

Akademia Górniczo-Hutnicza w Krakowie
Wydział Geologii, Geofizyki i Ochrony Środowiska



Sprawozdanie z Projektu 3
w ramach przedmiotu Przetwarzanie Danych Środowiskowych

autorzy: Maciej Bąk, Filip Giermek gr.1

Kraków/Olszyny, 2.05.2020

Przedmiot analizy

Przedmiotem analizy niniejszego projektu jest tabela wartości rocznych przyrostów modrzewi rosnących na terenie Nadleśnictwa Śnieżka z lat 1869-2011, oraz tabele danych wartości średnich temperatur i sum opadów na terenie Nadleśnictwa w latach 1900-2000.

0.1. Przygotowanie danych i bibliotek

Importujemy dane do R. Dla plików z danymi klimatycznymi konieczne było zastosowanie argumentu `header = F`. Następnie dodajemy nazwy kolumn do danych klimatycznych. Aby to zrobić, najpierw tworzymy wektor zawierający nazwy miesięcy, a następnie z wykorzystaniem funkcji `colnames()` nadajemy nazwy kolumnom.

```
library(tidyr)

library(dplyr)

library(ggplot2)
library(amap)
library(stats)

sn2_res <- read.delim(file = "sn2_res.txt")
sn2_t <- read.delim(file = "sn2_160_101_t.txt", header = F)
sn2_p <- read.delim(file = "sn2_160_101_p.txt", header = F)

nazwy_miesiecy <- c("Rok", "Styczeń", "Luty", "Marzec", "Kwiecień", "Maj", "Czerwiec", "Lipiec", "Sierpień", "Wrzesień", "Październik", "Listopad", "Grudzień")

colnames(sn2_t) <- nazwy_miesiecy
colnames(sn2_p) <- nazwy_miesiecy
```

0.2. Przycięcie danych

Przed przystąpieniem do analizy należy ustalić wspólny okres dla wszystkich danych. Dane klimatyczne obejmują lata 1901 - 2000, dlatego też "przytniemy" dane dendrochronologiczne do tych ram czasowych. Wykorzystamy do tego funkcję `subset()` wraz z prostym warunkiem logicznym. Dane w przewarżającej części są kompletne. Jedynym wyjątkiem jest kolumna X4sn2_14 - dane urywają się na roku 1971 co sprawia, że kolumna ta nie nadaje się do analizy. Z uwagi na duże braki w danych nierozsądnym byłoby zastępowanie ich średnią pozostałych, kompletnych wierszy. Usuwamy ją za pomocą komendy `select()`.

```
sn2_res_cut <- subset(sn2_res, year >= 1901 & year <= 2000)
sn2_res_cut <- select(sn2_res_cut, 1:13, 15)
#zmiana nazw wierszy
row.names(sn2_res_cut) <- c(1:100)
```

1.1. Uśrednienie serii przyrostów

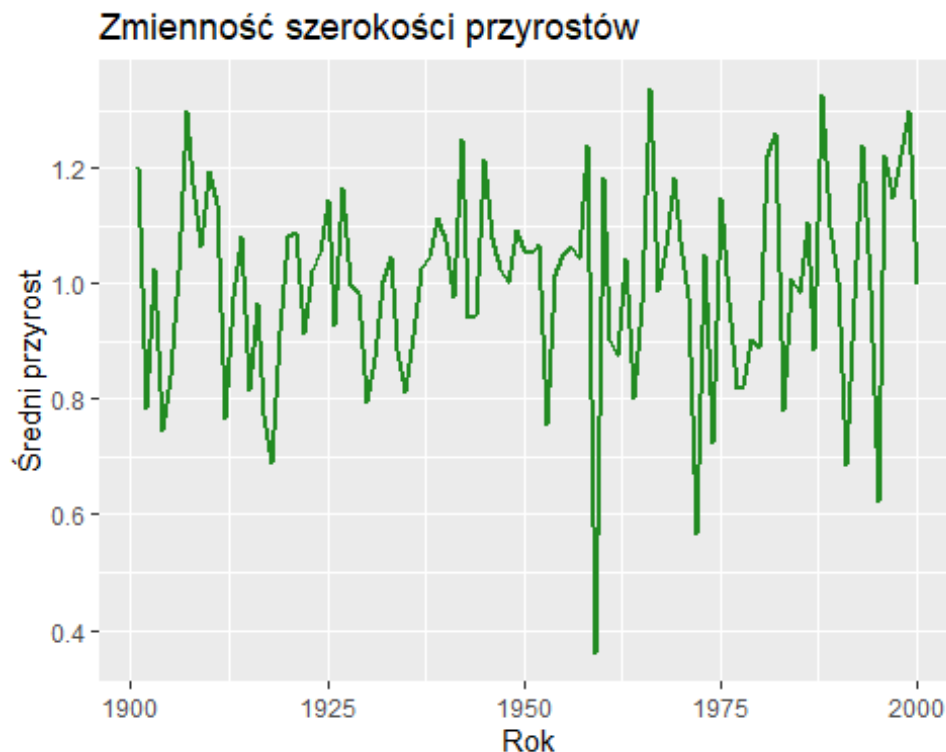
Do ramki danych `sn2_res_cut` dodamy kolumnę, która będzie zawierała chronologię stanowiskową. Wykorzystamy funkcję `rowMeans()`. Chronologia obejmuje lata 1901-2000 i została utworzona na podstawie 13 serii.

```
sn2_res_cut$srednia <- rowMeans(sn2_res_cut[,2:14])
```

1.2 Wykres zmienności szerokości przyrostów

Utworzymy wykres liniowy prezentujący zmienność szerokości przyrostów na przestrzeni lat.

```
ggplot(sn2_res_cut, aes(year, srednia)) +
  geom_line(color = "forestgreen", size = 1) +
  ylab("Średni przyrost") +
  xlab("Rok") +
  ggtitle("Zmienność szerokości przyrostów")
```



wykr. 1.2

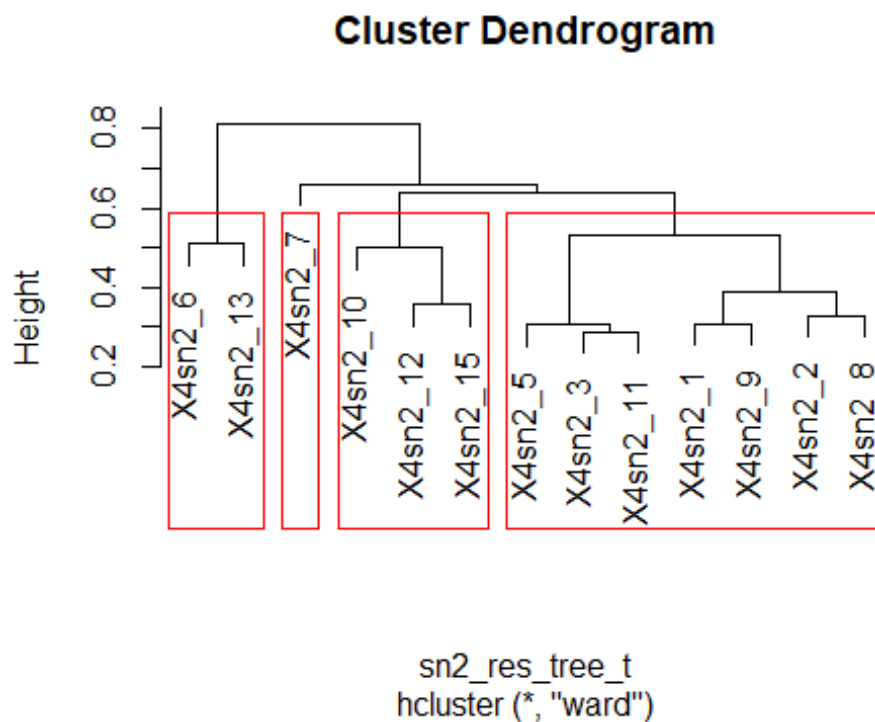
2.1 Grupowanie serii przyrostów (hierarchiczne drzewo)

Aby móc utworzyć hierarchiczne drzewo edytujemy ramkę danych. Najpierw tworzymy ramkę bez kolumn lat i średniej i wykonujemy transpozycję.

```
sn2_res_tree <- sn2_res_cut[,2:14]  
sn2_res_tree_t <- t(sn2_res_tree)
```

Następnie z wykorzystaniem funkcji z pakietu *amap()* tworzymy drzewo i wydzielamy grupy.

```
tree_przyr <- hcluster(sn2_res_tree_t, method = "correlation", link="ward")  
plot(tree_przyr)  
rect.hclust(tree_przyr, k=4, border="red")
```



wykr. 2.1

2.2 Liczenie średnich grupowych

Najpierw utworzymy wektory z kolejnych kolumn, które posłużą do tworzenia wykresów dla grup.

```
rok <- sn2_res_cut$year
k_1 <- sn2_res_cut$X4sn2_1
k_2 <- sn2_res_cut$X4sn2_2
k_3 <- sn2_res_cut$X4sn2_3
k_5 <- sn2_res_cut$X4sn2_5
k_6 <- sn2_res_cut$X4sn2_6
k_7 <- sn2_res_cut$X4sn2_7
k_8 <- sn2_res_cut$X4sn2_8
k_9 <- sn2_res_cut$X4sn2_9
k_10 <- sn2_res_cut$X4sn2_10
k_11 <- sn2_res_cut$X4sn2_11
k_12 <- sn2_res_cut$X4sn2_12
k_13 <- sn2_res_cut$X4sn2_13
k_14 <- sn2_res_cut$X4sn2_14
k_15 <- sn2_res_cut$X4sn2_15
```

Następnie tworzymy ramki klastrów złożone analogicznie jak grupy na drzewie. (z pominięciem kolumny X4sn2_7 – tworzyła samotnie jedną grupę)

```
klaster1_1 <- data.frame(rok, k_6, k_13)
klaster1_2 <- data.frame(rok, k_10, k_12, k_15)
klaster1_3 <- data.frame(rok, k_5, k_3, k_11, k_1, k_9, k_2, k_8)
```

2.3 Uśrednieni przyrostów w klastrach

Tworzymy w każdej ramce kolumnę *srednia*.

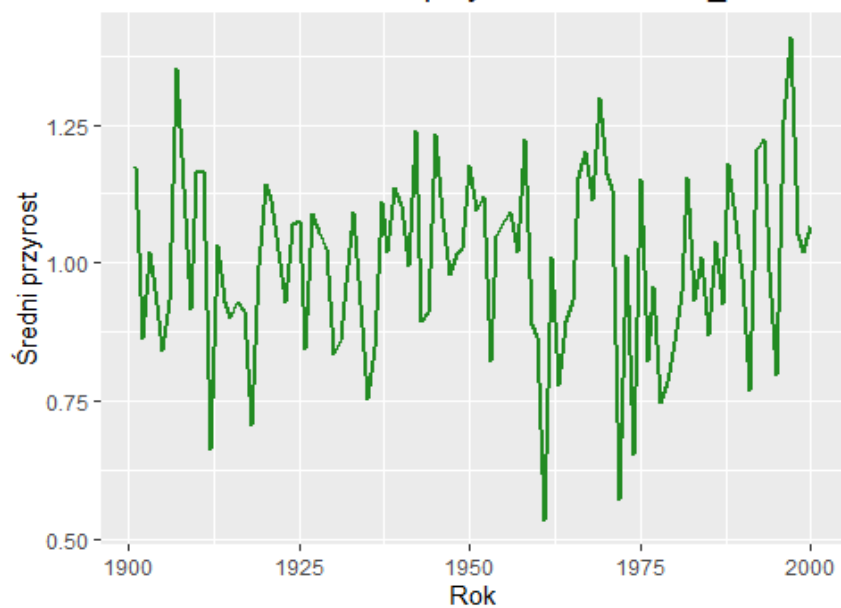
```
klaster1_1$srednia <- rowMeans(klaster1_1[,2:3])
klaster1_2$srednia <- rowMeans(klaster1_2[,2:4])
klaster1_3$srednia <- rowMeans(klaster1_3[,2:8])
```

2.4 Wykres liniowy zmienności szerokości przyrostów.

Postępujemy podobnie jak w punkcie 1.3

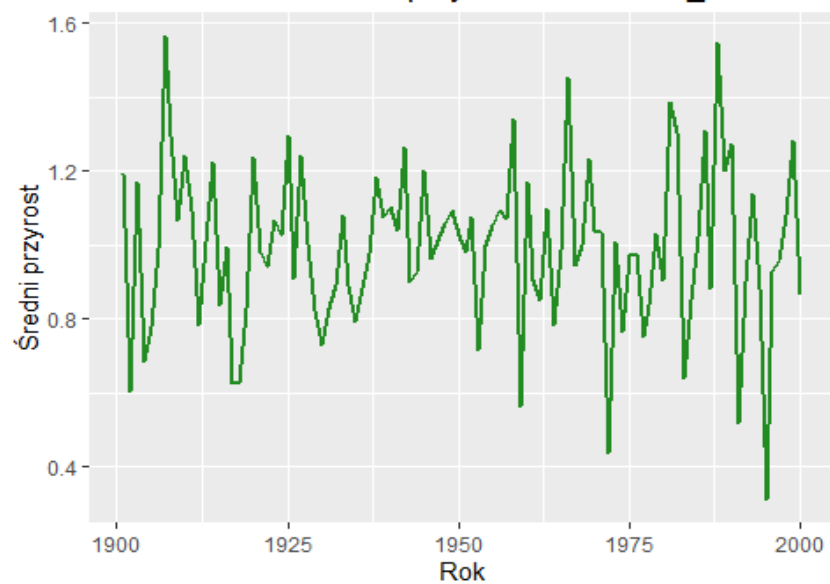
```
ggplot(klaster1_1, aes(rok, srednia)) +
  geom_line(color = "forestgreen", size = 1) +
  ylab("Średni przyrost") +
  xlab("Rok") +
  ggtitle("Zmienność szerokości przyrostów klaster 1_1")
```

Zmienność szerokości przyrostów klaster 1_1



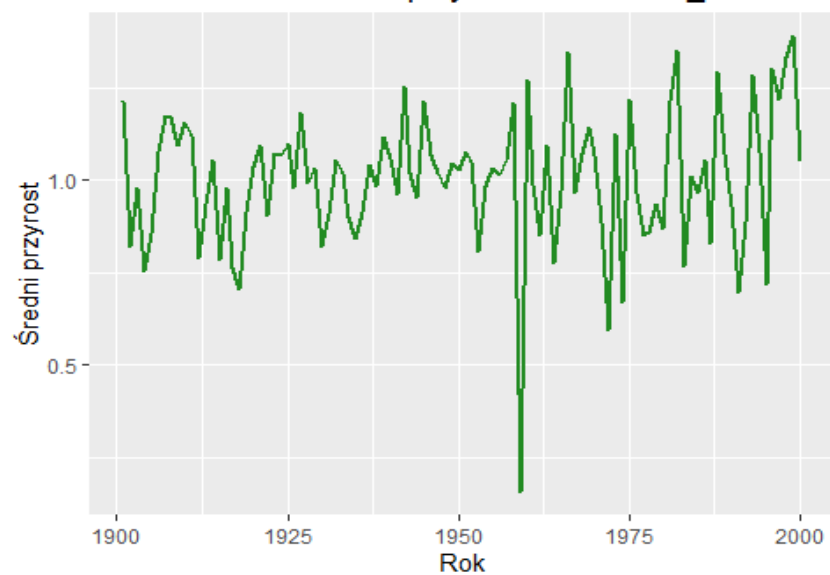
wykr. 2.4.1

Zmienność szerokości przyrostów klaster 1_2



wykr. 2.4.2

Zmienność szerokości przyrostów klaster 1_3



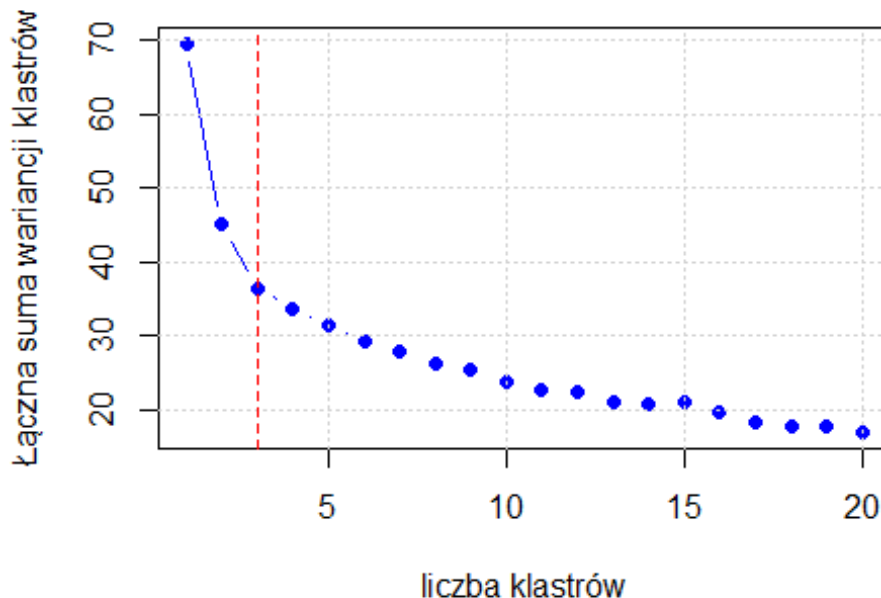
wykr. 2.4.3

2.5 Grupowanie serii przyrostów (metoda k-means)

Najpierw wyznaczamy optymalną liczbę klastrow metodą "łokcia" i za pomocą funkcji `abline()` zaznaczamy linią poszukiwaną wartość.

```
k <- 20 # maksymalna liczba klastrow
wektor_k <- sapply(1:k,function(k){kmeans(sn2_res_tree,k)$tot.withinss}) # wektor wart. k
plot(1:k, wektor_k, type="b", pch = 16, xlab="liczba klastrow", ylab="Łączna suma wariancji klastrow", col="blue")
grid()

abline(v = 3, lty = 2, col="red")
```



wykr. 2.5

Następnie za pomocą funkcji `kmeans()` przydzielamy wektory do odpowiednich grup.

```
sn2_res_tree_t_kmeans<-kmeans(sn2_res_tree_t, 3)
print(sn2_res_tree_t_kmeans)
```

Tworzymy ramki danych 3 wydzielonych klastrow (aby policzyć średnie).

```
klaster2_1 <- data.frame(rok, k_9, k_10, k_12, k_15)
klaster2_2 <- data.frame(rok, k_1, k_2, k_7)
klaster2_3 <- data.frame(rok, k_3, k_5, k_6, k_8, k_11, k_13)
```

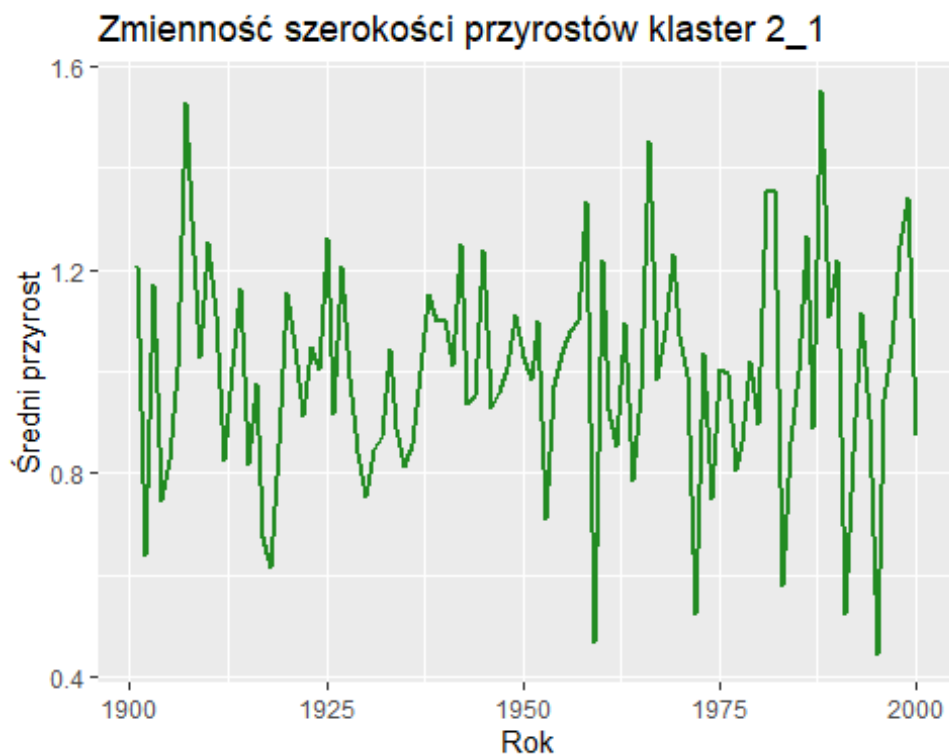
Liczmy średnie i dodajemy je w kolumnie.

```
klaster2_1$srednia <- rowMeans(klaster2_1[,2:5])  
klaster2_2$srednia <- rowMeans(klaster2_2[,2:4])  
klaster2_3$srednia <- rowMeans(klaster2_3[,2:7])
```

2.6 Utworzenie wykresów zmienności szerokości przyrostów

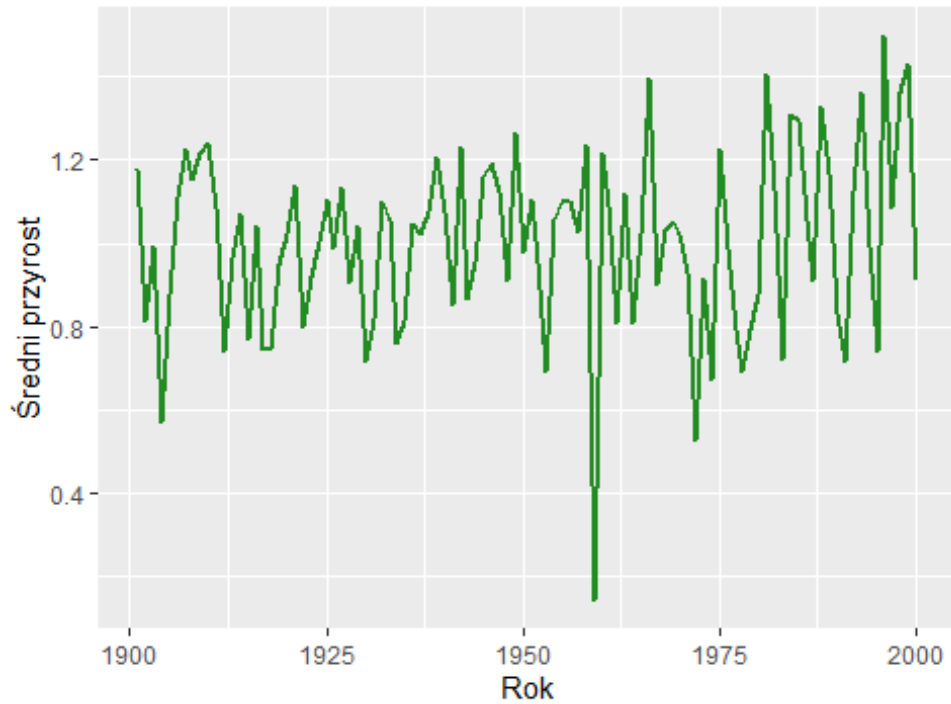
Postępujemy podobnie jak w punkcie 1.2.

```
ggplot(klaster2_1, aes(rok, srednia)) +  
  geom_line(color = "forestgreen", size = 1) +  
  ylab("Średni przyrost") +  
  xlab("Rok") +  
  ggtitle("Zmienność szerokości przyrostów klaster 2_1")
```



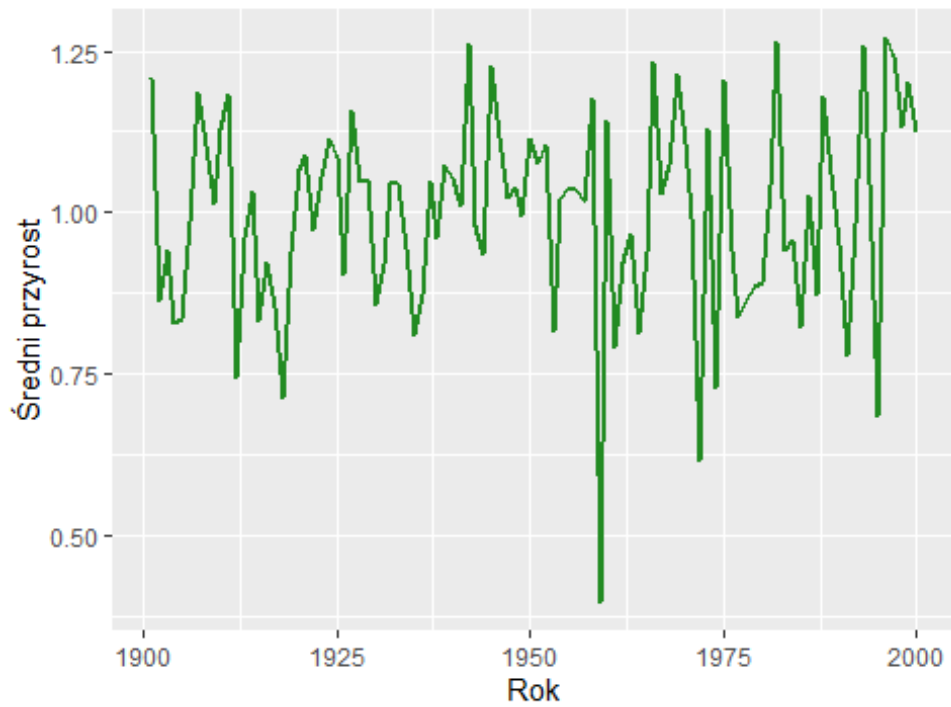
wykr. 2.6.1

Zmienność szerokości przyrostów klaster 2_2



wykr. 2.6.2

Zmienność szerokości przyrostów klaster 2_3



wykr. 2.6.3

3.1 Przygotowanie danych

Na początku utworzymy nowe *data.frame* zawierające dane o opadach i temperaturze dla następujących miesięcy: kwiecień, maj, czerwiec, lipiec, sierpień i wrzesień.

```
opady_cut <- select(sn2_p, 5:10)
temperatura_cut <- select(sn2_t, 5:10)
```

3.2 Analiza korelacji

Następnie wykonamy analizę korelacji (Pearsona) między danymi dendro a klimatycznymi dla chronologii z punktów 1, 2a oraz 2b polecenia. Wykorzystamy do tego funkcje *cor()*. Najpierw tworzymy zmienne zawierające poszczególne korelacje, następnie łączymy je w dwie ramki danych osobno dla danych o opadach i danych o temperaturze. Do połączenia wykorzystamy funkcje *rbind()*. Dla czytelności nadamy nazwy poszczególnym wierszom z wykorzystaniem *rownames()*.

```
miesiace <- c("kwiecień", "maj", "czerwiec", "lipiec", "sierpień", "wrzesień")
#Chronologia z punktu 1
ch_1_o <- cor(sn2_res_cut$srednia, opady_cut, method = "pearson")
ch_1_t <- cor(sn2_res_cut$srednia, temperatura_cut, method = "pearson")

#chronologie z punktu 2a
ch_1_1_o <- cor(klaster1_1$srednia, opady_cut, method = "pearson")
ch_1_1_t <- cor(klaster1_1$srednia, temperatura_cut, method = "pearson")
ch_1_2_o <- cor(klaster1_2$srednia, opady_cut, method = "pearson")
ch_1_2_t <- cor(klaster1_2$srednia, temperatura_cut, method = "pearson")
ch_1_3_o <- cor(klaster1_3$srednia, opady_cut, method = "pearson")
ch_1_3_t <- cor(klaster1_3$srednia, temperatura_cut, method = "pearson")

#chronologie z punktu 2b
ch_2_1_o <- cor(klaster2_1$srednia, opady_cut, method = "pearson")
ch_2_1_t <- cor(klaster2_1$srednia, temperatura_cut, method = "pearson")
ch_2_2_o <- cor(klaster2_2$srednia, opady_cut, method = "pearson")
ch_2_2_t <- cor(klaster2_2$srednia, temperatura_cut, method = "pearson")
ch_2_3_o <- cor(klaster2_3$srednia, opady_cut, method = "pearson")
ch_2_3_t <- cor(klaster2_3$srednia, temperatura_cut, method = "pearson")

#tabela dla opadów
korelacje_opady <- rbind(ch_1_o, ch_1_1_o, ch_1_2_o, ch_1_3_o, ch_2_1_o, ch_2_2_o, ch_2_3_o)
rownames(korelacje_opady) <- c("ch_1_o", "ch_1_1_o", "ch_1_2_o", "ch_1_3_o", "ch_2_1_o", "ch_2_2_o", "ch_2_3_o")

write.table(korelacje_opady, file="korelacje_opady.txt")
```

#tabela dla temperatury

```
korelacje_temperatury <- rbind(ch_1_t, ch_1_1_t, ch_1_2_t, ch_1_3_t, ch_2_1_t, ch_2_2_t, ch_2_3_t)
rownames(korelacje_temperatury) <- c("ch_1_t", "ch_1_1_t", "ch_1_2_t", "ch_1_3_t", "ch_2_1_t", "ch_2_2_t", "ch_2_3_t")
```

```
write.table(korelacje_temperatury, file="korelacje_temperatury.txt")
```

*na czerwono zaznaczono wartości współczynników korelacji dla których
p-value<0.05, co oznacza że te współczynniki wykazują istotność
(istotność liczona w punkcie 3.3)*

chronologia	Kwiecień	Maj	Czerwiec	Lipiec	Sierpień	Wrzesień
ch_1_o	-0,075	0,048	0,107	0,223	-0,101	0,041
ch_1_1_o	0,051	0,039	0,052	0,241	-0,077	0,008
ch_1_2_o	-0,024	0,018	0,070	0,244	-0,073	-0,029
ch_1_3_o	-0,103	0,053	0,115	0,184	-0,117	0,081
ch_2_1_o	-0,032	0,023	0,085	0,221	-0,068	0,023
ch_2_2_o	-0,145	0,070	0,172	0,228	-0,068	0,064
ch_2_3_o	-0,050	0,046	0,059	0,175	-0,135	0,032

Tabela współczynników korelacji – opady (rys. 3.2.1)

chronologia	Kwiecień	Maj	Czerwiec	Lipiec	Sierpień	Wrzesień
ch_1_t	-0,002	0,218	0,013	-0,112	0,078	0,152
ch_1_1_t	-0,054	0,232	0,025	0,017	0,109	0,069
ch_1_2_t	-0,071	0,181	-0,024	-0,149	-0,047	0,035
ch_1_3_t	0,038	0,200	0,030	-0,098	0,102	0,198
ch_2_1_t	-0,077	0,176	0,002	-0,141	-0,040	0,030
ch_2_2_t	0,084	0,175	-0,013	-0,116	0,134	0,205
ch_2_3_t	0,008	0,241	0,038	-0,059	0,131	0,195

Tabela współczynników korelacji – temperatura (rys. 3.2.2)

3.3 Test istotności korelacji

Po obliczeniu współczynnika korelacji Pearsona należy zbadać jego istotność. Wykorzystamy funkcję `cor.test()`. W tej funkcji domyślnie stosowana jest korelacja Pearsona. Całość wykonamy w pętlach. Pętla wypisuje wynik testu korelacji tylko wtedy, gdy korelacja jest istotna ($pvalue < 0.05$). w ten sposób będziemy w stanie w szybki sposób zidentyfikować istotne wartości.

#dla chronologii z punktu 1

```
for (i in 1:6) {  
  tmp1 <- cor.test(sn2_res_cut$srednia, opady_cut[,i])  
  if(tmp1[["p.value"]] < 0.05){  
    print(tmp1)  
  }  
  tmp2 <- cor.test(sn2_res_cut$srednia, temperatura_cut[,i])  
  if(tmp2[["p.value"]] < 0.05){  
    print(tmp2)  
  }  
}
```

#dla chronologii z punktu 2a

```
for (i in 1:6) {  
  tmp1 <- cor.test(klaster1_1$srednia, opady_cut[,i])  
  if(tmp1[["p.value"]] < 0.05){  
    print(tmp1)  
  }  
  tmp2 <- cor.test(klaster1_2$srednia, opady_cut[,i])  
  if(tmp2[["p.value"]] < 0.05){  
    print(tmp2)  
  }  
  tmp3 <- cor.test(klaster1_3$srednia, opady_cut[,i])  
  if(tmp3[["p.value"]] < 0.05){  
    print(tmp3)  
  }  
  tmp4 <- cor.test(klaster1_1$srednia, temperatura_cut[,i])  
  if(tmp4[["p.value"]] < 0.05){  
    print(tmp4)  
  }  
  tmp5 <- cor.test(klaster1_2$srednia, temperatura_cut[,i])  
  if(tmp5[["p.value"]] < 0.05){  
    print(tmp5)  
  }  
  tmp6 <- cor.test(klaster1_3$srednia, temperatura_cut[,i])  
  if(tmp6[["p.value"]] < 0.05){  
    print(tmp6)  
  }  
}
```

#dla chronologii z punktu 2b

```
for (i in 1:6) {  
  tmp1 <- cor.test(klaster2_1$srednia, opady_cut[,i])  
  if(tmp1[["p.value"]] < 0.05){  
    print(tmp1)  
  }  
  tmp2 <- cor.test(klaster2_2$srednia, opady_cut[,i])  
  if(tmp2[["p.value"]] < 0.05){  
    print(tmp2)  
  }  
  tmp3 <- cor.test(klaster2_3$srednia, opady_cut[,i])  
  if(tmp3[["p.value"]] < 0.05){  
    print(tmp3)  
  }  
  tmp4 <- cor.test(klaster2_1$srednia, temperatura_cut[,i])  
  if(tmp4[["p.value"]] < 0.05){  
    print(tmp4)  
  }  
  tmp5 <- cor.test(klaster2_2$srednia, temperatura_cut[,i])  
  if(tmp5[["p.value"]] < 0.05){  
    print(tmp5)  
  }  
  tmp6 <- cor.test(klaster2_3$srednia, temperatura_cut[,i])  
  if(tmp6[["p.value"]] < 0.05){  
    print(tmp6)  
  }  
}
```

3.4 Wykresy

Tworzymy korelogramy dla kolejnych chronologii i temperatur, a także chronologii i opadów. Z uwagi na to, że specyfika danych sprawiła, że musielibyśmy wykonać aż 84 wykresy, zdecydowaliśmy się na pominięcie wykresów dla których korelacja nie jest istotna ($p.value \geq 0.05$). Dla usprawnienia pracy zastosowaliśmy pętle *for* oraz instrukcję warunkową *if*. Szczegółowo omówimy polecenia zastosowane dla chronologii z pierwszej części zadania, ponieważ dla kolejnych chronologii schemat postępowania jest analogiczny.

Wspólnym wektorem wykorzystywanym we wszystkich pętlach jest wektor zawierający nazwy miesięcy. Wykorzystamy go przy nadawaniu nazw plikom, które będą zawierać poszczególne wykresy. Następnie, do wektora *chronologia1* przypisujemy kolumnę *srednia* z ramki danych *sn_2_res_cut* zawierającą chronologię obliczoną w zadaniu pierwszym.

Możemy teraz przystąpić do omówienia pętli. Pętla wykonuje się dla wartości zmiennej *i* od 1 do 6 (za każdym przejściem pętli zmienna jest inkrementowana). Do zmiennych *tmp_1* oraz *tmp_2* przypisujemy kolejne wektory z *data.frame*, które zawierają dane klimatyczne. Pozwoli to na stworzenie tymczasowych ramek danych *tmp_df1* oraz *tmp_df2*, które zawierają kolumnę z chronologią oraz odpowiednio jedną kolumnę z danymi o opadach i temperaturze.

W zmiennych *cor_o* i *cor_t* zapisujemy wynik testu korelacji. Komendy służące do tworzenia wykresu zostały zamknięte w instrukcji warunkowej *if*. Polecenia w niej zawarte są wykonywane tylko wtedy, gdy *p.value* jest mniejsze niż 0.05. Z uwagi na to, że wyniki testu korelacji zapisywane są w postaci listy, należało zastosować odpowiednie odwołanie do elementu listy zawierającego *p.value*.

Wykonywanie wykresu odbywa się w sposób standardowy. Wykorzystujemy pakiet *ggplot2* oraz funkcję *ggplot()*. w połączeniu z odpowiednimi komenadami. Tworzymy korelogram. Na osi x umieszczamy chronologię, a na osi y dane klimatyczne: sumę opadów atmosferycznych i temperaturę. *Geom_smooth* dodaje linię trendu. Komenda *print* służy do wyświetlenia narysowanego wykresu. Zapisywanie wykresu obsługuje funkcja *ggsave*. Połączenie z funkcją *paste0* pozwala na nadawanie odpowiednich nazw plików z wykorzystaniem wcześniej stworzonego wektora zawierającego nazwy miesięcy. Analogicznie postępujemy dla danych o temperaturze.

Schemat komend powtarza się dla chronologii w części 2a oraz 2b.

```

miesiace <- c("kwiecień", "maj", "czerwiec", "lipiec", "sierpień", "wrzesień")

#wektor do nadawania nazw plikom

#----- dla chronologii z punktu1-----#

chronologia1 <- sn2_res_cut$srednia #wydzielenie wektora zawierającego chronologię

for(i in 1:6)
{
  #tmp dla danych klimatycznych
  tmp_1 <- opady_cut[,i]
  tmp_2 <- temperatura_cut[,i]
  #tmp_df do tworzenia wykresów
  tmp_df1 <- data.frame(chronologia1, tmp_1)
  tmp_df2 <- data.frame(chronologia1, tmp_2)
  #zmienne dla testu korelacji
  cor_o <- cor.test(chronologia1, tmp_1)
  cor_t <- cor.test(chronologia1, tmp_2)

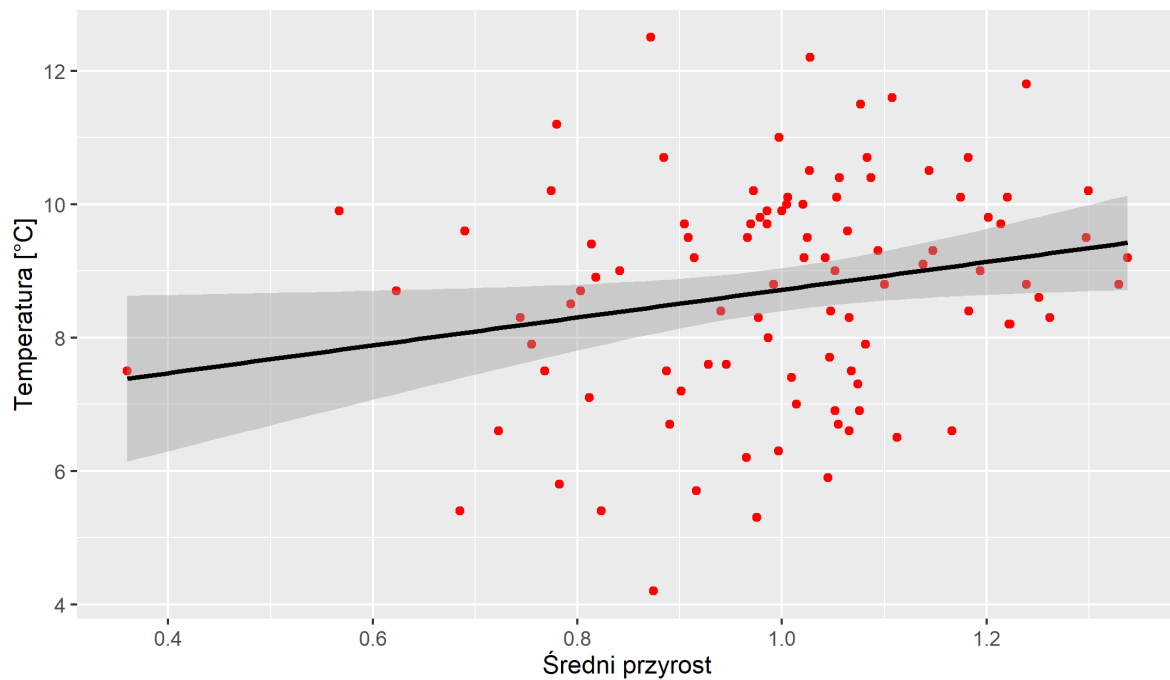
  #-->jeśli test wykaże istotność, to tylko wtedy warunek if pozwoli na narysowanie wykresu)<--#

  #opady dla poszczególnych miesięcy
  if(cor_o[["p.value"]] < 0.05){
    tmp_plot1 <- ggplot(tmp_df1, aes(chronologia1, tmp_1)) +
      geom_point(color = "blue") +
      geom_smooth(method = "lm", color = "black") +
      ylab("Suma opadów [mm]") +
      xlab("chronologia1")
    print(tmp_plot1)
    ggsave(tmp_plot1, filename = paste0("chrono1_opady_",miesiace[i],".png" ))
  }

  #temperatura dla poszczególnych miesięcy
  if(cor_t[["p.value"]] < 0.05){
    tmp_plot2 <- ggplot(tmp_df2, aes(chronologia1, tmp_2)) +
      geom_point(color = "red") +
      geom_smooth(method = "lm", color = "black") +
      ylab("Temperatura [°C]") +
      xlab("chronologia1")
    print(tmp_plot2)
    ggsave(tmp_plot2, filename = paste0("chrono1_temp_",miesiace[i],".png" ))
  }
}

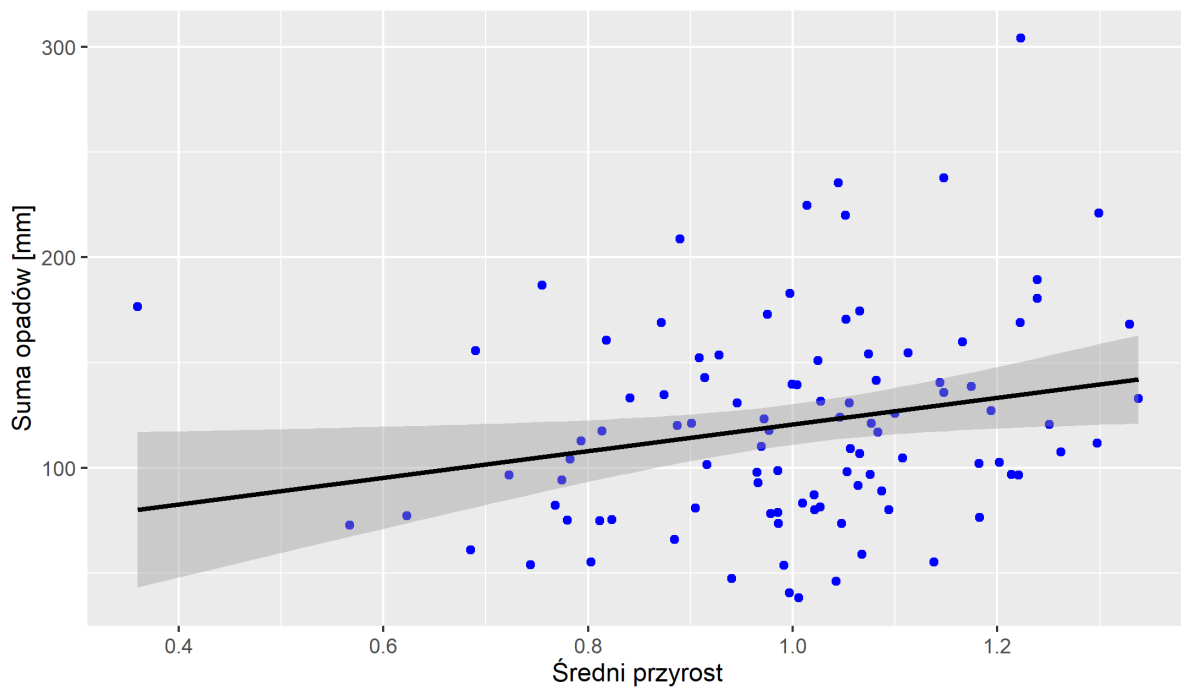
```

maj cała tabela



wykr. 3.4.1

lipiec cała tabela



wykr. 3.4.2


```

#----- dla chronologii z punktu2-----#

#zmienne dla obliczonych chronologii
kl1_1 <- klaster1_1$srednia
kl1_2 <- klaster1_2$srednia
kl1_3 <- klaster1_3$srednia

#----- dla klastra 1_1-----#

for(i in 1:6){

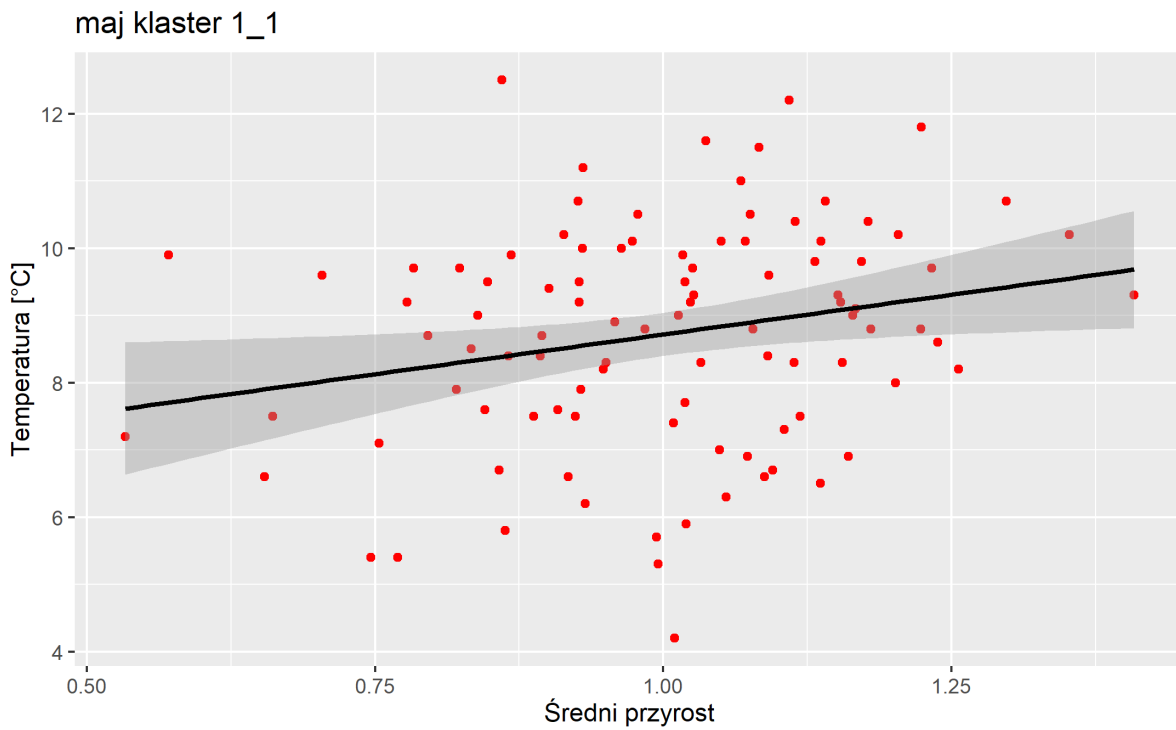
  tmp_1 <- opady_cut[,i]
  tmp_2 <- temperatura_cut[,i]
  tmp_df1 <- data.frame(kl1_1, tmp_1)
  tmp_df2 <- data.frame(kl1_1, tmp_2)

  cor_o <- cor.test(kl1_1, tmp_1)
  cor_t <- cor.test(kl1_1, tmp_2)

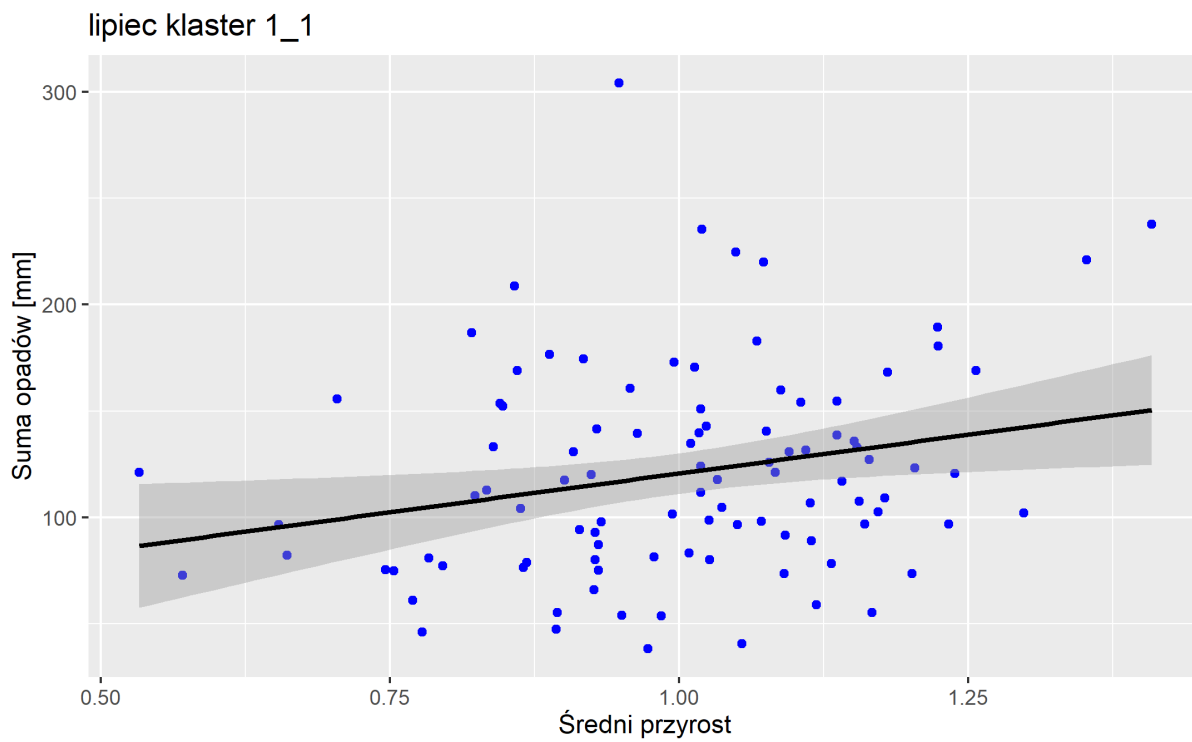
  if(cor_o[["p.value"]] < 0.05){
    tmp_plot1 <- ggplot(tmp_df1, aes(kl1_1, tmp_1)) +
      geom_point(color = "blue") +
      geom_smooth(method = "lm", color = "black") +
      ylab("Suma opadów [mm]") +
      xlab("klaster1_1")
    print(tmp_plot1)
    ggsave(tmp_plot1, filename = paste0("klaster1_1_opady_", miesiace[i], ".png" ))
  }

  if(cor_t[["p.value"]] < 0.05){
    tmp_plot2 <- ggplot(tmp_df2, aes(kl1_1, tmp_2)) +
      geom_point(color = "red") +
      geom_smooth(method = "lm", color = "black") +
      ylab("Temperatura [°C]") +
      xlab("klaster1_1")
    print(tmp_plot2)
    ggsave(tmp_plot2, filename = paste0("klaster1_1_temp_", miesiace[i], ".png"))
  }
}

```



wykr. 3.4.3



wykr. 3.4.4

```

#----- dla klastra 1_2-----#

for(i in 1:6){

  tmp_1 <- opady_cut[,i]
  tmp_2 <- temperatura_cut[,i]
  tmp_df1 <- data.frame(kl1_2, tmp_1)
  tmp_df2 <- data.frame(kl1_2, tmp_2)

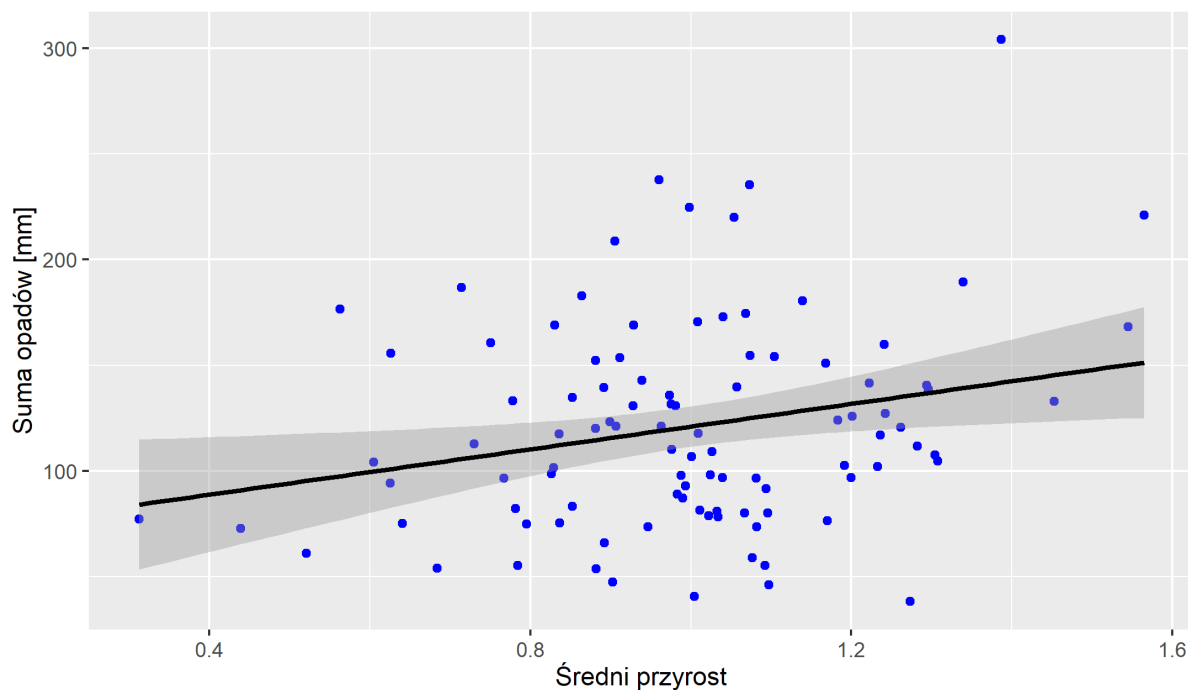
  cor_o <- cor.test(kl1_2, tmp_1)
  cor_t <- cor.test(kl1_2, tmp_2)

  if(cor_o[["p.value"]] < 0.05){
    tmp_plot1 <- ggplot(tmp_df1, aes(kl1_2, tmp_1)) +
      geom_point(color = "blue") +
      geom_smooth(method = "lm", color = "black") +
      ylab("Suma opadów [mm]") +
      xlab("klaster1_2")
    print(tmp_plot1)
    ggsave(tmp_plot1, filename = paste0("klaster1_2_opady_", miesiace[i], ".png" ))
  }

  if(cor_t[["p.value"]] < 0.05){
    tmp_plot2 <- ggplot(tmp_df2, aes(kl1_2, tmp_2)) +
      geom_point(color = "red") +
      geom_smooth(method = "lm", color = "black") +
      ylab("Temperatura [°C]") +
      xlab("klaster1_2")
    print(tmp_plot2)
    ggsave(tmp_plot2, filename = paste0("klaster1_2_temp_", miesiace[i], ".png" ))
  }
}

```

lipiec klaster 1_2



wykr. 3.4.5

```

#----- dla klastra 1_3-----#

for(i in 1:6){

  tmp_1 <- opady_cut[,i]
  tmp_2 <- temperatura_cut[,i]
  tmp_df1 <- data.frame(kl1_3, tmp_1)
  tmp_df2 <- data.frame(kl1_3, tmp_2)

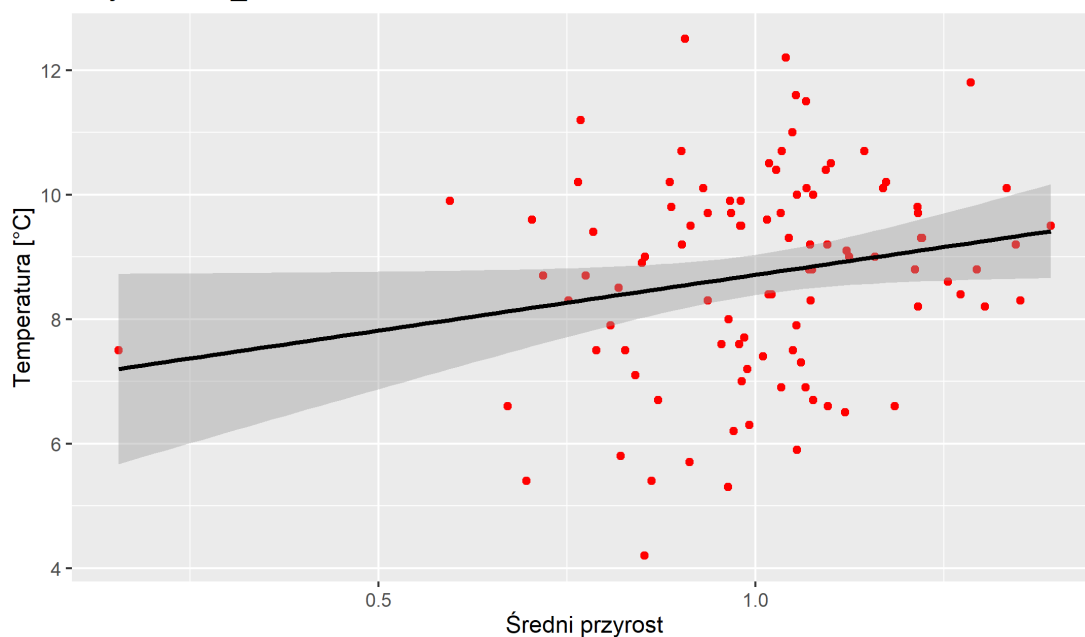
  cor_o <- cor.test(kl1_3, tmp_1)
  cor_t <- cor.test(kl1_3, tmp_2)

  if(cor_o[["p.value"]] < 0.05){
    tmp_plot1 <- ggplot(tmp_df1, aes(kl1_3, tmp_1)) +
      geom_point(color = "blue") +
      geom_smooth(method = "lm", color = "black") +
      ylab("Suma opadów [mm]") +
      xlab("klaster1_3")
    print(tmp_plot1)
    ggsave(tmp_plot1, filename = paste0("klaster1_3_opady_", miesiace[i], ".png" ))
  }

  if(cor_t[["p.value"]] < 0.05){
    tmp_plot2 <- ggplot(tmp_df2, aes(kl1_3, tmp_2)) +
      geom_point(color = "red") +
      geom_smooth(method = "lm", color = "black") +
      ylab("Temperatura [°C]") +
      xlab("klaster1_3")
    print(tmp_plot2)
    ggsave(tmp_plot2, filename = paste0("klaster1_3_temp_", miesiace[i], ".png" ))
  }
}

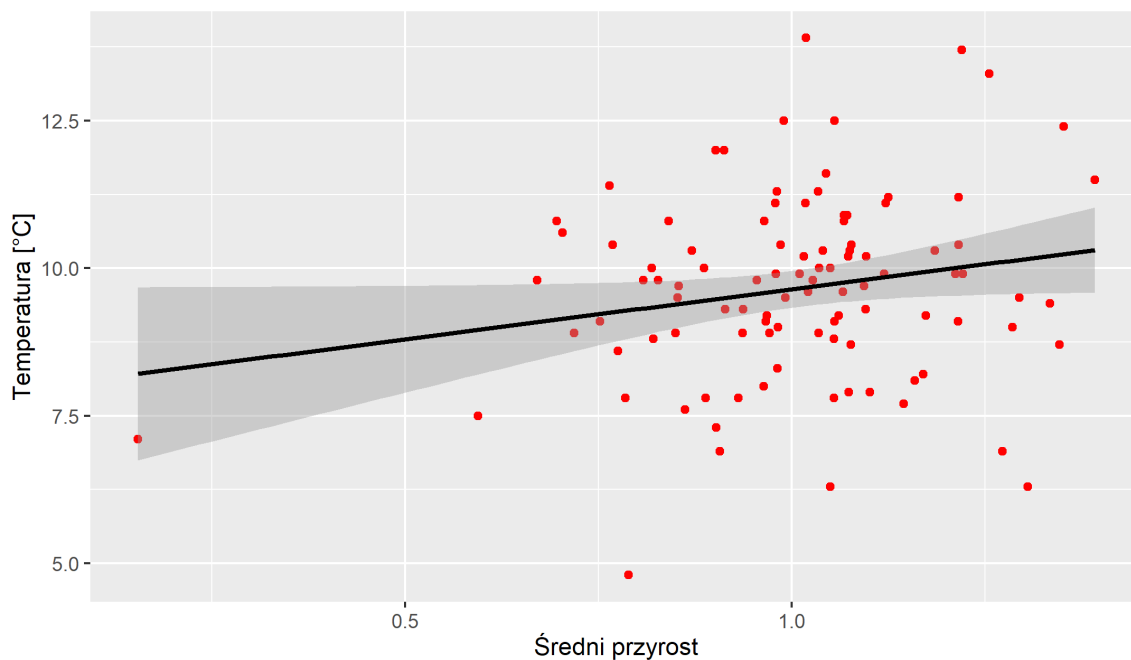
```

maj klaster 1_3



wykr. 3.4.6

wrzesień klaster 1_3



wykr. 3.4.7

```
#----- dla chronologii z punktu3-----#
```

```
#zmienne dla obliczonych chronologii
```

```
kl2_1 <- klaster2_1$srednia
```

```
kl2_2 <- klaster2_2$srednia
```

```
kl2_3 <- klaster2_3$srednia
```

```
#----- dla klastra 2_1-----#
```

```
for(i in 1:6){
```

```
  tmp_1 <- opady_cut[,i]
```

```
  tmp_2 <- temperatura_cut[,i]
```

```
  tmp_df1 <- data.frame(kl2_1, tmp_1)
```

```
  tmp_df2 <- data.frame(kl2_1, tmp_2)
```

```
  cor_o <- cor.test(kl2_1, tmp_1)
```

```
  cor_t <- cor.test(kl2_1, tmp_2)
```

```
  if(cor_o[["p.value"]] < 0.05){
```

```
    tmp_plot1 <- ggplot(tmp_df1, aes(kl2_1, tmp_1)) +
```

```
    geom_point(color = "blue") +
```

```
    geom_smooth(method = "lm", color = "black") +
```

```
    ylab("Suma opadów [mm]") +
```

```
    xlab("klaster2_1")
```

```
    print(tmp_plot1)
```

```
    ggsave(tmp_plot1, filename = paste0("klaster2_1_opady_", miesiace[i], ".png" ))
```

```
  }
```

```
  if(cor_t[["p.value"]] < 0.05){
```

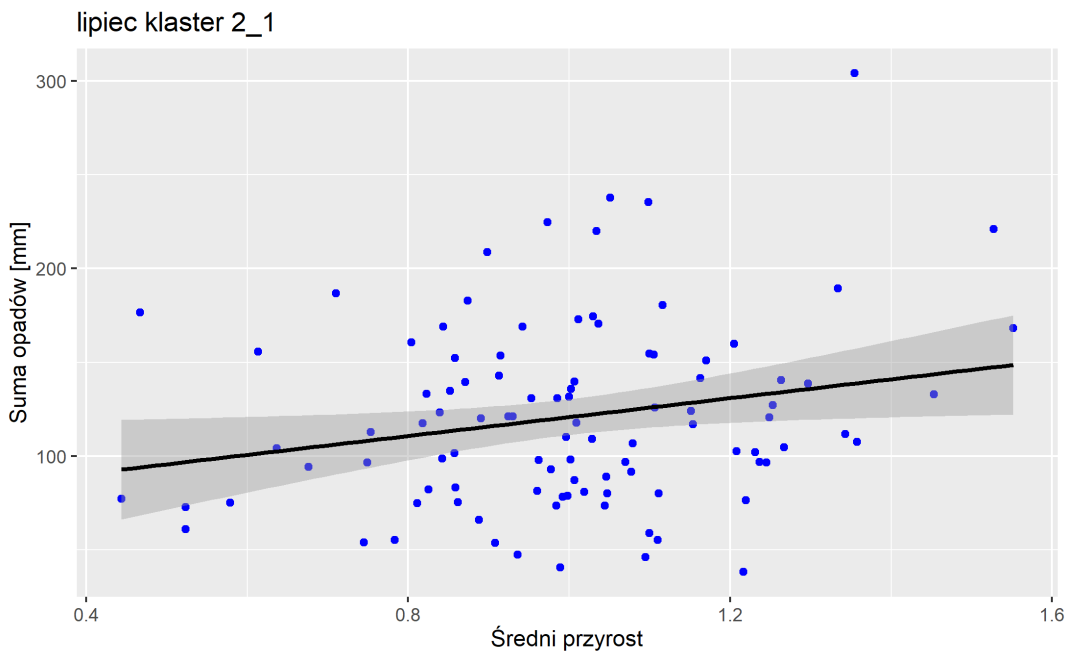
```
    tmp_plot2 <- ggplot(tmp_df2, aes(kl2_1, tmp_2)) +
```

```
    geom_point(color = "red") +
```

```

geom_smooth(method = "lm", color = "black") +
ylab("Temperatura [°C]") +
xlab("klaster2_1")
print(tmp_plot2)
ggsave(tmp_plot2, filename = paste0("klaster2_1_temp_", miesiace[i], ".png" ))
}
}

```



wykr. 3.4.8

#----- dla klastra 2_2-----#

```

for(i in 1:6){
  tmp_1 <- opady_cut[,i]
  tmp_2 <- temperatura_cut[,i]
  tmp_df1 <- data.frame(kl2_2, tmp_1)
  tmp_df2 <- data.frame(kl2_2, tmp_2)

  cor_o <- cor.test(kl2_2, tmp_1)
  cor_t <- cor.test(kl2_2, tmp_2)

  if(cor_o[["p.value"]] < 0.05){

    tmp_plot1 <- ggplot(tmp_df1, aes(kl2_2, tmp_1)) +
      geom_point(color = "blue") +
      geom_smooth(method = "lm", color = "black") +
      ylab("Suma opadów [mm]") +
      xlab("klaster2_2")
    print(tmp_plot1)
  }
}

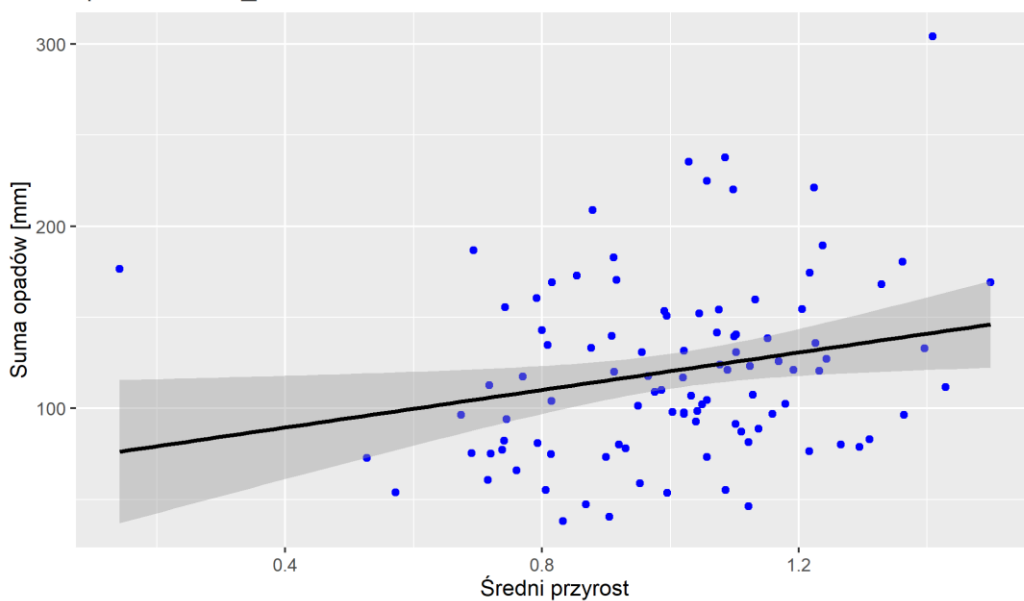
```

```

ggsave(tmp_plot1, filename = paste0("klaster2_2_opady_", miesiace[i], ".png" ))
}
if(cor_t[["p.value"]] < 0.05){
  tmp_plot2 <- ggplot(tmp_df2, aes(kl2_2, tmp_2)) +
  geom_point(color = "red") +
  geom_smooth(method = "lm", color = "black") +
  ylab("Temperatura [°C]") +
  xlab("klaster2_2")
  print(tmp_plot2)
  ggsave(tmp_plot2, filename = paste0("klaster2_2_temp_", miesiace[i], ".png" ))
}
}

```

lipiec klaster 2_2



wykr. 3.4.9

wrzesień klaster 2_2



wykr. 3.4.10



wykr. 3.4.11

#----- dla klastra 2_3-----#

```
for(i in 1:6){
  tmp_1 <- opady_cut[,i]
  tmp_2 <- temperatura_cut[,i]
  tmp_df1 <- data.frame(kl2_3, tmp_1)
  tmp_df2 <- data.frame(kl2_3, tmp_2)

  cor_o <- cor.test(kl2_3, tmp_1)
  cor_t <- cor.test(kl2_3, tmp_2)

  if(cor_o[["p.value"]] < 0.05){
    tmp_plot1 <- ggplot(tmp_df1, aes(kl2_3, tmp_1)) +
      geom_point(color = "blue") +
      geom_smooth(method = "lm", color = "black") +
      ylab("Suma opadów [mm]") +
      xlab("klaster2_3")
    print(tmp_plot1)
    ggsave(tmp_plot1, filename = paste0("klaster2_3_opady_", miesiace[i], ".png" ))
  }
  if(cor_t[["p.value"]] < 0.05){
    tmp_plot2 <- ggplot(tmp_df2, aes(kl2_3, tmp_2)) +
      geom_point(color = "red") +
      geom_smooth(method = "lm", color = "black") +
      ylab("Temperatura [°C]") +
      xlab("klaster2_1")
    print(tmp_plot2)
    ggsave(tmp_plot2, filename = paste0("klaster2_3_temp_", miesiace[i], ".png" ))
  }
}
```


4. Objaśnienie przynależności kolumn do grup/klastrów

Klastry:

1_1 <- X4sn2_6; X4sn2_13

1_2 <- X4sn2_10; X4sn2_12; X4sn2_15;

1_3 <- X4sn2_5; X4sn2_3; X4sn2_11; X4sn2_1; X4sn2_9; X4sn2_2; X4sn2_8;

2_1 <- X4sn2_9; X4sn2_10; X4sn2_12; X4sn2_15;

2_2 <- X4sn2_1; X4sn2_2; X4sn2_7;

2_3 <- X4sn2_3; X4sn2_5; X4sn2_6; X4sn2_8; X4sn2_11; X4sn2_13;

Grupy użyte w tabeli korelacji Pearsona:

ch_1_o/t <- kolumna wartości średnich przyrostów z głównej tabeli i kolumna wartości meteo (o-opady, t-temperatura)

ch_ "*numer klastra*" _o/t <- kolumna wartości średnich przyrostów z klastra i kol. wartości meteo (o-opady, t-temperatura)

4. Obserwacje i wnioski

Po pomyślnie przeprowadzonej analizie w programie R możemy przystąpić do wnioskowania i opisywania obserwacji. Praca z danymi dendrochronologicznymi okazała się niezwykle ciekawym i cennym doświadczeniem. Początkowo nie zdawaliśmy sobie sprawy z tego, że drzewa mogą wykazywać aż taką zmienność pomiędzy średnimi rocznymi przyrostami. Świetnie obrazuje to wykres 1.2. Można na nim wyraźnie zaobserwować wartości odstające – takie jak na przykład w okolicach roku 1960, kiedy to średni roczny przyrost dla drzew rosnących na obszarze, który odejmuje analiza był najmniejszy. W stosunku do roku poprzedzającego i następnego wartość ta jest dość zaskakująca. Same średnie przyrosty zmieniają się w przedziale od ok. 0.4 cm aż do wartości przekraczających 1.2 cm. Jak widać zmienność przyrostów jest dość spora, co może świadczyć o dużej wrażliwości drzew na czynniki zewnętrzne oraz na warunki klimatyczne i pogodowe w danym roku. Wszystkie wykonane przez nas wykresy potwierdzają dość dużą zmienność średnich przyrostów. Okazuje się, że drzewa mogą być swego rodzaju wehikułem czasu, który pozwala nam wnioskować o warunkach jakie panowały w przeszłości. Średnie roczne przyrosty drzew niosą ze sobą wiele cennych informacji, jednak analizowanie ich i wnioskowanie na ich temat wymaga ogromnej wiedzy i jeszcze większego doświadczenia. Aby je zdobyć wystarczy być uważnym, wnikliwym i cierpliwym obserwatorem.

Tabele 3.2.1 i 3.2.2 przedstawiają wartości współczynników korelacji grup z danymi meteorologicznymi. Najpierw zwróćmy uwagę na miesiące i grupy dla których korelacja jest istotna. Dla opadów jest to jedynie lipiec, dla średniej temperatury maj i wrzesień. Dla opadów największą wartość korelacji dodatniej wykazuje ch_1_2_o w lipcu (0,244), a największą wartość korelacji ujemnej grupa ch_1_3_o w kwietniu (-0,145). Dla temperatur największa korelacja dodatnia zachodzi dla ch_2_3_t w maju (0,241), a największa ujemna dla tej samej grupy we wrześniu (0,195).

Istotna dodatnia korelacja z opadami na poziomie 0,2 zachodzi dla 4 z 7 chronologii w lipcu. Warto zaznaczyć, że ujemne wartości zachodzą dla wszystkich grup w kwietniu i sierpniu.

Średnie wartości temperatur nie wykazują takiej jednoznaczności ponieważ 4 na 7 chronologii wykazuje istotną korelację (również na poziomie 0,2) dla maja, a 2 chronologie (w tym jedna się powtarza) dla września. Większość bo aż 6 na 7 (dla jednego miesiąca) współczynników ujemnych zachodzi dla lipca.

Gdy zwrócimy uwagę na wartość bezwzględną obliczonych współczynników korelacji (tych, które wykazują istotność – kolor czerwony w tabelach) okazuje się, że wszystkie z nich opisują korelację słabą. Wartości te oscylują wokół 0.2 i są bardzo podobne dla wszystkich grup. Patrząc na współczynniki korelacji, które nie wykazały istotności ($p\text{-value} > 0,5$) dochodzimy do wniosku, że w ogólności ich moduł jest mniejszy od wartości modułów korelacji istotnych.

Możemy zaobserwować również kolumny wyróżniające się na tle innych. Dla tabeli z opadami (rys. 3.2.1) wartości wsp. korelacji Pearsona dla lipca są wyraźnie większe niż w pozostałych miesiącach. Z kolei dla tabeli z temperaturami wyróżniającym się miesiącem jest maj.

Patrząc na korelogramy (które wykonaliśmy tylko dla wartości istotnych) obserwujemy rosnące linie trendu (znajduje to potwierdzenie w dodatnich wartościach współczynnika). Jednak ich nachylenie nie jest duże. Mimo to, na tej podstawie możemy stwierdzić, że (w ogólności) wzrostowi wartości temperatury i opadów towarzyszy wzrost wartości opisującej średni przyrost modrzewi.

Wśród wykresów zmienności szerokości przyrostów zarówno dla klastrów utworzonych za pomocą drzewa hierarchicznego (klastry 1_1-3) jak i dla klastrów, które powstały metodą k-means (klastry 2_1-3) widać pewną tendencję zmiany trendu przyrostów w zestawieniu z wykresem dla całej populacji (wykr. 2.4.1). Po pierwsze, warto zauważyć zachowanie krzywej w okolicy 1960 roku. Dla większości zestawienia wartość w tym roku jest wyjątkowo niska (proporcjonalnie) w porównaniu z otoczeniem. Jedynie dla klastra 1_2 (wykr. 2.4.2) w tym roku wartość jest z jednej strony niska wobec otoczenia, jednak dla kolejnych lat pojawiają się jeszcze mniejsze wartości (najmniejsza dla 1998). Można uznać, że do wysokości 1960 roku wszystkie wykresy zachowują się podobnie. Natomiast od 1960 pojawiają się bardzo zróżnicowane pomiary. Najbliżej zgodności z ogólnym wykresem jest klaster 1_3 (wykr.2.4.3) natomiast klaster 1_2 całkowicie odbiega od tendencji. Świadczy to o zróżnicowanej reakcji modrzewi rosnących na terenie Nadleśnictwa Śnieżka na zmiany klimatyczne zachodzące na przestrzeni lat.

5. Bibliografia

- prezentacje z przedmiotu PDŚ autorstwa Pani dr inż. Małgorzaty Danek
- <https://rstudio.com/resources/cheatsheets/>
- <https://bookdown.org/ndphillips/YaRrr/loops.html>