Simple Plots in R

SCATTERPLOTS: To draw a scatterplot of \( n \) pairs of observations \((x_i, y_i)\), write your data into two vectors (one for \( x \) and one for \( y \)). The vectors have to have the same length (here \( n \)). The command \( \text{plot}(x,y) \) will generate a scatterplot of \( y \) against \( x \). The plot can be made pretty with a number of arguments that can be used to set titles for the axes, a main title, select the plotting symbols, their size, tick marks, etc.

Example:

\[
\begin{align*}
x & \leftarrow c(1,2,3) \\
y & \leftarrow c(4,3,5) \\
\text{plot}(x,y)
\end{align*}
\]

If your data is contained in a data frame named \texttt{data} with more than two variables all of the same length, the command \( \text{plot(data)} \) will create a matrix of scatterplots of each variable against each other variable.

BOXPLOT: To draw simple or a side-by-side boxplot for two (or more) variables, use the \texttt{boxplot()} command.

- If \( x \) is a vector of values, then \texttt{boxplot(x)} will produce a simple boxplot of the data. Outliers are shown as circles.

- If you have a data frame with several variables named \texttt{data} the command \texttt{boxplot(data)} will produce a side-by-side boxplot of all variables. The boxplot will be labeled with the variable names if they exist.

- If you have a model of the form \( \text{response} \sim \text{predictor} \) where the predictor is a categorical variable and the response is a quantitative variable in a data frame called \texttt{data}, the command \texttt{boxplot(response \sim predictor, data = data)} will draw a side-by-side boxplot of the response values for the different predictor categories.

HISTOGRAMS: For a vector \( x \) of numbers, the command \texttt{hist(x)} will produce a histogram of the data. You can specify the breakpoints between the bins of the histogram as an argument.

Example:

\[
\begin{align*}
x & \leftarrow \text{rnorm}(100,0,1) \\
bins & \leftarrow \text{seq}(-5,5,1) \\
\text{hist}(x, \text{breaks} = bins)
\end{align*}
\]

In every plot, you can label the axes, change the colors, symbols or fonts with the following arguments (this is 	extit{not} an exhaustive list!).

ARGUMENTS FOR PLOTS:

- \texttt{main} labels the main title of the plot (use: \texttt{main = ‘‘Title’’})
• `xlab`, `ylab` label the x- and y-axis, respectively (use `xlab = 'age'`)

• `xlim`, `ylim` determine the range of the x- and y-axis, respectively (use `xlab = c(-10,10)`)

• `type` in a scatterplot, you can “connect the dots” in different ways. Values for `type` are `p` (points), `l` (line), `o` (points and lines), `s` (stair steps), `h` (histogram like vertical lines) (use `type = 'l'`)

• `pch` determines the symbol R uses in a scatterplot to represent observations. For instance, `pch = 1` are open circle, `pch = 2` are open triangles (etc.). For a full list see [http://voteview.com/symbols_pch.htm](http://voteview.com/symbols_pch.htm)

• `cex` scaling coefficient for the symbols in the plot. `cex = 2` will make them twice as large, while `cex = 0.5` will make them half as large.

• `col` determines the color of the symbols. Colors are numbered (1 = black, 2 = red, 3 = green etc.) and named (use `col = 'aquamarine'`). For a complete list, google “R color chart”.

To see many more graphical parameters that can be changed by the user in R, type `?par`. Don’t go crazy on making your graphs pretty, though. Readability is the goal.

**Example:** To illustrate the use of the different graphics parameters, here is a scatterplot of two random normal vectors:

```r
x <- rnorm(100,0,1)
y <- x + rnorm(100,0,0.1)
plot(x,y, xlab = "random predictor", ylab = "random response", main = "Pretty Scatterplot", xlim = c(-3,3), col = "cornflowerblue", pch = 17, cex = 1.5)
```

![Pretty Scatterplot](attachment:image)