

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»
ІНСТИТУТ КОМП'ЮТЕРНИХ НАУК ТА ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ**

Кафедра інформаційних систем і мереж

**ВІЗУАЛІЗАЦІЯ ДАНИХ
Візуалізація категоріальних+числових даних засобами RStudio
методичні вказівки до лабораторної роботи №4**

Затверджено на засіданні

кафедри ІСМ

Протокол № ____ від _____ 2022 р.

Львів-2022

Візуалізація категоріальних+числових даних засобами R: Методичні вказівки до виконання лабораторної роботи №4 / Укладач: В.А. Андруник – Львів: Видавництво Національного університету ”Львівська політехніка”, 2022. – 19 с.

Укладач

Андруник В.А., канд.техн. наук, доцент

Відповідальний за випуск

Литвин В.В., доктор техн. наук., професор

Рецензенти

Висоцька В.А., канд. техн. наук, доцент каф. ІСМ

Шестакевич Т.В., канд. техн. наук., доцент каф. ІСМ

Мета роботи: набуття практичних навичок опрацювання та візуалізації категоріальних та числових даних в середовищі RStudio.

Короткі теоретичні відомості

Категорійні – це такі дані, що мають обмежену або скінчену область визначення. Еквівалентне означення категоріальний даних – це набір даних, що можливо розбити на групи (класи, категорії).

Відповідно до NOIR (див. ЛР 2), до категоріальних належать дані типу nominal та ordinary. Різниця між типами полягає у можливості виділення порядку всередині групи.

Приклади категоріальних даних: стать, вікова група, національність, рівень освіти, колір, група крові, жанри, місяці, сезони, періоди, наявність/відсутність будь – чого тощо.

Основні ознаки та властивості категоріальних даних:

- Розділяють дані на групи, підгрупи;
- Вільні від типу даних, тобто категорії можуть бути задані числами, логічними значеннями, стрічками, текстом тощо.
- Можливо побудувати однозначне відображення категорій даних у скінчену множину чисел, векторів, проте ці об'єкти не мають математичного сенсу;
- Є якісними, а не кількісними даними;
- Піддаються візуалізації з допомогою спеціальних типів графіків.

Типи категоріальних даних:




- За кількістю класів: бінарні та мультиномінальні.
- За наявністю порядку: номінальні та впорядковані.

Подання категоріальних даних іноді вимагає побудови числових еквівалентів для кожної категорії. Як було згадано, категоріальні дані піддаються однозначному відображенню у множину чисел або векторів.

Подання категорії одним числом називається label encoding (кодування мітками). Зазвичай обирається послідовність чисел від 0 до $n-1$, або від 1 до n , де n – кількість класів.

Кодування векторами буває кількох типів. Найпростіший – one hot encoding (унітарне кодування), при якому кожному класу у відповідність ставиться розріджений одиничний вектор (вектор, де всі елементи крім одного рівні 0). Інший спосіб кодування векторами – multi hot encoding, при якому вектор може містити кілька ненульових елементів. Якщо ж координати вектору можуть мати певний сенс (наприклад класи, близькі за якісною ознакою, будуть ближчими у певному базисі) то таке кодування називається embedding.

Кодування мітками

	Encoding
	0
	1
	2

Унітарне кодування




	Cat	Dog	Zebra
	1	0	0
	0	1	0
	0	0	1

Рис. 1. Різниця між кодування мітками та унітарним

Візуалізація категоріальних даних можлива з допомогою спеціальних типів графіків, що відображають пропорційність груп, тобто категорій.

Якщо набори даних складаються виключно з категоріальних змінних, то, зазвичай, такі набори зберігають лише кількість (або частоту), з якою певна комбінація категорій зустрічається. Якщо ж датасет містить числові дані, то з кожним записом асоціюється певна категорія. В такому випадку підрахунок кількості записів певної групи відбувається безпосередньо перед візуалізацією.

Приклади візуалізації категоріальних даних

Приклади графіків виконанні на датасеті BrokenMarriage, що містить дані про розірвані шлюби різних типів. Підготовка набору даних та завантаження необхідних пакетів:

```
library(ggplot2)
library(dplyr)
library(vcd)
df <- BrokenMarriage
```

BARPLOT (стовпчикова діаграма)

Представляє значення сутностей, що використовують бруски різної довжини.

Барплот показує зв'язок між категоріальною та числовою змінною, що зазвичай є кількістю екземплярів певної категорії. Кожна сутність категоріальної змінної представлена у вигляді бруска-рядка. Розмір бруска представляє її числове значення. Відомий приклад стовпчикової діаграми є статево-вікова піраміда.

У ggplot2 для створення стовпчикової діаграми використовується геометрія geom_bar. Якщо ggplot2 має власноруч підрахувати кількість екземплярів, то необхідно задати параметр stat = "count", якщо ж довжина бруска залежить від заданої змінної, то stat необхідно присвоїти значення "identity".

```
df %>%
  ggplot(aes(x = rank, y = Freq)) +
    geom_bar(aes(fill=gender), size=1.5, stat="identity", color="black") +
    scale_fill_brewer(palette = 'Paired') +
    theme_bw()
```

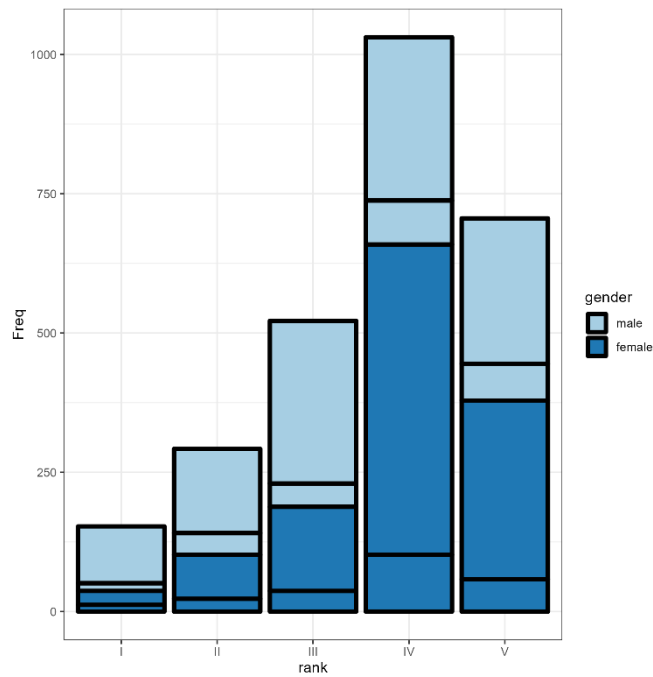


Рис. 2. Varplot

Існує достатня кількість параметрів, що дозволяють регулювати вигляд і подання діаграми.

Наприклад, параметр `position` задає тип накопичення стовпців. Перейти від стовпців з накладанням, до послідовних можливо присвоївши `position = "dodge"`, а якщо встановити параметру значення `"fill"`, то отримана діаграма визначає пропорції підгруп всередині кожного стовпця:

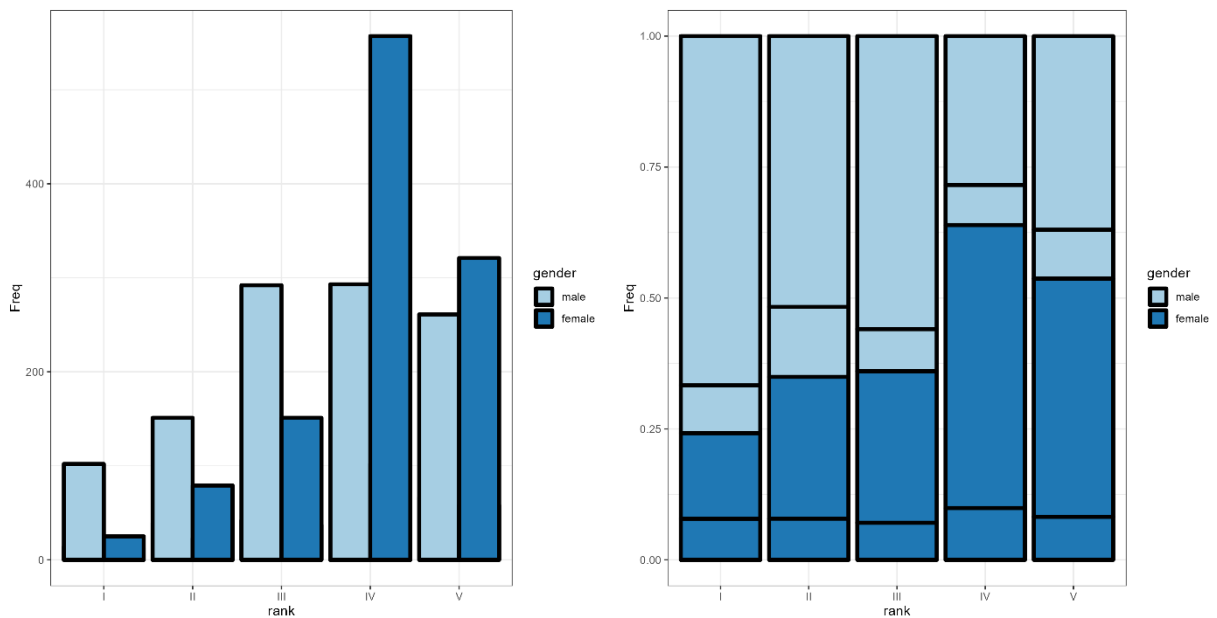


Рис. 3. Varplot з `position` рівним `"dodge"` та `"fill"`

PIE CHART & DONUT CHART (кругова та кільцева діаграми)

Обидва типи діаграм поділені на фрагменти для ілюстрації числової пропорції. У випадку pie chart поділяється круг, а donut chart – кільце.

Ggplot2 не містить готових реалізацій цих діаграм, проте їх легко отримати перетворивши стовпчикову діаграму у полярні координати з допомогою coord_polar.

Щоб отримати pie chart треба задати geom_bar без естетики x, і в полярних координатах перетворити у на кут повороту.

```
df %>%
  group_by(rank) %>%
  summarise(Freq = sum(Freq)) %>%
  ggplot(aes(x = "", y = Freq, fill=rank)) +
  geom_bar(stat="identity", color="white", lwd=1) +
  geom_label(aes(label = Freq), position= position_stack(vjust = 0.5)) +
  scale_fill_brewer(palette = 'Blues') +
  coord_polar(theta = "y") +
  theme_void()
```

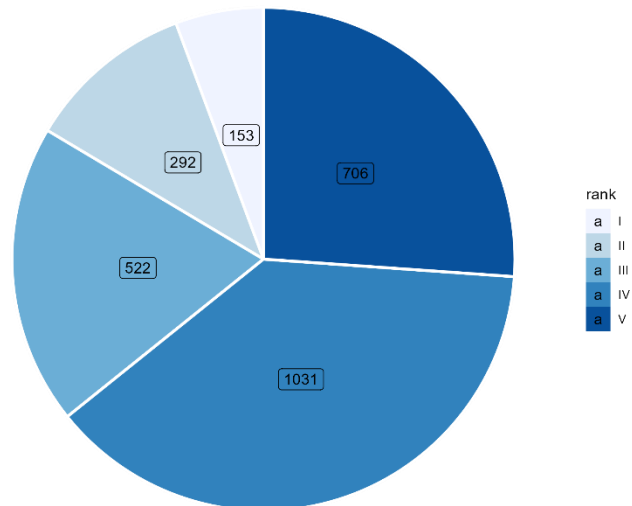


Рис. 4. Pie chart

Щоб отримати donut chart необхідно додати додаткову змінну у фрейм даних – ширину кільця, тоді, аналогічно до pie chart, створити графік (geom_col) і перевести у полярну систему координат:

```
df %>%
  group_by(rank) %>%
  summarise(Freq = sum(Freq)) %>%
  mutate(size=4.5) %>%
  ggplot(aes(x = size, y = Freq, fill=rank)) +
  geom_col(color="white", lwd=1) +
  geom_label(aes(label = Freq), position= position_stack(vjust = 0.5)) +
  scale_fill_brewer(palette = 'Blues') +
  coord_polar(theta = "y") + xlim(c(3, 5)) +
  theme_void()
```

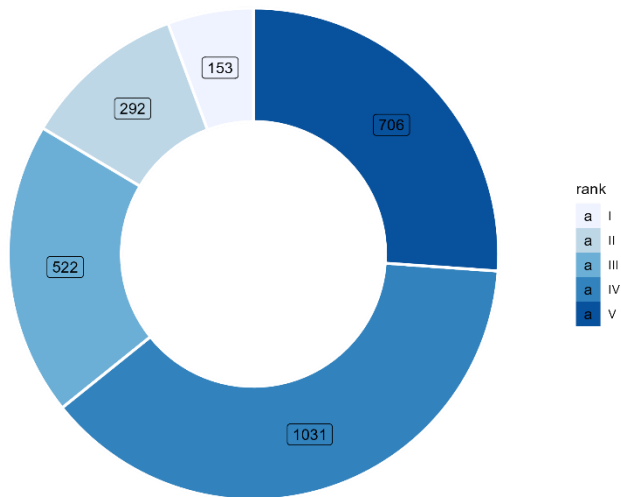


Рис. 5. Donut chart

CIRCULAR BARPLOT (Кругові стовпчикові діаграми)

Якщо виникає необхідність порівняти величину певного параметру по різним класам чи категоріям, то існує ще один варіант кругової діаграми – колова карта. В такій карті довжина кола відповідає за величину змінної для певної групи.

Цей тип не рекомендується, якщо потрібно точно порівняти значення групи. Проте, при правильним налаштуванням об'єкту aes ця діаграма дуже добре показує, як групи організовані в підгрупи.

```
df %>%
  group_by(rank) %>% summarise(Freq = sum(Freq)) %>%
  arrange(Freq) %>%
  mutate(order = seq_len(length(rank))) %>%
  ggplot(aes(x=reorder(rank, order),y=Freq,fill=reorder(rank, order))) +
  geom_bar(stat="identity", color="darkblue", lwd=0.5) +
  ylim(c(0, 1300)) + scale_fill_brewer(palette = 'Blues') +
  coord_polar(theta = "y", direction=1 ) + theme_bw()
```

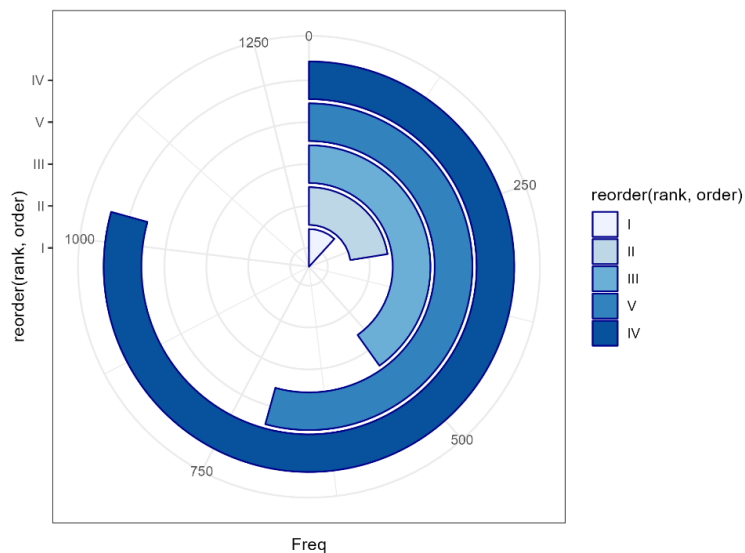


Рис. 6. Варіант кругової стовпчикової діаграми

Найпростіший варіант перетворення barplot у полярну систему координат породжує інший вид кругової стовпчикової діаграми:

```
df %>%
  ggplot(aes(x = rank, y = Freq)) +
    geom_bar(aes(fill=gender), stat="identity", color="black", lwd=0.5) +
    scale_fill_brewer(palette = 'Paired') +
    coord_polar(theta = "x", direction=1 ) +
    theme_bw()
```

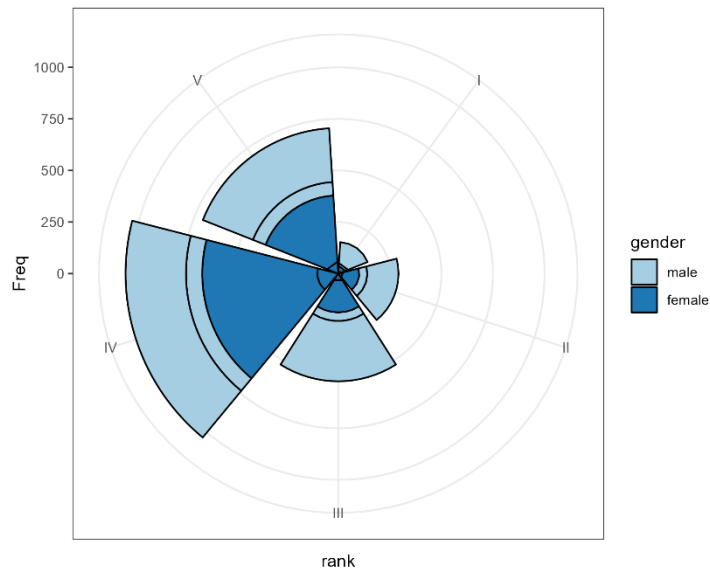


Рис. 7. Інший варіант кругової стовпчикової діаграми

LOLLIPOP (візуалізація “льодяниками”)

В загальному випадку, lollipop plot це просто barplot, де стовпець замінено на лінію та крапку. Проте, за оскільки такий спосіб візуалізації має легшу форму, він більш прийнятний для великої кількості даних.

Щоб отримати lollipop з допомогою ggplot2 необхідно окремо створити точки та сегменти ліній. Прийнятою практикою є сортування порядку класів за величиною, що відповідає за довжину лінії. Щоб отримати горизонтальну версію графіку необхідно додати шар coord_flip().

```
df %>%
  mutate(rank = paste(rank, gender)) %>%
  group_by(rank) %>%
  summarise(Freq=sum(Freq)) %>%
  arrange(Freq) %>%
  mutate(order = seq_len(length(rank))) %>%
  ggplot(aes(x = reorder(rank, order), y = Freq)) +
    geom_segment(aes(xend=rank, y=0, yend=Freq),color="#098080",lwd=0.7) +
    geom_point(color="#096060", size=5, shape=19) +
    geom_label(aes(x = rank, y = Freq, label=Freq), vjust = -0.4) +
    theme_bw() +
    # coord_flip() # - для горизонтальної версії
```

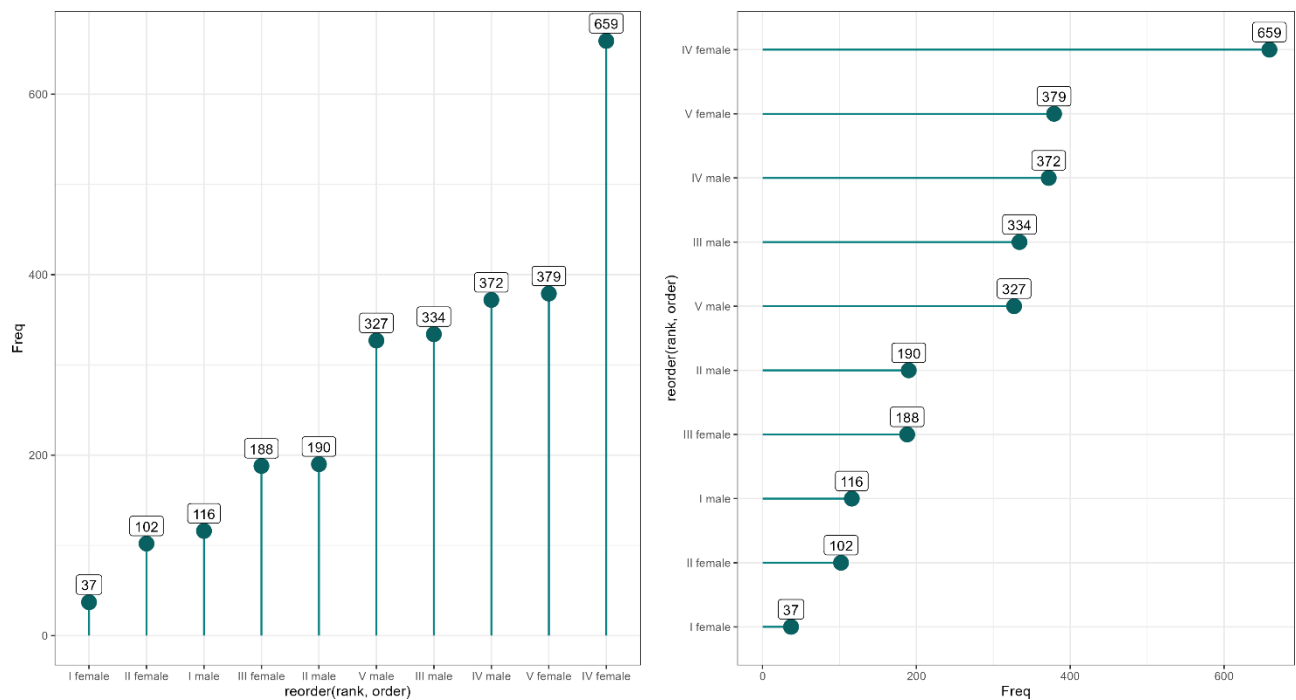



Рис. 8. Вертикальний та горизонтальний варіанти lollipop ді
Початкова координата для лінії не обов'язково має бути нулем, за потреби її можна зміщувати, наприклад:

```
df %>%
  mutate(rank = paste(rank, gender)) %>%
  group_by(rank) %>%
  summarise(Freq=sum(Freq)) %>%
  arrange(Freq) %>%
  mutate(order=seq_len(length(rank)),type=ifelse(Freq>mean(Freq), 0, 1)
%>% factor()) %>%
  ggplot(aes(x = reorder(rank, order), y = Freq)) +
  geom_segment(aes(xend=reorder(rank, order), y=mean(Freq), yend=Freq),
color = "#098080", lwd=0.7) +
  geom_hline(aes(yintercept = mean(Freq))) +
  geom_point(aes(color=type), size=5, shape=19) +
  scale_color_manual(values=c("#096060", "#cc1f12")) +
  geom_label(aes(x = rank, y = Freq, label=Freq), vjust = -0.6) +
  theme_bw() + coord_flip()
```

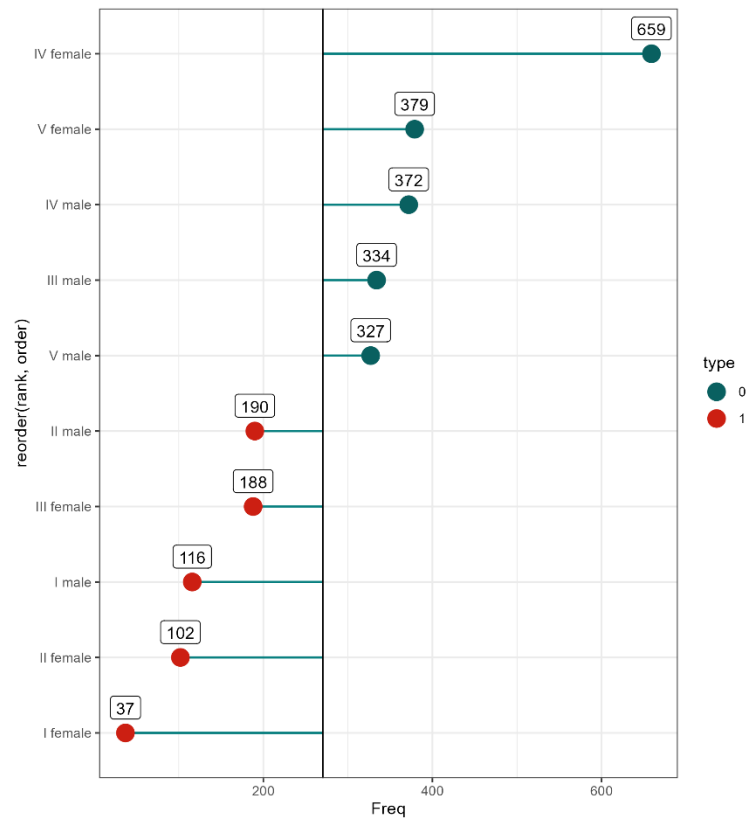


Рис. 9. Видозмінений варіант lollipop діаграми

Іноді важливо відобразити певну величину в межах кількох підгруп або комбінації категорій. Основними рішення цієї задачі є теплові карти, діаграми деревовидні карти та хмари слів.

HEATMAP (теплова карта)

Теплова карта - це графічне зображення даних, де окремі значення, що містяться в матриці, представлені у вигляді кольорів. У прикладі таким значенням є частота, а вимірами матриці є категорії трьох полів. ggplot2 має спеціальний шар для створення теплових карт – `geom_tile`, який потребує три естетики у об'єкті `aes`, а саме: `x`, `y`, та `fill`.

```
df %>%
  mutate(type=paste(ifelse(broken == 'yes', 'broken', 'not broken'),
gender)) %>%
  ggplot(aes(x = rank, y = gbtype, fill = Freq)) +
  geom_tile() +
  geom_text(aes(label=Freq), color="white") +
  scale_fill_viridis_c()
```

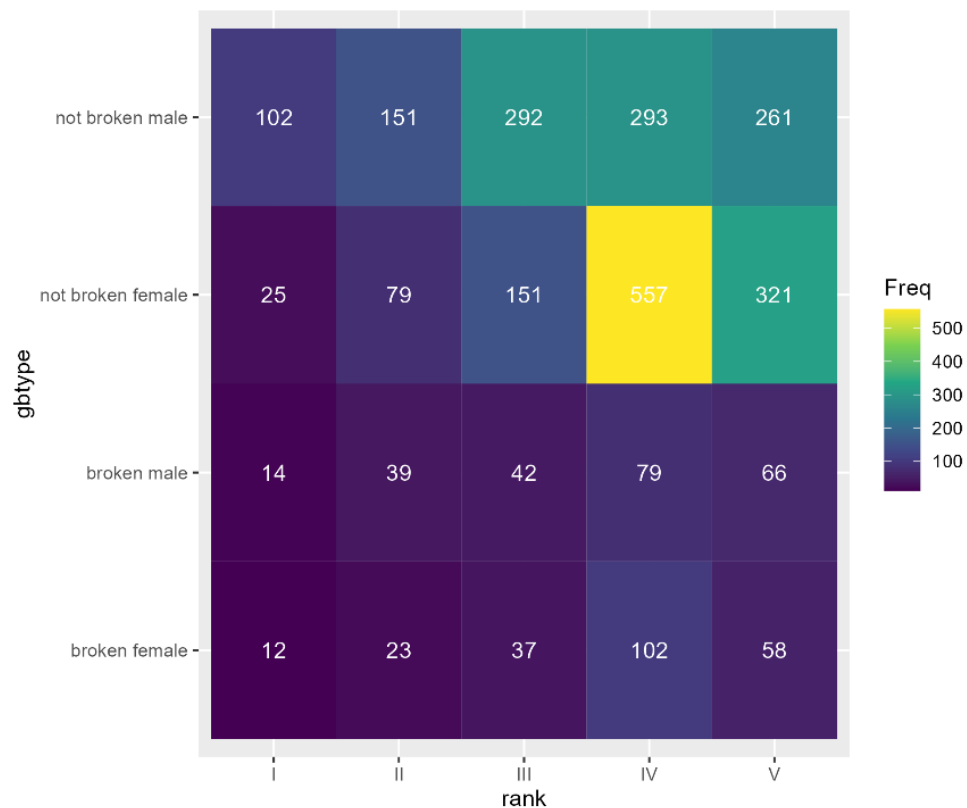


Рис. 10. Теплова карта по даним

TREEMAP (Деревовидна карта)

Treemap є узагальненням теплової карти, адже у ньому важливі не тільки кольори, а і площа кожного прямокутника. Формально treemap відображає ієрархічні дані у вигляді набору вкладених прямокутників. Деревовидна карта дозволяє використовувати як наявний простір, так і колір, тому вона корисна, коли треба чітко виділяти підгруп та усі можливі комбінації груп.

Ggplot2 немає вбудованої реалізації даного типу графіків, для цього існує пакет – розширення treemapify, що містить геометрію geom_treemap.

Простий приклад treemap з допомогою treemapify:

```
library(treemapify)
df %>%
  mutate(type = paste(ifelse(broken == 'yes', 'broken', 'not broken'),
gender, rank)) %>%
  ggplot(aes(area=Freq, fill=type, label=paste(type, Freq, sep="\n"))) +
  geom_treemap() +
  geom_treemap_text(
    colour = "white",
    place = "centre"
  )
```

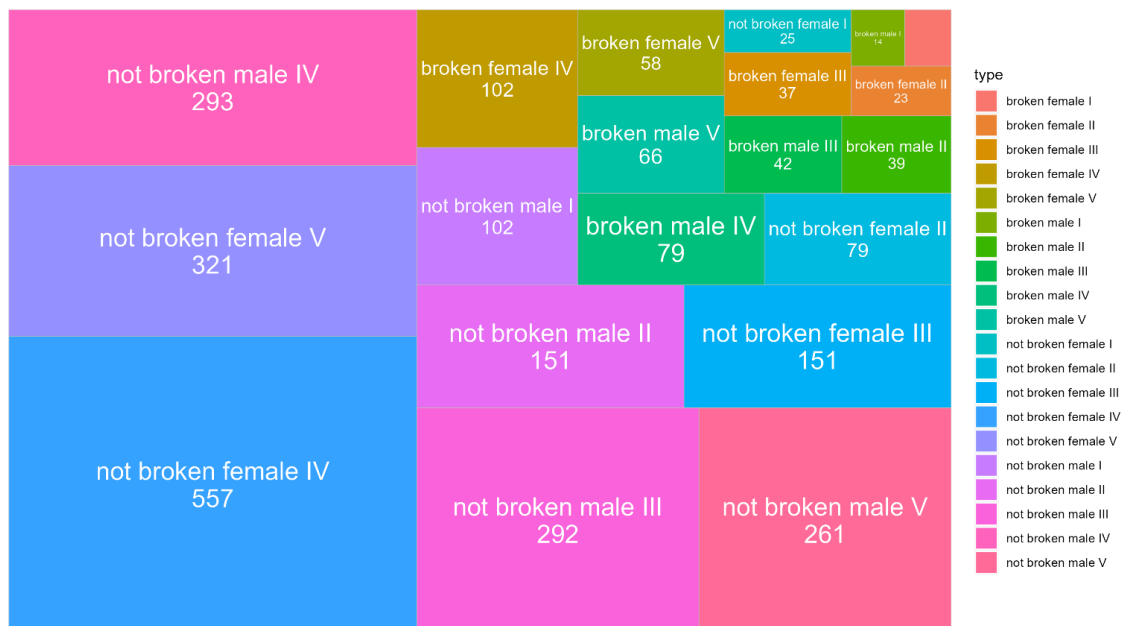


Рис. 11. Деревовидна діаграма з текстовими мітками

Пакет `treemapify` надає достатньо гнучкий API, що дозволяє створювати і більш складні візуалізації, якщо існує така потреба. Наприклад, розділити плитки на групи, додати підписи на задньому фоні тощо:

```
df %>%
  mutate(type = paste(ifelse(broken == 'yes', 'broken', 'not broken'),
gender)) %>%
  ggplot(aes(area=Freq, fill=Freq, label=paste(type, Freq, sep="\n"),
subgroup=rank)) +
  scale_fill_gradient2() +
  geom_treemap() +
  geom_treemap_subgroup_border(colour = "white", size = 5) +
  geom_treemap_subgroup_text(
    place = "centre", grow = TRUE,
    alpha = 0.25, colour = "black",
    fontface = "italic"
  ) +
  geom_treemap_text(colour = "black", place = "centre")
```

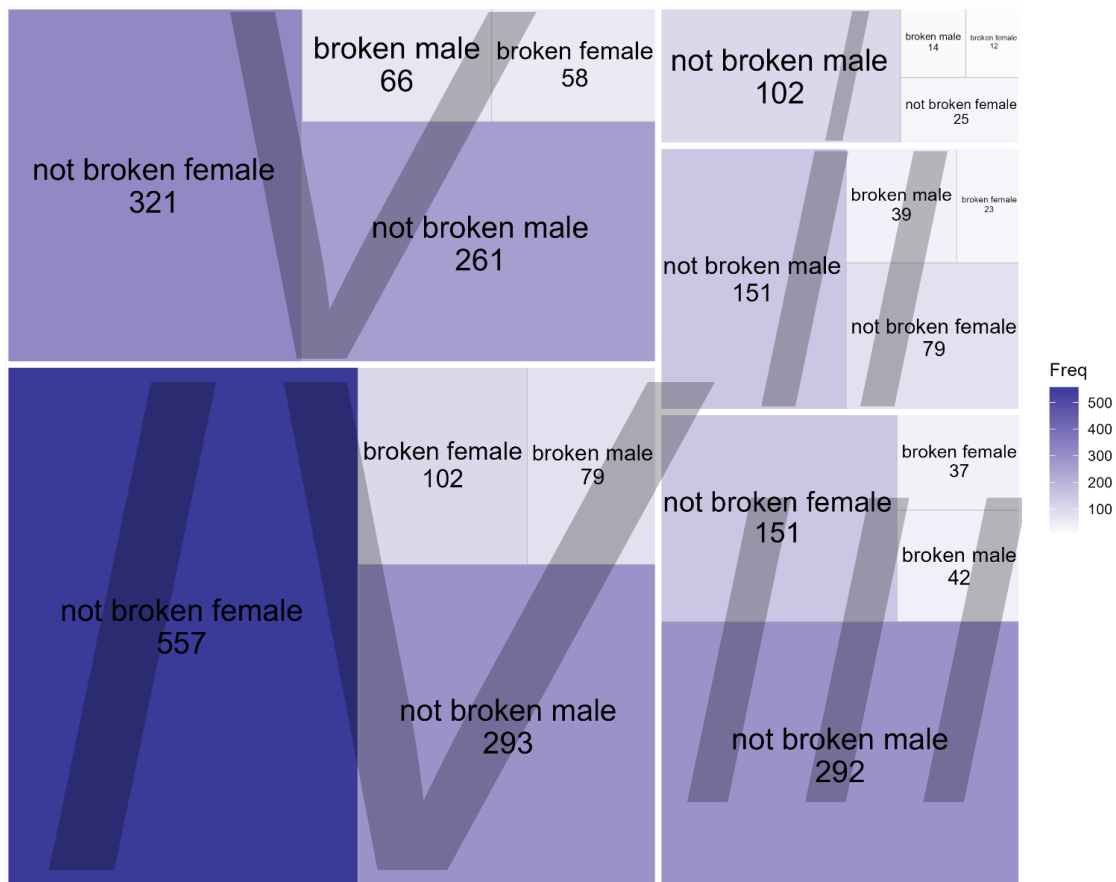


Рис. 12. Модифікована treemap

WORD CLOUD (“Хмара” слів)

Wordcloud відображає список слів, важливість кожного показано розміром або кольором шрифту. Цей формат корисний для швидкого сприйняття найвизначніших термінів. Така візуалізація добре відома та привертає увагу. Проте, має ряд недоліків. Наприклад, довгі слова будуть більш помітними на малюнку незалежно від їх виникнення.

Щоб створити хмару “слів” з допомогою ggplot2 необхідно встановити додатковий пакет ggwordcloud.

```
df %>%
  mutate(type = paste(ifelse(broken == 'yes', 'broken', 'not broken'),
gender, rank)) %>%
  ggplot(aes(label=type, size = Freq, color=rank)) +
  geom_text_wordcloud() +
  scale_color_brewer(palette = 'Paired') +
  scale_size_area(max_size = 20) +
  theme_minimal()
```

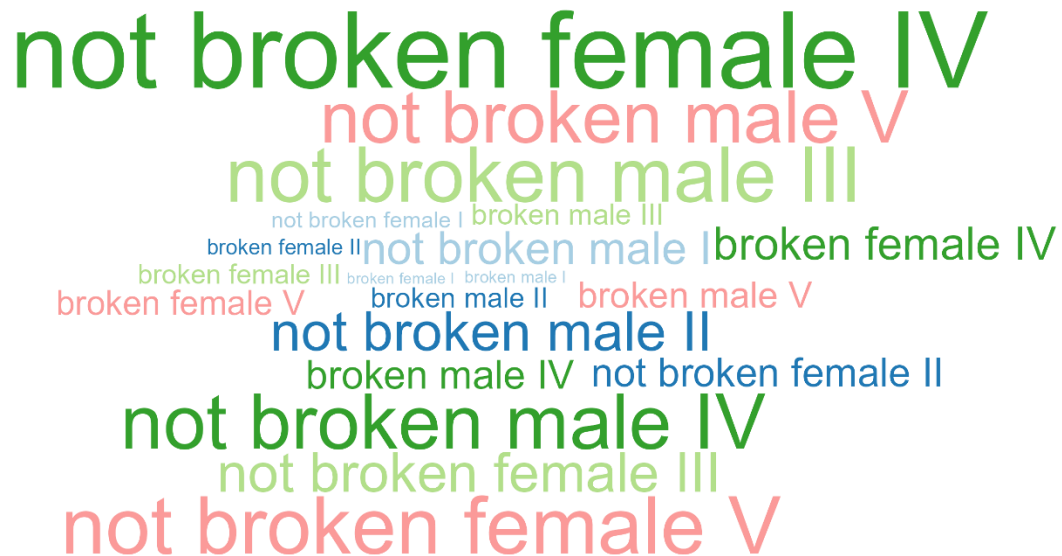


Рис. 13. Хмара слів по всім можливим комбінаціям категорій

Попередні приклади прекрасно працюють, якщо необхідно відобразити тільки одну некатегоріальну змінну залежно від наявних груп. Проте, часто кількість показників та категорій є більшою, і для візуалізації їх комбінації існують інші підходи.

Новий датасет, для побудови прикладів, є відома альтернатива даним iris – penguins:

<https://www.kaggle.com/datasets/ashkhagan/palmer-penguins-datasetalternative-iris-dataset>

Датасет містить три категоріальні змінні: вид, місце походження та стать пінгвінів, та чотири числових параметри – різні характеристики пінгвінів.

```
df <- read.csv("penguins.csv") %>% drop_na()
```

Візуалізація числових даних по категоріям

Перший важливий спосіб візуалізації комбінації числових та категоріальних даних – це використовувати відомі способи візуалізації числових даних, розбитих по категоріям. Це можуть бути точкові діаграми, boxplot, violinplot, гістограми, діаграми щільності тощо. Для прикладу, точкова діаграма (з “тремтінням”) geom_jitter, доповнена боксплотом або скрипковою діаграмою, по різним видам пінгвінів може бути задана так:

```
df %>%
  ggplot(aes(x = species, y = body_mass_g)) +
  geom_jitter(aes(color = sex, shape=island), size=2) +
  scale_color_brewer(palette = 'Paired') +
  # geom_boxplot(alpha=0.3, notch=T) + # - для боксплоту
  geom_violin(alpha=0.3, fill="grey") +
  theme_bw()
```

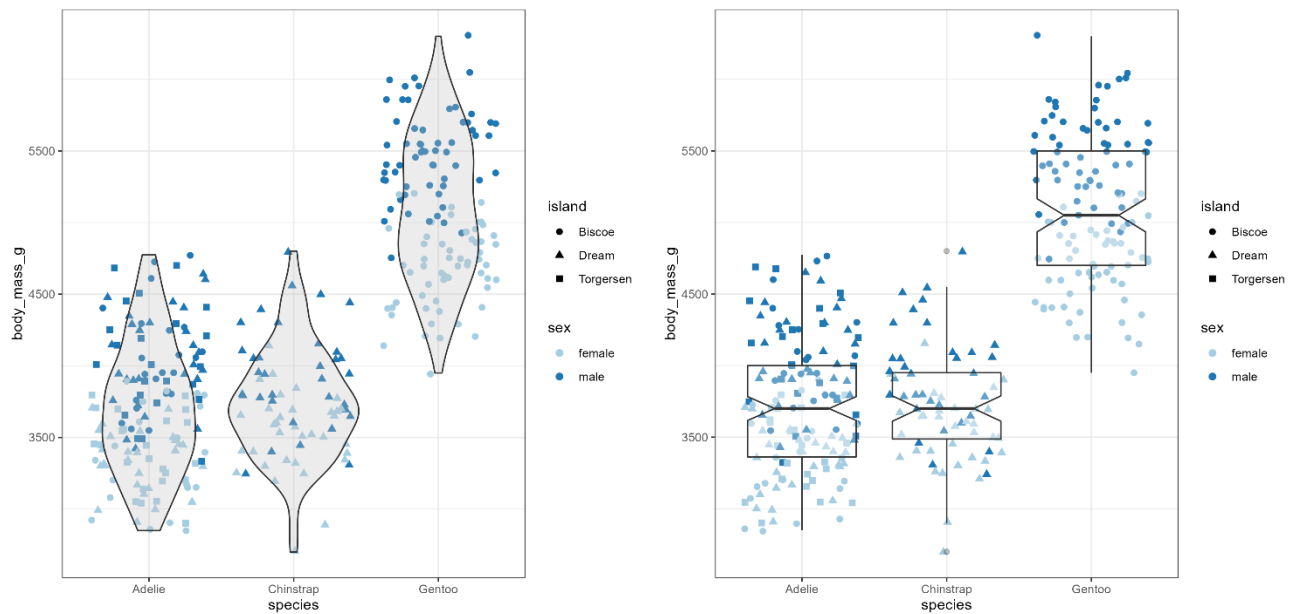


Рис. 14. Скрипкова діаграма та boxplot маси пінгвінів по їх видам

Вибір типу візуалізації числової змінної, розбитої по категоріям, залежить від задачі, що ставиться перед візуалізацією даних.

RADARCHART / SPIDER PLOT (радіолокаційна діаграма)

Радіолокаційна діаграма - це двовимірний тип діаграми, призначений для побудови однієї або декількох серій значень на кількох загальних кількісних змінних, поданих у полярній системі координат. Кожна змінна має свою вісь, усі осі є спільними в центрі фігури. Відомий приклад такої діаграми – рози вітрів.

Для того, щоб побудувати цей тип діаграм у ggplot2 існує пакет ggradar. Для прикладу, побудовано діаграму для кількох випадкових записів із датасету:

```
df %>%
  mutate(species = paste(species, island, sex)) %>%
  as_tibble() %>%
  select(c(1, 3:6)) %>%
  mutate_at(vars(-species), rescale) %>%
  slice_sample(n=10) %>%
  ggradar(fill=T, fill.alpha=0.1)
```

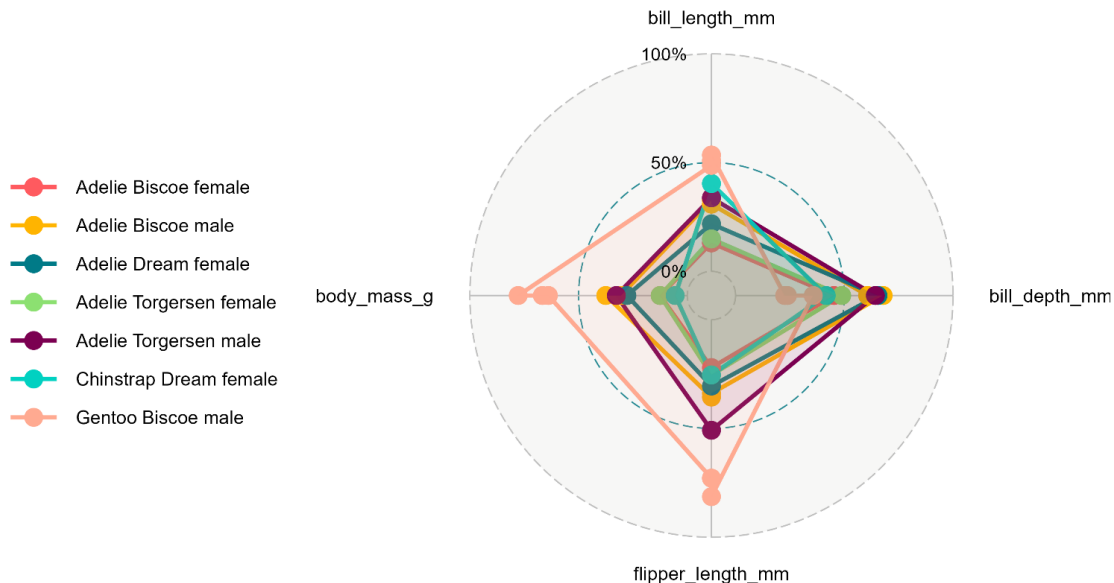


Рис. 15. Радіальна діаграма для невеликої вибірки даних по пінгвінам

PARALLEL PLOT (діаграма паралельних координат)

Діаграма паралельних координат дозволяє візуалізувати багатовимірні дані, порівнювати особливості кількох окремих спостережень на наборі числових змінних. Кожна вертикальна смуга являє собою змінну і зазвичай має свою шкалу. Кожна паралельна група може бути як числовим, так і категоріальним полем.

Такий тип діаграм можна розглядати як декартовий варіант `radarchart`.

Найлегший спосіб побудувати `parallel plot` з допомогою `ggplot2` це використати пакет `GGally`, що надає шар `ggparcoord`.

Проста діаграма, побудована на 100 випадкових записах з датасету:

```
library(GGally)
df %>%
  mutate(island = factor(island)) %>%
  slice_sample(n=100) %>%
  ggparcoord(
    groupColumn = "species",
    columns=2:6,
  ) + scale_color_brewer(palette = "Paired")
```

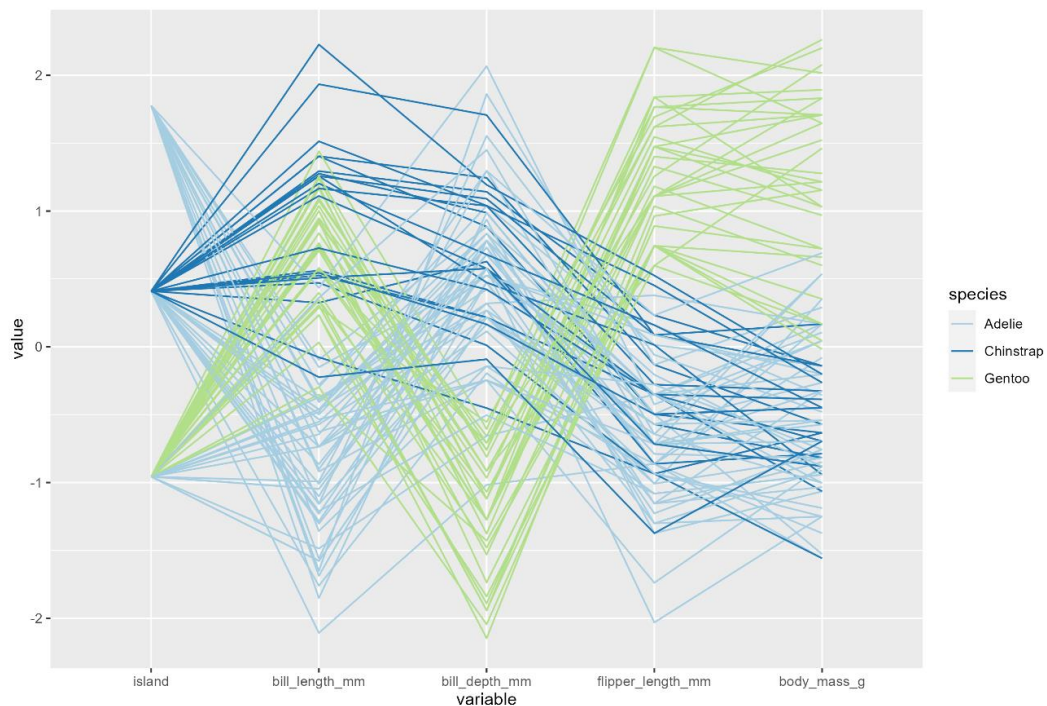



Рис. 16. Базова діаграма паралельних координат

Наявні параметри дозволяють регулювати появу діаграми, наприклад, додавати точки даних (showPoints), задавати поля, що враховувати при побудові (columns), або проводити сплайн інтерполяцію між точками (splineFactor) тощо.

```
df %>%
  mutate(island = factor(island)) %>%
  slice_sample(n=100) %>%
  ggparcoord(
    groupColumn = "species", columns=2:6,
    splineFactor = 10, showPoints = T,
  ) + scale_color_brewer(palette = "Paired") + theme_bw()
```

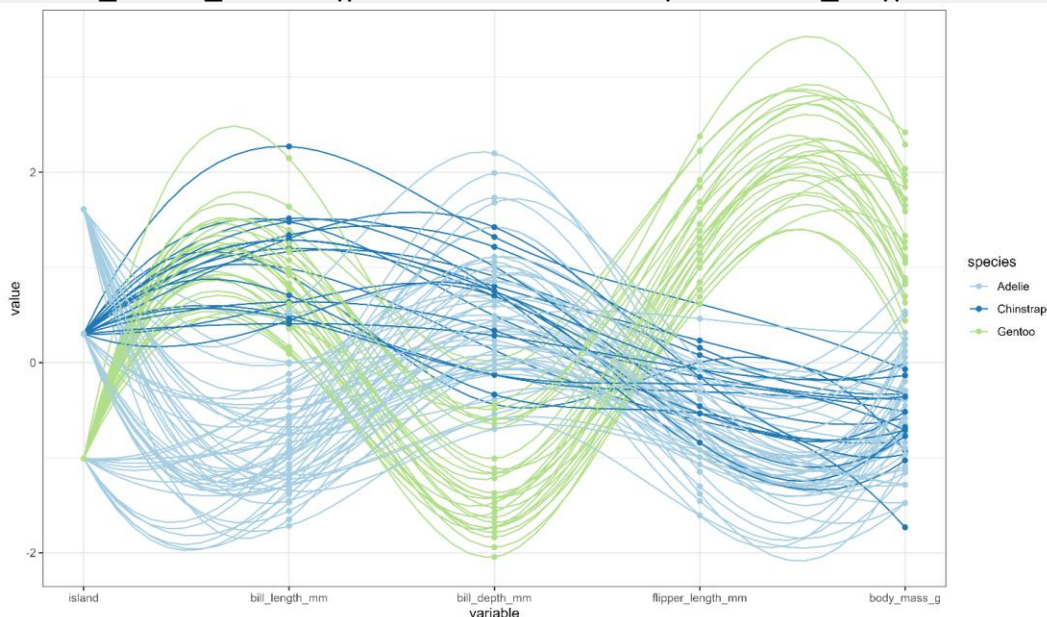


Рис. 17. Вдосконалена діаграма

Якщо кількість даних дуже велика, або категорії не мають чіткої роздільності по даним, то існує можливість розбиття графіку на окремі діаграми для кожного категорії. Для цього існує шар `facet_wrap`.

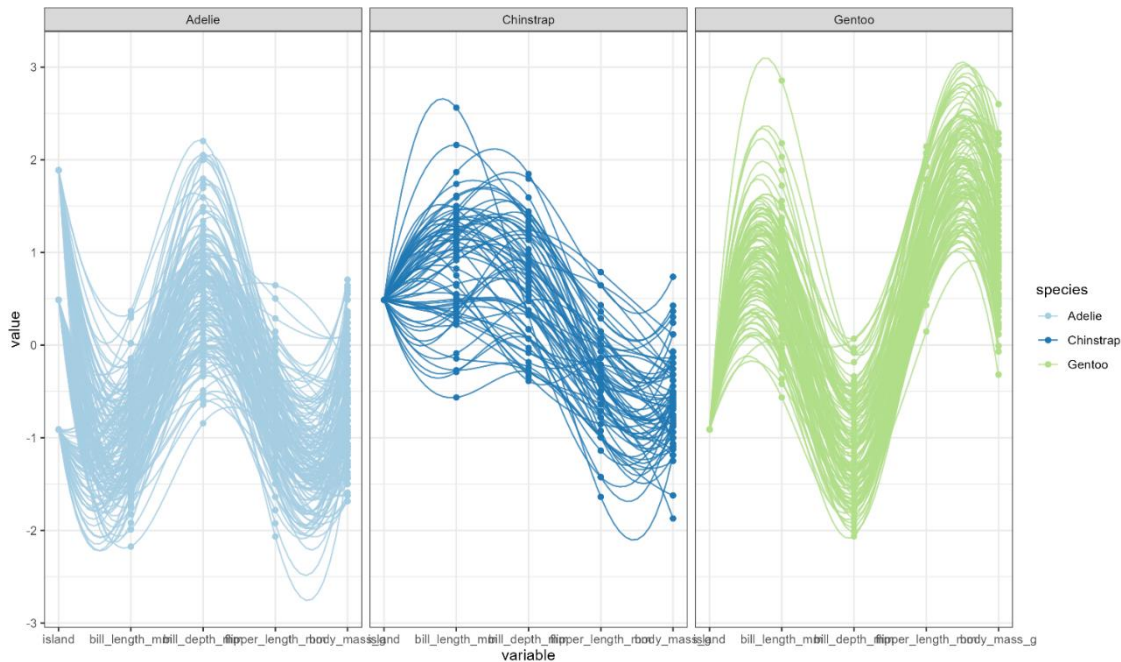


Рис. 18. Розбиття діаграми по категоріям

Існує велика кількість варіантів та додаткових типів діаграм та графіків, що можуть бути використані для візуалізації категоріальних даних. До них належать:

- Vien Diagram – діаграми Вена;
- Sunburst – радіальна версія treemap;
- Mosaic plot – мозаїчна діаграма;
- Dendrogram – дендограми;
- Circular packing – круговий еквівалент treemap;
- Мережні діаграми, як Sankey diagram (діаграма Санкея) або графі тощо.

Також, окрім ggplot2 існують й інші потужні бібліотеки, що дозволяють візуалізувати дані, наприклад plotly, esquisse, Lattice тощо.

Завдання

1. Ознайомитися з теоретичними відомостями та лекційним матеріалом.
2. Знайти або створити набір даних, про Клас, що розбитий на Категорії та додаткового містить числові змінні. Побудувати п'ять діаграм певного типу, з Таблиці 1, відповідно до варіантів завдань з Таблиця 2.

Таблиця 1 – Типи діаграм

Номер	Тип діаграми	Номер	Тип діаграми
1	Barplot	7	Treemap
2	Lollipop	8	Radarchart
3	Circular packing	9	Parallel plot
4	Donut	10	Heatmap
5	Pie chart	11	Scatter plot
6	World cloud	12	Boxplot

Таблиця 2 – Варіанти завдань

№	Клас	Можливі категорії			Типи діаграм
1	Техніка	Тип (смартфон, ноутбук,...)	Пам'ять	Виробник	1,3,8,10,11
2					1,4,6,7,8
3					1,3,7,10,12
4					1,5,6,8,11
5	Автомобілі	Бренд	Тип	Тип трансмісії (механіка,автомат,...)	2,3,7,10,11
6					2,5,9,10,12
7					2,3,6,7,12
8					2,4,7,9,11
9	Кінематограф	Тип (фільм, серіал,...)	Жанр	Мова оригіналу	1,3,6,8,11
10					1,5,8,10,11
11					1,3,6,7,12
12					1,4,8,10,12
13	Робота	Посада, роль	Місце	Вимоги	2,3,7,9,11
14					2,5,8,10,12
15					2,3,5,9,12