Scientific Programming – Data visualization in R

Exercises:
Use basic R plotting commands for most of these exercises; except for exercises 7 using `ggplot` is optional (of course, if you later want to do more exercises with `ggplot`, you can also implement exercises 1 to 6 in `ggplot`!)

1) Scatter plots
Load the GEO dataset `GSE102484` from GEO as we have seen in the exercise sheet for data processing and convert it to an expression matrix

- Produce a (nice) scatterplot of the first two probesets (row) in the matrix and see if there is a correlation between the expression levels of the two probesets. You can additionally compute different correlation coefficients using the function `cor` (see `?cor`).
  As labels for the x- and y-axis you can use the row names (probeset IDs).
- How about the first two samples (columns)? Are the gene expression levels in these two samples correlated? Produce a (nice) plot and compute the Pearson correlation and the Spearman rank correlation.
  Hint: use an appropriate symbol for plotting, e.g., a dot (`pch="."`). Try different symbols and symbol sizes (cex=…) to get a feeling for how best to display such a large number of data points (>50,000 probesets for each sample!). [A good compromise would, for example, be `pch=19, cex=0.1`]
- Produce a pair-wise scatterplot of the first 5 samples (columns) to immediately see whether they all show correlated gene expression levels. (This may take a while. Hint: use a small plot symbol here, e.g., the dot.)

2) Dot charts
We haven't seen this chart type in the lecture, but it might be useful, so give it a try:
- Briefly have a look at `?dotchart`
- Make a simple test: Take the expression values of the first probeset (row) for the first twenty samples (columns) and make dot chart … This is an interesting way, for example, to show differences in runtimes or performances of programs, differences in scores, or like in this case differences in expression levels of the same gene in different samples.

3) Multiple plots on one page and creation of a PDF file
- Create a page with 2 rows and 3 columns, such that in the upper row you have three scatterplots for different pairs of probesets (rows of the expression matrix) and in the lower row histograms of the corresponding differences in expression levels.
- Do the same but save the output into a PDF file.

4) Box and violin plots:
- Load a numeric dataset of your liking (e.g., some gene expression data), or create some random values from different distributions as seen in the lectures. Take about three to five groups of values (e.g., genes, samples, or random distributions) and produce:
  - Different box plots (vertical and horizontal; different colors; with and without notch) using `plot` and `boxplot`. Note: not all options are possible (or easy to realize) with the two commands.
  - Different violin plots (vertical and horizontal; different colors …)

    - For both types of plots: read the documentation or online help and maybe find some examples on the web to see what other parameters can be specified or used.
    - Regarding colors: especially when having many groups, finding a high number of distinct colors which are suitable to be used together is not always straight forward, but R offers many different ways of automatically generating **color palettes** (vectors of colors that can be used with the `col` parameter of `plot` and `boxplot`). Look up a few possibilities and test them. With the plots generated above.

5) Bar plots:

- Take some of the data from the previous exercise and generate different kinds of bar plots.

6) Heat maps:

- Load the GSE124571 dataset from GEO using the GEOquery package.
- Exclude sample 21 (GSM3536990 or “replicate sCJD_2”).
- Find the probes with the 100 lowest variances and produce a heat map … this will not look spectacular (mostly noise), but illustrates how important it is to use the right genes/probes when trying to distinguish between sample classes.
- Read the documentation for the `heatmap` function. Can you use different color palettes? Can you change the algorithms used for clustering rows and columns?
- Try to produce a heat map also using the function `heatmap.2` of the CRAN package “gplots”.
- Read the documentation for `heatmap.2` to find out if it offers more features.

7) ggplot2:

- Use the **examples from the lectures** and display the results stepwise. For example, after declaring the aesthetic, you can get a preliminary plot (before adding a geometry) by just typing the name of the plot object: `p`
- This helps you to better understand the stepwise construction of figures with ggplot2!
- Try to find information to build some other plot types we have already seen with the built-in graphics functions of R (bar plot, box plot, violin plot). Try to create some of these plots and compare their quality with those obtained with the standard functions.
- For example: some nice examples on the many ways in which you can customize violin plots can be found here: http://www.sthda.com/english/wiki/ggplot2-violin-plot-quick-start-guide-r-software-and-data-visualization

8) The R Graph Gallery:

- Plotting in R is a **VAST** topic which we cannot cover completely. But it is good to know what kinds of plots are possible. Have a look at the following website: https://www.r-graph-gallery.com/
- One of the most important issues is to first figure out how to best visualize your data, i.e., what the most expressive plot type is!