Number of inversions in multiset permutations

Janosch Ortmann, Zhaleh Safikhani, Petr Smirnov, Ian Smith

December 2018

1 Introduction

More context: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4314453/ (Kang et al., 2015)

The concordance index (CI) is a non-parametric metric for comparison between two orderings on a set. In practice, this metric can be used to determine whether a candidate biomarker is informative of a clinical outcome, like sensitivity to anti-cancer drugs. The metric is applied two a numeric response vector and a numeric predictor vector; both take real numbers and can be tied. For example, the response vector might be the area-above-dose response curve (AAC) of cancer cell lines to a particular anti-cancer drug (e.g. lapatinib), and the candidate biomarker might be the gene expression of EGFR in TPM from RNA-Seq data. The bioinformatic question is then to compute the predictive value of the candidate biomarker for the response vector, assign significance and quantify effect size, and test for generalizability in other data sets. Concordance index is a metric for testing significance, though it also has an effect size interpretation. Concordance index is also analgoous to the area under the ROC curve if the predictor vector or candidate biomarker is considered as a classifier or regressor for the response vector. Concordance index - in its vanilla form - is also essentially equivalent to (a linear transformation of) Kendall's Tau.

As discussed in Obuchowski (2006), the area under the ROC curve is an obvious metric for identifying predictors for binary response variables. However, many response variables are continuous - for instance IC50 or AAC for drug response data, or survival time for clinical trials. Continuous-valued data can be binarized by imposing a threshold, but this choice is often arbitrary. Obuchowski points out through simulation that binarization creates problems and incorrectly estimates the association between the two variables - so the goal is to use a metric that operates on continuous valued inputs.

Insert more context here - in particular, why are we using concordance index instead of Spearman.

Let the two input vectors (e.g. gold-standard response vector and predictor) be x and y, let the length of each vector be N, and denote particular elements by the subscripts i and j. The vectors x and y refer to the same items - for example, cell lines or patients - and can be thought of as two different functions on those items. The ordering of the items in x and y is arbitrary, but the values of x and y are paired. As in reference https://papers.nips.cc/paper/3375-on-ranking-in-survival-analysis-bounds-on-the-concordance-index.pdf, formally, we define concordance index as:

$$CI(x,y) \equiv \frac{1}{E}\sum_{ij} 1(x_i < x_j, y_i < y_j) \tag{1}$$

CI is defined as the probability that a pair of elements i and j will have the same ordering in x and y. The normalization divides by the total number of pairs, i.e. $C(N,2)$. CI takes values from $[0,1]$, is 0.5 for uncorrelated inputs, 0 for perfectly anticorrelated (e.g. $y=-x$), and 1 for perfectly correlated inputs. CI is non-parametric; it only depends on the relative ordering of x and y, not on their absolute values. CI on the ranks of x and y returns the same value as on x and y.

Conceptually, it is sufficient to calculate CI by comparing all $C(N,2)=\frac{N(N-1)}{2}$ pairs of elements of x and y. This is $O(N^2)$ and unnecessarily expensive. An *inversion* in discrete mathematics is a pair of elements in a permutation that are in the wrong order, and there is considerable literature on analysis of inversions.

In addition to the value of the CI itself, it is necessary to compute the variance or standard error of the calculation both to (1) assign significance or a p-value and (2) determine if the CI of one predictor is statistically greater than that of another - or greater than 0.5 and informative at all. This turns out to be a particularly challenging problem and one that the majority of our efforts have been devoted to solving. The gold standard method for verifying our null model is permutation testing, but this isn't a practical solution. To illustrate this, we attempted to compute CI on 100 drugs in 500 cell lines using the gene expression of each of 20,000 genes as candidate biomarkers and assign p-values to each of the 100 x 20,000 tests. Using one million permutations per problem requires $2x10^{12}$ computations of $O(n \log n)$ for $n = 500$ and is intractable.

Finally, there are four different variations of concordance index that we sought to implement, devise a null model for, and compute variance:

1. Vanilla Concordance Index (CI) - the two vectors are composed of unique elements, i.e. there does not exist i,j such that $x_i = x_j$ or $y_i = y_j$. By ranking the two vectors, they may be considered permutations of 1:n.

2. Concordance Index with ties (CI) - the two vectors may internally have elements with the same value. That is, $\exists i, j | x_i = x_j$. This problem breaks down further into two cases: (1) the tie structure can be represented by one multiset - the pairing of elements is transitive, or (2) $\exists i, j, k | i \cong j, j \cong k, i! \cong k$, where $\cong$ denotes a tie in at least one of the vectors.

3. Modified Concordance Index (mCI) - mCI differs from CI in that we define two thresholds t_x and t_y such that if $|x_i - xj| < t_x$ or $|y_i - y_j| < t_y$, the pair is considered a tie and ignored in the CI calculation. The denominator in the count of inversions is the number of valid pairs, i.e. those with differences in x and y exceeding the thresholds. As with non-transitive tied CI, mCI's tie structure does not form an equivalence relation and lacks transitivity.

4. Kernelized MCI (kmCI) - kernelized MCI assigns a weight to all pairs that is a function of one or both of the differences between the values. Larger differences in x and y can be considered a pair with greater confidence. The choice of kernel is monotone increasing with delta, for instance a sigmoid. mCI can be thought of as a special case of kmCI where the kernel is a heavyside step function. kmCI assigns a positive weight to all pairs that aren't explicitly tied.

At present, we have computed the null distribution for vanilla CI (no ties) and CI with transitive ties.

1.1 Computing Concordance Index

We independently rederived a method for computing CI in $O(N \log(N))$ time based on merge sort and implemented this in fastCI. The procedure is to (1) sort x, (2) put the elements of y in the same order as those of x, and (3) merge sort y, counting how many inversions are made with each successive merge operation. The details are here. In the literature, counting inversions (the third operation) has been implemented in $O(n \log \log n)$ and $O(n\sqrt{n})$ time Chan and Patrascu (2010). As computing CI still requires initially sorting one of the vectors, as the ordering of the items in the two vectors is arbitrary, the run time of CI should still be $O(N \log N)$.

1.2 Variance

2 Practical Notes

2.1 Generating Polynomial

As shown below in the Formal Analysis section, the simplest way to generate the probability distribution on the number of inversions from a set or multiset is to use a generating polynomial or an ordinary generating function. The distribution on the permutations which result in a particular inversion number is represented as coefficients in a polynomial on x; the number (or probability, if normalized) of permutations yielding k inversions is the coefficient of x^k. This allows exact recursive and iterative methods for adding elements by multiplying polynomials. This is equivalent to convolutions on discrete probability distributions.

For case 1 - CI with no ties, the null hypothesis that every permutation is equally likely yields a special case in which the distribution is a product of polynomials with coefficients 1, i.e. of the form:

$$z_n \equiv \sum_{k=0}^{n-1} x^k = 1 + x + x^2 + ... + x^{n-1} \tag{2}$$

We attempted three solutions for multiplying polynomials:

1. `polynom` package - an R library for representing and operating on polynomials. This package turns out to be slow and somewhat inexact, and was not used. Polynomial multiplication by `polynom`

2. Differences on CDFs: for the special case of the z_n polynomials with all 1s, the multiplication of a general polynomial $g(x)$ by z_n is equivalent to the difference of two offset CDFs on the coefficients of $g(x)$. This approach is considerably faster than more general multiplication, but restricted to z_n.

3. Fourier transformation - polynomial multiplication, or convolution, can be accomplished in $O(n)$ time; once transformed, polynomial multiplication is elementwise instead of convolutional. It introduces some precision problems and very slightly complex coefficients. The naive application of FFT for polynomial multiplication would be to take the two factors, transform them, multiply, and inverse transform them. We implemented a method by which the sequence of polynomials to multiply is generated, transformed and multiplied simultaneously, and inverse transformed once to further optimize the process.

Computing the distribution on inversions for case 2 - CI with ties requires dividing polynomials. It turns out that in all cases enumerated below - with the D_n ratios of $\prod_{k=n}^{n+m} z_k$ to $\prod_{k=1}^{m+1} z_k$, for each polynomial in the denominator, there always exists a polynomial in the numerator which is a polynomial multiple of the denominator such that the coefficients are integral and the remainder is 0. Doing any polynomial division - in FFT space or using `polynom` - was prohibitively inaccurate, so we implemented a method to simplify the D_ns to just the product of the quotients, eliminating the division entirely.

Open problems:

1. Numerical precision in the tails of the polynomial expansion for n! ¿ .Machine$double.xmax ($10^{308}$ gives n = 172). Our implementation maintains the coefficients as a probability distribution, such that they sum to 1, but the coefficients at the tails get small rapidly. For CI without ties, there is exactly 1 way to have 0 (or choose(n,2)) inversions and n! total permutations, so the coefficient is $1/n!$. One solution is to put a lower bound on the coefficients, but this will require breaking up the FFT multiplication for sufficiently large values of n.

2. Asymmetry in the distribution - for values of n greater than 15, the coefficients of the distribution - which should be symmetric - develop asymmetries that become larger with n. It is likely that this reflects numerical precision near 1, as the trick for multiplying z_ns uses CDFs. One solution is to compute the left side of the distribution and mirror it. These differences are very small - no greater than 10^{-17}, but inelegant. They may also introduce errors which compound.

3. Intransitive ties (case 2b) turns out to be exceptionally challenging to compute the null distribution for. We have not as yet come up with a closed form to represent complex tie structures. MCI's tie structure is even more complicated than generalized CI with ties, as it cannot be represented by a multiset on either x or y.

4. Null distribution for kmCI.

5. Generalized variance for the CI distributions for any association, not just the null hypothesis.

6. Normal approximation and an examination of the values of n for which the approximation is valid.

3 Formal Analysis

The best way to think about permuting elements with ties between them is as follows: Let the set of distinct elements to be permuted be denoted $E = \{e_1, ..., e_n\}$ and let $a_j \in \mathbb{Z}_{>0}$ denote the multiplicity of element e_j, that is how often it appears. Thus there are $\alpha := \sum_{j=1}^{n} \alpha_j$ elements in total. We denote by $M = \{e_1^{\alpha_1}, ..., e_n^{\alpha_n}\}$ the multiset containing all elements (with ties).

3.1 Exact expressions - explicit formula

In Margolius (2001) we have the following result for *sets*, that is $\alpha_j = 1$ for all j (and hence $\alpha = n$): let $I_n(k)$ denote the number of inversions of S with k inversions then

$$\Phi_n(x) := \sum_{j=1}^{\binom{n}{2}} I_n(x)x^k = \prod_{j=1}^{n} \sum_{k=1}^{j-1} x^k. \tag{3}$$

It turns out that an analogous result can be obtained for multisets. The original reference is to a paper from 1915 – see [Mac15] in Remmel and Wilson (2015) – but it's easier to read in modern notation. Let $\mathrm{inv}(\sigma)$ denote the number of inversions of a permutation of the multiset (set with ties) S. In Remmel and Wilson (2015) the *distribution* of inv is shown to be

$$D_M(x) = \sum_{\sigma \in S_M} x^{\mathrm{inv}(\sigma)} = \begin{bmatrix} \alpha \\ \alpha_1 ... \alpha_n \end{bmatrix}_x = \frac{\alpha!_x}{\alpha_1!_x .. \alpha_n!_x} \tag{4}$$

with the *q-factorial* being defined by

$$m!_x = \prod_{k=1}^{r} \left(1 + x + ... + x^{k-1}\right) \tag{5}$$

(The expression on the right-hand side of (4) is also called the *q-multinomial coefficient*. Observe that, by splitting the sum over S_M according to the number of inversions,

$$D_M(x) = \sum_{k \geq 0} \sum_{\substack{\sigma \in S_M \\ \mathrm{inv}(\sigma)=k}} x^{\mathrm{inv}\,\sigma} = \sum_{k \geq 0} \sum_{\substack{\sigma \in S_M \\ \mathrm{inv}(\sigma)=k}} x^k = \sum_{k \geq 0} I_M(k)x^k \tag{6}$$

where $I_M(k)$ denotes the number of permutations of the multiset M with k inversions. Thus, (4) is the exact multiset analogue of (3).

3.2 Recursive formula

The paper Margolius (2001) also has a recursion formula, expressing $I_n(k)$ in terms of the $I_{n-1}(j)$: in terms of the generating function this reads

$$\Phi_n(x) = \left(\sum_{k=0}^{n-1} x^k \right) \Phi_{n-1}(x). \tag{7}$$

The proof proceeds by looking at permutations of the first $n-1$ elements and then inserting the last element at all possible position. By keeping track of how many extra inversions this insertion introduces, we arrive at (7).

This argument extends rather well to the case with ties: let M be the multiset as described in the introduction and denote by M^- the set obtained from M by removing one occurrence of e_n. That is, if $M = e_1^{\alpha_1}, .., e_n^{\alpha_n}$, then

$$M^- = e_1^{\alpha_1}, e_2^{\alpha_2}, \dots, e_{n-1}^{\alpha_{n-1}}, e_n^{\alpha_n - 1}, \tag{8}$$

and in particular if $|M| = n$ then $|M^-| = n - 1$. We can give the following analogue of (7):

$$D_M(x) = \begin{bmatrix} \alpha \\ \alpha_n \end{bmatrix}_x D_{M^-}(x) = \frac{\alpha!_x}{(\alpha - \alpha_n)!_x \alpha_n!_x} D_{M^-}(x), \tag{9}$$

with $m!_x$ defined in (5).

3.3 Gaussian approximation

In Margolius (2001), asymptotics are also discussed. It seems like there is also a Gaussian approximation result for the inversions in the multiset case, see Conger and Viswanath (2007).

References

TM Chan and M Patrascu. Counting inversions, offline orthogonal range counting, and related problems. *Proceedings of the Twenty-First Annual ACM- . . .* , pages 161–173, 2010. doi: 10.1137/1.9781611973075.15. URL `http://www.siam.org/journals/ojsa.php` `http://dl.acm.org/citation.cfm?id=1873616`.

Mark Conger and D. Viswanath. Normal Approximations for Descents and Inversions of Permutations of Multisets arXiv : math / 0508242v2 [math . PR] 29 Sep 2006. *Journal of Theoretical Probability*, 20(2):1–16, may 2007. ISSN 0894-9840. doi: 10.1007/s10959-007-0070-5. URL `http://link.springer.com/10.1007/s10959-007-0070-5`.

Le Kang, Weijie Chen, Nicholas A Petrick, and Brandon D Gallas. Comparing two correlated C indices with right-censored survival outcome: A one-shot nonparametric approach. *Statistics in Medicine*, 34(4):685–703, feb 2015. ISSN 10970258. doi: 10.1002/sim.6370. URL `http://www.ncbi.nlm.nih.gov/pubmed/25399736` `http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC4314453`.

Barbara H Margolius. Permutations with inversions. *Journal of Integer Sequences*, 4(2), 2001. ISSN 15307638. URL `https://cs.uwaterloo.ca/journals/JIS/VOL4/MARGOLIUS/inversions.pdf`.

Nancy A. Obuchowski. An ROC-type measure of diagnostic accuracy when the gold standard is continuous-scale. *Statistics in Medicine*, 25(3):481–493, feb 2006. ISSN 02776715. doi: 10.1002/sim.2228. URL `http://www.ncbi.nlm.nih.gov/pubmed/16287217` `http://doi.wiley.com/10.1002/sim.2228`.

Jeffrey B. Remmel and Andrew Timothy Wilson. An extension of MacMahon's equidistribution theorem to ordered set partitions. *Journal of Combinatorial Theory. Series A*, 134:242–277, aug 2015. ISSN 10960899. doi: 10.1016/j.jcta.2015.03.012. URL `https://linkinghub.elsevier.com/retrieve/pii/S0097316515000497`.