**CONFIRMATORY STRATEGIES – NOTES 2020\_10\_07**

**CREATE TEMPLATES FOR PLOTS**

Many plots have the same structure, so it is helpful to have standard templates that, given some input, save the plots as PDFs. For now, it is sufficient to display the plot in the JupiterNotebook. Each script should be divided into two parts:

* INPUT PART – set the variables of interest (title, values, parameters…)
* PLOTTING PART – create the plot using the input from above

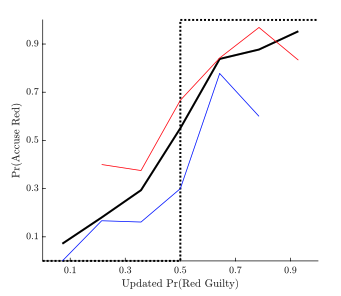
**NOTE: SAVE LINKS TO ONLINE RESOURCES**

For each template indicate the online reference you used (some webpage that contains the list of options for the plot), for example for heatmap you might use this: https://seaborn.pydata.org/generated/seaborn.heatmap.html

These are the most common plots.

**2-D line [to show the relation between two continuous variables]**

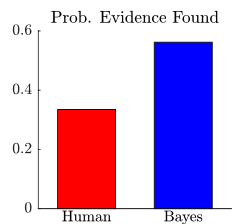
EXAMPLE: Pr(accuse red) as a function of Pr(red)



Input: title (e.g. empty in this figure), x-values (vector), y-values (matrix, one row for every line to plot), colors (one color for every line), line width (one for every line, e.g. 1 or 2), line style (one for every line, e.g. solid or dotted), range x-axis (e.g. 0 to 1), range y-axis (e.g. 0 to 1), tick x-axis (e.g. 0.1:0.2:0.9), tick y-axis (e.g. 0.1:0.2:0.9), label x-axis, label y-axis

**Barplot [to show the comparison between different groups, e.g. human vs model]**

EXAMPLE: Human vs Model prediction for the probability of finding the final evidence

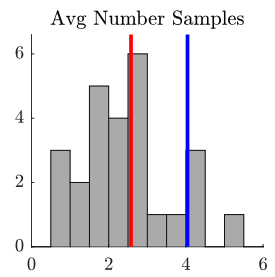


Input: title, y-values (vector, one value for every bar), colors (one color for every bar), range y-axis (e.g. 0 to 0.6), tick y-axis (e.g. 0:0.2:0.6), names x-axis (one name for every bar), label y-axis (e.g. empty in this figure)

NOTE: the template should be flexible to have 2 or more bars next to each other

**Histogram [to show distributions – vertical lines represent comparison between different groups, similar to the values comparisons shown in a barplot]**

EXAMPLE: Distribution of the average number of samples



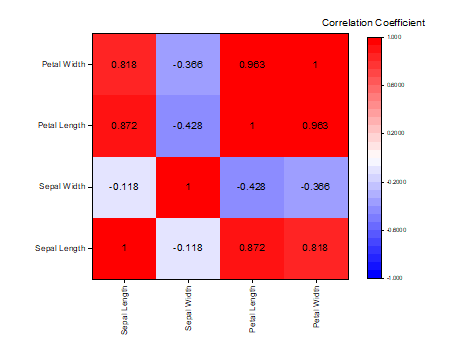
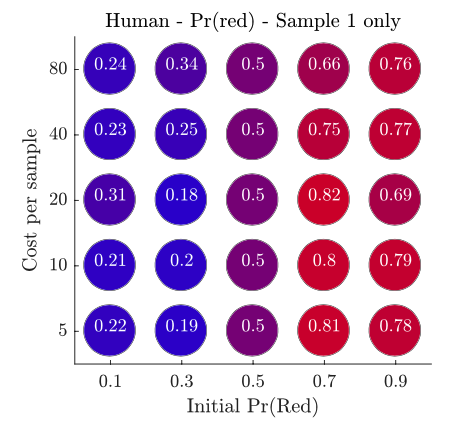
Input: separate different “parts”

* GENERAL: title, range x-axis (e.g. 0 to 6), range y-axis (e.g. 0 to 7), tick x-axis (e.g. 0:2:6), tick y-axis (e.g. 0:2:10), label x-axis, label y-axis
* HISTOGRAM: values-vector (vector, one value per participant), number of bins (integer number, e.g. 10), histogram-color
* VERTICAL LINES: x-values (vector), colors (one color for every line), line width (one for every line, e.g. 1 or 2), line style (one for every line, e.g. solid or dotted)

**Heatmap [to display a 3-dimensional distribution]**

EXAMPLE: Probability of voting RED as a function of prior (x-axis) and cost (y-axis).

On the right, a more classic example of heatmap in Python

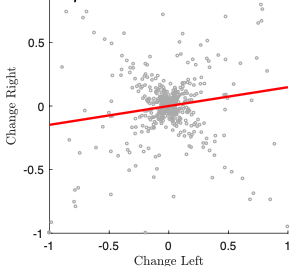
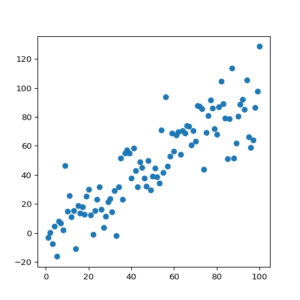


Input:

* GENERAL: title, tick x-axis (e.g. 0.1:0.2:0.9), tick y-axis (e.g. [5,10,20,40,80]), label x-axis, label y-axis
* HEATMAP: values (matrix, each cell is a value to display), color palette (there are default ones, for example 0-1 with 1=black and 0=white, or 0.1-0.9 with 0.1=blue and 0.9=red)

**Scatterplot [to display the relation between two variables]**

EXAMPLE: Pr(confirmatory) vs Avg number of samples  
The figures below are different plots, from other papers, just to give you an idea

Input:

* GENERAL: title, range x-axis, range y-axis, tick x-axis (e.g. 0.1:0.2:0.9), tick y-axis (e.g. 0:1:10), label x-axis, label y-axis
* SCATTERPLOT: values (matrix with two columns=variables, each row is a participant), dot size, dot color
* NOTE: ignore the extra red line that is displayed in the first plot