Genomics 97 (2011) 58–68

Contents lists available at ScienceDirect

Genomics

journal homepage: www.elsevier.com/locate/ygeno

Methods

Ranking analysis of correlation coefficients in gene expressions

Yuan-De Tan

College Of Life Science, Hunan Normal University, Changsha, 410081, PR China

ARTICLE INFO

Article history:
Received 20 February 2010
Accepted 17 September 2010
Available online 1 October 2010

Keywords:
Microarray
Gene expression
Correlation coefficient
Ranking analysis
FDR
Cancers

ABSTRACT

Development of statistical methods has become very necessary for large-scale correlation analysis in the current "omic" data. We propose ranking analysis of correlation coefficients (RAC) based on transforming correlation matrix into correlation vector and conducting a "locally ranking" strategy that significantly reduces computational complexity and load. RAC gives estimation of null correlation distribution and an estimator of false discovery rate (FDR) for finding gene pairs of being correlated in expressions obtained by comparison between the ranked observed correlation coefficients and the ranked estimated ones at a given threshold level. The simulated and real data show that the estimated null correlation distribution is exactly the same with the true one and the FDR estimator works well in various scenarios. By applying our RAC, in the null dataset, no gene pairs were found but, in the human cancer dataset, 837 gene pairs were found to have positively correlated expression variations at FDR$\leq$5%. RAC performs well in multiple conditions (classes), each with 3 or more replicate observations.

A great advance of "omic" technologies has led to an unprecedented development of large-scale data, for example, microarray data, microRNA data, and protein array data. The large-scale omic data let us take a global insight into complex biological procedures, interactions between drugs and proteins, and pathological mechanisms of complex diseases such as diabetes, stroke, heart disease, hypertension, and various cancers. For microarray data, gene expression profiles provide a clue to cluster or classify the functional genes into groups because functional genes in a group possibly have the same or similar expression patterns under various conditions [1–5]. The similar expression patterns may be described by correlated expressions, including coexpressions [6,7] and coregulations of gene expressions [6–8]. The correlated expressions between genes can be measured by Pearson correlation coefficients [9–11]. By using correlation of gene expressions, one can build clusters or networks of functional genes [9–12]. But like differential expressions of genes, there also exist noises in the correlated expressions of genes. In other words, there are many spurious correlated expressions in microarray data due to expression noise. We therefore reasonably believe that the current various gene expression networks based on correlation coefficients might have spurious connections between some genes or spurious correlations, which lead to misclassification of functional genes. Therefore, to test for the correlation coefficients between genes in expressions variation is necessary. Conventionally, one uses correlation analysis to draw a distinction between genes that are coregulated or coexpressed and those that do not have a common expression pattern. However, large-scale data challenge the conventional correlation analysis because a single threshold α, as a probabilistic criterion for determining whether a single null hypothesis is acceptable or not, is not valid for testing a large-scale number of hypotheses. For example, in testing for 10,000 hypotheses, at least 500 hypotheses are expected to be significant by chance at $\alpha = 0.05$. Such results, due to too many false positives, cannot be acceptable in statistics. Although there have been a variety of statistical methods for identification of genes differentially expressed between treatments or conditions, no methods for large-scale correlation analysis have been proposed so far. The main reason is that ranking is indispensable in the large-scale statistical analysis because multiple-test procedures such as Bonferroni (B) procedure and Benjanimini-Hochberg (BH) procedure [13] need to rank a set of p-values while ranking analysis methods such as significance analysis of microarrays (SAM) [14], ranking analysis of microarray data (RAM) [15] need to sort a set of statistics such as t-statistics or modified t-statistics. However, ranking a large two-dimensional correlation matrix would lead to the problem of over-memory and/or over-time (see Discussion section). In this article, we propose a "locally ranking" strategy to greatly reduce complexity of ranking a matrix and use a dissection approach to estimate the null correlation distribution. In addition, we also develop a new approach to estimate false discovery rate (FDR) because the current multiple-testing procedures and ranking analysis methods are not appropriate to our ranking correlation analysis.

1. Methods

1.1. Ranking analysis of correlation coefficients

Let x_{ick} be the kth expression value of gene i under condition c where $i = 1, \ldots, N$ (number of genes detected on arrays), $k = 1, \ldots, M_c$ (number of replicate observations in expressions of gene i under

E-mail address: tanyuande@hotmail.com.

0888-7543/$ – see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.ygeno.2010.09.002

condition c) and $c = 1, \ldots, C$ (number of experimental conditions). Then a model for the kth expression value of gene i under condition c is

$$x_{ick} = \mu_i + \tau_{ic} + e_{ick} \tag{1}$$

where μ_i is the expression expectation of gene i under the null hypothesis; τ_{ic}, effects of condition c on expression variation of gene i; and e_{ick}, the special expression noise of observation k of gene i under condition c. Eq. (1) does not include association between genes. Practically, genes would be correlatively expressed or coexpressed if the conditions have the same or similar regulation effects on their expression variations. If the conditions show up-regulation effects on gene i but down-regulation effect on gene j, and vice versa, then their expressions would be negatively correlated. Therefore, we can use the traditional correlation coefficient

$$\rho_{ij} = \frac{\sum_{c=1}^{C}\sum_{k=1}^{M_c}\dfrac{(x_{ick}-\mu_i)\left(x_{jck}-\mu_j\right)}{CM_c}}{\sqrt{\sigma_i^2\sigma_j^2}} \tag{2}$$

to measure the expression association between genes i and j ($i<j$) where σ_i^2 and σ_i^2 are the expression variances of genes i and j, respectively, in population. According to the model above, the correlation coefficient in Eq. (2) may be dissected as

$$\rho_{ij} = \rho\left(\tau_i\tau_j\right) + \rho\left(\tau_i e_j\right) + \rho\left(e_i\tau_j\right) + \rho\left(e_i e_j\right). \tag{3}$$

The detail derivation of Eq. (3) can be found in Appendix A. It can be seen from Eq. (3) that if the condition effects (τ) do not simultaneously change expressions of genes i and j, that is, $\tau_{ic}=0$ for gene i but $\tau_{jc}\neq 0$ for gene j, or $\tau_{ic}\neq 0$ for gene i but $\tau_{jc}=0$ for gene j, or $\tau_{ic}=0$ and $\tau_{jc}=0$ for both genes, then $\rho_{ij}=\rho(\tau_i e_j)+\rho(e_i e_j)$ or $\rho_{ij}=\rho(e_i\tau_j)+\rho(e_i e_j)$, or $\rho_{ij}=\rho(e_i e_j)$. Therefore, $E(\rho_{ij})=\rho(\tau_i e_j)+\rho(e_i\tau_j)+\rho(e_i e_j)$, which is the expectation of expression correlation between genes i and j under the null hypothesis that genes i and j do not simultaneously respond to the condition effects (τ) in differential expression. In classical correlation analysis, correlation coefficient between a pair of uncorrelated variables is expected to be zero in large samples and hence we test if a single observed correlation coefficient is zero at a given significance level. However, expression noise may not completely be a random and independent variable because microarray experiment is often conducted in small samples, almost all of the correlation coefficients between genes under null hypotheses are significantly unequal to zero but expected to follow a null distribution with mean of zero and variance >0. On the other hand, for expression associations between many thousands of genes in microarray experiments, a single significance test at a given probabilistic level is meaningless. So, to address these two problems, we have to consider another strategy, a ranking analysis strategy. Given a threshold Δ, a pair of genes i^* and j^* ($i^*<j^*$) are interestingly correlated in expressions if and only if

$$\begin{aligned} &R_{i^*j^*}-E\left(\rho_{i^*j^*}\right)>\Delta \quad \text{for} \quad R_{i^*j^*}>0 \quad \textit{and} \quad E\left(\rho_{i^*j^*}\right)>0 \quad \textit{or} \\ &E\left(\rho_{i^*j^*}\right)-R_{i^*j^*}>\Delta \quad \text{for} \quad R_{i^*j^*}<0 \quad \textit{and} \quad E\left(\rho_{i^*j^*}\right)<0, \end{aligned} \tag{4}$$

where R is an observed correlation coefficient, * represents an ordered sequence in which the R or r values are ordered from smallest to largest, i^*j^* is the i^*j^* th gene pair or variable pair in the ordered sequences. Δ is a threshold chosen to classify a set of observed R-values into non-interesting group and interesting group. By changing threshold value, we can obtain a series of non-interesting and interesting groups of gene pairs for their correlated expressions.

1.2. Estimate of the null correlation distribution

$E(\rho_{ij})$ is unknown and hence $E(\rho_{i^*j^*})$ in Eq. (4) is also unknown. To make Eq. (4) work, we need to estimate the null correlation distribution. In Eq. (1), for gene i, the expression expectation μ_i may be estimated by the observed overall mean $\overline{x}_i$ and the condition effect τ_{ic} may be estimated by intra-group mean – overall mean, that is $\tau_{ic} = \overline{x}_{ic} - \overline{x}_i$ and expression noise e_{ick} is estimated by an observation value – intra-group mean, $e_{ick} = x_{ick} - \overline{x}_{ic}$. In addition, σ_i^2 in Eq. (2) is also estimated by $s_i^2 = \sum_{c=1}^{C}\sum_{k=1}^{m_c}\left(x_{ick}-\overline{x}_i\right)^2/(Cm_c-1)$ where m_c is a sample size under condition c. Thus, ρ in Eq. (2) may be estimated by

$$R_{ij} = R\left(\tau_i\tau_j\right) + R\left(\tau_i e_j\right) + R\left(e_i\tau_j\right) + R\left(e_i e_j\right). \tag{5}$$

Derivation of Eq. (5) can be found in Appendix B. Therefore, $E(\rho_{ij})$ may be estimated by

$$r_{ij} = R\left(\tau_i e_j\right) + R\left(e_i\tau_j\right) + R\left(e_i e_j\right). \tag{6}$$

For a single $E(\rho_{ij})$ value, r_{ij} may not be a good estimator due to error fluctuation, but for a distribution, r_{ij} has the same distribution with $E(\rho_{ij})$, so, after ranking them, $r_{i^*j^*}$ indeed is a desirable estimate of $E(\rho_{i^*j^*})$ (see Results).

1.3. Strategy for ranking a correlation matrix

As correlation coefficients between pairs of variables form a two-dimensional matrix, ranking a two-dimensional matrix is more difficult than ranking a one-dimensional vector and this leads to computer memory overflow error when the number of the correlated variables is large. To address this technical difficulty, we propose a "locally ranking" strategy, which consists of five steps:

Step 1. Transform two-dimensional correlation matrix into one-dimensional correlation vector: $R_{ij}\rightarrow R_s$, where s stands for ij, $i<j$. We code $s=1$ for 12, $s=2$ for 13, ..., $s=S$ for $(N-1)N$. Ranking one-dimensional correlation coefficient vector R_s can avoid the memory overflow error. In the next step, we solve the problem of computational speed because a large number of pairs of variables would make computational speed down.

Step 2. Divide the interval between −1 and 1 into G ordered subintervals, (r_{11},r_{12}), (r_{21},r_{22}), $\cdots$, (r_{g1},r_{g2}), $\cdots$, (r_{G1},r_{G2}), which depends on number (N) of variables (genes). A pair of variables is assigned to rank g (or subinterval g) if their correlation coefficient value falls into the subinterval (r_{g1},r_{g2}), $g=1, 2, \ldots, G$. That is, if $R_s\in(r_{g1},r_{g2})$, we then set $R_s=R_{gt}$ where r_{g1} and r_{g2} are lower and upper boundaries of subinterval g, and t is the tth pair in subinterval g, $t=1,2,\ldots,T_g$.

Step 3. Sort the pairs of variables within subinterval $g=1$ by R_{1t} values. Thus, we have a suborder of correlation coefficients, denoted by R_{1t^*}, within subinterval $g=1$. Asterisk (*) represents an order in which element values are ranked from smallest to largest.

Step 4. Repeat step 3 until $g=G$.

Step 5. Sequentially connect the G suborders to form a whole-ordered correlation vector.

Since subintervals $(r_{11},r_{12})<(r_{21},r_{22})<\cdots<(r_{g1},r_{g2})<\cdots<(r_{G1},r_{G2})<\Lambda(r_{G1},r_{G2})$ are also an ordered sequence, after steps 3, 4, and 5, the whole vector is ordered.