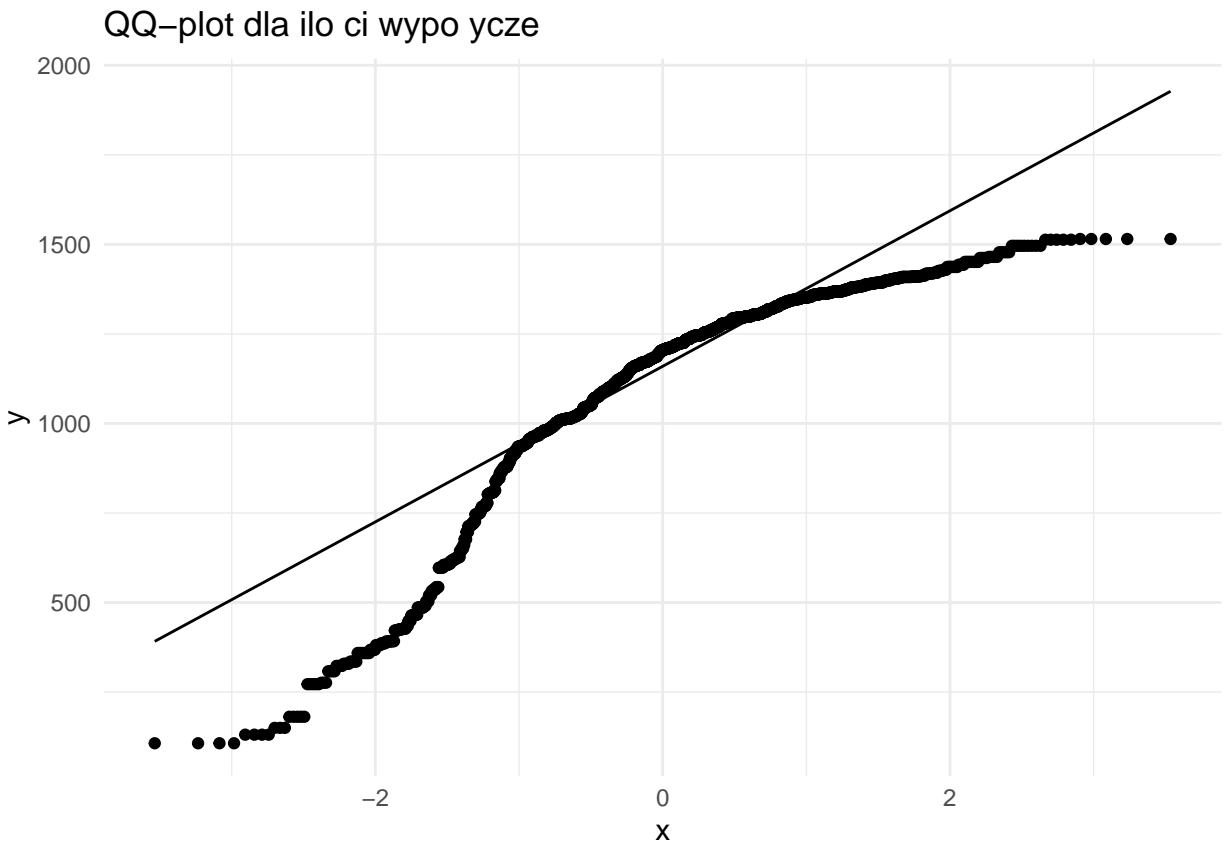
```
par(mfrow=c(2,2))  
termplot(lm)
```



- Deszcz: widać istotny spadek liczby wypożyczeń w dni deszczowe.
- Średnia temperatura: istnieje wyraźny trend wzrostowy, co potwierdza dodatni wpływ temperatury na wypożyczenia.
- Średnia wilgotność: niewielki, ale zauważalny wzrost liczby wypożyczeń przy wzrastającej wilgotności.
- Średnia prędkość wiatru: prawie płaski trend, co potwierdza jego niski wpływ.

```
ggplot(trips, aes(sample = trip_count)) +  
  stat_qq() +  
  stat_qq_line() +
```

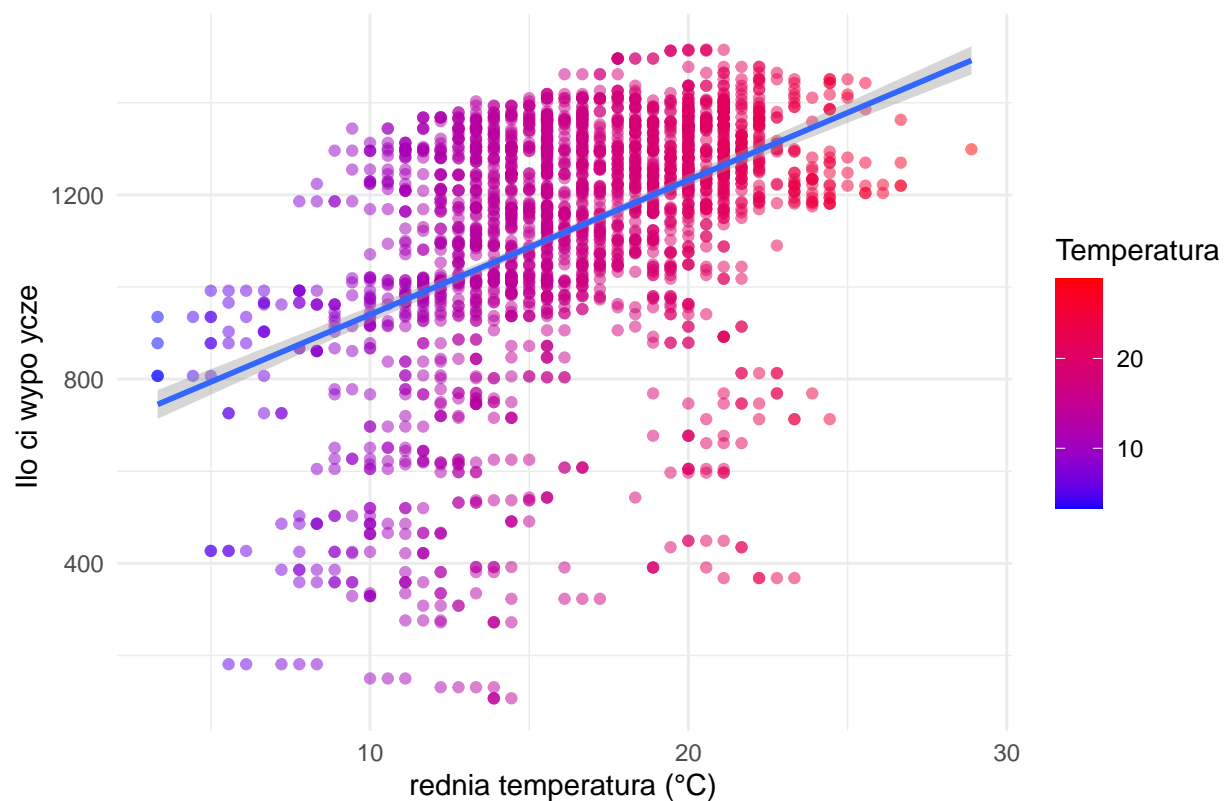
```
labs(title = "QQ-plot dla ilości wypożyczeń") +  
theme_minimal()
```



QQ-plot sugeruje, że reszty modelu nie są idealnie normalnie rozłożone – widoczna jest asymetria (dolny lewy róg odchyła się od linii), co sugeruje potencjalne niedopasowanie modelu do danych. Oznacza to, że model może mieć problem z dopasowaniem ekstremalnych wartości (np. bardzo niskich i wysokich liczby wypożyczeń).

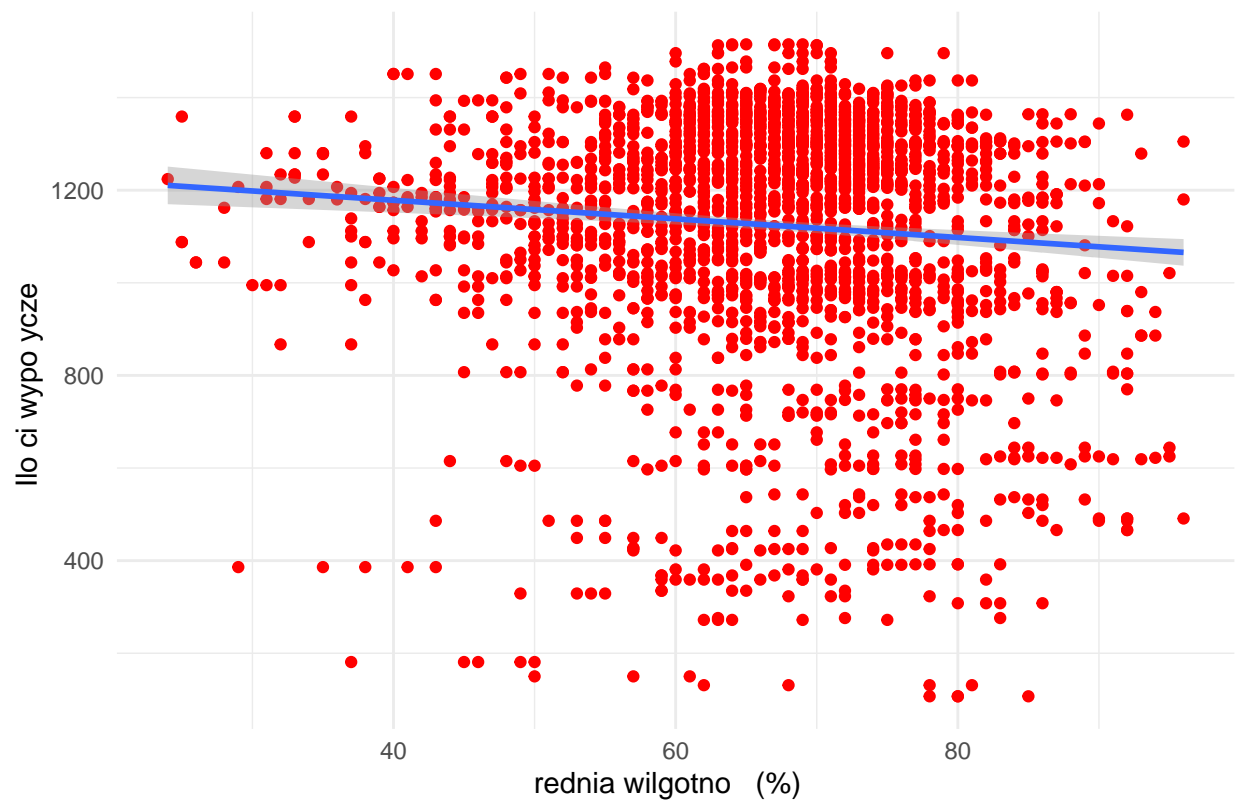
```
ggplot(data = trips, aes(mean_temperature, trip_count, color = mean_temperature)) +  
  geom_point(alpha=0.5) +  
  scale_color_gradient(low = "blue", high = "red") +  
  geom_smooth(method = "lm") +  
  theme_minimal() +  
  labs(title = "Zależność temperatury od ilości wypożyczeń w dni powszednie",  
        y = "Ilości wypożyczeń",  
        x = "Średnia temperatura (°C)",  
        color = "Temperatura")
```

Zależność temperatury od ilości wypożyczeń w dni powszednie



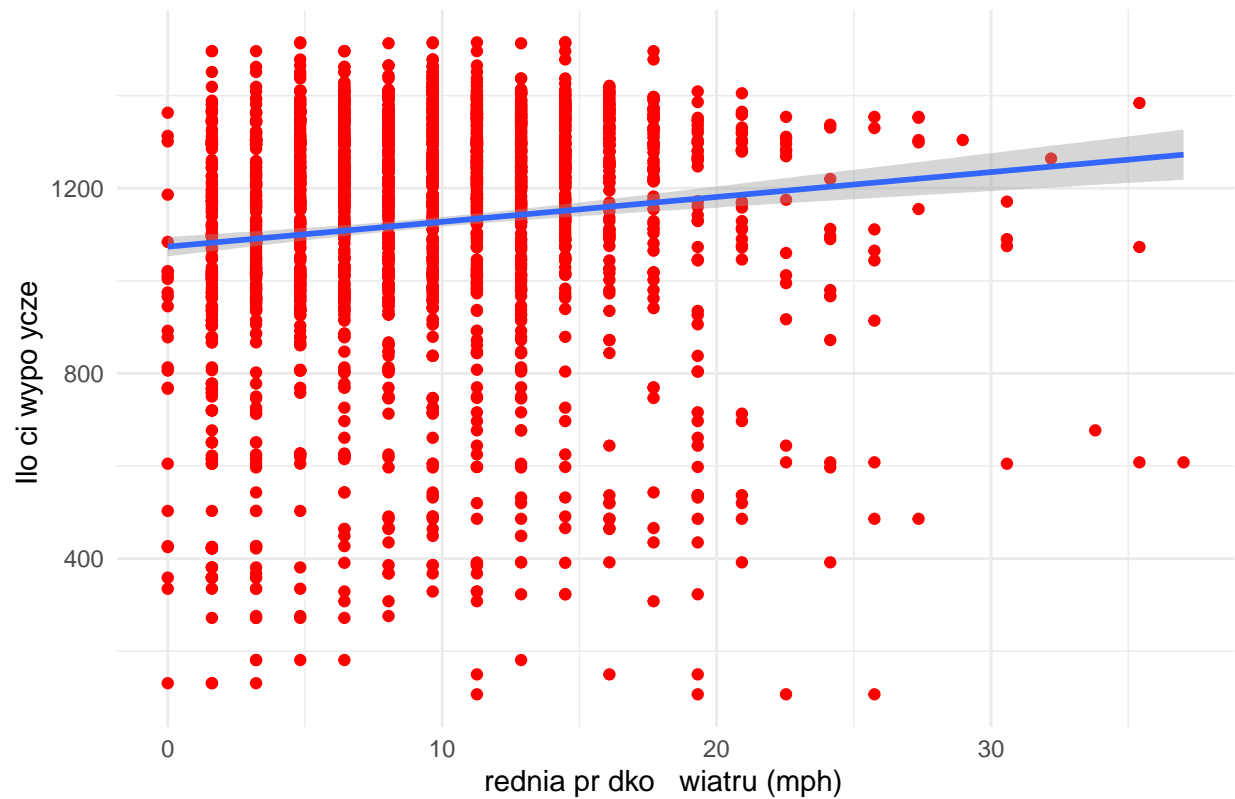
```
ggplot(data = trips, aes(mean_humidity, trip_count)) +
  geom_point(color = "red") +
  geom_smooth(method = "lm") +
  theme_minimal() +
  labs(title = "Zależność wilgotności od ilości wypożyczeń w dni powszednie",
        y = "Ilości wypożyczeń",
        x = "Średnia wilgotność (%)")
```

Zależność wilgotności od ilości wypożyczeń w dni powszednie



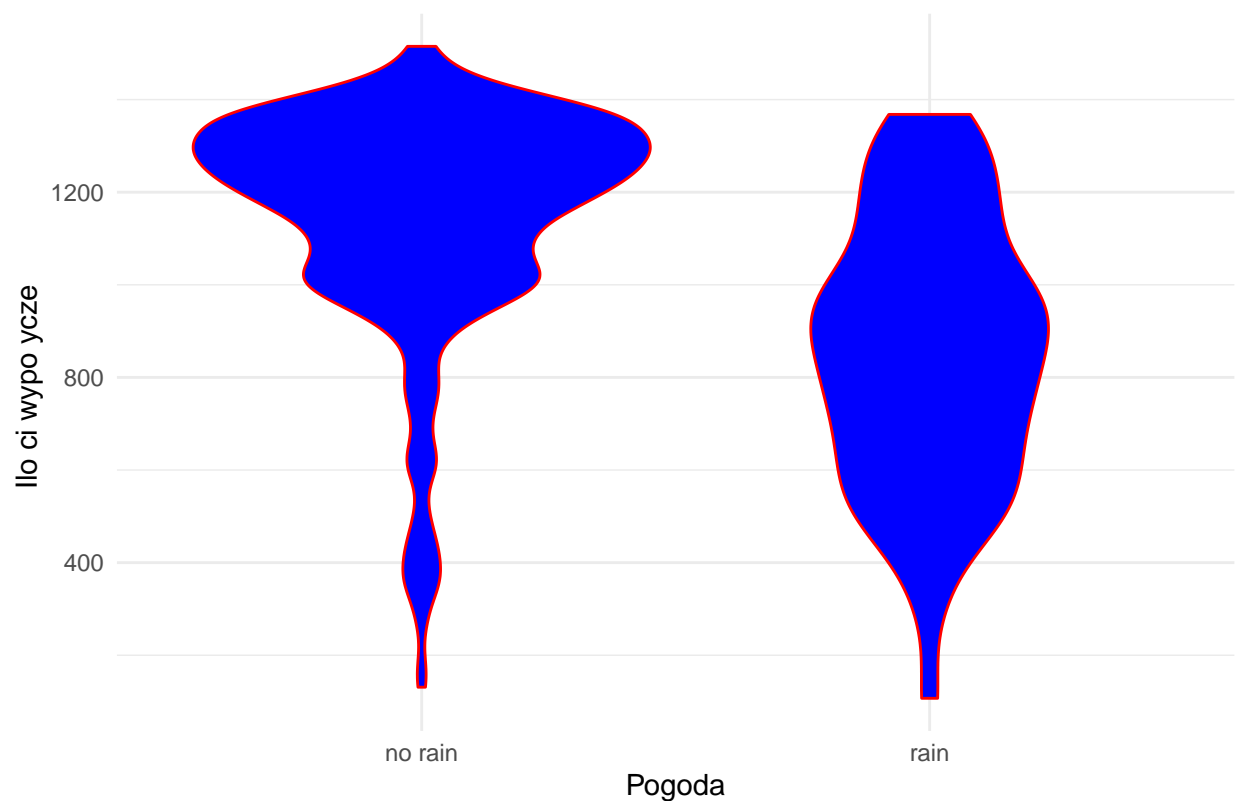
```
ggplot(data = trips, aes(mean_wind_speed, trip_count)) +
  geom_point(color = "red") +
  geom_smooth(method = "lm") +
  theme_minimal() +
  labs(title = "Zależność prędkości wiatru od ilości wypożyczeń w dni powszednie",
        y = "Ilości wypożyczeń",
        x = "Średnia prędkość wiatru (mph)")
```

Zależność prędkości wiatru od ilości wypożyczeń w dni powszednie



```
ggplot(data = trips, aes(rain, trip_count)) +
  geom_violin(color = "red", fill = "blue") +
  theme_minimal() +
  labs(title = "Zależność deszczu od ilości wypożyczeń w dni powszednie",
       y = "Ilości wypożyczeń",
       x = "Pogoda")
```

## Zależność deszczu od ilości wypożyczeń w dni powszednie



```
t.test(trips_grouped$trip_count[trips_grouped$day_type == "Dzień powszedni"],
       trips_grouped$trip_count[trips_grouped$day_type == "Weekend"])
```

```
##
## Welch Two Sample t-test
##
## data: trips_grouped$trip_count[trips_grouped$day_type == "Dzień powszedni"] and trips_grouped$trip_
## t = 142.08, df = 2817.8, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  723.7741 744.0301
## sample estimates:
## mean of x mean of y
##  909.7436 175.8415
```

1. Hipoteza zerowa:

- Średnia liczba wypożyczeń rowerów w dni powszednie i w weekendy jest taka sama.

2. Hipoteza alternatywna:

- Średnia liczba wypożyczeń rowerów w dni powszednie różni się od tej w weekendy.

- Wartość t-testu jest bardzo wysoka, co pokazuje istotną różnicę między średnimi.
- Bardzo niska wartość p, oznacza, że istnieją bardzo silne dowody przeciwko hipotezie zerowej.

## Regresja liniowa z regulacją L1

```
X <- model.matrix(trip_count ~ rain + mean_temperature + mean_humidity + mean_wind_speed, data = trips)
y <- trips$trip_count

lasso_model <- glmnet(X, y, alpha = 1)

cv_lasso <- cv.glmnet(X, y, alpha = 1, nfolds = 10)

best_lambda <- cv_lasso$lambda.min

lasso_final_model <- glmnet(X, y, alpha = 1, lambda = best_lambda)

coef(lasso_final_model)
```

```
## 5 x 1 sparse Matrix of class "dgCMatrix"
##              s0
## (Intercept)  627.504502
## rainrain    -252.507461
## mean_temperature  25.935626
## mean_humidity   1.280861
## mean_wind_speed  1.248064
```

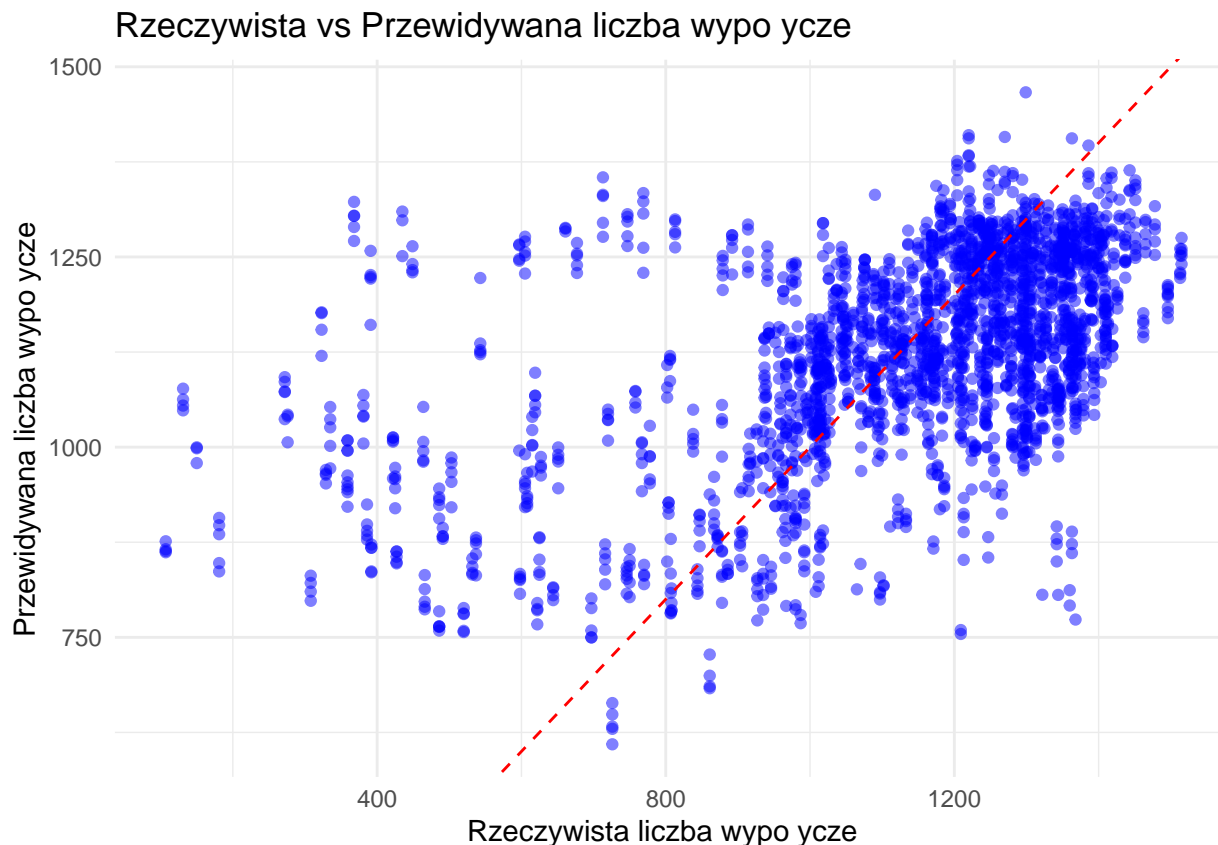
```
predictions <- predict(lasso_final_model, newx = X)
```

```
predictions <- predict(lasso_final_model, newx = X)
```

## Wykres rzeczywistych vs przewidywanych wartości

```
ggplot(data = trips, aes(x = y, y = predictions)) +
  geom_point(color = "blue", alpha = 0.5) +
  geom_abline(slope = 1, intercept = 0, color = "red", linetype = "dashed") +
  labs(title = "Rzeczywista vs Przewidywana liczba wypożyczeń",
       x = "Rzeczywista liczba wypożyczeń",
       y = "Przewidywana liczba wypożyczeń") +
  theme_minimal()
```





- Są widoczne rozproszenia danych wokół linii idealnego dopasowania, szczególnie dla mniejszych wartości (np. poniżej 800). Pokazuje to, że model może mieć trudności z dokładnym przewidywaniem niższych liczb wypożyczeń.
- Większych wartości (powyżej 1000) są bardziej przewidywane przez model są bardziej zgodne z rzeczywistością, choć występuje nadal pewne odchylenie.
- Istnieją wartości, gdzie model znacząco różni się od rzeczywistości, co sugeruje, że mogą występować jeszcze inne czynniki wpływające na liczbę wypożyczeń (np. sezonowość, specjalne wydarzenia)

## Podsumowanie

Analiza statystyczna danych dotyczących wypożyczeń rowerów wykazała istotne zależności pomiędzy liczbą wypożyczeń a czynnikami pogodowymi.

- Temperatura: Wzrost temperatury do 20-20 C sprzyja wypożyczeniom, natomiast w niskich ( $<15^{\circ}\text{C}$ ) i bardzo wysokich temperaturach ( $>25^{\circ}\text{C}$ ) liczba wypożyczeń spada.
- Dzień tygodnia: W dni powszednie liczba wypożyczeń jest znacznie wyższa niż w weekendy, co potwierdził test t-Studenta ( $p < 0.001$ ). Średnia liczba wypożyczeń wynosi ok. 910 w dni robocze i 176 w weekendy.
- Długość wypożyczeń: Większość wypożyczeń jest krótka, szczególnie w dni robocze, gdzie są bardziej skoncentrowane czasowo. W weekendy długości wypożyczeń są bardziej zróżnicowane, co sugeruje użytkowanie rekreacyjne.
- Godziny szczytu: W dni robocze występują dwa wyraźne szczyty o godzinie 8:00 i 17:00, zgodne z typowymi godzinami szczytu. W weekendy rozkład wypożyczeń jest bardziej równomierny.

- Najwyższa aktywność: Najwięcej wypożyczeń odnotowano w dni powszednie , a najmniej w weekendy, co sugeruje, że system jest wykorzystywany głównie do codziennych dojazdów.