

## Course: R Language in Computational Probability and Statistics

### Lecture 8: High-level plotting functions

Lecturer: Nataliia Krulhova

#### Функції високого рівня

Найбільш потужним та популярним інструментом для візуалізації даних в R є пакет **ggplot2**. З його допомогою можна створювати якісні, естетичні графіки з високим рівнем налаштувань. Створення графіка з пакетом **ggplot2** починається з виклику функції **ggplot()**, в яку подаються дані та загальна структура графіка. Зазвичай, це встановлення естетичних атрибутів, таких як колір, форма, розмір тощо.

В базовій графіці ми повинні спочатку використати функцію високого рівня, щоб встановити загальну структуру графіка, таку як його розмірність, масштаби осей, назви тощо. Серед найбільш вживаних функцій високого рівня можна відзначити **plot()**, **scatterplot()**, **hist()**, **boxplot()**, **pairs()**.

Після створення основної структури графіка до нього можна додавати додаткові елементи, такі як текст, лінії, легенду тощо, використовуючи функції низького рівня. До таких функцій належать **lines()**, **points()**, **legends()** та **axis()**.

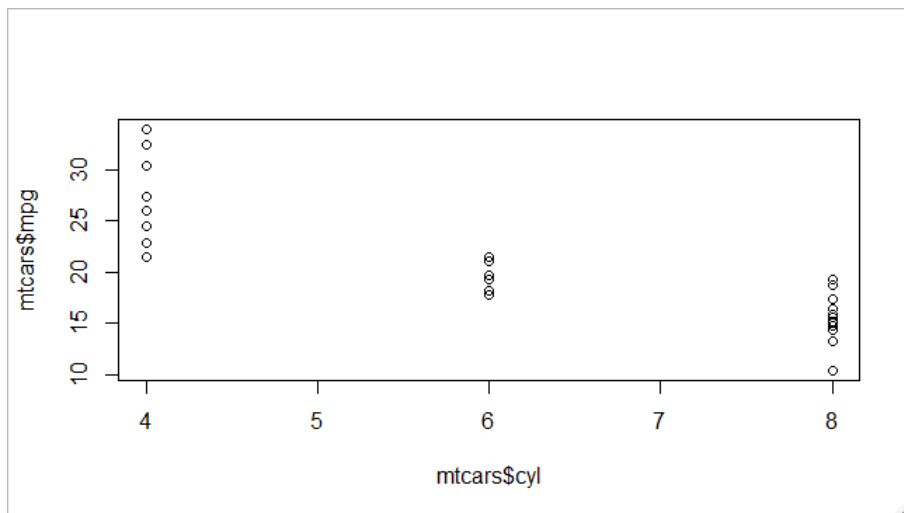
Управління деталями графічного зображення здійснюється за допомогою набору параметрів, які визначають колір, тип, розміри символів або маркерів, товщину та стиль ліній, штрихування, рамку графіка і багато іншого.

#### Функція **plot()**

Функція **plot()** є ключовою у створенні графіків в R. Вона визначає загальну структуру графіка, і тип графіка, який буде створено, залежить від класу об'єктів, переданих цій функції в якості аргументів. Тому за допомогою **plot()** можна створити графіки різного типу та форми.

**Приклад.** Побудуємо графік залежності змінної **mpg** від змінної **cyl** в наборі даних **mtcars**.

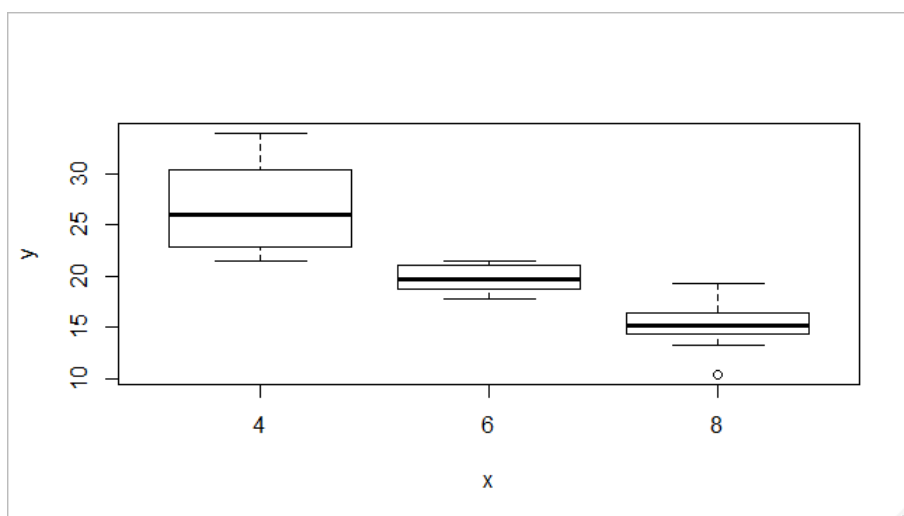
```
> plot(mtcars$cyl,mtcars$mpg)
```



Оскільки змінна **cyl** приймає невелику кількість значень, то її можна перетворити на категоріальну (факторну). Тепер змінна по осі X в нас факторна. У випадку факторної змінної R буде діаграму «скриня з вусами» для кожної категорії факторної змінної.

**Приклад.**

```
> plot(factor(mtcars$cyl),mtcars$mpg)
```



### Параметри функції plot()

Параметри функції **plot()** дозволяють налаштувати вигляд графіка.

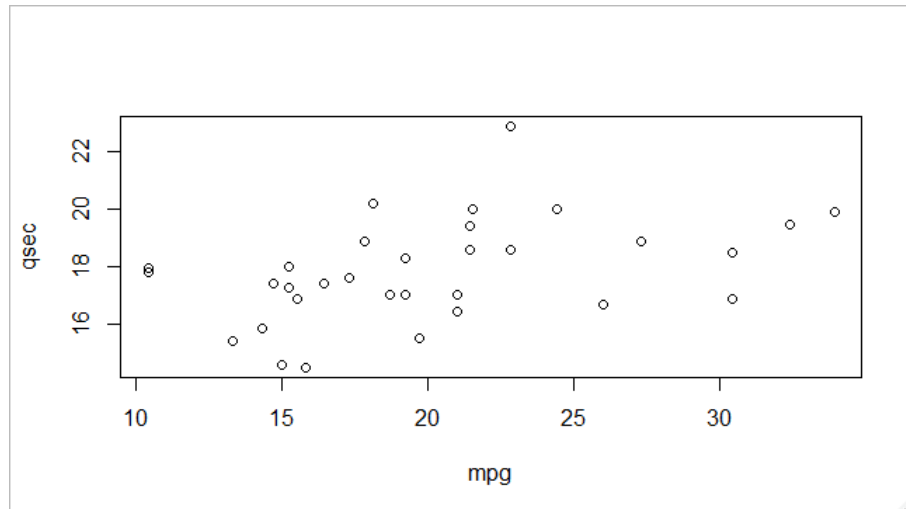
Розгляньмо деякі з них.

*Параметри **xlab** і **ylab**.*

Параметри **xlab** і **ylab** змінюють назви осей **X** та **Y** відповідно.

### Приклад.

```
> plot(mtcars$mpg,mtcars$qsec,xlab="mpg",ylab="qsec")
```



### Параметр *type*.

Параметр **type** змінює вигляд точок на графіку.

Він приймає наступні значення.

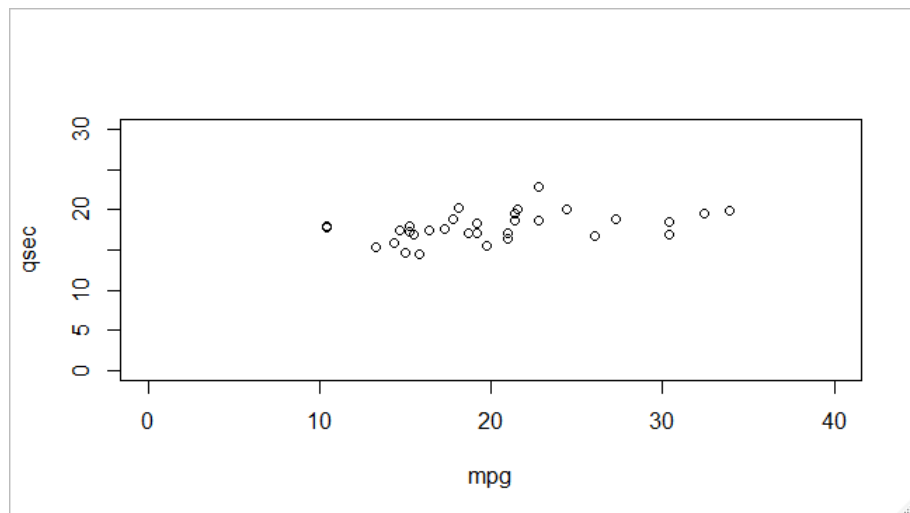
Значення	Опис
<b>b</b>	Точки і лінії
<b>h</b>	Гістограма
<b>l</b>	Лінія
<b>n</b>	Дані не відображаються
<b>o</b>	Точки зверху ліній
<b>p</b>	Точки (значення по замовчуванню)
<b>s</b>	Ступінчата лінія

### Параметри *xlim* та *ylim*.

Встановлюють діапазон значень по кожній осі.

### Приклад.

```
> plot(mtcars$mpg,mtcars$qsec,xlab="mpg",ylab="qsec",xlim=c(0,40),  
ylim=c(0,30))
```



*Параметр axes.*

Цей параметр відповідає за побудову на графіку осей координат. Якщо він приймає значення **TRUE**, то на графіку відображаються вісі координат, якщо **FALSE**, то - ні.

*Параметр sub.*

За допомогою цього параметра задається підпис графіка під зображенням.

*Параметр asp.*

Задає коефіцієнт відношення між масштабними одиницями осі ординат та абсцис. Якщо задати значення 1 для цього параметру, то масштаб по осях буде однаковий, якщо не підключати параметр **asp**, то R буде автоматично визначати найбільш зручне для розміщення графіку значення параметру.

*Параметр log.*

Визначає вибір осі, по якій задається логарифмічний масштаб. Значення параметру **log="x"** задає логарифмічний масштаб по осі абсцис, **log="y"** – по осі ординат, **log="xy"** — по обох осях.

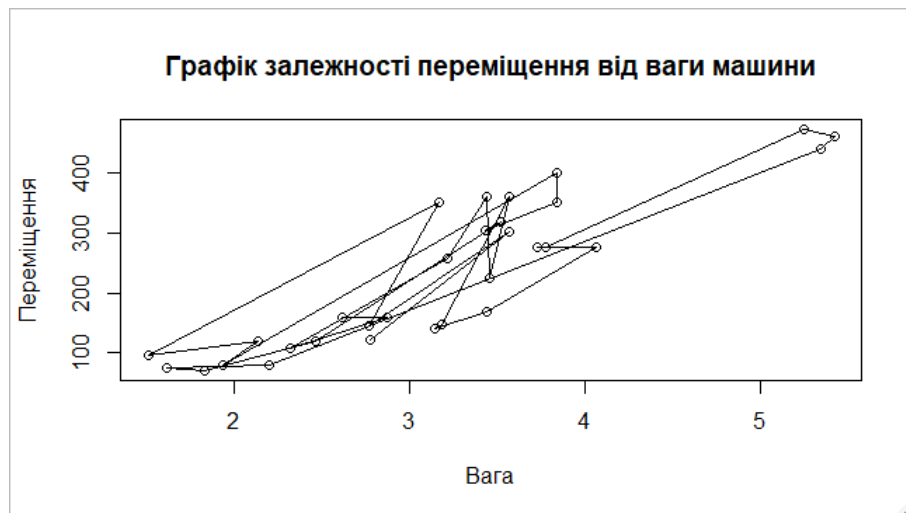
*Параметр lwd* — контролює ширину ліній.

*Параметр main.*

Цей параметр формує заголовок графіка. За стандартом назва рисунка розташовується в верхній частині.

**Приклад.**

```
> plot(mtcars$wt,mtcars$disp,ylab="Переміщення",xlab="Вага",type="o",
main="Графік залежності переміщення від ваги машини")
```



*Параметр pch.*

Відповідає за тип маркера. Приймає числові значення від 1 до 25.

**Приклад.**

```
> plot(mtcars$wt,mtcars$disp,ylab="Переміщення",xlab="Вага",type="o",main="Графік залежності переміщення від ваги машини",pch=5)
```



В наступному прикладі буде виведено стандартні маркери з їх кодами.

**Приклад.**

```
> x<-c(rep(1,5),rep(2,5),rep(3,5),rep(4,5),rep(5,5))
> y<-rep(1:5,5)
> plot(x,y,pch=1:25,col=1:25,cex=2,axes=F,xlab="",ylab="",ylim=c(0,6))
> text(x+0.1,y,1:25,cex=1)
```

◇5	⊕10	■15	●20	▽25
×4	⊕9	⊠14	●19	△24
+3	*8	⊠13	◆18	◇23
△2	⊠7	⊠12	▲17	□22
○1	▽6	⊠11	●16	○21

Якщо в якості значення параметра **pch** передати змінну символічного типу, то точки будуть зображуватись не маркером, а відповідним символом.

**Приклад.**

```
> plot(mtcars$wt,mtcars$disp,ylab="Переміщення",xlab="Вага",type="o",
main="Графік залежності переміщення від ваги машини",pch="7")
```



Параметр **sex** (**sex=число**) визначає розмір маркера.

Колір маркера можна задати наступним чином:

° за назвою: **col = "red"** або **col = "green"** або **col = "black"**.

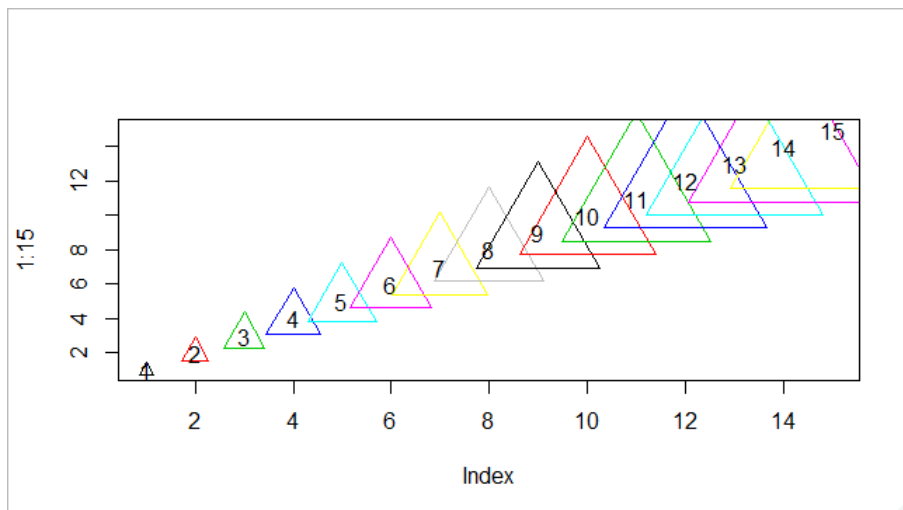
В R є 657 стандартних кольорів. Їх назви можна вивести на екран за допомогою функції **colors()**;

°безпосереднім переліком основних компонент: червоний , зелений , синій в RGB спектрі, наприклад:"#RRGGBB";

°за кодом, приміром: **col = 2**(червоний), **col = 3**(зелений).

**Приклад:**

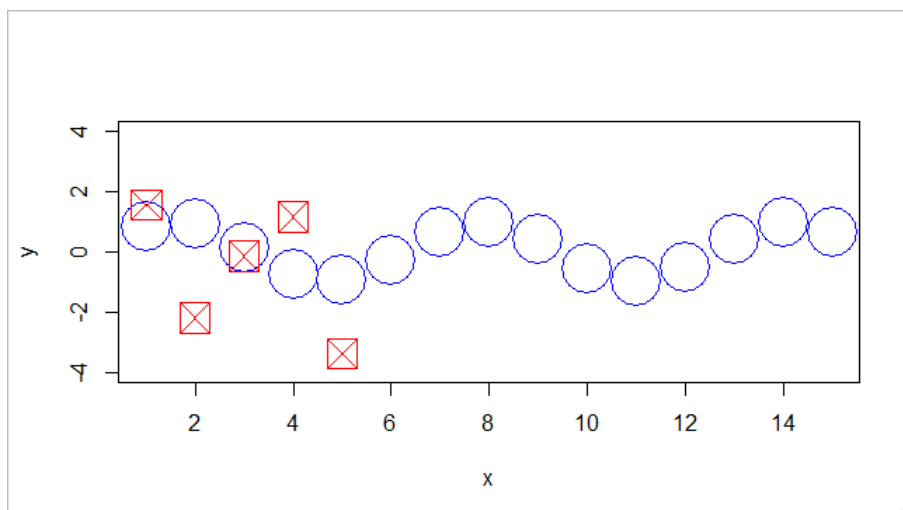
```
> plot(1:15, pch=24, sex=1:15, col=1:15)
> text(1:15)
```



Функція **plot()** в першу чергу створює вікно виводу, після чого додає у нього графічні об'єкти. Якщо потім буде потрібно додати ще графічний об'єкт до графіку, то не можна використовувати для цього функцію високого пріоритету, а необхідно використовувати функцію низького пріоритету.

**Приклад.**

```
> x<-c(1:15)
> y<-sin(x)
> plot(x,y,cex=5,col="blue",ylim=c(-4,4))
> points(x[1:5],tan(x[1:5]),cex=3,col="red",pch=7)
```

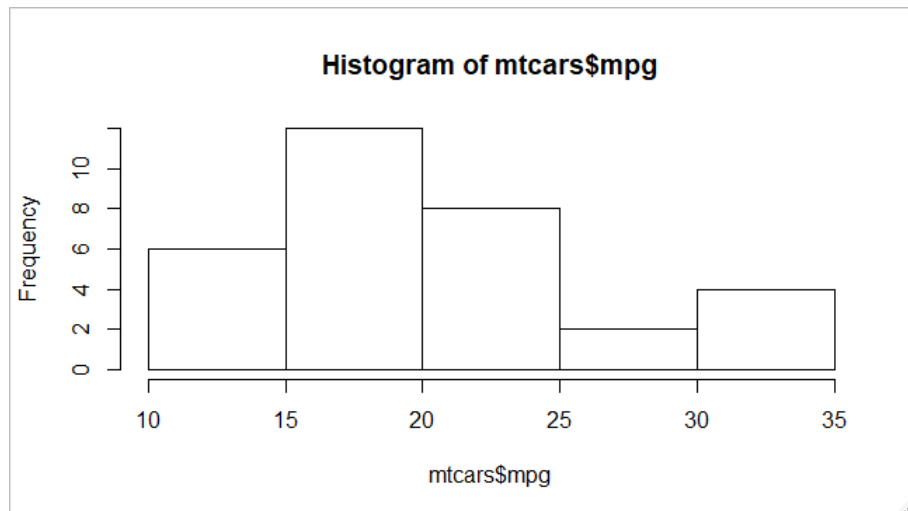


### Гістограма

В системі **R** для побудови гістограми використовують функцію **hist()**. Її аргумент – ім'я об'єкту, що аналізується. Якщо вказати тільки ім'я змінної в якості параметру, то система автоматично визначить кількість інтервалів розбиття.

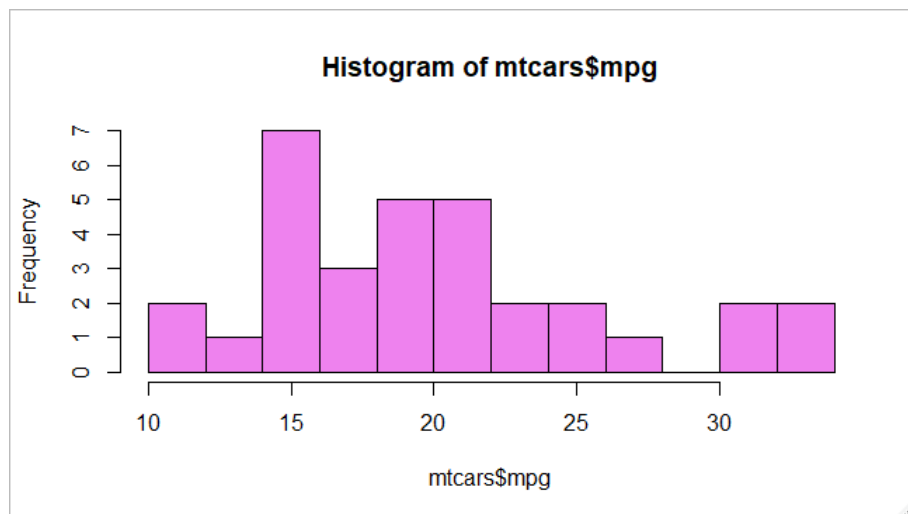
### Приклад.

```
> hist(mtcars$mpg)
```



**Приклад.** Побудуйте гістограму частот, змінивши кількість інтервалів розбиття.

```
> hist(mtcars$mpg, breaks=10,col="violet")
```



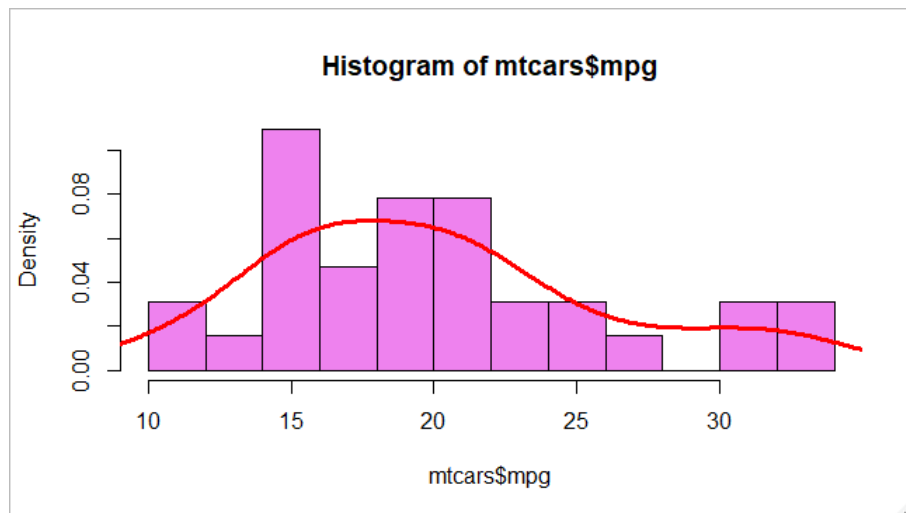
Параметр **freq = FALSE** дозволяє побудувати гістограму відносних частот. Площа гістограми дорівнює одиниці.

Функція **density()** дає оцінку щільності розподілу.

Цілком логічно сумістити гістограму з графіком щільності.

### Приклад.

```
> hist(mtcars$mpg, breaks=10,col="violet",freq=F)  
> lines(density(mtcars$mpg),lwd=3,col="red")
```



### Скриня з вусами(box-whisker plots)

Діаграма розмаху (скриня з вусами) - задає візуальну статистичну характеристику генеральної сукупності, основні графічні характеристики зображено на рисунку.

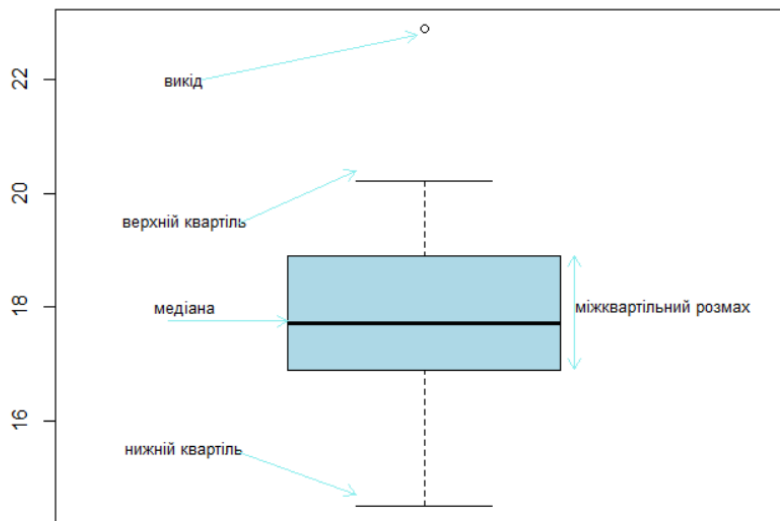
#### Приклад.

```

> boxplot(mtcars$sqsec,col="lightblue",main="Скриня з вусами")
> text(0.65,15.5,"нижній кuartіль",sex=0.8)
> text(0.65,18,"медіана",sex=0.8)
> text(0.65,19.5,"верхній кuartіль",sex=0.8)
> text(0.65,22,"викід",sex=0.8)
> text(1.35,18,"міжкuartільний розмах",sex=0.8)
> arrows(x0=0.725,y0=15.45,x1=0.9,y1=14.7,col = "darkslategray2",
length = 0.1, code=2)
> arrows(x0=0.625,y0=17.75,x1=0.8,y1=17.75,col = "darkslategray2",
length = 0.1, code=2)
> arrows(x0=0.725,y0=19.45,x1=0.9,y1=20.4,col = "darkslategray2",
length = 0.1, code=2)
> arrows(x0=0.675,y0=22,x1=0.99,y1=22.8,col = "darkslategray2",
length = 0.1, code=2)
> arrows(x0=1.22,y0=16.9,x1=1.22,y1=18.9,col = "darkslategray2",
length = 0.1, code=3)

```

Скрина з вусами



### Функція `curve()`

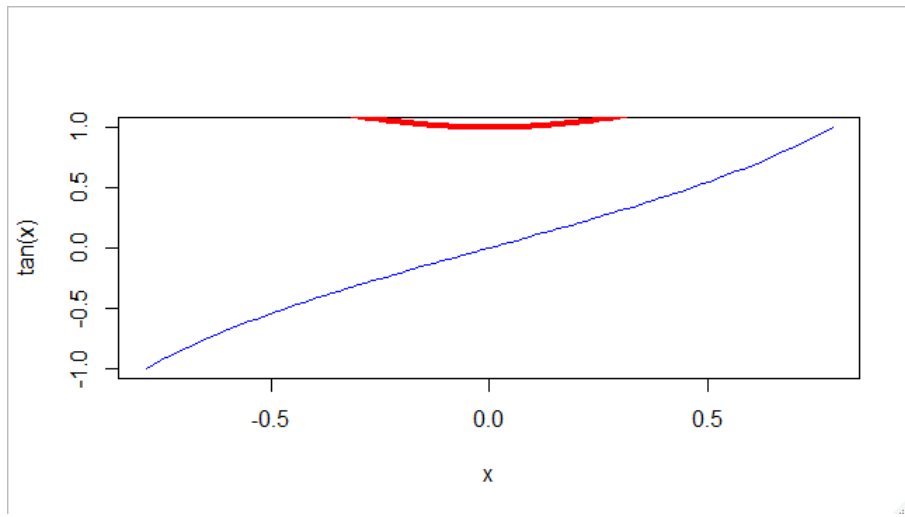
Функція `curve()` будує графік заданої функції на деякому інтервалі.

Аргументи функції:

- **expr** — аналітичний вираз від  $x$  або ім'я функції, графік якої будується.
- **from, to** — числові границі відрізка, на якому будується графік. Якщо ці параметри не задані, то границі визначаються значенням аргументу  $x$ . Якщо  $x$  не визначено, то графік будується на відрізку  $[0; 1]$ .
- **n** — числовий аргумент (число точок у **expr**).
- **add** — логічний аргумент (**add=TRUE** – графік додається до існуючого, **add=FALSE** – створюється нове вікно).

**Приклад.** Будуємо графік функції і її похідної:  $f_1(x) = \tan(x)$ ,  $f_2(x) = \frac{1}{\cos^2(x)}$

```
> curve(tan(x),-pi/4,pi/4,col="blue")  
> curve(1/(cos(x))^2,add=T,col="red",lwd=4)
```



Всі малюнки у лекції були згенеровані в RStudio.