Visualizing dataset to apply machine learning

Jan-Philipp Kolb

30 Mai, 2019

Exercises: Visualizing dataset to apply machine learning

- Exercise based on r-exercises - visualizing for ml

Exercise 1

Create a variable “x” and attach to it the input attributes of the “iris” dataset. HINT: Use columns 1 to 4.

Exercise 2

Create a variable “y” and attach to it the output attribute of the “iris” dataset. HINT: Use column 5.

Exercise 3

Create a whisker plot (boxplot) for the variable of the first column of the “iris” dataset. HINT: Use `boxplot()`.

Exercises

Exercise 4

Now create a whisker plot for each one of the four input variables of the “iris” dataset in one image. HINT: Use par().

Exercise 5

Create a barplot to breakdown your output attribute. HINT: Use plot().

Exercise 6

Create a scatterplot matrix of the “iris” dataset using the “x” and “y” variables. HINT: Use featurePlot().

Exercise 7

Create a scatterplot matrix with ellipses around each separated group. HINT: Use plot=“ellipse”.

Exercises

Exercise 8

Create box and whisker plots of each input variable again, but this time broken down into separated plots for each class. HINT: Use plot="box".

Exercise 9

Create a list named "scales" that includes the "x" and "y" variables and set relation to "free" for both of them. HINT: Use `list()`

Exercise 10

Create a density plot matrix for each attribute by class value. HINT: Use `featurePlot()`.

Solutions

Solution Exercise 1

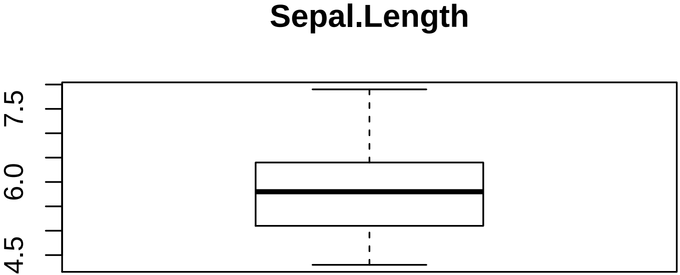
```
library(caret)
data(iris)
validation <- createDataPartition(iris$Species, p=0.80,
                                   list=FALSE)
validation20 <- iris[-validation,]
iris <- iris[validation,]
x <- iris[,1:4]
```

Solution Exercise 2

```
library(caret)
y <- iris[,5]
```

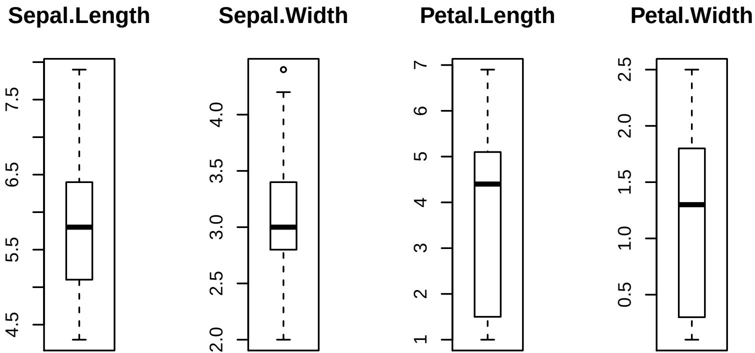
Solution Exercise 3

```
library(caret)
boxplot(x[,1], main=names(iris)[1])
```



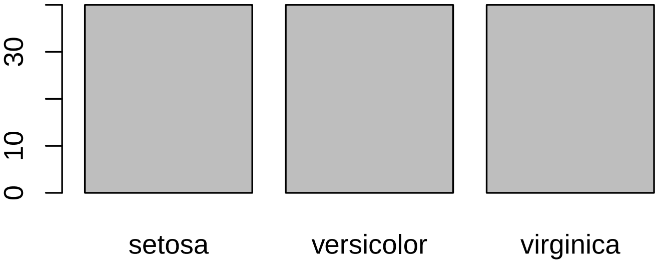
Solution Exercise 4

```
library(caret)
par(mfrow=c(1,4))
for(i in 1:4) {
  boxplot(x[,i], main=names(iris)[i])
}
```



Solution Exercise 5

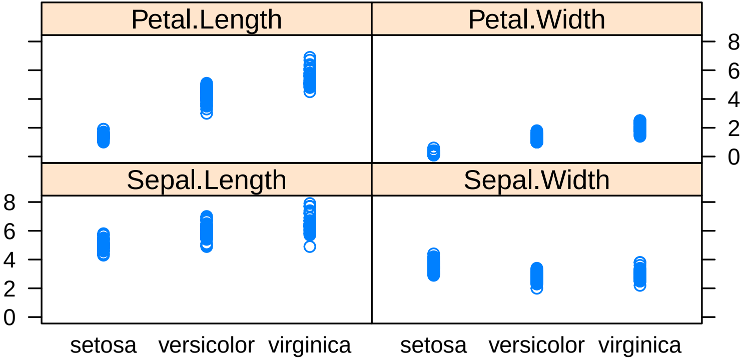
```
library(caret)
plot(y)
```



Solutions - Visualizing ML

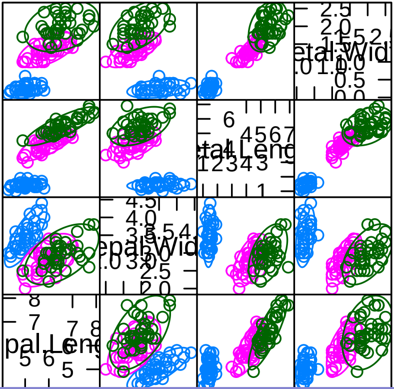
Solution Exercise 6

```
library(caret)
featurePlot(x=x, y=y)
```



Solution Exercise 7

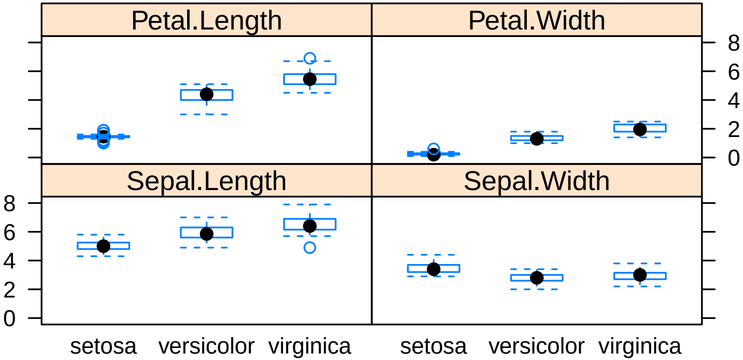
```
# install.packages("ellipse")
library(ellipse)
library(caret)
featurePlot(x=x, y=y,plot="ellipse")
```



Solutions - Visualizing ML

Solution Exercise 8

```
library(caret)
featurePlot(x=x, y=y, plot="box")
```



Solutions - Visualizing ML

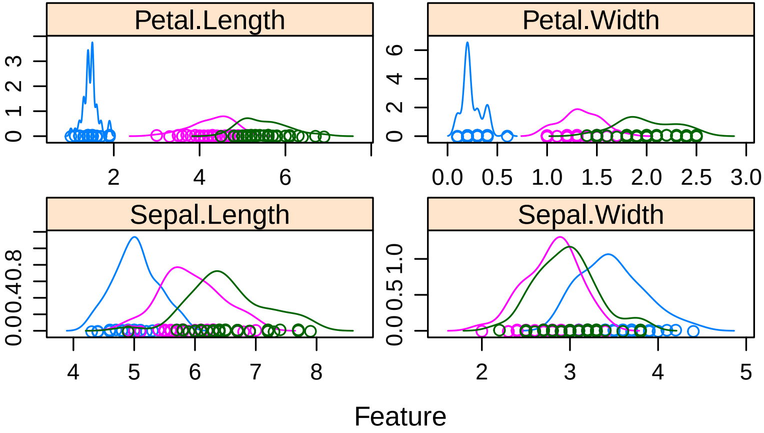
Solution Exercise 9

```
library(caret)
scales <- list(x=list(relation="free"), y=list(relation="free"))
```

Solution Exercise 10

```
library(caret)
scales <- list(x=list(relation="free"), y=list(relation="free"))
featurePlot(x=x, y=y, plot="density", scales=scales)
```

Solution Exercise 10


Petal.Length
Petal.Width
Sepal.Length
Sepal.Width
Feature