BIOINFORMATICS

Vol. 00 no. 00 2015
Pages 1–3

# A simple analytical formula to compute the residual Mutual Information between pairs of data vectors

Jens Kleinjung [1,*], Anthony C.C. Coolen [2]

[1]The Francis Crick Institute, Mill Hill Laboratory, London NW7 1AA, U.K.
[2]Institute for Mathematical and Molecular Biomedicine, Hodgkin Building, room HB 4.5N, Guy's Campus, London SE1 1UL, U.K. and
Department of Mathematics, Strand Building, room S5.26, Strand Campus, London WC2R 2LS, U.K.

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Associate Editor: XXXXXXX

**ABSTRACT**

**Summary:** The Mutual Information of pairs of data vectors, for example sequence alignment positions or gene expression profiles, is a quantitative measure of the interdependence between the data. However, data vectors based on a finite number of samples retain non-zero Mutual Information values even for completely random data, which is referred to as background or residual Mutual Information. Estimates of the residual Mutual Information have so far been obtained through heuristic or numerical approximations. Here we introduce a simple analytical formula for the computation of the residual Mutual Information that yields precise values and does not require the joint probabilities between the vector elements as input.

**Availability and Implementation:** A C program *arMI* is available at http://mathbio.crick.ac.uk/wiki/Software#arMI. Using an input alignment in FASTA format or alternatively an internally created random alignment of specified length and depth, the program computes three types of Mutual information: *(i)* Shannon's Mutual Information between all pairs of alignment columns; *(ii)* the numerical residual Mutual Information by using the same formula on the randomised (shuffled) data; *(iii)* the analytical residual Mutual Information introduced here. The package depends on the GNU Scientific Library, which is used for vector and matrix operations, factorial expressions and random number generation (**?**). Reference alignments and result data are included in the program package in the folder 'tests'. The R environment was used for statistics and plotting (**?**).

**Contact:** Jens.Kleinjung@crick.ac.uk

**Supplementary Material:** A detailed derivation of the analytical formula is given in the Supplementary Material.

Structural and functional constraints have shaped biomolecules and their functions over evolutionary times. This is reflected for example in positional conservation of nucleotide and amino acid sequences across multiple species or in gene expression profiles present in related cell states. These types of correlation patterns can be detected by computing the Mutual Information ($I_{x,y}$) between pairs of data vectors. Shannon's Mutual Information $I_{x,y} = \sum_{x,y \in S} p(x,y) \;\; log\left(\frac{p(x,y)}{p(x)\,p(y)}\right)$ (**?**) scales the joint probability $p(x,y)$ to observe a specific element pair $x$ and $y$ with the marginal probabilities $p(x)$ and $p(y)$. $S$ is a base set of symbols (alphabet) onto which the vector elements are mapped; for biological sequence alignments that corresponds generally to 4 nucleotides (DNA, RNA) or 20 amino acids (proteins) and for expression profiles typically to the number of genes under study. We will use $I_{x,y}$ is this sense throughout the paper.

A critical question for all methods using correlation signals is whether the signal strength of the observed $I_{x,y}$ is above the background or residual Mutual Information $I^r_{x,y}$, which is the Mutual Information between two fully independent alignment positions. The theoretically desirable $I^r_{x,y}$ value of zero is obtained with a hypothetical random sample of infinite size, while in real biological data two error sources lead to non-zero background levels: i) under–sampling due to finite sample size and ii) redundancy among data due to their phylogenetic or functional relatedness, both yielding sampled frequency probabilities $\hat{p}(x)$, $\hat{p}(y)$ and $\hat{p}(x,y)$ differing from expectation values. Heuristic methods have been developed to estimate $I^r_{x,y}$ or to derive covariation values that have been corrected for background signals. The 'average product correlation' evaluates a form of excess $I_{x,y}$ of two alignment positions *versus* the average $I_{x,y}$ over all pairs of alignment positions (**?**). Alternatively, the covariance of alignment positions can be quantified *via* estimation of the sparse inverse covariance (**?**). A term for the expected systematic error of $I_{x,y}$ has been proposed by **?**. A numerically inspired estimator of $I^r_{x,y}$ is the $I_{x,y}$ value obtained from randomised (shuffled) data (**?**). In the following we will use this numerical residual Mutual Information $I^{nr}_{x,y}$ for comparison.

Here we present a simple analytical formula to compute the analytical residual Mutual Information $I^{ar}_{x,y}$ that has been derived from Shannon's formula under the basic assumption that $\hat{p}(x)$ and $\hat{p}(y)$ were statistically independent of each other, which is the essential condition to obtain $I^r_{x,y}$ values (see Supplementary

*to whom correspondence should be addressed

© Oxford University Press 2015.

Material for the detailed derivation):

$$\hat{I}^{ar}_{x,y} = log(N) + \sum_{x \in S} \sum_{y \in S} \hat{p}(x)\,\hat{p}(y) \sum_{n=0}^{N-1} \binom{N-1}{n} \cdot log(1+n) \Big( (\hat{p}(x)\,\hat{p}(y))^n (1-\hat{p}(x)\,\hat{p}(y))^{N-1-n} - \hat{p}(x)^n\,(1-\hat{p}(x))^{N-1-n} - \hat{p}(y)^n\,(1-\hat{p}(y))^{N-1-n} \Big) . \quad (1)$$

Equation 1 depends only on the sampled element probabilities $\hat{p}(x)$ and $\hat{p}(y)$ and on the vector length (or sample size) $N$. The assumption of statistical independence has led to the elimination of the joint probabilities $p(x, y)$ occurring in Shannon's $I_{x,y}$ formula, simplifying the input to the probabilities of the base set symbols. This has favourable practical implications, for example in sequence analysis and design, where $I^r_{x,y}$ can now be controlled by variation of the composition of alignment profiles without the need to actually create these alignments.

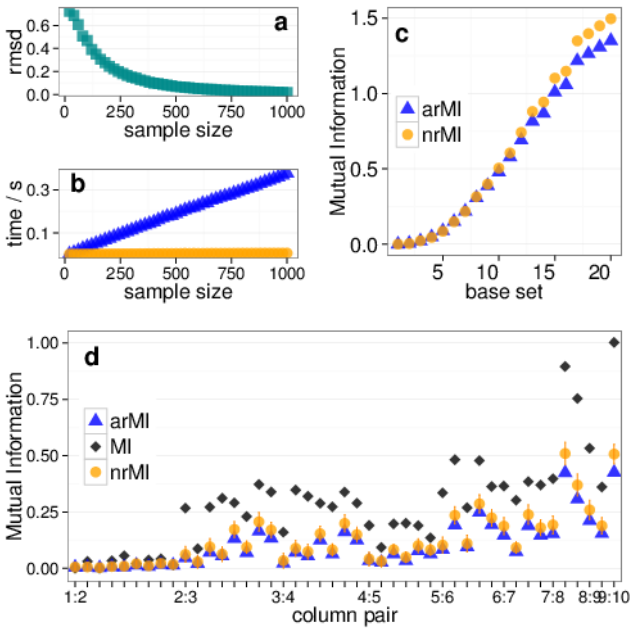



**Fig. 1.** (a) Root mean square deviation between $I^{ar}_{x,y}$ and $I^{nr}_{x,y}$ values for random sequence alignments of length 10 and depth (sample size) 20 to 1000 and (b) the computational time per vector pair ($I^{ar}_{x,y}$: blue triangles, $I^{nr}_{x,y}$: orange circles). (c) $I^{ar}_{x,y}$and $I^{nr}_{x,y}$base set sizes 1 to 20 with even element probabilities. (d) Comparison between $I_{x,y}$, $I^{ar}_{x,y}$and $I^{nr}_{x,y}$ values of an alignment of 100 sequences of the protein Ras (residues 57-66).

A direct comparison between the $I^{ar}_{x,y}$ and $I^{nr}_{x,y}$ measure, unbiased by input data correlation, was obtained from a random sequence alignment of length 10 and depths (sample sizes) between 20 and 1000. Figure 1a shows the root mean square deviation between $I^{ar}_{x,y}$ and $I^{nr}_{x,y}$ over all computed values. It is apparent that the numerical estimator $I^{nr}_{x,y}$ significantly overestimates $I^r_{x,y}$ below sample sizes of about 500. However, the computational time per pair of data vectors is almost unaffected by the sample size for $I^{nr}_{x,y}$ (Figure 1b), while it increases linearly with the sample size for the analytical estimator $I^{ar}_{x,y}$. The absolute value of $I^r_{x,y}$depends on the distribution of the element probabilities in the base set as shown in Figure 1c, where the total probability is evenly spread over varying base set sizes from 1 to 20 elements (sample size 100). The absolute $I^r_{x,y}$ values and also the difference between $I^{ar}_{x,y}$ and $I^{nr}_{x,y}$ increase with the size of the base set.

To illustrate the application of $I^{ar}_{x,y}$ (equation 1) on biological data, a short and gap–less alignment of the switch II region (residues 57–66) of the Ras protein was taken from the Pfam database (**?**). The chosen sample size was 100 to emphasise the differences between $I^{ar}_{x,y}$ and $I^{nr}_{x,y}$. We computed the three described types of $I_{x,y}$ between all column pairs: *(i)* Shannon's $I_{x,y}$ of the original alignments, yielding the biological correlation signal (uncorrected for $I^r_{x,y}$); *(ii)* $I^{nr}_{x,y}$ by application of Shannon's $I_{x,y}$ formula on shuffled alignment columns, where randomisations were repeated 100 times to estimate the mean and variance; *(iii)* $I^{ar}_{x,y}$ computed according to equation 1. Figure 1d shows the resulting $I_{x,y}$ (black diamonds), $I^{nr}_{x,y}$ (orange circles, mean±sd) and $I^{ar}_{x,y}$ (blue triangles) values plotted over the array of all column combinations (1:2, 1:3,...,1:10, 2:3, 2:4,...,9:10) for sample size 100. $I_{x,y}$ values fluctuate depending on the correlation between the particular combinations of alignment columns. arMI yields the analytically correct residual values, while $I^{nr}_{x,y}$ overestimates the background correlation as described above.

In conclusion, the analytical $I^{ar}_{x,y}$ equation 1 is a precise and practical estimator of the residual $I_{x,y}$, in particular for sample sizes below 500, where the usually employed numerical randomisation deviates from appreciably from the expectation values. The results suggest a pragmatic strategy for the computation of $I^r_{x,y}$, which is to use the analytical formula for smaller samples and the numerical approach for larger samples, because that strategy should yield a high precision of the resulting $I^r_{x,y}$ at low computational costs across a wide range of sample sizes. Due to the fact that the joint probabilities $p(x, y)$ have been eliminated from the analytical equation, the $I^{ar}_{x,y}$ measure provides a means to explore $I^r_{x,y}$ by varying vector compositions without explicitly pairing the vector elements.

## 0.1 Time Complexity

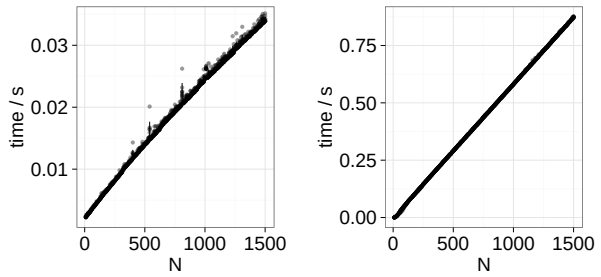
The time complextity of the underlying algorithms has two main components, the combinatorics of the column pair comparisons and the MI calculation. The former, $N * (N-1)/2$ pair comparisons, have a time complexity of $O(N^2)$, which applies to both, $I^{nr}_{x,y}$ and $I^{ar}_{x,y}$ computations.

Disregarding the combinatorial part, we focus on the MI computation of single column pairs. The MI computation is dominated by random shuffling in the case of $I^{nr}_{x,y}$ and by the evaluation of polynomial terms in the case of $I^{ar}_{x,y}$.

Random shuffling was performed using the GSL function 'gsl_ran_shuffle', which is an implementation of the Durstenfeld version of the FisherYates shuffle. The algorithm has time complexity O(N), where $N$ is the number of elements in the set (**?**) or the sample size in our context.

Contrastingly, the computational time spent on the $I^{ar}_{x,y}$ computation is dominated by the polynomials $p(x)p(y)^n$ and $p(x)p(y)^{N-1-n}$, which are evaluated $(N-1)$ times. Therefore,

the time complexity is also O(N), but the actual time spent on the respective subroutines is considerably different, with random shuffling being about 25 times faster than evaluation of polynomial terms at large N (Fig. 2).




(a) nrMI (b) arMI

**Fig. 2.** Computational time for the computation of (a) $I^{nr}_{x,y}$ and (b) $I^{ar}_{x,y}$ for a single column pair of sample size $N$. Each computation is repeated 10 times. The standard deviation is indicated by vertical bars.

## ACKNOWLEDGEMENT

*Funding*: JK acknowledges support by the Francis Crick Institute (U117581331).