Plotting raw data in RStudio

Following are instructions on how to use the function supplied for this project to create plots of your raw data. It assumes you have installed RStudio as described in the document “Setup for RStudio”. See the document “Data management in RStudio” for how to manage your data in data frames.

The ability of R to produce publication-quality pdf plots of statistical data is a major benefit of the software. The graphics engine is based on a software package called ggplot2 written by Hadley Wickham of Rice University. It is an implementation of Wilkinson’s The Grammar of Graphics (2005), which specifies a graphics programming language. The gg in ggplot2 stands for Grammar of Graphics.

The philosophy of ggplot2 is to give the user the flexibility of creating custom plots by combining or layering the plotting components from a general set of components. The advantage of this approach is that you are not restricted to a limited set of predetermined plot types. Instead, you can combine the components in an unlimited way to produce virtually any type of plot. The disadvantage is that it takes more effort to compose your plot. You can use the function provided for this course to create your plots of the raw data without understanding any of the details of ggplot2. Inspect the function source in dp4dsFunctions.R if you are interested in the details.

A few of the types of components that you can combine are

- Aesthetics — the general visual attributes of plot elements. Examples include $x$ and $y$ position of elements, size of elements, shape of elements, and color of elements.

- Geometries — the general type of plot elements. Examples include points, lines, line segments, bars, and text.

- Statistics — the results of computations from the data that are displayed in the plot. Examples include max, min, and mean displayed in a box plot and smooth displayed as a best-fit continuous curve.

- Coordinates — the coordinate system of the plot. Examples are cartesian, polar for pie charts, and map for cartographic displays.

1. To plot raw data:
Suppose your data frame is named asgnData and you have an independent variable called NumSorted and five dependent variables called InsertAsgn, SelectAsgn, HeapAsgn, MergeAsgn, and QuickAsgn. (The following data are fictitious.)

```r
> asgnData
   NumSorted InsertAsgn SelectAsgn HeapAsgn MergeAsgn QuickAsgn
1      200       99999     88888     77777     66666     55555
2      400       99999     88888     77777     66666     55555
3      800       99999     88888     77777     66666     55555
4     1600       99999     88888     77777     66666     55555
5     2400       99999     88888     77777     66666     55555
6     3200       99999     88888     77777     66666     55555
7     4000       99999     88888     77777     66666     55555
8     4800       99999     88888     777777     66666     55555
9     5600       99999     88888     777777     66666     55555
10    6400       999999     88888     777777     66666     55555
```
The function `dp4dsRawPlot()` provided for this project makes plots a family of raw data in a line plot for a given data frame assuming the first column is the independent (x) variable and the remaining columns are the dependent (y) variables. It has five parameters:

- A data frame.
- The label for the x-axis.
- The label for the y-axis.
- The label for the legend.
- A vector of labels of the dependent variables for the legend.

2. **To install the function:**
   Open the file `dp4dsFunctions.R` in RStudio. Click the source button in the upper right corner of the edit pane.

3. **To call the function:**
   Enter the following on the RStudio command line. Do not enter the +. It is the R command line continuation prompt.

   ```r
   > dp4dsRawPlot(asgnData, "Number sorted", "Number of assignments", "Algorithm", + c("Insert sort", "Selection sort", "Heap sort", "Merge sort", "Quick sort"))
   ``

   The following plot is produced.

4. **To save the plot:**
   With the Plot tab selected and the plot showing in the pane, click Export → Save as PDF . . . . A useful feature is the ability to set a custom size for the plot in the PDF size field. This feature will scale the plot without distorting the text. Because the plot is generated with vector graphics, you can zoom in to inspect the details of the plot without encountering any jaggies that would be present if you saved it as a screenshot.

5. **To plot a different set of data:**
   If you want to plot a different set of data, make a new data frame with the data you want to plot. Keep the first variable as the independent variable. See the document “Data management in RStudio”.

---

Plotting raw data in RStudio  CoSc 320, Data Structures