Report worksheet 7

Joaquín Rapela

March 22, 2023

Question 1: mutual information: one trial length

Function `def H(x):` computes the entropy of the probability distribution given in variable `x`. This distribution can be univariate, e.g., `x=p(y)`, or bivariate, e.g., `x=p(y, z)`. Function `def mi(C):` computes mutual information from the expression $I(X, Y) = H(X) + H(Y) - H(X, Y)$, using the previous function `def H(x):` to calculate the marginal and joint entropies.

Because X and Y are independent random variables, there mutual information is zero. However, we don't get zero, since the code above uses the plugin estimate of mutual information, which is biased above, and returns a positive mutual information estimate.

Question 2: mutual information: multiple trial lengths

Figure 1 shows the results of estimating mutual information (using the plugin method) for different numbers of trials.
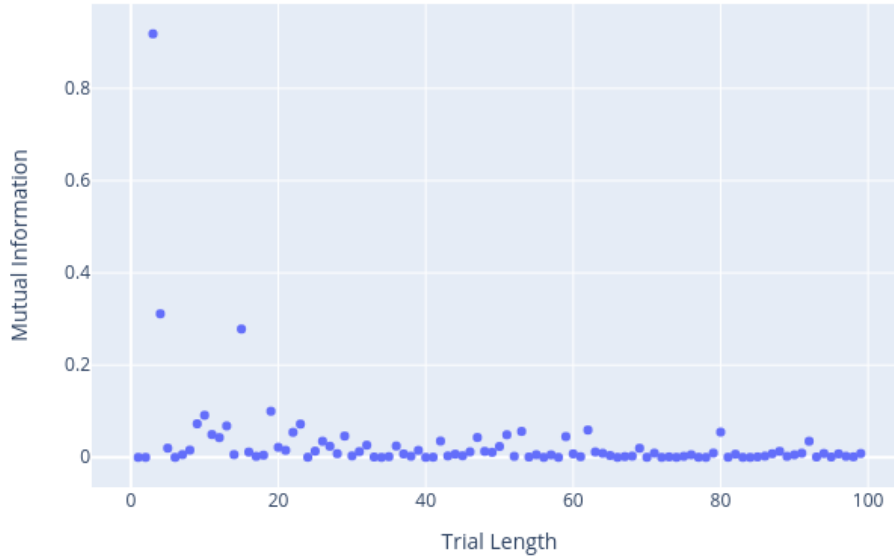
Figure 1: Estimates of mutual information, using the plugin method, for different number of trials. Generated with this script using parameters `--n_repeats 1`. Click on the image to see its interactive version.

Question 3: mutual information: multiple trial lengths

As the number of trials increases, the plugin estimate of mutual information should approach its true value. In agreement with this, Figure 1 shows that, as the number of trials increases, the estimated mutual information approches its true zero value.

For one trial the mutual information estimate is zero, because the probability estimates for one trial are those of a deterministic random variable, and the marginal and joint entropies of deterministic random variables are zero.

We cannot compute mutual information for an experiment with zero trials.

Question 4: computing mutual information: multiple trial lengths, averaged

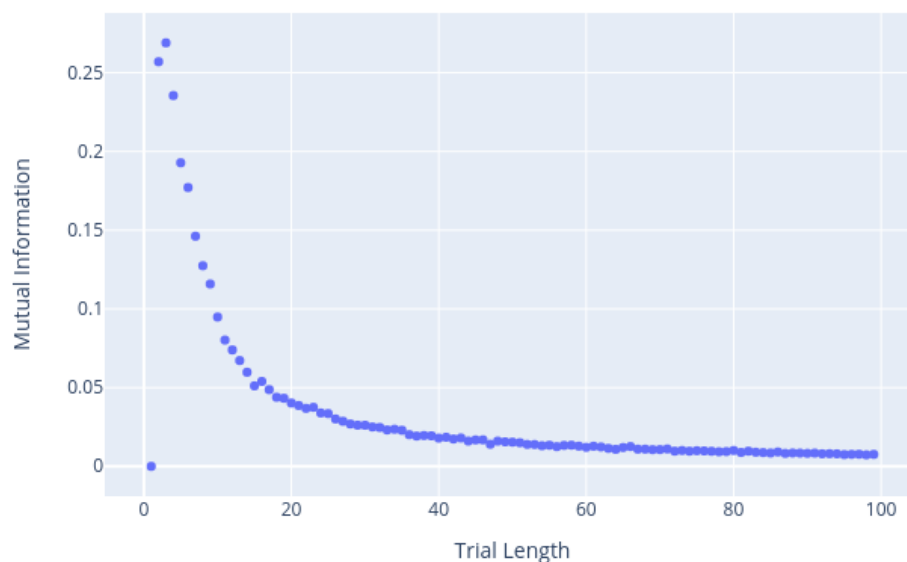Figure 2 shows averaged estimates of mutual information across multiple trial lengths.

Figure 2: Averaged estimates of mutual information, using the plugin method, for different number of trials. Generated with this script using parameters `--n_repeats` 1000. Click on the image to see its interactive version.

Question 5: cross-validated mutual information: one trial length

The code to estimate cross-validated mutual information is the same as the one used above, with the exception that entropies are estimated with cross validation. Two sets of probabilities are calculated, one set is used for the probability component of the entropy, and the other set for the log-progability component. Thus, if both estimates of probability were accurate, the cross-valiated estimates should be identical to the non-cross-validated ones.

We get estimates smaller than zero because, differently from the plugin estimation method, cross-validation estimation is biased below.

Question 6: cross-validated mutual information: multiple trial lengths

Figure 3 shows cross-validated estimates of mutual information for different numbers of trials.
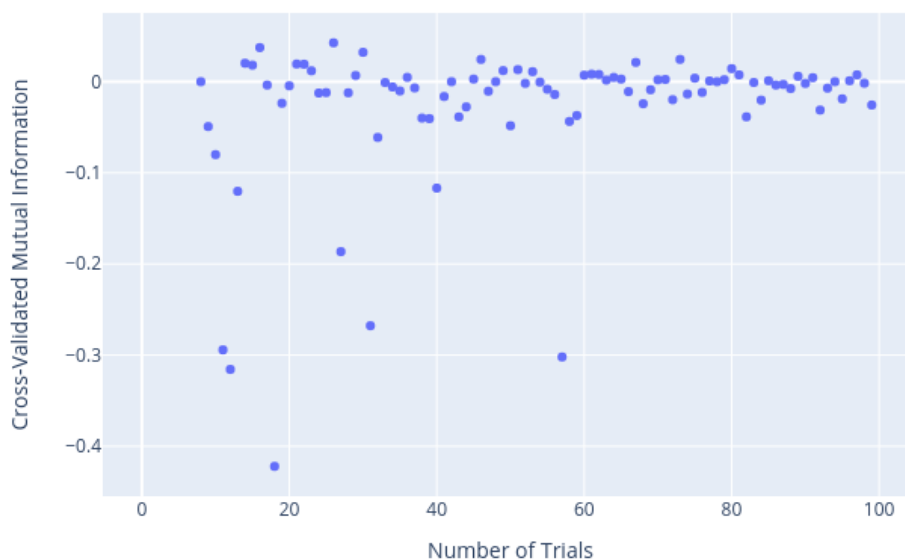
3

Figure 3: Cross-Validated estimates of mutual information for different number of trials. Generated with this script using parameters `--n_repeats 1`. Click on the image to see its interactive version.

Question 7: why do we get error in the function $\log_2$ during the estimation of the cross-validated mutual information?

Because sometimes we compute $\log_2$ of zero values, since the `nonzero` variable is set based on the probability used in the probability part of the entropy (and not based on the probability used to calculated the log-probability part of the entropy).

Question 8: computing mutual information: multiple trial lengths, averaged, cross-validated

Figure 4 shows averaged cross-validated estimates of mutual information across multiple trial lengths.
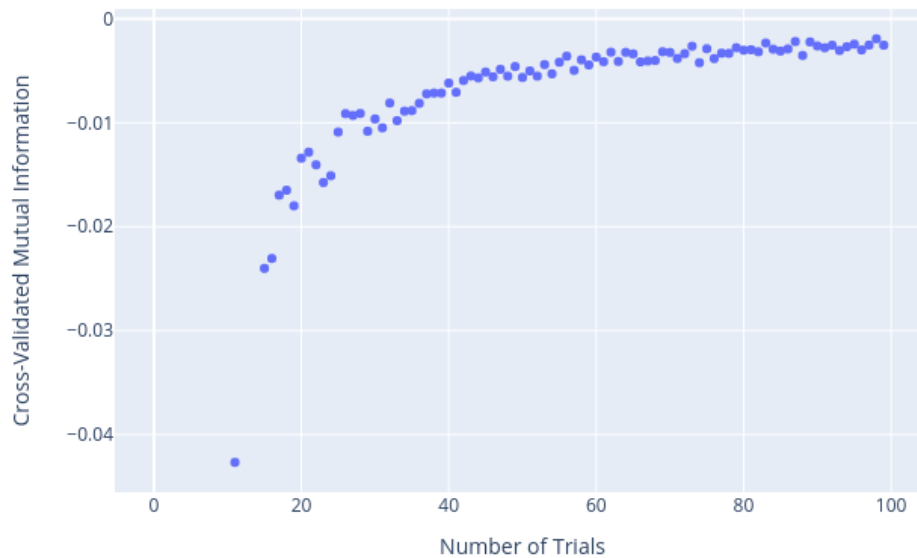
Figure 4: Averaged cross-validated estimates of mutual information for different number of trials. Generated with this script using parameters `--n_repeats 1000`. Click on the image to see its interactive version.

Question 8: why the shape of the mutual information estimates in Fig. 4

Because cross-validated estimates of mutual information are biased below.