

13.07.2023, Kraków

# Projekt zaliczeniowy

## Modelowanie w naukach o Ziemi

Wykonały:

Zuzanna Sabat 410176

Emilia Brandys 409963

## 1. Wstęp

Celem projektu było przeanalizowanie zmian zasięgu lodu morskiego na Antarktydzie. Zasięg lodu morskiego jest istotnym wskaźnikiem zmian klimatycznych i ma ważne konsekwencje dla ekosystemu Antarktydy. Ćwiczenie polegało na zbadaniu dynamiki jego zmian na przestrzeni dekad, w różnych szerokościach geograficznych. Wykorzystano dostępne dane z lat 1978-2009, aby skonstruować modele matematyczne oraz wizualizacje minimalnego oraz rzeczywistego zasięgu lodu morskiego na Antarktydzie. Kod i wykresy zostały wykonane w środowisku RStudio przy użyciu bibliotek „tidyverse”, „lubridate”, „matrixStats”, „animation” i „ggplot2”.

## 2. Opis

Dane z pliku `daily_ice_edge.csv` (Rys. 1) przedstawiające zasięg lodu morskiego na Antarktydzie zależny od czasu oraz długości geograficznych zapisano do zmiennej `data` za pomocą funkcji `read.csv()`. Następnie funkcją `data.frame()` stworzono nową ramkę danych, która przechowuje dane z pliku niezawierające pierwszej kolumny `Date`, aby móc swobodnie operować wartościami. Początkowo badania zapisywano w pliku co dwa dni, aż do 9 lipca 1987 roku, po którym pojawiały się już codziennie. Macierz uzupełniono tak, aby nie było niespójności, a więc w dni, w które dane nie były pomierzone oszacowano wartość zasięgu lodu. W tym celu stworzono nową ramkę danych `data_update`, do której zapisywano dane ze zmiennej `data`, a także wyinterpolowano nową wartość, obliczając średnią z dwóch sąsiadujących wierszy. Aby zachować przejrzystość, funkcją `colnames()` zmieniono domyślne nazwy kolumn w ramce `data_update` na oryginalne z ramki `data`.

`#zapisanie danych do zmiennej data`

```
data<-read.csv("daily_ice_edge.csv",header=TRUE,encoding="UTF-8")
```

```
data_pom=data[,2:362]
```

`#uzupełnienie danych tak, aby początkowe dane również były codziennie`

```
data_update<- data.frame(matrix(nrow=0, ncol=361))
```

```
for (i in 1:1589)
```

```
{
```

```
  data_update[nrow(data_update)+1,]=data_pom[i,]
```

```
  data_update[nrow(data_update)+1,]=(data_pom[i,]+data_pom[i+1,])/2
```

```
}
```

```
for (i in 1590:9530)
```

```
{
```

```
  data_update[nrow(data_update)+1,]=data_pom[i,]
```

```
}
```

```
colnames(data_update)<-colnames(data[2:362])
```

```
#zmiana na macierz
matrix_data<-as.matrix(data_update)
```

Date	longitude_0E	longitude_1E	longitude_2E	longitude_3E	longitude_4E
26-Oct-1978	-56.87729	-56.87296	-56.85567	-56.62469	-56.16289
28-Oct-1978	-56.87729	-57.08573	-56.64327	-56.20085	-55.95161
30-Oct-1978	-56.87729	-57.08573	-56.64327	-56.20085	-55.74054
01-Nov-1978	-57.30311	-57.08573	-56.85567	-56.62469	-55.74054
03-Nov-1978	-57.30311	-57.29872	-57.06830	-57.04942	-56.37441
05-Nov-1978	-57.72983	-57.51193	-57.06830	-57.26213	-57.01027
07-Nov-1978	-57.72983	-57.51193	-57.49422	-57.47505	-57.22266
09-Nov-1978	-57.94353	-57.51193	-57.49422	-57.47505	-57.43527
11-Nov-1978	-57.94353	-57.93903	-57.49422	-57.47505	-57.64810
13-Nov-1978	-58.15744	-58.15291	-57.49422	-57.47505	-57.01027

Rys 1. Fragment pliku daily\_ice\_edge.csv przedstawiający wartości szerokości geograficznych, do których sięga lód morski dla kolejnych długości geograficznych.

Następnie za pomocą funkcji **colMins()** stworzono zmienną przechowującą wartości minimalne zasięgu lodu morskiego dla każdej z długości geograficznych (min\_value) oraz wektor południków od -180 do 180 - longitude. Funkcja **na.omit()** została użyta w ramach zabezpieczenia przed wierszami z brakującymi wartościami, natomiast zasadniczo takie wiersze nie wystąpiły w pliku. Połączenie tych elementów w ramce danych pozwoliło na późniejsze przedstawienie minimalnego zasięgu lodu morskiego na Antarktydzie (Rys 2.) za pomocą funkcji **ggplot()**. Aby wykres ilustrował mapę Antarktydy dodano funkcję **coord\_polar()**, dzięki której przekształcono współrzędne z układu kartezjańskiego na układ polarny, co w praktyce oznaczało przedstawienie szerokości geograficznych w formie okręgu, gdzie każdy punkt na okręgu reprezentuje konkretny równoleżnik oraz funkcję **ylim(-90,0)**, która ogranicza wyświetlane wartości na osi pionowej. Ograniczenie to jest szczególnie przydatne w kontekście zasięgu lodu, ponieważ pozwala na lepszą wizualizację danych.

```
#zmienna przechowujaca min zasieg dla kazdego kata
```

```
min_value <- colMins(matrix_data, na.rm = TRUE)
```

```
#wektor dlugosci geograficznych
```

```
longitude <- c(-180:180)
```

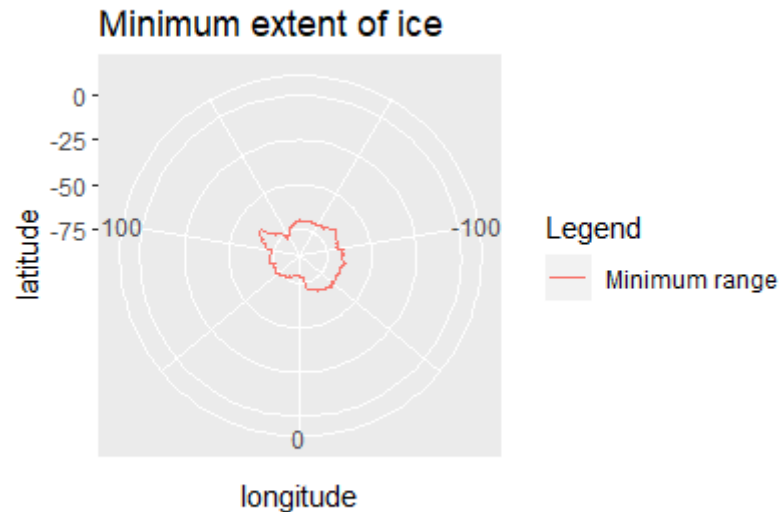
```
min_range_df <- data.frame(longitude = longitude, latitude =
min_value)
```

```
# usuniecie wierszy z brakującymi danymi
```

```
#min_range_df <- na.omit(min_range_df)
```

```
min_range_chart <- ggplot(min_range_df, aes(x = longitude, y =
latitude, col = "Minimum range"))+ggtitle("Minimum extent of
ice")+geom_path()+coord_polar()+ylim(-90,0)
```

min\_range\_chart



Rys 2. Wykres przedstawiający minimalny zasięg lodu morskiego na Antarktydzie.

Kolejnym krokiem było utworzenie ramki danych z jedną kolumną *Data*, aby wyświetlić poszczególne daty na wykresie. Dane przekonwertowano funkcją `as.date()` na typ reprezentujący datę, stworzono wektor dat od 1978-10-26 do 1987-07-08 funkcją `seq()`, a następnie zapisano go jako tekst do kolumny *Data* w nowej ramce danych. Stworzono nową ramkę danych na pozostałe daty z oryginalnej ramki danych *data*, a następnie połączono je ze sobą za pomocą funkcją `rbind()`.

#zapis daty

```
start_date <- as.Date("1978-10-26")
end_date <- as.Date("1987-07-08")
dates <- seq(start_date, end_date, by = "day")

dates_1 <- data.frame(Data = as.character(dates))
# wczytanie późniejszych dat
dates_2 <- data.frame(matrix(nrow = 7941, ncol = 1))
o <- 1

for (t in 1590:9530) {
  dates_2[o, 1] <- data[t, 1]
  o <- o + 1
}

dt <- format(dates_2, format = "%Y-%m-%d")
```

```
colnames(dates_2)<-"Data"
```

```
df_combined <- rbind(dates_1, dates_2)
```

Następnym punktem projektu było stworzenie modelu matematycznego dla zasięgu lodu w funkcji czasu dla każdego kąta z osobna oraz rzeczywistego zasięgu lodu. W tym celu został utworzony blok animacji, a w nim pętla while iterująca po pomocniczej zmiennej „i” do momentu spotkania ostatniej wartości danych przypisanych do zmiennej cols. Rzeczywisty zasięg lodu uzyskano poprzez stworzenie nowej ramki danych, na którą składają się długości geograficzne oraz kolejne wartości z macierzy matrix\_data przedstawiające szerokość geograficzną. Aby zbudować model matematyczny przekonwertowano wartości południków i równoleżników zapisanych w stopniach na radiany. Przedział długości geograficznej od -180 do 180 stopni został przekształcony na przedział od  $-\pi$  do  $\pi$  radianów, a przedział szerokości geograficznej od -90 do 90 na przedział od  $-\frac{\pi}{2}$  do  $\frac{\pi}{2}$ . Następnie za pomocą funkcji **sin()** i **cos()** zostały wyliczone wartości sinusa równoleżników i cosinusa południków. Przekształcenia te miały na celu konwersję między układem kartezjańskim, a układem współrzędnych biegunowych. W dalszej kolejności przy użyciu funkcji **lm()** dopasowano model liniowy regresji do danych szerokości geograficznych przy użyciu zmiennych niezależnych cos\_longitud i sin\_latitude. Aby uzyskać podsumowanie modelu wykorzystano funkcję **summary()** (Rys 3.), która pomaga ocenić jak dobrze model liniowy wyjaśnia zmienność danych. Najważniejszym jej elementem jest współczynnik F-statistic, który jest testem istotności globalnej modelu. Wartość p-value okazała się mniejsza niż  $2.2e-16$ , co daje silne dowody na istotność modelu. Na podstawie dopasowanego modelu linear utworzono funkcję **predict()**, aby obliczyć przewidywane wartości szerokości geograficznej dla nowych danych związanych z długością geograficzną. Następnie utworzono pętlę iterującą przez każdą kolumnę ice\_range\_model\_df i warunek, który sprawdza czy wartość przewidywana jest mniejsza od wartości minimalnego zakresu dla danego kąta geograficznego. Jeżeli warunek się zgadza to przypisywana jest wartość minimalnego zakresu, aby zapewnić, że modelowany zakres nie będzie mniejszy niż rzeczywisty zakres lodu, a tym samym, że model będzie realistyczny. Na samym końcu utworzono wykres na który nałożono minimalny i rzeczywisty zasięg lodu oraz matematyczny model. Tak jak w przypadku wykresu modelu minimalnego dołączono funkcje coord\_polar() oraz ylim(). Stworzono również kolorową legendę oraz nadano tytuł wykresu stworzony przy pomocy zmiennej j, który zmienia się w zależności od dnia agregacji danych. Animację utworzono w przyspieszeniu równym 0,2 pod nazwą ice\_animation.gif. Dodatkowo dołączono pasek postępu wykonania animacji dzięki funkcji **txtProgressBar()** i wyświetlono przykładowy kadr animacji (Rys 4.).

```
i = 1  
j = 0  
cols = 100
```

```
prog_bar <- txtProgressBar(min = 0, max = cols, style = 3)
```

```
saveGIF({  
  
  step <- (-1)
```

```

while (i < cols){

  ice_range_df <- data.frame(longitude = longitude, latitude =
matrix_data[i,])

  #model matematyczny
  cos_longitud <- cos(2*pi*ice_range_df$longitude/360)
  sin_latitude <- sin(2*pi*ice_range_df$latitude/180)

  #dopasowanie modelu liniowego
  linear <- lm(ice_range_df$latitude~cos_longitud+sin_latitude)
  summary(linear)

  prediction <- predict(linear, newdata =
data.frame(ice_range_df$longitude))

  ice_range_df$prediction <- prediction

  ice_range_model_df <- data.frame(longitude = longitude, latitude
=prediction)

  for (k in 1:361) {
    #modelowany zasięg nie jest mniejszy niz rzeczywisty
    if (ice_range_model_df[k,2]<min_range_df[k,2]) {
      ice_range_model_df[k,2]<-min_range_df[k,2]
    }
  }
  full_map <- ggplot(ice_range_df, aes(x = longitude, y = latitude,
col = "Actual range"))+geom_path()+geom_line(data =
ice_range_model_df, aes(x = longitude, y = latitude, col =
"Mathematical model"))+geom_line(data = min_range_df, aes(x =
longitude, y = latitude, col = "Minimum range"))+coord_polar()+ylim(-
90,0)+ggtitle(df_combined[j,1])+labs(color = 'Legend')

  print(full_map)

  step <- step+1

  setTxtProgressBar(prog_bar, step)

  i <- i + 1
  j <- j + 1
}
},interval = 0.2, movie.name = "ice_animation.gif")

print(full_map)

```

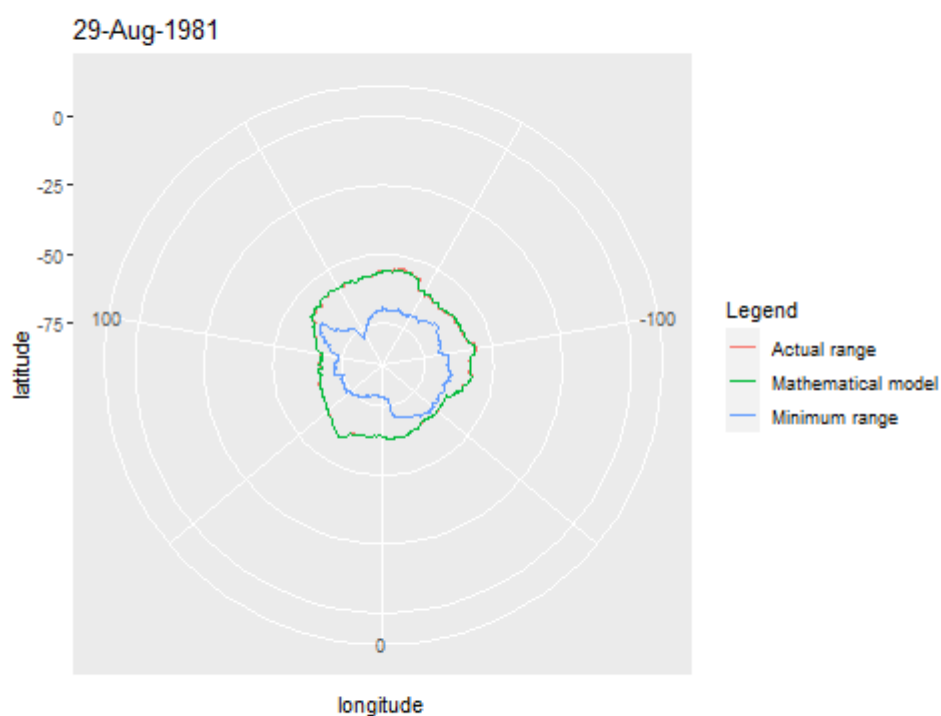
```
Call:
lm(formula = ice_range_df$latitude ~ cos_longitud + sin_latitude)

Residuals:
    Min       1Q   Median       3Q      Max
-0.35937 -0.20135 -0.07867  0.10679  1.31965

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -93.75431    0.11040  -849.236 < 2e-16 ***
cos_longitud   0.09505    0.02448   3.884 0.000123 ***
sin_latitude  -37.37138    0.16185  -230.899 < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3044 on 358 degrees of freedom
Multiple R-squared:  0.9942,    Adjusted R-squared:  0.9942
F-statistic: 3.081e+04 on 2 and 358 DF,  p-value: < 2.2e-16
```

Rys 3. Szczegółowe informacje na temat dopasowania modelu regresji liniowej dla modelu matematycznego.



Rys 4. Przykładowy widok minimalnego i rzeczywistego zasięgu lodu morskiego oraz modelu matematycznego na Antarktydzie z dnia 29 sierpnia 1981 roku zaczerpnięty z animacji ice\_animation.gif.

Na samym końcu obliczono średnie wartości zasięgu lodu na Antarktydzie dla każdego dnia w ciągu roku. Zastosowano do tego pętlę for, w której tworzony był wektor jako sekwencja indeksów odpowiadająca danemu dniu w ciągu roku. W kolejnej pętli dla każdego dnia wyliczone zostały współrzędne longitude\_polar i latitude\_polar na podstawie wcześniej obliczonych średnich wartości zasięgu. Finalnie dane te zostały przeliczone na współrzędne polarne i wizualizowane.

```
degrees <- seq(1:361)
#Średni zasięgu w ciągu roku
daily_mean <- data.frame(matrix(NA, nrow=0, ncol=361))
day <- 1
for(i in 1:365){
  vector <- seq(day,11119,365)
  temporary <- matrix_data[vector, ]
  daily_mean[i, ] <- colMeans(temporary[, ])
  day <- day + 1
}
View(daily_mean)
for(i in 1:365){
  degrees <- seq(1:362)

  ice_yearly <- as.numeric(daily_mean[i, ])
  longitude_polar <- ice_yearly*cos(degrees*pi/180)
  latitude_polar <- ice_yearly*sin(degrees*pi/180)

  # Stworzenie wykresu dla danej klatki animacji
  map_df <- data.frame(x = longitude_polar, y = latitude_polar)
  map <- ggplot(map, aes(x = x, y = y)) +
    geom_line(color = "blue") +
    ggtitle(paste("Day:", i))
```

### 3. Wnioski

Analiza przeprowadzonych modeli wskazuje na wahania zasięgu lodu morskiego na Antarktydzie w ciągu roku. Stworzona animacja, która przedstawia zmiany rzeczywistego i wymodelowanego zasięgu lodu na przestrzeni lat pozwoliła lepiej dostrzec te różnice. Warto zauważyć, że model matematyczny stosunkowo dokładnie pokrywał się z rzeczywistym zasięgiem lodu morskiego. Tego typu badania pozwalają lepiej zrozumieć mechanizmy kształtujące zasięg lodu i przewidywać ewentualne skutki jego topnienia i wpływu na poziom mórz i oceanów.