Domača naloga 6

Alen Kahteran

3. 11. 2020

Zagon potrebnih knjižnic ter pridobivanje podatkov

```
library(tidyverse)
data("iris")
iris = cbind(row_num=rownames(iris), iris)
df=tibble(iris)
```

Uporaba sample_n() funkcije iz dplyr paketa.

Princip je podoben kot za funkcijo sample(), le da dobimo rezultat v obliki `tbl_df`. `sample_n(df, 6, replace=TRUE)` izbere 6 zapisov izmed vseh možnih z enako verjetnostjo. S funkcijo `sample()` bi to lahko naredili na tak način da najprej naredimo to na vektorju indeksov in nato te indekse izberemo iz prvotnega `data.frame`.

```
# set the seed for reproducability
set.seed(8)

# an example of selecting 6 sample with function sample(), which is equivalent to below
# df_6 <- df[sample(x=1:nrow(df), size=6, replace=FALSE), ] %>%
#     # create unique identifier which is also a title
#     mutate(facet_name=paste0(Species, ", (", row_num, ")")) %>%
#     # remove unnecessary columns
#     select(-Sepal.Length, -Sepal.Width, -Petal.Width, -Species, -row_num)

# select 6 random iris flowers
df_6 <- sample_n(df, 6, replace=FALSE) %>%
    # create unique identifier which is also a title
    mutate(facet_name=paste0(Species, ", (", row_num, ")")) %>%
    # remove unnecessary columns
    select(-Sepal.Length, -Sepal.Width, -Petal.Width, -Species, -row_num)
```

Definicija funkcije za generiranje x in y vrednosti za pravilen izris

```
# generate x and y valus based on species
convert_to_xy <- function(row) {

    # create correct sequence of phi values.
    phi <- seq(0,
               ifelse(strsplit(row[2][1], " ")$facet_name[1] == "versicolor,", 2 * pi, pi),
               0.001)

    if (strsplit(row[2][1], " ")$facet_name[1] == "versicolor,") {
```

```
        x <- as.numeric(row[1]) * cos(phi) * sin(3 * phi) ^2
        y <- as.numeric(row[1]) * sin(phi) * sin(3 * phi) ^2
    } else {
        x <- as.numeric(row[1]) * cos(phi) * sin(3 * phi)
        y <- as.numeric(row[1]) * sin(phi) * sin(3 * phi)
    }

    # output a list of lists
    return(list(facet_name=rep(as.character(row[2]), length(phi)),
                x=x,
                y=y))
}
```

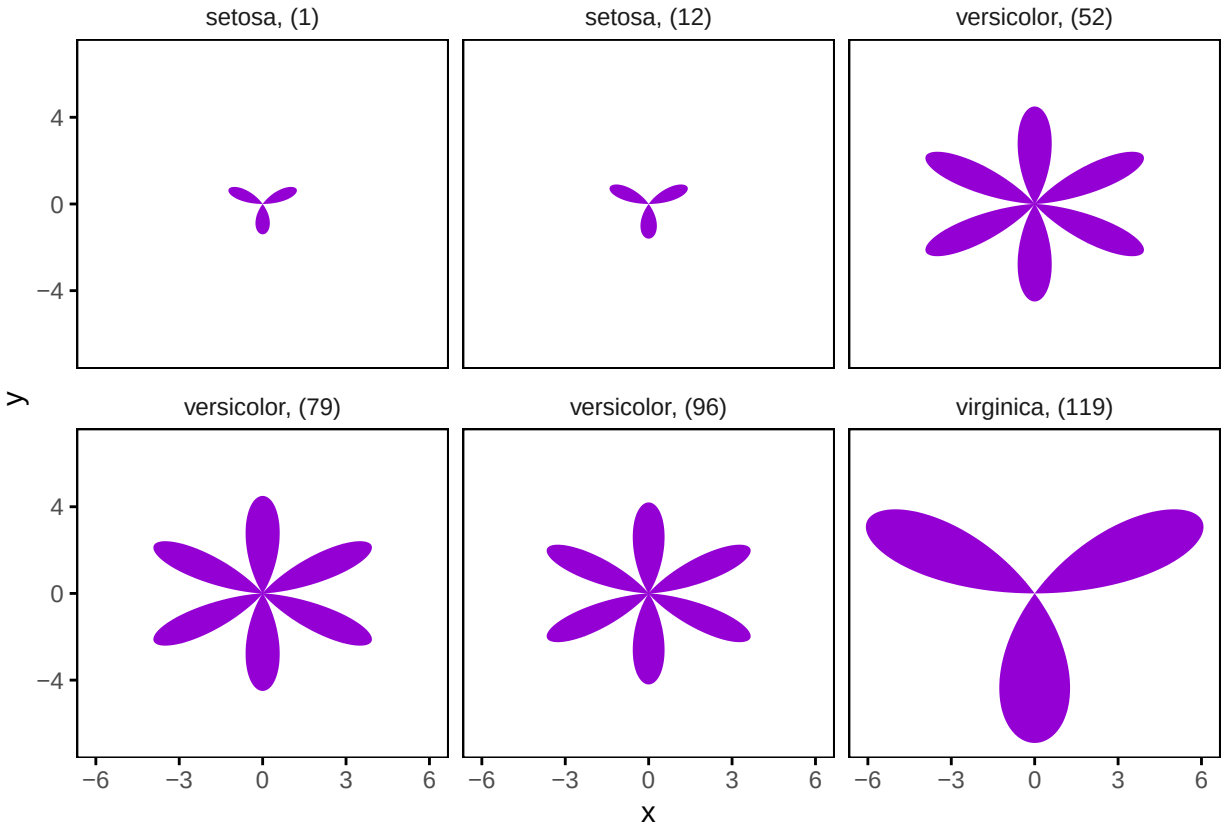
Preoblikovanje seznama seznamov ki ga vrne apply funkcija

```
# correctly reshape the data to long format
df_6_gg <- transpose(apply(df_6, 1, convert_to_xy)) %>%
    as_tibble() %>%
    unnest(cols=c(facet_name, x, y))
```

Risanje grafov

```
fw <- ggplot(df_6_gg, aes(x=x, y=y)) +
    geom_polygon(fill="#9400d3") +
    # setting ylim value so (0,0) is in the middle, x is taken care of by facet_wrap
    ylim(c(min(df_6_gg$y), abs(min(df_6_gg$y))))

fw + facet_wrap(vars(facet_name),
                nrow=2,
                scales="fixed") +
    theme_minimal() +
    theme(panel.grid = element_blank(), # remove the grid
          panel.background = element_rect(colour = "black"), # add black panel outline
          axis.ticks = element_line()) # add axis ticks
```


setosa, (1)
setosa, (12)
versicolor, (52)
versicolor, (79)
versicolor, (96)
virginica, (119)
y
x
4
0
−4
−6
−3
0
3
6