

Київський національний університет  
імені Тараса Шевченка

Звіт

до лабораторної роботи 5,  
дисципліни: “Технології аналізу та візуалізації даних”,  
студента: Євчика Олексія,  
групи Інформатика, Магістри

## Тема:

Радіальні діаграми (полігони), рельєфні діаграми та теплові карти

## Мета:

Опанувати побудову та інтерпретацію:

- радіальних діаграм (radar / spider / полігональні);
- рельєфних діаграм (поверхня, ізолінії, заповнені контури);
- теплових карт (heatmaps) для матриць/таблиць та просторових полів.

Навчитися коректно масштабувати дані, добирати кольорові палітри й уникати типових помилок візуалізації.

## Практичне завдання (Варіант 6):

```
packages c(iris, mmsb, and pheatmap others required but are not installed. install Don't Show Again)
# =====
# 1. Завантаження бібліотек
# =====
# install.packages(c("fmsb", "tidyverse", "ggplot2", "plotly", "reshape2", "pheatmap", "sca

library(fmsb)
library(tidyverse)
library(scales)
library(ggplot2)
library(reshape2)
library(plotly)
library(pheatmap)

..

13
14 # =====
15 # 2. Радіальна діаграма (fmsb)
16 # =====
17
18 df <- data.frame(
19   Accuracy = c(85, 78),
20   Recall = c(80, 82),
21   Precision = c(88, 75),
22   F1 = c(84, 78),
23   AUC = c(90, 81),
24   Kappa = c(72, 68)
25 )
26 rownames(df) <- c("Model_A", "Model_B")
27
28 df_scaled <- rbind(apply(df, 2, max), apply(df, 2, min), df)
29
30 radarchart(
31   df_scaled,
32   axistype = 1,
33   pcol = c("#1b9e77", "#d95f02"),
34   pfcol = scales::alpha(c("#1b9e77", "#d95f02"), 0.3),
35   plwd = 2,
36   title = "Радіальна діаграма: профілі метрик"
37 )
38
39 legend("topright", legend = rownames(df), col = c("#1b9e77", "#d95f02"), lwd = 2, bty = "n")
40
41 "
```

```

41 # =====
42 # 3. Радіальна діаграма 0..1 нормована
43 # =====
44
45 norm01 <- function(x) (x - min(x)) / (max(x) - min(x) + 1e-9)
46 df01 <- as.data.frame(lapply(df, norm01))
47 df01_scaled <- rbind(apply(df01, 2, max), apply(df01, 2, min), df01)
48
49 radarchart(df01_scaled, axistype = 1, title = "Радіальна (нормована 0..1)")
50
51 # =====
52 # 4. Рельєфні графіки (volcano)
53 # =====
54
55 z <- volcano
56 x <- 1:nrow(z)
57 y <- 1:ncol(z)
58
59 contour(x, y, z, main = "Ізолінії (volcano)", xlab = "X", ylab = "Y")
60
61 filled.contour(x, y, z, color.palette = terrain.colors,
62               plot.title = title(main = "Заповнені контури (volcano)"))
63
64 # 3D поверхня (base R)
65 persp(x, y, z, theta = 135, phi = 25, col = "lightblue", shade = 0.5,
66       ticktype = "detailed", xlab = "X", ylab = "Y", zlab = "Z",
67       main = "3D-поверхня (volcano)")
68
69 # plotly 3D поверхня
70 plot_ly(z = ~volcano) %>%
71   add_surface() %>%
72   layout(title = "Plotly Surface: volcano")
73
74 # =====
75 # 5. Рельєф у ggplot2 (filled contours)
76 # =====
77
78 df_volc <- melt(volcano)
79 colnames(df_volc) <- c("X", "Y", "Z")
80
81 ggplot(df_volc, aes(X, Y, z = Z)) +
82   geom_contour_filled() +
83   labs(title = "ggplot2: заповнені контури", fill = "Висота") +
84   coord_fixed()
85
86 ..

```

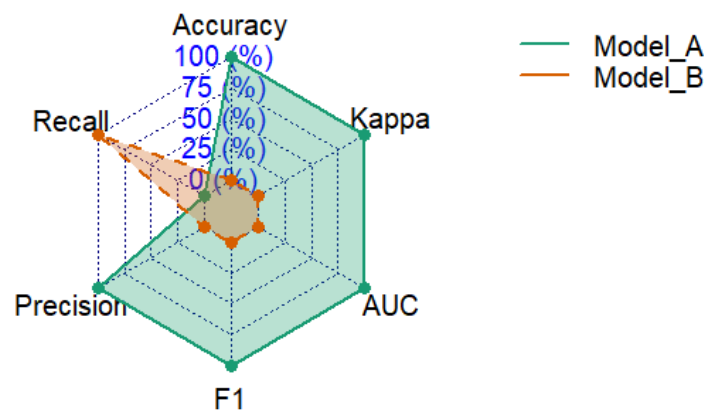
```

86 # =====
87 # 6. Heatmap кореляцій (ggplot2)
88 # =====
89
90 data(mtcars)
91 cor_m <- cor(mtcars)
92 mcor <- melt(cor_m)
93
94 ggplot(mcor, aes(Var1, Var2, fill = value)) +
95   geom_tile() +
96   scale_fill_gradient2(low = "#313695", mid = "white", high = "#a50026", midpoint = 0) +
97   theme_minimal(base_size = 12) +
98   theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
99   labs(title = "Теплова карта кореляцій mtcars", x = "", y = "", fill = "r")
100
101 # =====
102 # 7. Heatmap з кластеризацією (pheatmap)
103 # =====
104
105 pheatmap(cor_m,
106   color = colorRampPalette(c("#313695", "white", "#a50026"))(100),
107   display_numbers = TRUE, number_format = "%.2f",
108   main = "pheatmap: кореляції mtcars",
109   clustering_distance_rows = "euclidean",
110   clustering_distance_cols = "euclidean",
111   clustering_method = "complete")
112
113

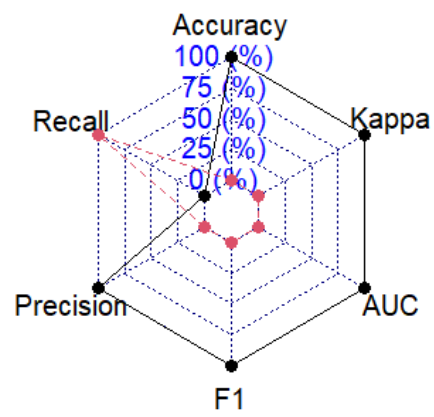
```

Результат виконання:

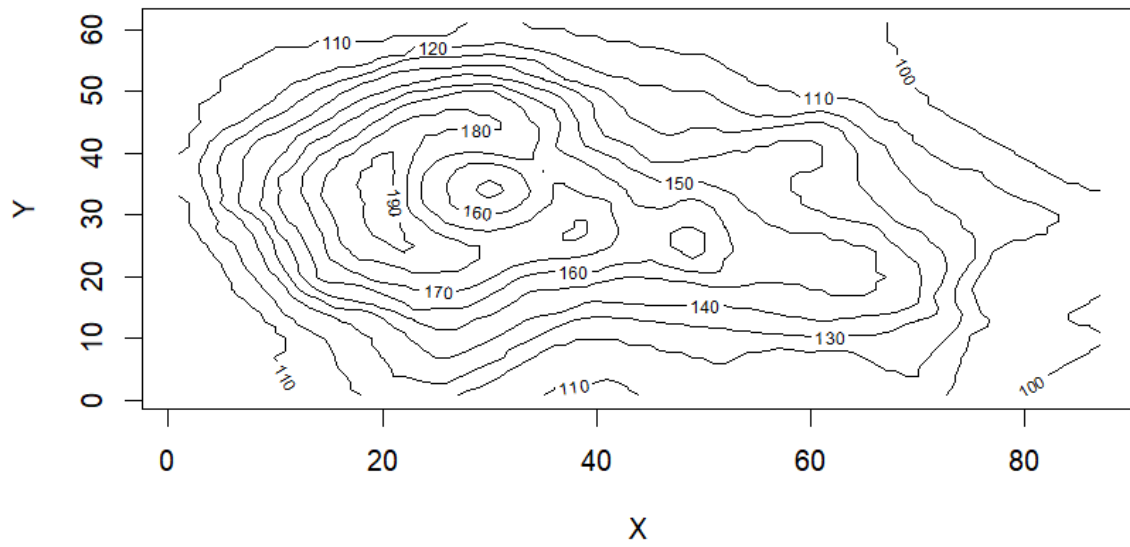
### Радіальна діаграма: профілі метрик



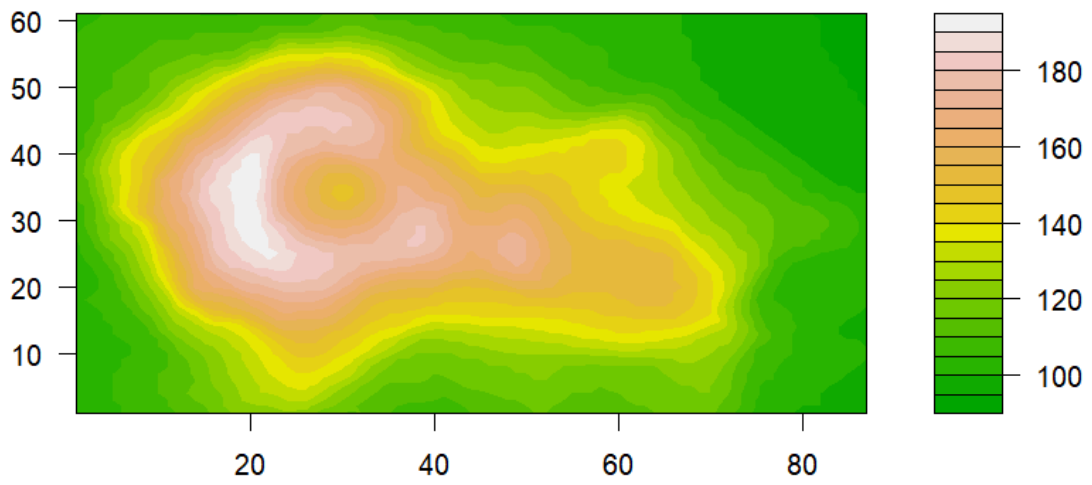
### Радіальна (нормована 0..1)



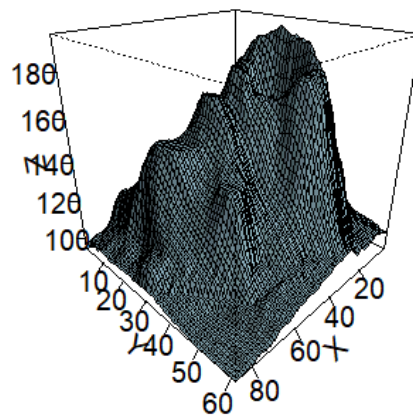
### Ізолінії (volcano)

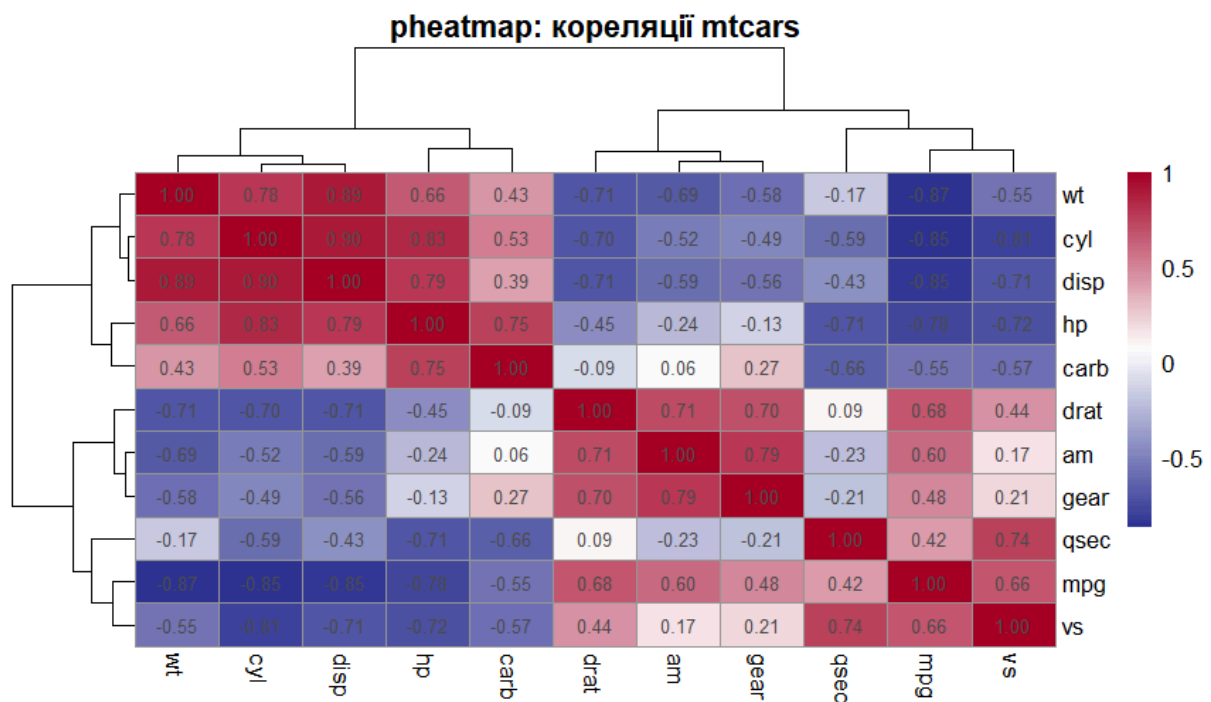
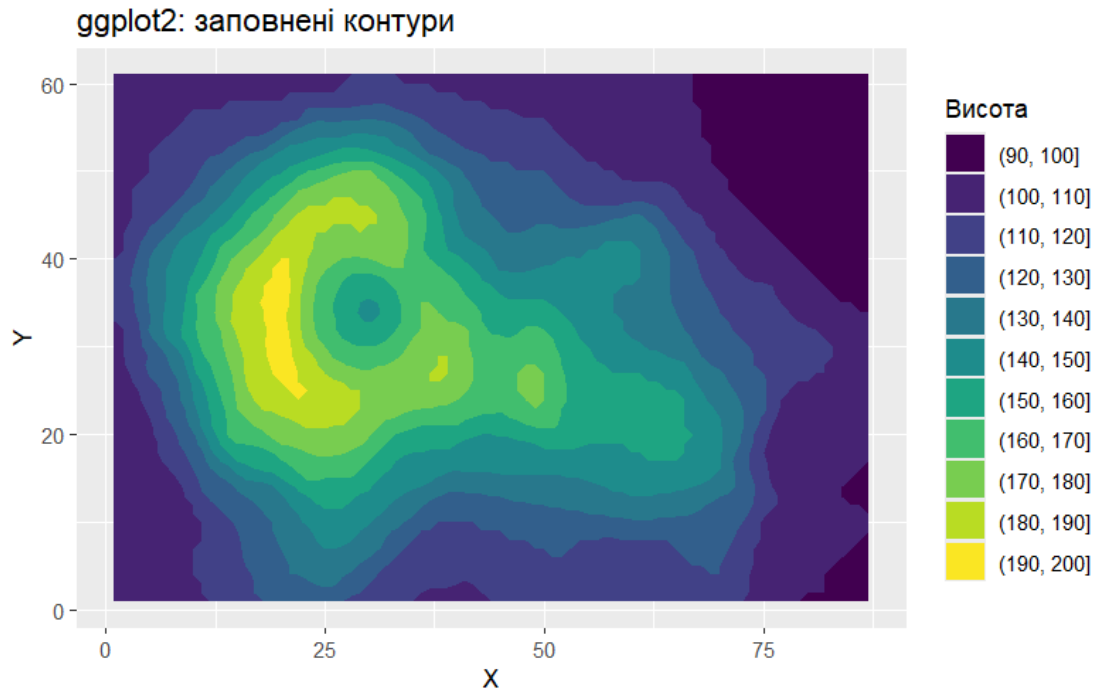


**Заповнені контури (volcano)**



**3D-поверхня (volcano)**





## Індивідуальне завдання (варіант 6)

### Filled contour: палітра

**Мета:** Вибрати послідовну палітру, дружню до дальтонізму.

**Кроки:** filled.contour + viridisLite::viridis; поясніть вибір.

**Очікувано:** читабельна карта з легендою.

Завантажуємо бібліотеку якщо не встановлена.

```

1 if (!requireNamespace("viridisLite", quietly = TRUE)) {
2   install.packages("viridisLite")
3 }
4
5 library(viridisLite)
6
7 z <- volcano
8 x <- seq_len(nrow(z))
9 y <- seq_len(ncol(z))
10

```

Записуємо дані.

Нижче код програми з коментарями.

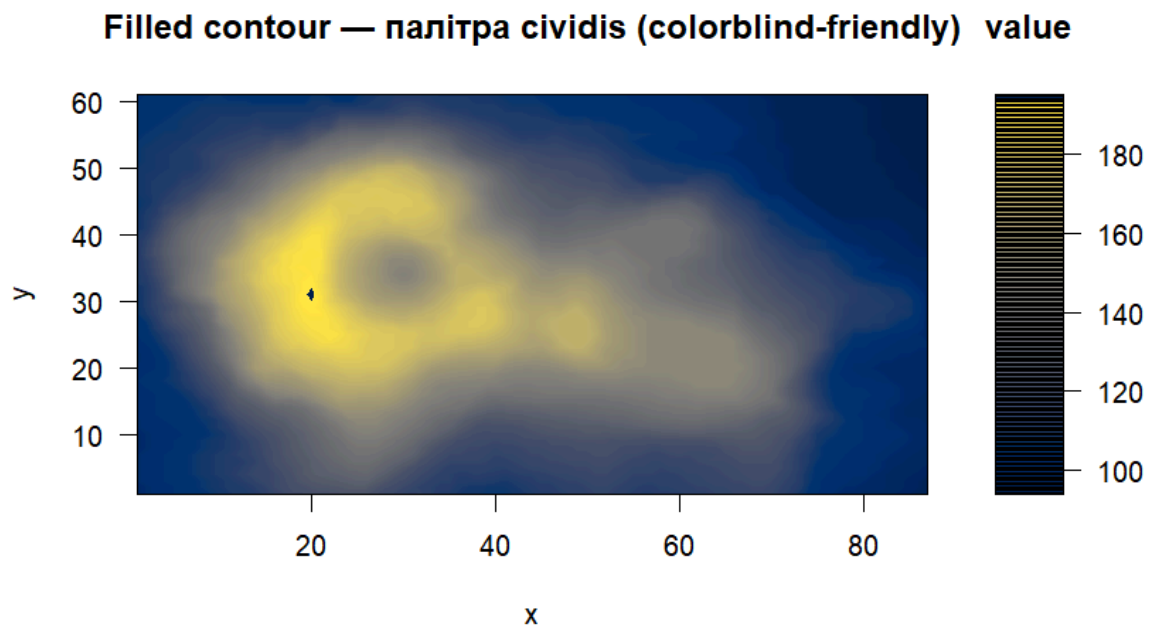
```

10
11 # Параметри палітри
12 ncols <- 200 # кількість відтінків у палітрі (чим більше – тим пл
13 palette_cols <- viridis(n = ncols, option = "cividis") # "cividis" – very colorblind-friendl
14 # Альтернативи: option = "viridis", "magma", "plasma", "inferno" (усі з viridis сім'ї; "civic
15
16 # Межі значень для легенди
17 zlim <- range(z, na.rm = TRUE)
18 at_ticks <- pretty(zlim, n = 6) # розумні відмітки на легенді
19
20 # Малюємо filled.contour з власною легендою (key.axes)
21 filled.contour(
22   x = x, y = y, z = z,
23   color.palette = function(n) palette_cols, # подаємо нашу палітру
24   nlevels = ncols,
25   zlim = zlim,
26   plot.title = title(main = "Filled contour – палітра cividis (colorblind-friendly)",
27                       xlab = "x", ylab = "y"),
28   plot.axes = {
29     axis(1); axis(2)
30     # додаткові осі/мітки можна додати тут
31   },
32   key.title = title(main = "value"), # заголовок колірної шкали
33   key.axes = {
34     # намалювати осі з бажаними відмітками у легенді
35     axis(4, at = at_ticks, labels = format(at_ticks, digits = 3), las = 1)
36   }
37 )
38

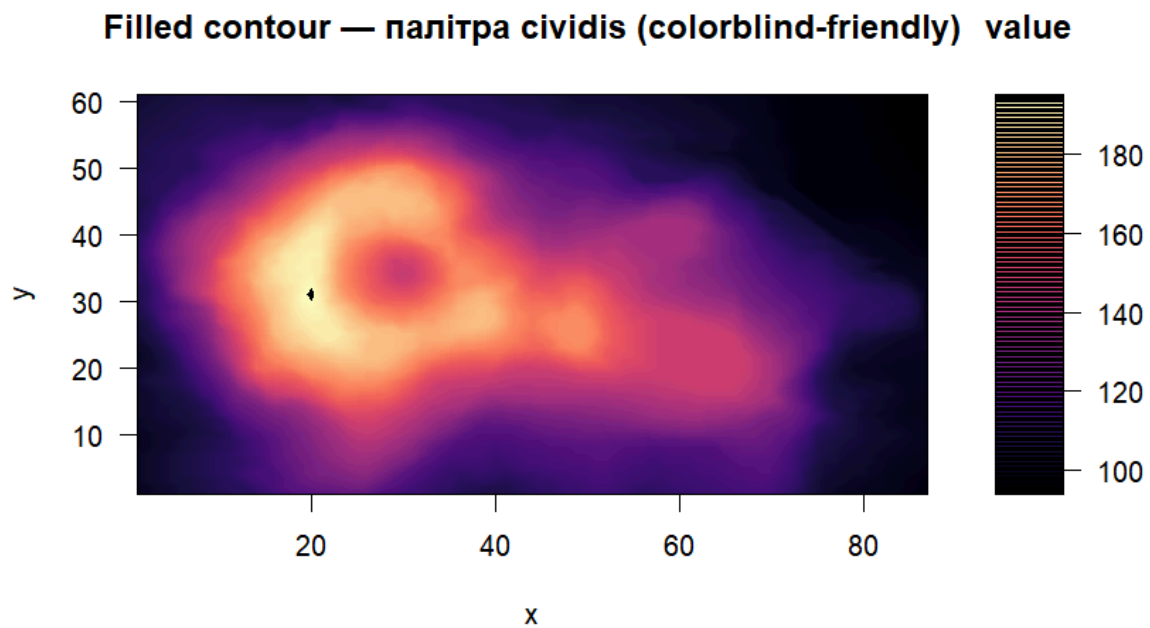
```

Результат:





Варіант з палітрою “магма”



**Чому я обрав viridis / cividis**

Перцепційно-однорідні: зміни кольору відповідають рівномірним змінам сигналу — це допомагає уникнути хибного сприйняття "псевдо-контурів".

Дружні до дальтонізму: сімейство *viridis* (особливо *cividis*) було розроблене з урахуванням проблем дальтонізму — кольори розрізняються за яскравістю і добре читаються.

Підходять для друку та чорно-білих відтінків: палітра зберігає порядок інтенсивностей і при монохромному друці.

Можна задавати багато відтінків (я використав `ncols = 200`) — отримуємо плавний градієнт без різких переходів.

### **Поради для читабельності карти**

Використовуйте достатньо рівнів (`nlevels`) і відтінків (`ncols`) щоб уникнути бандингу (ступінчастих переходів).

Додайте чіткі підписи осей і заголовки легенди (як у прикладі).

Якщо потрібно підкреслити конкретні інтервали — розгляньте `levels` або кастомні `breaks` (але зберігайте послідовність тонів).

Для публікацій/презентацій перевіряйте карту в сірих тонах (щоб впевнитися, що інформація не втратиться при друці).