Statystyka Projekt

Jakub Jodrzyczyk

2024-05-30

Instalowanie i importowanie potrzebnych paczek

```
#install.packages('readr')
#install.packages('ggplot2')
#install.packages('moments')

library(readr)
library(ggplot2)
library(moments)
```

Opis paczek

- readr - Pakiet readr jest częścią tidyverse, która jest kolekcją pakietów R zaprojektowanych do pracy z danymi. readr dostarcza szybki i przyjazny sposób na wczytywanie danych z płaskie tekstowych i csv, jest znacznie szybszy i bardziej elastyczny niż base R.
- ggplot2 - To jeden z najpopularniejszych pakietów do wizualizacji danych w R. ggplot2 pozwala na tworzenie skomplikowanych wykresów w dość prosty sposób, używając gramatyki grafiki. Jest to część tidyverse, co oznacza, że dobrze integruje się z innymi pakietami z tego zbioru.
- moments - Pakiet moments jest używany do obliczania momentów statystycznych: skośności (skewness) i kurtozy (kurtosis), które są miarami symetrii i wypukłości rozkładu danych odpowiednio. Te metryki są używane do analizy charakteru rozkładu danych.

Opis użytych danych

Dane jakie użyje w projekcie pochodzą z serwisu Kaggle, zawierają minimalną temperaturę zarejestrowaną danego dnia i dotyczą okresu od 1981 do 1990 roku

Link do dataset'u: https://www.kaggle.com/datasets/paulbrabban/daily-minimum-temperatures-in-melbourne

Na potrzeby projektu będziemy brać uwagę tylko na temperaturę z pierwszego roku, czyli na pierwsze 365 rekordów.

Importowanie danych

W poniższym kodzie importujemy dane przy pomocy paczki readr. Zmienamy typ pierwszej kolumny z formatu character, interpretowanego jako tekst, na format date, który jest interpretowany jako data. By funkcja zadziałała poprawnie podajemy oryginalny format daty czyli %Y-%m-%d. Typ drugie kolumny został zmieniony na numeric. Jeśli w wartościach znajduje się taka, która nie będzie się dało przekonstruktować to zostanie ona zastąpiona wartością NA.

```
data <- read_csv("D:/Dry Fabularne/Studia 2/Semestr 2/daily-minimum-temperatures-in-me.csv",
    col_types = cols(Date = col_date(format = "%Y-%m-%d"),
        `Daily minimum temperatures in Melbourne, Australia, 1981-1990` = col_number()),
    na = "NaN")

## Warning: One or more parsing issues, call `problems()` on your data frame for details,
## e.g.:
##   dat <- vroom(...)
##   problems(dat)
```

Modyfikowanie danych

Następnie przyciemamy nasz dataset by zawierał pierwsze 365 rekordów i zmieniamy nazwy kolumn na bardziej czytelne

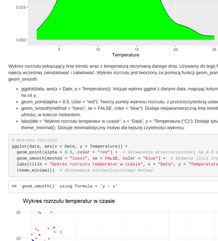
```
data=data[1:365, ]
colnames(data)<- c("Date","Temperature")
```

Wizualizacja danych

Rysujemy histogram przedstawiający rozkład temperatur w złożicie danych. Używamy do tego funkcji ggplot z pakietu ggplot2, który należy wcześniej zainstalować i załadować. Histogram jest tworzony za pomocą funkcji geom_histogram.

- ggplot(data, aes(x = Temperature)): Inicjuje wykres ggplot z danymi data i mapuje kolumnę Temperature na oś x.
- geom_histogram(binwidth = 1, fill = "blue", color = "black"): Tworzy histogram z szerokością kosza równą 1, wypełnionym kolorem niebieskim i czarnymi krawędziami.
- ggtitle("Histogram temperatur"): Dodaje tytuł do wykresu.

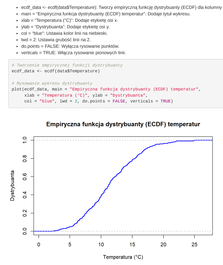
```
# Histogram
ggplot(data, aes(x = Temperature)) +
  geom_histogram(binwidth = 1, fill = "blue", color = "black") +
  ggtitle("Histogram temperatur")
```



Rysujemy wykres gęstości, który przedstawia estymowaną funkcję gęstości rozkładu temperatur w złożicie danych. Używamy do tego funkcji ggplot z pakietu ggplot2, który należy wcześniej zainstalować i załadować. Wykres gęstości jest tworzony za pomocą funkcji geom_density.

- ggplot(data, aes(x = Temperature)): Inicjuje wykres ggplot z danymi data i mapuje kolumnę Temperature na oś x.
- geom_density(fill = "green"): Tworzy wykres gęstości, wypełniony kolorem zielonym.
- ggtitle("Wykres gęstości temperatur"): Dodaje tytuł do wykresu.

```
# Wykres gęstości
ggplot(data, aes(x = Temperature)) +
  geom_density(fill = "green") +
  ggtitle("Wykres gęstości temperatur")
```

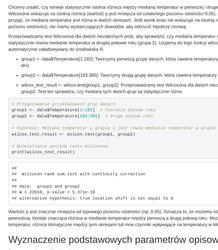


Wykres rozrzutu pokazujący linie trendu wraz z temperatura otrzymaną danego dnia. Używamy do tego funkcji ggplot z pakietu ggplot2, który należy wcześniej zainstalować i załadować. Wykres rozrzutu jest tworzony za pomocą funkcji geom_point, a linia trendu za pomocą geom_smooth.

- ggplot(data, aes(x = Date, y = Temperature)): Inicjuje wykres ggplot z danymi data, mapując kolumnę Date na oś x i kolumnę Temperature na oś y.
- geom_point(alpha = 0.5, color = "red"): Tworzy punkty wykresu rozrzutu, z przezroczystością ustawioną na 0.5 i czerwonym kolorem.
- geom_smooth(method = "loess", se = FALSE, color = "blue"): Dodaje niepararametryczną linię trendu (metoda LOESS) bez przedziałów ufności, w kolorze niebieskim.
- labs(title = "Wykres rozrzutu temperatur w czasie", x = "Data", y = "Temperatura (°C)"): Dodaje tytuł wykresu oraz etykiety osi x i y.
- theme_minimal(): Stosuje minimalistyczny motyw dla lepszej czytelności wykresu.

```
# Wykres rozrzutu
ggplot(data, aes(x = Date, y = Temperature)) +
  geom_point(alpha = 0.5, color = "red") +  # Ustawienie przezroczystości na 0.5 dla lepszego efektu wizualnego
  geom_smooth(method = "loess", se = FALSE, color = "blue") +  # Dodanie linii trendu
  labs(title = "Wykres rozrzutu temperatur w czasie", x = "Data", y = "Temperatura (°C)") +
  theme_minimal()  # Stosowanie minimalistycznego motywu

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

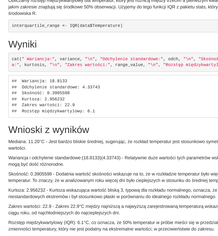


Tworzymy i rysujemy empiryczną funkcję dystrybuanty (ECDF). Używamy do tego funkcji ecdf i plot z pakietu stats, który jest automatycznie zaladowywany do środowiska R.

- ecdf_data <- ecdf(data$Temperature): Tworzy empiryczną funkcję dystrybuanty (ECDF) dla kolumny Temperature.
- main = "Empiryczna funkcja dystrybuanty (ECDF) temperatur": Dodaje tytuł wykresu.
- xlab = "Temperatura (°C)": Dodaje etykietę osi x.
- ylab = "Dystrybuanta": Dodaje etykietę osi y.
- col = "blue": Ustawia kolor linii na niebieski.
- lwd = 2: Ustawia grubość linii na 2.
- do.points = FALSE: Wyłącza rysowanie punktów.
- verticals = TRUE: Włącza rysowanie pionowych linii.

```
# Tworzenie empirycznej funkcji dystrybuanty
ecdf_data <- ecdf(data$Temperature)

# Wykreślenie wykresu dystrybuanty
plot(ecdf_data, main = "Empiryczna funkcja dystrybuanty (ECDF) temperatur",
     xlab = "Temperatura (°C)", ylab = "Dystrybuanta",
     col = "blue", lwd = 2, do.points = FALSE, verticals = TRUE)
```



Hipoteza test T-Studenta

Na wykresie gęstości możemy zauważyć, że najczęściej pojawiają się wyniki w okolicy 10 stopni Celsjusza. Na podstawie tego możemy stworzyć hipotezę.

Hipotezy

- Hipoteza zerowa (H0): Średnia temperatura jest równa 10°C.
- Hipoteza alternatywna (H1): Średnia temperatura jest różna od 10°C.

Test t-Studenta dla próby zależy (one-sample t-test) służy do sprawdzenia, czy średnia wartości w danym zbiorze danych (próbie) jest statystycznie różna od określonej wartości teoretycznej (hipotetycznej średniej). W naszym przypadku chcemy sprawdzić, czy średnia temperatura w zbiorze danych data jest różna od równo 10 stopniom Celsjusza.

Jeśli wynik testu t-Studenta wskazuje, że różnica jest istotna (wartość p jest mniejsza od ustalonego poziomu istotności, np. 0.05), możemy odrzucić hipotezę zerową i przyjąć, że średnia temperatura różni się od 10°C. Jeśli wynik testu nie wskazuje na istotną różnicę (wartość p jest większa od poziomu istotności), nie mamy wystarczających dowodów, aby odrzucić hipotezę zerową.

Użyjemy do tego funkcji t.test z paczki stats, która jest zautamtycznie zaladowana do środowiska R:

- x_test_result <- t.test(data$Temperature, mu = 10): Przeprowadza test t-Studenta dla jednej próby. Argument mu = 10 określa wartość hipotetycznej średniej temperatury, którą testujemy. Domyślny przedział ufności to 0.95.
- print(t_test_result): Wyświetla wyniki testu t-Studenta, w tym wartość statystyki testowej, stopnie swobody, wartość p, średnią próby oraz przedział ufności dla średniej.

```
# Hipoteza: Średnia temperatura jest równa 10 stopniom Celsjusza
t_test_result <- t.test(data$Temperature, mu = 10)

# Wyświetlenie wyników t-testu
print(t_test_result)
```

```
##
##  One Sample t-test
##
## data:  data$Temperature
## t = -6.683, df = 364, p-value = 8.862e-11
## alternative hypothesis: true mean is not equal to 10
## 95 percent confidence interval:
##  11.07988 11.96372
## sample estimates:
## mean of x
##  11.52179
```

Wartość p jest mniejsza od 0.05 więc możemy uznać że są podstawy do odrzucenia H0. Test jednoczesnie pokazał nam że na 95% średnia temperatura w Melbourne wynosi od 11.07 do 11.96 stopni Celsjusza.

Hipoteza test Wilcox

Na wykresie rozrzutu możemy zauważyć, że mniej więcej w środku naszej próbki danych wygada zima. Na podstawie tego stwórzmy daną hipotezę.

Hipotezy

- Hipoteza zerowa (H0): Mediana temperatur w grupie 1-pierwsza połowa roku) jest równa medianie temperatur w grupie 2-(druga połowa roku).
- Hipoteza alternatywna (H1): Mediana temperatur w grupie 1 jest różna od mediany temperatur w grupie 2.

Chcemy ustalić, czy istnieje statystycznie istotna różnica między medianą temperatur w pierwszej i drugiej połowie roku. Jeśli wynik testu Wilcoxona wskazuje na istotną różnicę (wartość p jest mniejsza od ustalonego poziomu istotności 0.05), możemy odrzucić hipotezę zerową i przyjąć, że mediana temperatur jest różna w dwóch okresach. Jeśli wynik testu nie wskazuje na istotną różnicę (wartość p jest większa od poziomu istotności), nie mamy wystarczających dowodów, aby odrzucić hipotezę zerową.

Przeprowadzimy test Wilcoxona dla dwóch niezależnych prób, aby sprawdzić, czy mediana temperatur w pierwszej połowie roku (grupa 1) jest statystycznie równa medianie temperatur w drugiej połowie roku (grupa 2). Użyjemy do tego funkcji wilcox.test z pakietu stats, który jest automatycznie zaladowywany do środowiska R.

- group1 <- data$Temperature[1:182]: Tworzymy pierwszą grupę danych, która zawiera temperatury z pierwszej połowy roku (pierwsze 182 dni).
- group2 <- data$Temperature[183:365]: Tworzymy drugą grupę danych, która zawiera temperatury z drugiej połowy roku (ostatnie 183 dni).
- wilcox_test_result <- wilcox.test(group1, group2): Przeprowadzamy test Wilcoxona dla dwóch niezależnych prób na grupach group1 i group2. Test ten sprawdza, czy mediany tych dwóch grup są statystycznie różne.

```
# Przygotowanie przykładowych grup danych
group1 <- data$Temperature[1:182]  # Pierwsza połowa roku
group2 <- data$Temperature[183:365]  # Druga połowa roku

# Hipoteza: Mediana temperatur w grupie 1 jest równa medianie temperatur w grupie 2
wilcox_test_result <- wilcox.test(group1, group2)

# Wyświetlenie wyników testu Wilcoxona
print(wilcox_test_result)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  group1 and group2
## W = 22910, p-value = 5.371e-10
## alternative hypothesis: true location shift is not equal to 0
```

Wartość p jest znacznie mniejsza od tysięcznego poziomu istotności (np. 0.05). Oznacza to, że możemy odrzucić hipotezę zerową o dużą pewnością, istnieje znacząca różnica w medianie temperatur między pierwszą a drugą połową roku. Może to wskazywać na sezonowe zmiany temperatur, różnica istnieniczne między tymi okresami lub inne czynniki wpływające na temperatury w Melbourne w analizowanym okresie.

Wyznaczenie podstawowych parametrów opisowych

By wyznaczyć wariancję, minimalną i maksymalną oraz średnią, medianę i kwartyle użyjemy funkcji summary z pakietu base, który jest automatycznie zaladowywany do środowiska R.

```
summary(data$Temperature)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   2.10    9.30   11.20   11.52   14.40   25.80
```

Obliczamy wariancję złożicie danych, co pozwala ocenić, jak bardzo wartości są rozproszone od średniej. Użyjemy do tego funkcji var z pakietu stats, który jest automatycznie zaladowywany do środowiska R.

```
variance <- var(data$Temperature)
```

Obliczamy odchylenie standardowe temperatur w złożicie danych poprzez wyciągnięcie pierwiastka kwadratowego z wariancji. Obliczca średnią różnice danych od średniej. Użyjemy do tego funkcji sqrt z pakietu base, który jest automatycznie zaladowywany do środowiska R.

```
odch <- sqrt(variance)
```

Obliczamy skośność rozkładu temperatur, która wskazuje na asymetrię wartości wokół średniej, pokazując, czy rozkład jest bardziej rozciągnięty po jednej stronie średniej niż po drugiej. Użyjemy do tego funkcji skewness z pakietu moments.

```
skewness <- skewness(data$Temperature)
```

Obliczamy kurtozę dla rozkładu temperatur, która mierzy stopień koncentracji danych wokół średniej, pokazując, czy rozkład jest bardziej "szczytowy" i ma cięższe ogony w porównaniu do normalnego rozkładu. Użyjemy do tego funkcji kurtosis z pakietu moments.

```
kurtosis <- kurtosis(data$Temperature)
```

Obliczamy zakres wartości temperatur w złożicie danych, zwracając wektor z minimalną i maksymalną temperatura, co pokazuje bezpośrednio rozstęp temperatur obserwowanych w danych.Użyjemy do tego funkcji range z pakietu stats, który jest automatycznie zaladowywany do środowiska R.

```
range <- range(data$Temperature)
range_value <- range[1]-range[2]
```

Obliczamy rozstęp międzykwartylowy dla temperatur, który jest różnicą między trzecim a pierwszym kwartylem danych, pomagając zrozumieć, w jakim zakresie znajduje się środkowe 50% obserwacji. Użyjemy do tego funkcji IQR z pakietu stats, który jest automatycznie zaladowywany do środowiska R.

```
interquartile_range <- IQR(data$Temperature)
```

```
cat("Wariancja:", variance, "\n", "Odchylenie standardowe:", odch, "\n", "Skośność:", skewness, "\n", "Kurtoza:", kurtosis, "\n", "Zakres wartości:", range_value, "\n", "Rozstęp międzykwartylowy:", interquartile_range)
```

```
## Wariancja: 18.8133
## Odchylenie standardowe: 4.33743
## Skośność: 0.3905998
## Kurtoza: 2.956232
## Zakres wartości: 22.9
## Rozstęp międzykwartylowy: 6.1
```

Wnioski z wyników

Mediana: 11.20°C - Jest bardzo bliskie średniej, sugerując, że rozkład temperatur jest stosunkowo symetryczny bez wyraźnie odstających wartości.

Wariancja i odchylenie standardowe (18.8133)(4.33743) - Relatywnie duże wartości tych parametrów wskazują na to, że temperatury w ciągu roku mogą być dość różnorodne.

Skośność: 0.3905599 - Dodatnia wartość skośności wskazuje na to, że w rozkładzie temperatur były więcej wartości po stronie wyższych temperatur. To znaczy, że w analizowanym roku więcej dni było cieplejszych w stosunku do średniej temperatury.

Kurtoza: 2.956232 - Kurtoza wskazująca a wartość bliską 3, typowa dla rozkładu normalnego, oznacza, że rozkład temperatur nie wykazywał nierozkładnerdowych ekstremów i był stosunkowo płaski w porównaniu do idealnego rozkładu normalnego.

Zakres wartości: 22.9 °C - Zakres 22.9 °C między najniższą a najwyższą zarejestrowaną temperaturą wskazuje na znaczną zmienność temperatur w ciągu roku, od najchłodniejszych do najcieplejszych dni.

Rozstęp międzykwartylowy (IQR): 6.1°C, co oznacza, że 50% temperatur w próbie mieści się w przedziale od 9.30°C do 14.40°C. Jest to wskaźnik zmienności temperatury, który nie jest podatny na ekstremalne wartości, w przeciwieństwie do zakresu.

Analizując kurtozy i zakres wartości, można przypuszczać sądzić, że te dwie wartości dostarczają sprzecznych informacji, ale tak naprawdę oba te parametry wskazują na różne aspekty rozkładu temperatur. Możemy się domyślić że wartościach ekstremalnie są rzadkie przez co wynik kurtozy jest blisko trzech. Jednak zakres wartości nie bierze pod uwagę częstości występowania ekstremów. Innymi słowy, choć występuje duża różnica między najwyższą a najniższą temperaturą, większość dni charakteryzowała się temperaturami bliższymi średniej, co jest typowe dla umiarkowanego klimatu bez ekstremalnych skoków temperatur.

Podsumowując, dane pokazują, że temperatura w Melbourne charakteryzuje się średnią temperatura w okolicach 11°C z dość małym zakresem wahania przez rok. Rozkład temperatur jest tylko nieznacznie prawostronnie skośny i płaski.