PROJEKT ZALICZENIOWY

KAMILA SOĆKO NR. 402770, INŻYNIERIA I ANALIZA DANYCH

UŻYTE BIBLIOTEKI:

```
# ZALADOWANIE WYKORZYSTYWANYCH PAKIETOW
```{r}
library(dplyr)
library(tidyverse)
library(graphics)
library(magrittr)
library(lattice)
library(latticeExtra)
library(aplpack)
library(plotly)
library(ggplot2)
```
```

ETAP I – WCZYTANIE DANYCH I PODSUMOWANIE DANYCH

```
myData<-read.csv("Socko_dane_surowe.csv", skip=1, header=FALSE,
col.names=c("Period", "Sex", "Age", "Count"))
myData
```

Wczytałam plik .csv z danymi do zmiennej myData oraz nadałam nazwy kolumn dla danych. Wyrażeniem skip=1 omijam pierwszy wiersz, z wejściowymi nazwami kolumn, nazwałam je w kolejnej linijce.

Początek danych:

```
> myData
    Period    Sex         Age Count
1     2008   Male      Infant   198
2     2008   Male        1-4    45
3     2008   Male        5-9    18
4     2008   Male      10-14    33
5     2008   Male      15-19   132
6     2008   Male      20-24   153
7     2008   Male      25-29   120
8     2008   Male      30-34   123
9     2008   Male      35-39   189
10    2008   Male      40-44   249
11    2008   Male      45-49   366
12    2008   Male      50-54   477
13    2008   Male      55-59   711
14    2008   Male      60-64   978
15    2008   Male      65-69  1260
16    2008   Male      70-74  1506
17    2008   Male      75-79  2148
18    2008   Male      80-84  2529
```

```
exists('myData') && is.data.frame(get('myData'))
any(is.na(myData))
summary(myData)
str(myData)
typeof(myData)
length(myData)
class(myData)
head(myData)
tail(myData)
write.csv(myData, "Socko_dane_przeksztalcone.csv")
```

Sprawdziłam czy dane się wczytały i istnieją oraz czy są data.frame, a następnie sprawdziłam czy w moich danych występuje NA – nie występują.

```
[1] TRUE
[1] FALSE
```

Dokonałam podsumowania danych, sprawdziłam typ każdej z kolumn, typ danych, ilość kolumn.

```
     Period          Sex                 Age                  Count       
 Min.   :2008   Length:828         Length:828         Min.   :    6  
 1st Qu.:2011   Class :character   Class :character   1st Qu.:  117  
 Median :2014   Mode  :character   Mode  :character   Median :  366  
 Mean   :2014                                         Mean   : 1792  
 3rd Qu.:2016                                         3rd Qu.: 1635  
 Max.   :2019                                         Max.   :34260  
'data.frame':   828 obs. of  4 variables:
 $ Period: int  2008 2008 2008 2008 2008 2008 2008 2008 2008 2008 ...
 $ Sex   : chr  "Male" "Male" "Male" "Male" ...
 $ Age   : chr  "Infant" "1-4" "5-9" "10-14" ...
 $ Count : int  198 45 18 33 132 153 120 123 189 249 ...
[1] "list"
[1] 4
[1] "data.frame"
```

Oraz wyświetliłam początek i koniec moich danych:

```
> head(myData)
  Period  Sex    Age Count
1   2008 Male Infant   198
2   2008 Male    1-4    45
3   2008 Male    5-9    18
4   2008 Male  10-14    33
5   2008 Male  15-19   132
6   2008 Male  20-24   153
>
>
> tail(myData)
    Period   Sex          Age Count
823   2019 Total        80-84  4881
824   2019 Total        85-89  5889
825   2019 Total        90-94  4599
826   2019 Total        95-99  1890
827   2019 Total 100 and over   324
828   2019 Total        Total 34260
```

Zapisałam dane do pliku „Socko_dane_przekształcone.csv”

ETAP 2 – PRACA Z DANYMI – wykorzystanie pakietu tidyverse

1. *Wykonałam podsumowanie dla danych dot. różnych płci:*

```
male_data<-myData%>%filter(Sex=="Male")
female_data<-myData%>% filter(Sex=="Female")
total_data<-myData%>% filter(Sex=="Total")

print("PODSUMOWNANIE DLA MEZCZYZN: ")
summary(male_data)
print("PODSUMOWANIE DLA KOBIET")
summary(female_data)
print("DLA KAZDEJ Z PLCI:")
summary(total_data)
```

```
> print("PODSUMOWNANIE DLA MEZCZYZN: ")
[1] "PODSUMOWNANIE DLA MEZCZYZN: "
> summary(male_data)
     Period         Sex                 Age                 Count
 Min.   :2008   Length:276         Length:276         Min.   :    6
 1st Qu.:2011   Class :character   Class :character   1st Qu.:  117
 Median :2014   Mode  :character   Mode  :character   Median :  339
 Mean   :2014                                         Mean   : 1353
 3rd Qu.:2016                                         3rd Qu.: 1406
 Max.   :2019                                         Max.   :17583
> print("PODSUMOWANIE DLA KOBIET")
[1] "PODSUMOWANIE DLA KOBIET"
> summary(female_data)
     Period         Sex                 Age                 Count
 Min.   :2008   Length:276         Length:276         Min.   :    6.0
 1st Qu.:2011   Class :character   Class :character   1st Qu.:   63.0
 Median :2014   Mode  :character   Mode  :character   Median :  268.5
 Mean   :2014                                         Mean   : 1335.5
 3rd Qu.:2016                                         3rd Qu.: 1170.8
 Max.   :2019                                         Max.   :16677.0
> print("DLA KAZDEJ Z PLCI:")
[1] "DLA KAZDEJ Z PLCI:"
> summary(total_data)
     Period         Sex                 Age                 Count
 Min.   :2008   Length:276         Length:276         Min.   :   15.0
 1st Qu.:2011   Class :character   Class :character   1st Qu.:  194.2
 Median :2014   Mode  :character   Mode  :character   Median :  621.0
 Mean   :2014                                         Mean   : 2688.4
 3rd Qu.:2016                                         3rd Qu.: 2682.8
 Max.   :2019                                         Max.   :34260.0
>
```

2. *Używając „filter" wybrałam dane z 2019 roku dla mężczyzn i kobiet, gdy liczba zgonów była mniejsza od 100*

```
myData %>% filter(Period==2019 & Sex!="Total") %>% filter(Count<100)
```

Period <int>	Sex <chr>	Age <chr>	Count <int>
2019	Male	1–4	30
2019	Male	5–9	15
2019	Male	10–14	30
2019	Male	15–19	87
2019	Male	100 and over	54
2019	Female	1–4	24
2019	Female	5–9	12
2019	Female	10–14	24
2019	Female	15–19	45
2019	Female	20–24	54

1-10 of 13 rows Previous 1 2 Next

3. *Wybrałam kolumnę, która kończy się na „od"*

```
select(myData, ends_with("od"))
```

Period <int>
2008
2008
2008
2008
2008
2008
2008
2008
2008
2008

1-10 of 828 rows Previous 1 2 3 4 5 6 ... 83 Next

4. *Wybrałam kolumnę „Age" i wszystkie inne pasujące do niej kolumny z odpowiadającymi wierszami.*

```
select(myData, Age, everything())
```

Age <chr>	Period <int>	Sex <chr>	Count <int>
Infant	2008	Male	198
1–4	2008	Male	45
5–9	2008	Male	18
10–14	2008	Male	33
15–19	2008	Male	132
20–24	2008	Male	153
25–29	2008	Male	120
30–34	2008	Male	123
35–39	2008	Male	189
40–44	2008	Male	249

1-10 of 828 rows Previous 1 2 3 4 5 6 ... 83 Next

5. *Wybrałam dane, zmieniając jednej z kolumn nazwę.*

```
rename(myData, Gender = Sex)
```

Period <int>	Gender <chr>	Age <chr>	Count <int>
2008	Male	Infant	198
2008	Male	1-4	45
2008	Male	5-9	18
2008	Male	10-14	33
2008	Male	15-19	132
2008	Male	20-24	153
2008	Male	25-29	120
2008	Male	30-34	123
2008	Male	35-39	189
2008	Male	40-44	249

1-10 of 828 rows Previous 1 2 3 4 5 6 ... 83 Next

6. *Wyfiltrowałam dane – „Period, Age, Count" niezawierające lat 100 i wyżej, oraz wieku niemowlęcego oraz wyeliminowałam wiek oraz płeć – total, czyli łączne wartości. Pogrupowałam względem ilości i posortowałam względem ilości zgonów.*

```
myData %>%
  filter(Age!="100 and over" & Age!="Total" & Sex!= "Total" &
Age!="Infant")%>%
  select(Period, Age, Count) %>%
  group_by(Count) %>% arrange(Count)
```

Period <int>	Age <chr>	Count <int>
2010	5-9	6
2011	5-9	6
2011	5-9	9
2014	5-9	9
2009	5-9	12
2010	5-9	12
2012	5-9	12
2013	5-9	12
2013	5-9	12
2014	10-14	12

1-10 of 480 rows Previous 1 2 3 4 5 6 ... 48 Next

7. *Przefiltrowałam dane, tak aby mieć tylko łączne wartości zarówno dla kobiet jak i mężczyzn w wieku 2019 i posortowałam malejąco względem ilości zgonów.*

```
myData %>%
  filter(Sex=="Total") %>%
  filter(Period==2019) %>% arrange(desc(Count))
```

Period <int>	Sex <chr>	Age <chr>	Count <int>
2019	Total	85-89	5889
2019	Total	80-84	4881
2019	Total	90-94	4599
2019	Total	75-79	4110
2019	Total	70-74	3372
2019	Total	65-69	2466
2019	Total	95-99	1890
2019	Total	60-64	1818
2019	Total	55-59	1428
2019	Total	50-54	990

1-10 of 22 rows Previous 1 2 3 Next

8. *Używając „mutate" zmieniłam nazwę dla wartości Infant z kolumny Age na Niemowle.*

```
myData %>%
  filter(Age=="Infant")%>% mutate(Age="Niemowle")
```

Period <int>	Sex <chr>	Age <chr>	Count <int>
2008	Male	Niemowle	198
2008	Female	Niemowle	126
2008	Total	Niemowle	321
2009	Male	Niemowle	168
2009	Female	Niemowle	138
2009	Total	Niemowle	306
2010	Male	Niemowle	186
2010	Female	Niemowle	141
2010	Total	Niemowle	327
2011	Male	Niemowle	168

1-10 of 36 rows Previous 1 2 3 4 Next

9. *Wyświetliłam dane z 2008 roku zawierające informację na temat wieku, w którym była największa ilość zgonów u kobiet.*

```
myData %>% filter(Period==2008 & Sex=="Female") %>%
  filter(Age!="Total")%>%
  slice_max(Count)
```

Period <int>	Sex <chr>	Age <chr>	Count <int>
2008	Female	85-89	2847

Oraz najmniejsza ilość zgonów u mężczyzn w 2010 roku.

```
myData %>% filter(Period==2010 & Sex=="Male") %>%
  filter(Age!="Total")%>%
  slice_min(Count)
```

Period <int>	Sex <chr>	Age <chr>	Count <int>
2010	Male	5-9	12

10. *Do zmiennej specificCount wstawiłam dane, zawierające informacje na temat ilości zgonów w roku 2019 w zależności od wieku – nie biorąc pod uwagę niemowląt i ludzi w wieku 100 lub więcej. Dot. tylko kobiet. Wybrałam kolumnę Count oraz Age.*

```
specificCount<-myData %>%
  filter(Age!="100 and over" & Age!="Total" & Age!="Infant")%>%
  filter(Period==2019 & Sex=="Female") %>% select(Count, Age)
```

Count <int>	Age <chr>
24	1-4
12	5-9
24	10-14
45	15-19
54	20-24
60	25-29
87	30-34
84	35-39
159	40-44
282	45-49

1-10 of 20 rows Previous 1 2 Next

11. *Obliczyłam średnią dla danych wybranych z poprzedniego podpunktu dla Count, wartości zaokrągliłam do jednego miejsca po przecinku oraz wykonałam ogólne podsumowanie danych.*

```
srednia=round(mean(specificCount[1:20,1]), digits = 1)
srednia
summary(specificCount)
```

```
[1] 814.4
> summary(specificCount)
     Count            Age
 Min.   :  12.0   Length:20
 1st Qu.:  58.5   Class :character
 Median : 340.5   Mode  :character
 Mean   : 814.4
 3rd Qu.:1352.2
 Max.   :3084.0
```

ETAP 3 – WIZUALIZACJE

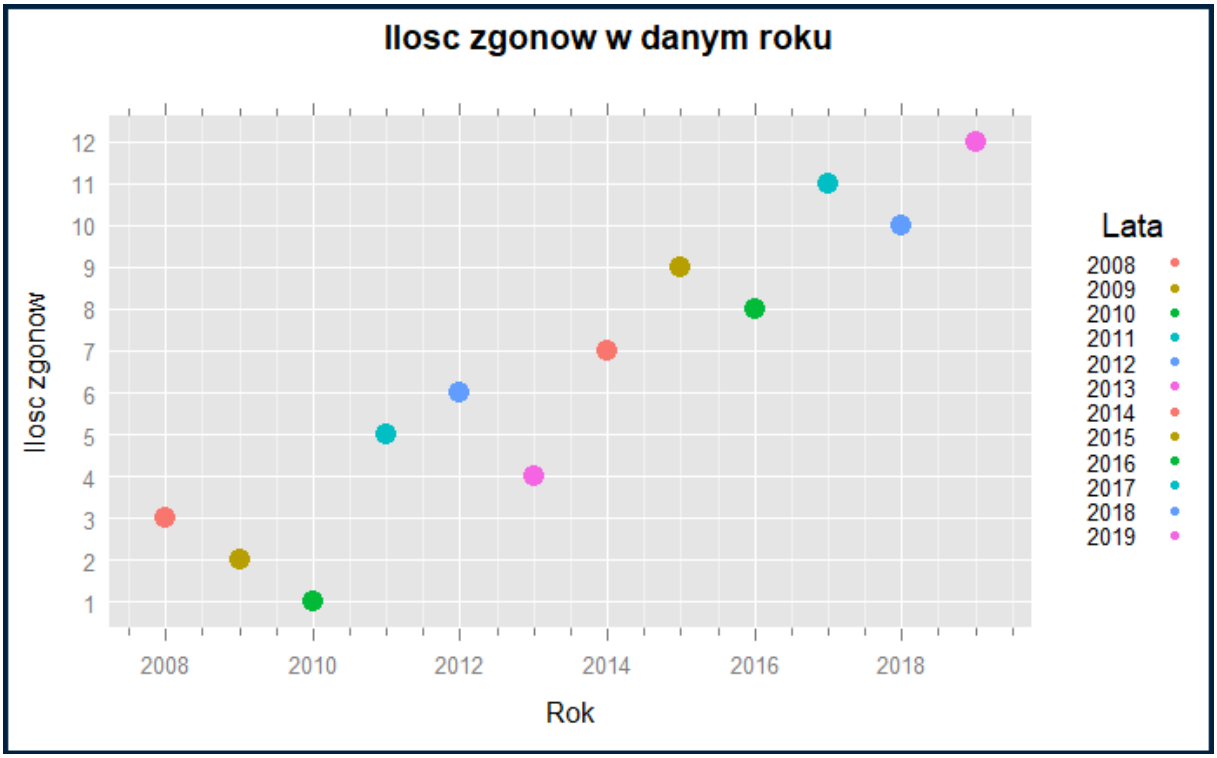
WYKRES 1 – wykorzystanie pakietu lattice oraz latticeExtra

Na początek wybrałam dane, z których skorzystałam przy wykonywaniu wykresu. Wybrałam dane zawierające informacje ogólne bez wyróżniania wieku oraz płci, pogrupowałam dane względem ilości zgonów.

```
period_count<-myData %>%
  filter(Age=="Total" & Sex=="Total") %>% select(Period,Count) %>%
group_by(Count)
period_count
```

Wykonałam wykres przedstawiający, ile było zgonów w danym roku. Skorzystałam z funkcji dotplot, dodałam kolorowe kropki oznaczające dany rok (jeden kolor na dwa okresy czasu), dzięki auto.key informacje są po prawo, mają tytuł „Lata" oraz została ustawiona wielkość czcionki dla tytułu legendy. Kropki również zostały delikatnie powiększone.

```
dotplot(Count ~ Period, period_count, group = Period,
 main = "Ilosc zgonow w danym roku",
 xlab = "Rok",
 ylab = "Ilosc zgonow",
 auto.key = list(space = "right", title = "Lata", cex.title=1.2),
 par.settings=ggplot2like(),
 lattice.options = ggplot2like.opts(),
 cex=1.5)
```



Na wykresie zauważamy, że najmniej zgonów było w roku 2010, a najwięcej w 2019.

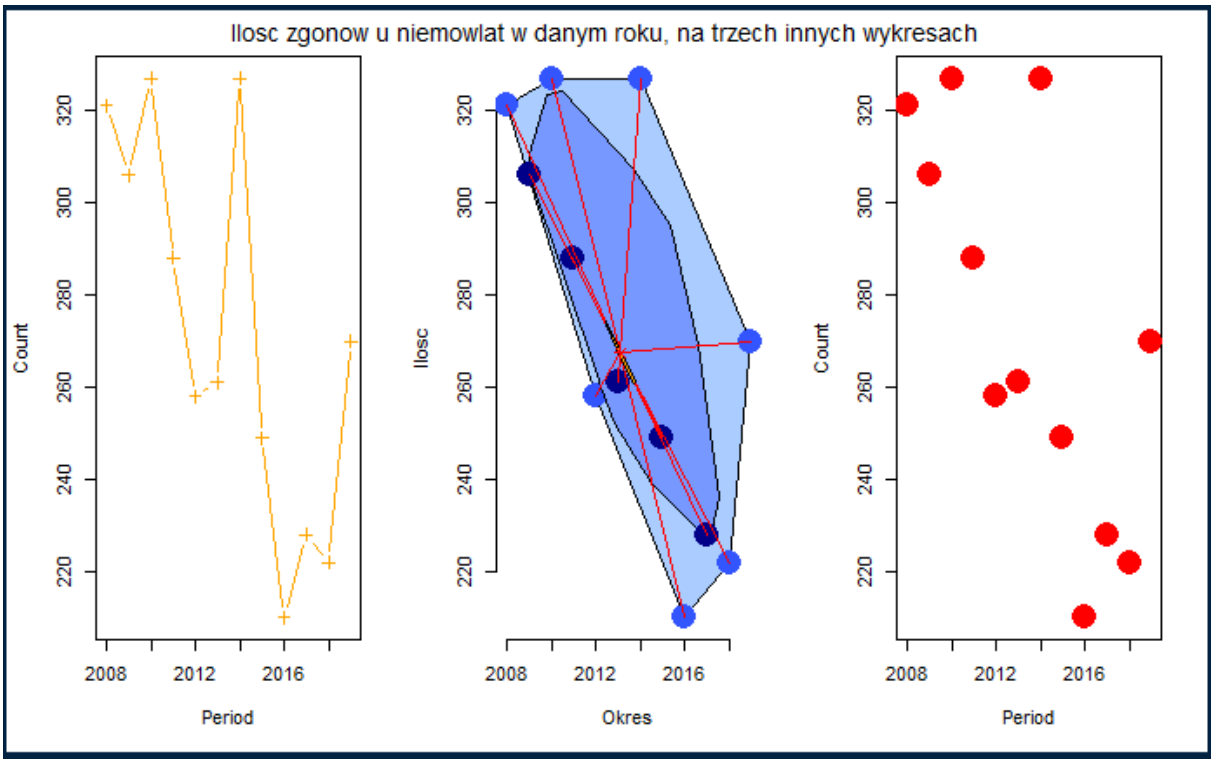
WYKRES 2 – wykorzystanie pakietu graphics i aplpack

Wybrałam dane (kolumny Period i Count) dotyczące niemowląt bez względu na płeć oraz pogrupowałam dane względem ilości zgonów.

```
total_count_period<-myData %>%
  filter(Age=="Infant" & Sex=="Total") %>% select(Period, Count) %>%
group_by(Count)
```

Wykonałam 3 wykresy przedstawiające to samo tylko w innej wersji graficznej. Wykorzystując funkcje plot ustawiłam wykres na połączone pomarańczowe krzyżyki. Funkcja bagplot – zmieniłam nazwy osi na polskie nazwy oraz powiększyłam kropki, które są połączone. Dodatkowo funkcją sunflowerplot stworzyłam wykres z czerwonymi punktami i również je powiększyłam. Dodałam tytuł do wykresów.

```
par(mfrow=c(1,3))
plot(total_count_period, col="orange",type="b",pch=3)
bagplot(total_count_period$Period,total_count_period$Count,cex=3,
xlab="Okres", ylab="Ilosc")
sunflowerplot(total_count_period, col="red", cex=3)
mtext("Ilosc zgonow u niemowlat w danym roku, na trzech innych
wykresach", outer=TRUE,  cex=0.9, line=-1.6)
```



W roku 2010 oraz 2014 była największa ilość zgonów u niemowląt, najmniej w 2016.

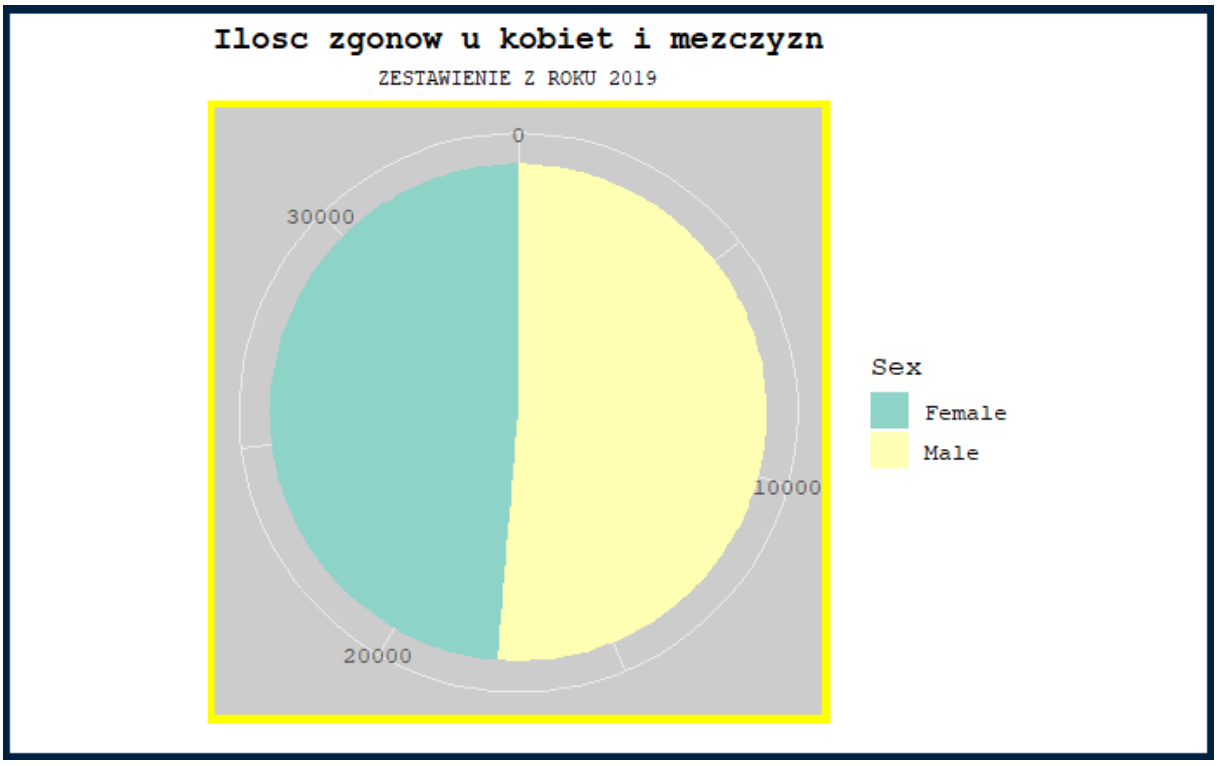
WYKRES 3 – WYKORZYSTANIE PAKIETU GGPLOT2

Wybrałam dane w zależności od płci oraz ilości zgonów w roku 2019.

```
count_sex<-myData%>% filter(Age=="Total" & Sex!="Total" & Period==2019)
%>% select(Count, Sex)
count_sex
```

Wykonałam wykres kołowy przedstawiający, ile mężczyzn i kobiet umarło w 2019. Ustawiłam w coord_polar, że wykres zaczyna się od 0, czyli od północnego punktu na wykresie. Wybrałam paletę kolorów „Set3" oraz ustawiłam inną czcionkę oraz rozmiar. Dodatkowo ustawiłam kolor oraz grubość obramowania wykresu i kolor środkowego tła. Usunęłam zbędne tytuły z osi. Pogrubiłam tytuł wykresu, wyśrodkowałam i zmieniłam rozmiar, co więcej – wykorzystując ggtitle dodałam podtytuł.

```
ggplot(count_sex, aes(x="", y=Count, fill=Sex)) + geom_bar(width = 1,
stat = "identity")+coord_polar("y", start=0)+
  scale_fill_brewer(palette="Set3")+theme_minimal(base_family = "mono",
base_size = 12) +
  theme(
  panel.background = element_rect(fill = "#CCCCCC",
                                  colour = "yellow",
                                  size = 3, linetype = "solid"),
  axis.title.x = element_blank(),
  axis.title.y = element_blank(),
  axis.ticks = element_blank(),
  plot.title=element_text(size=14, face="bold", hjust = 0.5),
  plot.subtitle = element_text(size=9, hjust=0.5)
  ) + ggtitle("Ilosc zgonow u kobiet i mezczyzn", subtitle = "ZESTAWIENIE
Z ROKU 2019")
```



Zauważamy, że w 2019 roku zmarło więcej mężczyzn niż kobiet, ale różnica nie jest diametralna.

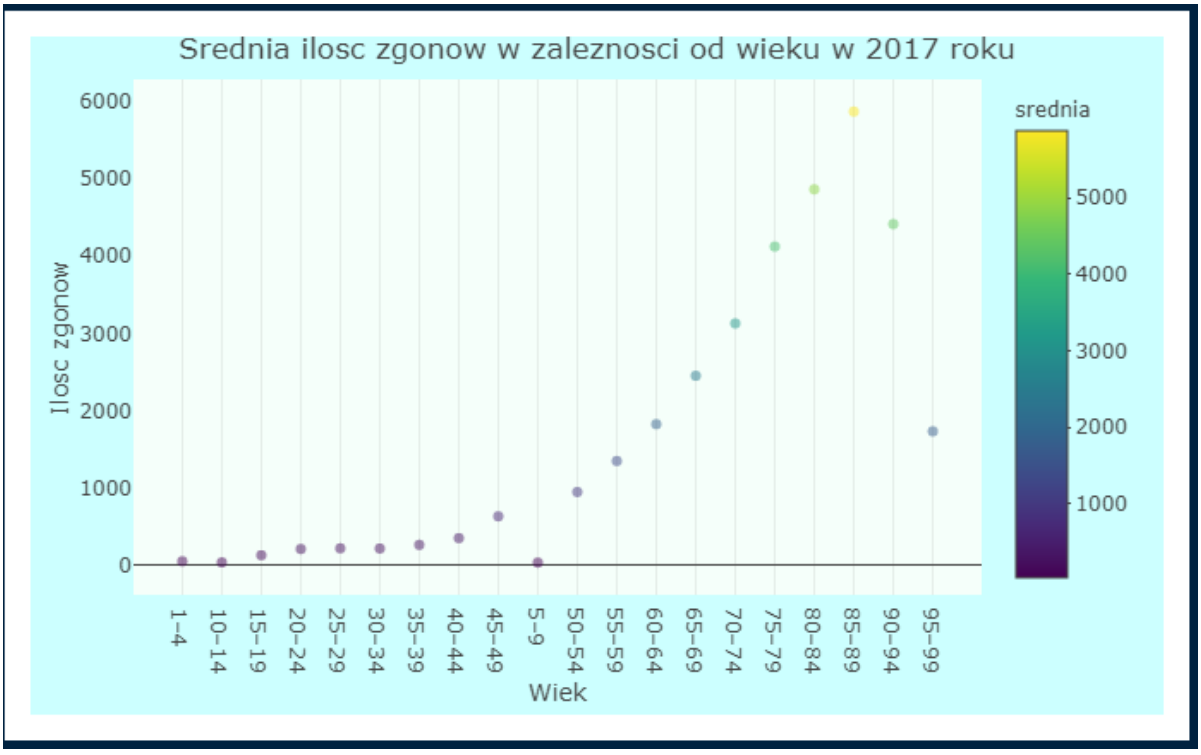
WYKRES 4 – wykorzystanie pakietu plotly

Do wykresu potrzebowałam danych, które będą zawierały informację na temat wszystkich ludzi w 2017 roku w zależności od wieku. (Nie brałam pod uwagę osób, które przeżyły 100 lub więcej lat oraz niemowląt). Pogrupowałam dane względem wieku oraz wyliczyłam średnią ilość zgonów dla danego wieku.

```
countAge<-myData %>%
  filter(Age!="100 and over" & Age!="Total" & Age!="Infant")%>%
  filter(Period==2017 & Sex=="Total") %>% group_by(Age)%>%
summarise(srednia=mean(Count))
countAge
```

Wykonałam wykres używając funkcji plot_ly, punkty ustawiłam na półprzezroczyste, dodałam nazwy osi oraz pionową siatkę do wykresu. Kolor tła na wykresie ustawiłam na mincream oraz kolor poza wykresem na jasnoniebieski.

```
countAge%>%
plot_ly(x=~Age, y=~srednia, color=~srednia) %>%
add_markers(marker=list(opacity=0.5), showlegend=FALSE) %>%
layout(xaxis = list(title="Wiek", showgrid=TRUE),
yaxis = list(title="Ilosc zgonow", showgrid=FALSE),
title = "Srednia ilosc zgonow w zaleznosci od wieku w 2017 roku",
paper_bgcolor="#CCFFFF", plot_bgcolor="mintcream")
```



Średnio najmniejsza ilość zgonów w 2017 roku była w przedziale wiekowym 5-9 lat, największa między 85 a 89 rokiem życia.

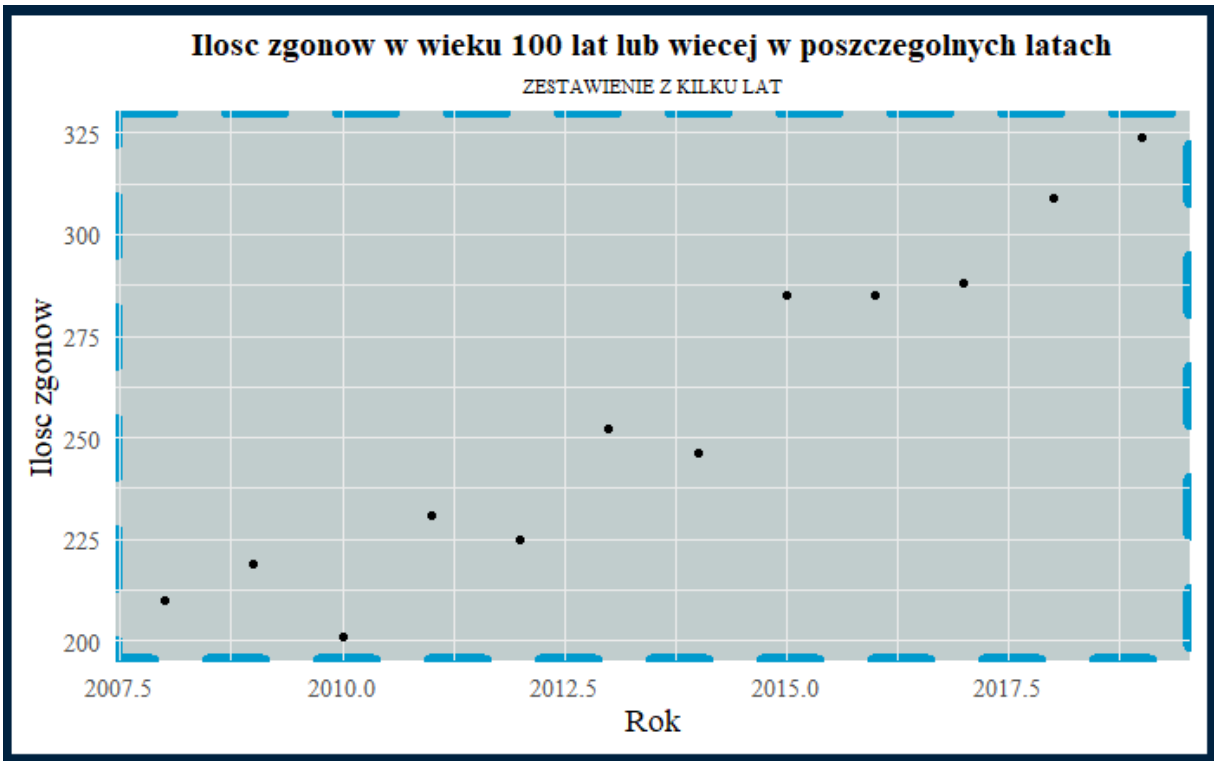
WYKRES 5 – wykorzystanie pakietu ggplot2

Do celów sporządzenia wykresu użyłam informacji na temat ludzi, którzy umarli w wieku 100 i powyżej. Wybrałam kolumnę Period i Count.

```
higher_age_total<-myData %>%
  filter(Age=="100 and over" & Sex=="Total")%>%
  select(Period,Count)
higher_age_total
```

Korzystając z ggplot wykonałam wykres punktowy przedstawiający ile osób w wieku 100 lub więcej umarło w poszczególnych latach. Ustawiłam kolor w środku wykresu na „azure3" oraz zrobiłam przerywane obramowanie kolorem „deepskyblue3". Tytuł wykresu pogrubiłam i wyśrodkowałam, dodałam również podtytuł do wykresu oraz nazwałam osie.

```
ggplot(higher_age_total, aes(x=Period, y=Count,
group=Count))+geom_point()+theme_minimal(base_family = "serif", base_size
= 14) +
  theme(
  panel.background = element_rect(fill = "azure3",
                                  colour = "deepskyblue3",
                                  size = 3, linetype = "dashed"),
  plot.title=element_text(size=14, face="bold", hjust = 0.5),
  plot.subtitle = element_text(size=8, hjust=0.5)
  ) + ggtitle("Ilosc zgonow w wieku 100 lat lub wiecej w poszczegolnych
latach", subtitle = "ZESTAWIENIE Z KILKU LAT") + xlab("Rok")+ylab("Ilosc
zgonow")
```



Zauważamy wzrost w ilości osób, które dożyły 100 lub więcej lat biorąc pod uwagę zgony między 2008 a 2019 rokiem.