Statistics and Its Interface Volume 8 (2015) 507–515

Estimation of gene co-expression from RNA-Seq count data

Alicia T. Specht and Jun Li*

Gene coexpression networks are widely used in understanding gene regulations, inferring gene functions, etc. The most straightforward way of constructing a coexpression network is to connect gene pairs whose expressions are highly correlated under different experimental conditions. Usually, this correlation is measured by the Pearson's correlation coefficient, which, however, does not directly apply to data generated from RNA-Seq technique. RNA-Seq data are non-negative integers which cannot be properly modeled by a Gaussian distribution, and moreover, these counts have mean values that are proportional to the sequencing depths, and thus there are no identically distributed "replicates." Directly normalizing counts by the corresponding sequencing depths and then using Pearson's correlation coefficient can be of low efficiency. We propose a generalization of the Pearson's correlation coefficient called iCC that can be directly applied to RNA-Seq data. On simulation data, iCC shows higher efficiency in distinguishing coexpressed gene pairs from unrelated gene pairs. In a real dataset, iCC generates a coexpression network that appears to more closely agree with experimentally validated networks than other methods. More generally, iCC can be used for calculating the correlation coefficient for any two series of random variables.

Keywords and phrases: Pearson's correlation coefficient, RNA-Seq, Coexpression network, Count data, Robust estimate.

1. INTRODUCTION

Coexpression is the simultaneous expression/silence, or simultaneously high/low expression, of two or more genes. The "guilt-by-association" heuristic has led to the use of Gene Coexpression Networks (GCNs), where genes that coexpress are believed to be associated with a common cellular function ([42]). Well-constructed GCNs are used to help understand molecular mechanisms underlying biological processes and to predict gene functions that are not previously known.

In GCNs, nodes represent genes and an edge between two nodes represents coexpression between a pair of genes. Computational inference of GCNs is based on a set of experiments each measuring the expression of a large set of genes by high-throughput techniques like microarrays. These experiments use samples from different tissues or different conditions, so genes that are coexpressed tend to have high/low expressions in the same experiment simultaneously. Many methods have been proposed for constructing GCNs based on microarray data (e.g. [39, 38, 24, 33, 9]). According to [5], these methods can be classified into four categories: correlation-based methods, probabilistic network-based approaches (mainly Bayesian networks), partial-correlation-based methods, and information-theory-based methods. Correlation-based methods remain the most straightforward among any of these methods. They calculate the (Pearson's or Spearman's rank) correlation coefficient between each pair of genes, and assign them as coexpressed if the coefficient is high enough.

In recent years, RNA-Seq (ultra high-throughput sequencing of transcriptomes) is taking the place of microarrays as the first-choice technology for measuring gene expression in a high-throughput manner. Perceived benefits to using RNA-Seq include efficient discovery of new genes/isoforms and a much larger dynamic range in measuring expression. RNA-Seq measures gene expression by the number of short sequences called "reads" that are mapped to each gene; so this measure is a "count" (nonnegative integer). Different from microarray data, in which gene expression is measured by real-valued numbers that are usually modeled by Gaussian distributions, the count data generated by RNA-Seq are better modeled by Poisson or negative binomial distributions. Moreover, different RNA-Seq experiments generate different total numbers of reads, corresponding to different "sequencing depths," which need to be used to normalize the counts so that expression measures from different experiments are comparable.

The key to building a correlation-based network is to calculate the correlation of expression of a pair of genes. The "count" nature and difference in sequencing depths place difficulties in the calculation of a correlation coefficient based on RNA-Seq data. Current attempts ([20, 22]) re-use methods developed for continuous data. These methods transform/normalize count data and then calculate the Pearson's correlation coefficient of the transformed/normalized data. This will be discussed in detail in Section 2.2. Recently, there have been efforts to better use the information contained in RNA-Seq data to capture

*Corresponding author.

the network structure, although none are correlation-based. These methods include a log-linear graphical model ([3]), a local Poisson graphical model ([4]), and a hierarchical Poisson log-normal model ([17]). Others attempt to work at the exon level by implementing canonical correlation analyses on linear combinations of expression levels at exon positions ([19, 14]). However, when considering the correlation of gene expression, there is still a need to develop statistical methods that handle count data with different sequencing depths more directly so that the information contained in the count data can be used more efficiently.

In this article, we propose a new definition of the correlation coefficient, which can be viewed as an extension of the regular Pearson's correlation coefficient (PCC). It extends the application of PCC to two series of random numbers with arbitrary distributions, discrete or continuous. This definition works on estimation of co-expression of RNA-Seq data, as well as many other situations. On both simulation data and real data, we have shown that our method of estimating correlation coefficients distinguishes coexpressed gene pairs from unrelated gene pairs more efficiently.

This article is arranged as follows. In Section 2, we discuss the limitation of PCC, propose our extended definition, and show how to apply it to RNA-Seq data. In Section 3, we apply our method on simulation data and compare it with two transformation-based methods. In Section 4, we apply these three methods on a real RNA-Seq datasets from *E. coli* and compare their performance. The conclusion is given in Section 5.

2. METHODS

2.1 Commonly-used negative binomial model for RNA-Seq data

Suppose we have data from p RNA-Seq experiments, each measuring the expression levels of n genes. Let x_{ij} be the number of reads mapped to gene i in experiment j, $i = 1, \ldots, p$, and $j = 1, \ldots, m$. It is a nonnegative integer and usually modeled by a negative binomial or Poisson distribution. As the Poisson distribution is a special case of the negative binomial distribution, hereby we use the negative binomial distribution for short. Let d_j be the sequencing depth of experiment j and ν_i be the expression of gene i,

$$X_{ij} \sim \mathrm{NB}(d_j\nu_i, \phi_i), \tag{1}$$

where NB means negative binomial, $d_j\nu_i$ is its mean, and ϕ_i is the dispersion parameter so that $\mathrm{var}(X_{ij}) = d_j\nu_i + \phi_i(d_j\nu_i)^2$. This model has been widely used for RNA-Seq data. Poisson distribution corresponds to $\phi_i = 0$.

In model 1, d_j is usually estimated beforehand using all data (See e.g. [35, 6, 26]) and assumed known. These depths can differ in several or tens of folds from experiment to experiment, even in the same dataset. So it is important to somehow "normalize" the effect of sequencing depths so that the counts are comparable. The unknown parameters in model 1 are ν_i and ϕ_i. Recently, many state-of-the-art methods have been developed to estimate ϕ_i more accurately than simply maximizing the likelihood of each individual gene (e.g. edgeR, DESeq, baySeq, NBPSeq).

2.2 Pearson's correlation coefficient and its limitation

Pearson's correlation coefficient (PCC) is perhaps the most widely used definition of the correlation coefficient. For two random variables X_1 and X_2 with means μ_1 and μ_2 and standard deviations σ_1 and σ_2, it is defined as

$$\rho = \frac{E[(X_1 - \mu_1)(X_2 - \mu_2)]}{\sigma_1\sigma_2}. \tag{2}$$

When this definition is applied on samples, one has m observations of X_1, denoted as $x_{11}, \ldots, x_{1m}$, and m observations of X_2, denoted as $x_{21}, \ldots, x_{2m}$, then

$$\hat{\rho} = \frac{\frac{1}{m}\sum_{j=1}^{m}(x_{1j} - \bar{x}_1)(x_{2j} - \bar{x}_2)}{S_{x_1} \cdot S_{x_2}}, \tag{3}$$

where $S_{x_1} = \sqrt{\frac{1}{m-1}\sum_{j=1}^{m}(x_{1j} - \bar{x}_1)^2}$, $S_{x_2} = \sqrt{\frac{1}{m-1}\sum_{j=1}^{m}(x_{2j} - \bar{x}_2)^2}$, $\bar{x}_1 = \frac{1}{m}\sum_{j=1}^{m} x_{1j}$, and $\bar{x}_2 = \frac{1}{m}\sum_{j=1}^{m} x_{2j}$.

Unfortunately, this definition does not apply to RNA-Seq data. In model 1, although $x_{i1}, \ldots, x_{im}$ are "replicates," in the sense that they all measure the expression of gene i, they are scaled by the sequencing depths and not directly comparable, and thus $\bar{x}_i = \frac{1}{m}\sum_{j=1}^{m} x_{ij}$ does not make sense and formula (3) does not work.

A simple remedy will be "normalizing" x_{ij} by d_j then using the normalized data, say x'_{ij}, for formula (3). We call these methods "normalization-based methods" or "transformation-based methods." The normalization is often done by the following two methods ([18]): (1) $x'_{ij} \leftarrow x_{ij}/d_j$, and (2) $x'_{ij} \leftarrow \sqrt{x_{ij}}/\sqrt{d_j}$. For the first method, x'_{ij} have the same mean but quite different variances. The second method tries to take care of the variance by taking the square root first, which is the "variance stabilization transformation"(VST) of the Poisson distribution (not the VST for the negative binomial distribution, though) and gives $\mathrm{var}(\sqrt{x_{ij}}) \approx 1/4$ when x_{ij} follows a Poisson distribution regardless of the mean. However, after dividing by $\sqrt{d_j}$ to stabilize the mean, x'_{ij} still have different variance. One may propose other ways to normalize the data, such as using the VST of negative binomial distributions, but it is generally impossible to normalize both mean and variance simultaneously as stabilizing one will impact the other.

2.3 A new definition of the correlation coefficient

We want to generalize the definition of PCC so that it works on count data with different means. Being a bit more ambitious, we want the new definition to work for the following more general case: two sequences of random variables $x_{11}, \ldots, x_{1m}$ and $x_{21}, \ldots, x_{2m}$, where each x_{ij} follows a different distribution (it can be an arbitrary distribution, continuous or discrete, but assumed known).

We start from the intuitive meaning of correlation: a positive/negative correlation means that x_{2j} should be on the bigger/smaller side (of its own distribution) when x_{1j} is on the bigger side (of its own distribution). For example, suppose x_{11} is Poisson-distributed with mean 10 and x_{21} is exponentially distributed with mean 100. If x_{11} and x_{21} are highly positively correlated, and we observe $x_{11} = 2$ (much smaller than its mean 10), then x_{21} is likely to be much smaller than its mean 100. Based on this idea, we let

$$p_{ij} = \Pr(X_{ij} < x_{ij}) \tag{4}$$

if X_{ij} is continuous at x_{ij}, and

$$p_{ij} = \Pr(X_{ij} < x_{ij}) + \frac{1}{2}\Pr(X_{ij} = x_{ij}) \tag{5}$$

if X_{ij} is discrete at x_{ij}.

When X_{ij} follows a continuous distribution (Gaussian or not), p_{ij} follows $\text{Uniform}(0, 1)$ distribution exactly. Otherwise, p_{ij} follows $\text{Uniform}(0, 1)$ approximately. Note that we use "mid-p-value" ([2]) in this case so that the approximation is better. As a result, positive/negative correlation means that p_{2j} is likely to be larger/smaller than 0.5 when p_{1j} is larger than 0.5. Taken one step further, since p_{ij}'s have the same distribution for different i's and j's, we can view them as x_{ij}'s and plug them into Equation (3) to calculate the correlation coefficient.

The above definition works for virtually any random variable X_{ij}, although it does not give the same estimate as the regular PCC even when all X_{ij} are Gaussian. This defect can be solved by letting

$$x'_{ij} = \Phi^{-1}(p_{ij}) \tag{6}$$

where Φ is the cumulative density function of the standard Gaussian distribution, and view x'_{ij}'s as x_{ij}'s and plug them into formula (3) to get ρ. Now our definition will agree exactly with the traditional PCC for Gaussian data.

On the above, we assume the distribution of X_{ij} is known. In real applications, like in RNA-Seq data, parameters in the distribution are unknown beforehand and need to be estimated from the data. So our new definition of correlation coefficients will be calculated following four steps: (1) estimate the distributional parameters of X_{ij}, (2) calculate p_{ij} according to Equation 4 or Equation 5, (3) convert p_{ij} to x'_{ij} using Equation 6, and (4) view x'_{ij} as x_{ij} and plug into Equation 3 to calculate $\hat{\rho}$. We call our method Distribution-**i**nversed and Gaussian-transformed **C**orrelation **C**oefficient, or iCC for short.

iCC is always applicable no matter how X_{ij} is distributed, even when the distribution of X_{ij} varies for different i and different j. It is worth noting that although our x'_{ij} is also "transformed" from x_{ij}, it is intrinsically different from the transformation-based methods we discussed in Section 2.2. Firstly, we make use of the distribution of x_{ij}, including mean and other parameters of the distribution, while transformation-based methods use only the distribution family but not the parameters. For example, when $x_{ij} \sim \text{Poisson}(\mu_{ij})$, our transformation needs μ_{ij}, which is usually estimated by using $x_{i1}, \ldots, x_{ij}$ combined, while transformation-based methods use the same square root transformation regardless of the value of μ_{ij}. Secondly, no matter how X_{ij} is distributed, our x'_{ij} follows the same standard Gaussian distribution, while x'_{ij} given by transformation-based methods follow transformation-dependent and often hard-to-describe distributions.

2.4 Robust estimation of parameters in RNA-Seq data

In this section, we focus on using iCC for calculating the correlation coefficient based on RNA-Seq data. The calculation is done by following the four steps described in Section 2.3. Steps 2 to 4 are straightforward, and the only task remaining is to estimate ν_i and ϕ_i in model 1. For simplicity, we use MLE (maximum likelihood estimate) based on counts for each individual gene.

It is now well known that RNA-Seq data usually contain outliers (e.g. [25, 1]), which can lead to failure of MLE and many other estimates. Outliers need to be identified and excluded from the calculation of correlation. To this end, we first calculate MLE using all samples and obtain p_{ij} accordingly. A p_{ij} that is too close to 0 or 1 is likely to be an outlier, so we identify x_{ij} as an outlier if $s_i = \min(p_{ij}, 1 - p_{ij}, 1 \leq j \leq m) < C$, where C is a constant. To determine C, we notice that if there is no outlier, we have $\Pr(s_i < C) = 1 - (1 - 2C)^m$. Thus, for $\alpha \in (0, 1)$, if we let $C = \frac{1}{2}[1 - (1 - \alpha)^{\frac{1}{m}}]$, we will have $\Pr(s_i < C) = \alpha$, which means that the false positive rate is controlled at level α. We use $\alpha = 0.05$ and remove all x_{ij}'s whose $p_{ij} < C$, and then recalculate the MLE. This procedure is iterated until no more outliers are identified.

3. SIMULATION STUDY

3.1 Simulating correlated data

To simulate correlated data from arbitrary distributions, a simple and effective method is the NORTA Algorithm ([13, 31, 32]). Using NORTA, we are able to simulate pairs of genes whose expression are correlated. We simulate pairs of genes (that is, $i = 1, 2$) across 20 experimental conditions ($m = 20$) with different sequencing depths. We assume the count X_{ij} follows a negative binomial distribution as in Equation 1. Sequencing depths $d_1, \ldots, d_{20}$ for different

Table 1. AUC of different methods for NORTA, simulation without outliers

		method	$\rho = .3$	$\rho = .5$	$\rho = .7$	$\rho = -.3$	$\rho = -.5$	$\rho = -.7$
$\phi = 0$	$\nu = 5$	iCC	**.6960** (.0015)	**.8906** (.0012)	**.9840** (.0004)	**.6949** (.0017)	**.8870** (.0012)	**.9838** (.0004)
		DIV	.6072 (.0023)	.7573 (.0028)	.9043 (.0024)	.6024 (.0024)	.7490 (.0032)	.8939 (.0030)
		VST	.6031 (.0024)	.7334 (.0031)	.8678 (.0031)	.5571 (.0027)	.6667 (.0040)	.8019 (.0048)
	$\nu = 20$	iCC	**.7056** (.0017)	**.9034** (.0011)	**.9880** (.0003)	**.7087** (.0018)	**.9020** (.0011)	**.9879** (.0002)
		DIV	.6197 (.0021)	.7893 (.0028)	.9313 (.0020)	.6228 (.0023)	.7877 (.0027)	.9301 (.0020)
		VST	.6162 (.0020)	.7811 (.0028)	.9251 (.0005)	.6125 (.0022)	.7721 (.0029)	.9181 (.0023)
	$\nu = 100$	iCC	**.7113** (.0016)	**.9057** (.0010)	**.9895** (.0003)	**.7110** (.0015)	**.9054** (.0010)	**.9893** (.0003)
		DIV	.6238 (.0024)	.7915 (.0029)	.9368 (.0019)	.6236 (.0022)	.7921 (.0027)	.9361 (.0022)
		VST	.6234 (.0024)	.7909 (.0029)	.9362 (.0004)	.6220 (.0023)	.7898 (.0027)	.9346 (.0022)
$\phi = .25$	$\nu = 5$	iCC	**.6885** (.0019)	**.8800** (.0015)	**.9781** (.0006)	**.6857** (.0021)	**.8733** (.0016)	**.9767** (.0007)
		DIV	.6454 (.0019)	.8229 (.0023)	.9559 (.0011)	.6425 (.0022)	.8177 (.0024)	.9479 (.0015)
		VST	.6451 (.0020)	.8086 (.0023)	.9372 (.0015)	.5916 (.0031)	.7400 (.0042)	.8841 (.0039)
	$\nu = 20$	iCC	**.7055** (.0019)	**.9010** (.0013)	**.9857** (.0004)	**.7058** (.0021)	**.8986** (.0012)	**.9855** (.0003)
		DIV	.6845 (.0017)	.8823 (.0012)	.9822 (.0004)	.6929 (.0018)	.8832 (.0011)	.9824 (.0003)
		VST	.6920 (.0018)	.8864 (.0014)	.9827 (.0004)	.6882 (.0021)	.8800 (.0015)	.9815 (.0004)
	$\nu = 100$	iCC	**.7106** (.0017)	.9049 (.0011)	.9883 (.0003)	**.7101** (.0018)	.9042 (.0011)	.9874 (.0003)
		DIV	.6951 (.0017)	.8917 (.0010)	.9865 (.0003)	.7009 (.0016)	.8953 (.0010)	.9863 (.0003)
		VST	.7098 (.0016)	**.9051** (.0010)	**.9896** (.0003)	.7099 (.0016)	**.9049** (.0011)	**.9892** (.0003)
$\phi = .5$	$\nu = 5$	iCC	**.6888** (.0021)	**.8819** (.0015)	**.9793** (.0005)	**.6825** (.0021)	**.8720** (.0016)	**.9769** (.0007)
		DIV	.6496 (.0017)	.8343 (.0018)	.9634 (.0009)	.6506 (.0019)	.8311 (.0020)	.9536 (.0012)
		VST	.6666 (.0018)	.8406 (.0019)	.9586 (.0010)	.6102 (.0027)	.7779 (.0035)	.9163 (.0027)
	$\nu = 20$	iCC	**.7059** (.0023)	**.9014** (.0014)	**.9864** (.0004)	**.7071** (.0021)	**.8991** (.0014)	**.9862** (.0003)
		DIV	.6764 (.0017)	.8753 (.0012)	.9803 (.0004)	.6900 (.0017)	.8776 (.0011)	.9793 (.0004)
		VST	.6995 (.0017)	.8943 (.0012)	.9857 (.0003)	.6952 (.0019)	.8887 (.0012)	.9844 (.0003)
	$\nu = 100$	iCC	**.7099** (.0019)	**.9045** (.0011)	.9881 (.0003)	**.7108** (.0019)	.9038 (.0012)	.9874 (.0003)
		DIV	.6808 (.0017)	.8780 (.0010)	.9827 (.0003)	.6913 (.0017)	.8835 (.0010)	.9815 (.0004)
		VST	.7089 (.0016)	.9042 (.0009)	**.9894** (.0003)	.7101 (.0016)	**.9047** (.0010)	**.9891** (.0003)

conditions were generated by taking 20 independent values $u_1, ..., u_{20} \sim$ uniform$(-3, 3)$ and then raising 2 to those values: i.e. $d_i = 2^{u_i}$. Gene expression (ν_i) are simulated to be 5, 20, or 100 to represent low, medium, and high expression, respectively. Three different levels of dispersion were used for the distribution of the counts, $\phi = 0$, $\phi = 0.25$, and $\phi = 0.5$, representing data that is not overdispersed (Poisson), moderately overdispersed, and heavily overdispersed. For correlation coefficient (ρ), we simulate seven different values: 0 (no correlation), ± 0.3 (low correlation), ± 0.5 (moderate correlation), and ± 0.7 (high correlation).

We also simulate data with outliers by randomly selecting one of the 20 count values for each gene to be an outlier and setting its expression to be $10\nu_i$, comparing with ν_i for other counts.

3.2 Performance of different methods

For both data without outliers and with outliers, the correlation was estimated using our iCC method, as well as the two normalization based methods described in Section 2.2: divide by the depth (denoted by "DIV" for short), and take the square root and then divide by the square root of the depth (denoted by "VST" for short). The two normalization-based methods are expected to work poorly for data with outliers as both of them do not handle outliers properly. To make them more competitive, we use the function cov.rob() from the R ([34]) package MASS ([40]) to estimate the correlation after the transformation. This R function computes a robust estimate of PCC with a high breakdown point.

A commonly used measure of performance of an estimator is the mean square error (that is, the mean of $(\hat{\rho} - \rho)^2$ across simulations), which, however, is not a fair measure here for comparing different methods: it is completely acceptable that other methods give $\hat{\rho}$ different from ρ, as far as they give different $\hat{\rho}$ for different ρ so that they can differentiate correlated gene pairs ($\rho \neq 0$) from unrelated gene pairs ($\rho = 0$). When constructing gene co-expression networks, one often sets a cutoff for the absolute value of the estimated correlation coefficient; all gene pairs with correlation above the threshold are treated as co-expressed, and other gene pairs are treated as not co-expressed. The sensitivity and specificity can be obtained accordingly. Using a series of cutoffs, one can then plot the ROC (Receiving Operating Characteristic) curve. The area under curve (AUC) serves as a direct measure of performance of different methods, and it works in our comparison. To this end, we simulate 500 uncorrelated gene pairs and 500 correlated gene pairs, estimate correlation for each pair, plot the ROC curve and calculate the AUC. This was repeated 100 times. Table 1 gives the average AUC for different settings of simulation parameters, including dispersion, gene expression, and cor-

Table 2. AUC of different methods for NORTA, simulation with outliers

		method	$\rho = .3$	$\rho = .5$	$\rho = .7$	$\rho = -.3$	$\rho = -.5$	$\rho = -.7$
$\phi = 0$	$\nu = 5$	iCC	**.6729** (.0019)	**.8589** (.0013)	**.9719** (.0006)	**.6699** (.0018)	**.8557** (.0013)	**.9690** (.0008)
		DIV	.5787 (.0019)	.6930 (.0019)	.8253 (.0017)	.5527 (.0022)	.6576 (.0026)	.7887 (.0031)
		VST	.5748 (.0019)	.6922 (.0020)	.8351 (.0015)	.5713 (.0019)	.6865 (.0019)	.8263 (.0020)
	$\nu = 20$	iCC	**.6847** (.0017)	**.8763** (.0012)	**.9812** (.0004)	**.6820** (.0018)	**.8754** (.0011)	**.9813** (.0003)
		DIV	.5712 (.0017)	.6837 (.0020)	.8197 (.0018)	.5672 (.0020)	.6792 (.0017)	.8175 (.0021)
		VST	.5760 (.0018)	.6932 (.0019)	.8344 (.0016)	.5748 (.0019)	.6923 (.0017)	.8331 (.0017)
	$\nu = 100$	iCC	**.6846** (.0017)	**.8799** (.0010)	**.9821** (.0004)	**.6858** (.0017)	**.8796** (.0011)	**.9824** (.0003)
		DIV	.5728 (.0019)	.6870 (.0020)	.8242 (.0023)	.5724 (.0021)	.6852 (.0023)	.8251 (.0022)
		VST	.5738 (.0018)	.6939 (.0018)	.8344 (.0016)	.5738 (.0019)	.6961 (.0021)	.8366 (.0015)
$\phi = .25$	$\nu = 5$	iCC	**.6230** (.0020)	**.7786** (.0021)	**.9120** (.0016)	**.6212** (.0021)	**.7732** (.0022)	**.9061** (.0018)
		DIV	.5919 (.0016)	.7131 (.0016)	.8512 (.0014)	.5599 (.0019)	.6729 (.0023)	.8203 (.0021)
		VST	.5830 (.0018)	.7083 (.0021)	.8485 (.0018)	.5715 (.0024)	.6872 (.0025)	.8275 (.0030)
	$\nu = 20$	iCC	**.6386** (.0021)	**.8013** (.0016)	**.9223** (.0012)	**.6358** (.0018)	**.8005** (.0016)	**.9196** (.0013)
		DIV	.5836 (.0019)	.7189 (.0015)	.8637 (.0011)	.5802 (.0018)	.7085 (.0017)	.8525 (.0013)
		VST	.5839 (.0020)	.7176 (.0017)	.8616 (.0012)	.5826 (.0019)	.7149 (.0017)	.8583 (.0015)
	$\nu = 100$	iCC	**.6370** (.0019)	**.8011** (.0015)	**.9217** (.0012)	**.6349** (.0018)	**.7968** (.0016)	**.9175** (.0011)
		DIV	.5861 (.0016)	.7160 (.0017)	.8650 (.0012)	.5849 (.0019)	.7145 (.0015)	.8598 (.0011)
		VST	.5900 (.0019)	.7211 (.0017)	.8654 (.0013)	.5865 (.0021)	.7233 (.0017)	.8637 (.0012)
$\phi = .5$	$\nu = 5$	iCC	**.6154** (.0023)	**.7665** (.0021)	**.9057** (.0016)	**.6138** (.0022)	**.7612** (.0022)	**.8986** (.0017)
		DIV	.5805 (.0021)	.6956 (.0019)	.8364 (.0016)	.5495 (.0024)	.6512 (.0028)	.7983 (.0027)
		VST	.5785 (.0017)	.6965 (.0017)	.8308 (.0018)	.5635 (.0021)	.6695 (.0026)	.8082 (.0029)
	$\nu = 20$	iCC	**.6290** (.0021)	**.7887** (.0016)	**.9198** (.0010)	**.6263** (.0022)	**.7873** (.0018)	**.9167** (.0012)
		DIV	.5720 (.0021)	.6987 (.0018)	.8496 (.0015)	.5710 (.0025)	.6899 (.0020)	.8383 (.0014)
		VST	.5776 (.0018)	.7061 (.0016)	.8481 (.0012)	.5760 (.0019)	.7026 (.0017)	.8465 (.0013)
	$\nu = 100$	iCC	**.6271** (.0024)	**.7864** (.0020)	**.9162** (.0012)	**.6238** (.0022)	**.7824** (.0016)	**.9122** (.0010)
		DIV	.5756 (.0018)	.6979 (.0019)	.8525 (.0014)	.5770 (.0021)	.6992 (.0020)	.8463 (.0016)
		VST	.5833 (.0017)	.7092 (.0016)	.8541 (.0013)	.5814 (.0019)	.7106 (.0016)	.8545 (.0012)

relation for the 500 correlated pairs (the correlation is always 0 for the 500 uncorrelated pairs). There are no outliers in these simulations. For each setting, the largest AUC, which corresponds to the best performance, is bolded. The standard error of each AUC is given in parenthesis.

It is easy to see that iCC outperforms both transformation-based methods in most simulation scenarios. This difference is usually large when the expression is low ($\nu = 5$). It is worth noting that performance for low-expression genes are very important as people have found that a large proportion of counts in RNA-Seq experiments are small, and also low-expression genes are hard to explore using other experimental techniques. The AUC of all methods increases as the expression of genes increases, and the difference between different methods becomes smaller. In several cases, iCC has smaller AUC than VST, but the difference is small and often statistically insignificant.

Table 2 gives the results when outliers are added in the simulation. It is clear that in all combinations of parameters, iCC has much higher AUC than the transformation-based methods. Again, the difference is largest for low-expression genes.

The significantly better performance of iCC on data without outliers shows that our definition of the correlation coefficient is more suitable to this type of data, and the outperformance of iCC on data with outliers further shows that our way of dealing with outliers is efficient. Note that this does not necessarily mean that the cov.rob() function in MASS package is not good at giving a robust estimate of correlation, as such an estimate often assumes identically distributed (Gaussian) data, which is surely not the case for the transformed data.

In the above computation, we assume that the dispersion parameter is unknown and we estimate it gene by gene simply by using MLE. It is expected that iCC performs even better if we can estimate the dispersion more accurately, say using advanced methods developed in recent years. The performance of transformation-based methods will not change as they do not use the estimated dispersion anyway.

3.3 Sensitivity to distributional assumptions

In order to apply iCC, one must make an assumption about the distribution of the data. For RNA-Seq data, we assume that counts follow a negative binomial distribution, which may not be the case for real data. To determine how heavily our method is affected by this distributional assumption for overdispersed data, we also simulated data by using a Poisson log-normal model [12], while still assuming a negative binomial distribution when applying iCC.

Again, we simulate pairs of genes ($i = 1, 2$) across 20 experimental conditions. First, vectors were independently sampled from normal distributions $z_{1j} \sim \mathrm{N}(\mu_1, \sigma^2)$ and

Table 3. AUC of different methods for Poisson log-normal, simulation without outliers

		method	$\rho = .4$	$\rho = .6$	$\rho = .8$	$\rho = -.4$	$\rho = -.6$	$\rho = -.8$
$\sigma = .1$	$\nu = 5$	iCC	**.7624** (.0026)	**.9175** (.0015)	**.9869** (.0005)	**.7657** (.0025)	**.9220** (.0014)	**.9873** (.0005)
		DIV	.5741 (.0023)	.6704 (.0024)	.7826 (.0027)	.6078 (.0017)	.7123 (.0019)	.8178 (.0017)
		VST	.5796 (.0023)	.6831 (.0026)	.8098 (.0024)	.6158 (.0021)	.7294 (.0025)	.8465 (.0018)
	$\nu = 10$	iCC	**.7456** (.0027)	**.8992** (.0020)	**.9784** (.0009)	**.7447** (.0025)	**.9004** (.0022)	**.9782** (.0008)
		DIV	.5773 (.0020)	.6618 (.0019)	.7736 (.0019)	.5869 (.0019)	.6797 (.0019)	.7909 (.0018)
		VST	.5813 (.0020)	.6683 (.0018)	.7817 (.0018)	.5831 (.0017)	.6784 (.0018)	.7937 (.0018)
	$\nu = 50$	iCC	**.6373** (.0036)	**.7598** (.0044)	**.8609** (.0040)	**.6363** (.0037)	**.7578** (.0044)	**.8598** (.0043)
		DIV	.5440 (.0020)	.5960 (.0027)	.6715 (.0034)	.5448 (.0020)	.5986 (.0024)	.6757 (.0034)
		VST	.5441 (.0020)	.5955 (.0025)	.6706 (.0033)	.5439 (.0018)	.5966 (.0024)	.6723 (.0033)
$\sigma = .2$	$\nu = 5$	iCC	**.6858** (.0032)	**.8272** (.0031)	**.9326** (.0023)	**.6854** (.0031)	**.8323** (.0030)	**.9356** (.0023)
		DIV	.5517 (.0020)	.6224 (.0022)	.7136 (.0027)	.5810 (.0019)	.6609 (.0024)	.7529 (.0024)
		VST	.5532 (.0021)	.6271 (.0023)	.7267 (.0027)	.5838 (.0019)	.6677 (.0027)	.7713 (.0029)
	$\nu = 10$	iCC	**.6457** (.0033)	**.7685** (.0040)	**.8740** (.0036)	**.6462** (.0032)	**.7682** (.0044)	.8732 (.0038)
		DIV	.5435 (.0020)	.5981 (.0025)	.6763 (.0029)	.5538 (.0022)	.6136 (.0025)	.6926 (.0033)
		VST	.5468 (.0021)	.6022 (.0025)	.6814 (.0032)	.5491 (.0021)	.6079 (.0025)	.6907 (.0033)
	$\nu = 50$	iCC	**.5588** (.0032)	**.6220** (.0042)	**.6891** (.0042)	**.5582** (.0028)	**.6200** (.0040)	**.6907** (.0046)
		DIV	.5151 (.0019)	.5340 (.0022)	.5683 (.0023)	.5144 (.0019)	.5382 (.0021)	.5707 (.0028)
		VST	.5155 (.0019)	.5340 (.0021)	.5668 (.0025)	.5148 (.0019)	.5360 (.0022)	.5666 (.0029)

Table 4. AUC of different methods for Poisson log-normal, simulation with outliers

		method	$\rho = .4$	$\rho = .6$	$\rho = .8$	$\rho = -.4$	$\rho = -.6$	$\rho = -.8$
$\sigma = .1$	$\nu = 5$	iCC	**.6964** (.0042)	**.8523** (.0049)	**.9494** (.0034)	**.6909** (.0054)	**.8451** (.0054)	**.9431** (.0041)
		DIV	.6122 (.0017)	.7232 (.0018)	.8318 (.0019)	.5834 (.0019)	.6907 (.0024)	.8047 (.0026)
		VST	.6187 (.0021)	.7361 (.0021)	.8545 (.0018)	.5845 (.0023)	.6957 (.0028)	.8219 (.0028)
	$\nu = 10$	iCC	**.6901** (.0054)	**.8499** (.0037)	**.9518** (.0026)	**.6852** (.0051)	**.8412** (.0046)	**.9482** (.0028)
		DIV	.5955 (.0020)	.7001 (.0020)	.8140 (.0021)	.5845 (.0020)	.6847 (.0024)	.8011 (.0023)
		VST	.5899 (.0018)	.6946 (.0021)	.8148 (.0020)	.5857 (.0017)	.6881 (.0022)	.8077 (.0023)
	$\nu = 50$	iCC	**.6241** (.0074)	**.7392** (.0087)	**.8553** (.0094)	**.6054** (.0078)	**.7297** (.0092)	**.8467** (.0089)
		DIV	.5546 (.0021)	.6201 (.0027)	.7063 (.0035)	.5526 (.0023)	.6174 (.0030)	.7064 (.0038)
		VST	.5535 (.0020)	.6174 (.0026)	.7051 (.0035)	.5533 (.0022)	.6176 (.0030)	.7054 (.0037)
$\sigma = .2$	$\nu = 5$	iCC	**.6489** (.0050)	**.7837** (.0052)	**.8960** (.0046)	**.6406** (.0059)	**.7742** (.0066)	**.8856** (.0057)
		DIV	.5877 (.0020)	.6759 (.0024)	.7765 (.0024)	.5601 (.0019)	.6441 (.0022)	.7447 (.0030)
		VST	.5858 (.0020)	.6792 (.0022)	.7850 (.0023)	.5611 (.0019)	.6454 (.0025)	.7486 (.0028)
	$\nu = 10$	iCC	**.6279** (.0059)	**.7585** (.0065)	**.8669** (.0056)	**.6250** (.0056)	**.7501** (.0067)	**.8633** (.0059)
		DIV	.5632 (.0021)	.6343 (.0027)	.7233 (.0033)	.5530 (.0019)	.6171 (.0027)	.7096 (.0033)
		VST	.5565 (.0022)	.6275 (.0028)	.7195 (.0034)	.5556 (.0019)	.6207 (.0026)	.7117 (.0035)
	$\nu = 50$	iCC	**.5465** (.0057)	**.5955** (.0081)	**.6679** (.0114)	**.5421** (.0064)	**.5971** (.0082)	**.6649** (.0112)
		DIV	.5241 (.0017)	.5500 (.0024)	.5929 (.0030)	.5222 (.0019)	.5464 (.0025)	.5896 (.0030)
		VST	.5232 (.0018)	.5466 (.0023)	.5874 (.0030)	.5221 (.0019)	.5469 (.0024)	.5883 (.0029)

$z_{2j} \sim \mathrm{N}(\mu_2, \sigma^2)$, where values of $\mu_i = \log(5)$, $\log(10)$, and $\log(50)$ were used to represent low, medium, and high expression, respectively, and $\sigma = 0.1$ or 0.2 to represent moderate or heavy dispersion at levels similar to those of the negative binomial based simulations. Then correlated pairs of p-values p_{1j} and p_{2j} were generated by sampling from a multivariate normal distribution with $\mu = \binom{0}{0}$ and $\Sigma = \left(\begin{smallmatrix}1 & \rho\\ \rho & 1\end{smallmatrix}\right)$, and then finding their respective p-values with the cumulative density function of the standard Gaussian distribution. Finally, the expression data was generated by applying to each p_{ij} the inverse cumulative density function for the Poisson distribution, with $\lambda_{ij} = d_j \exp(z_{ij})$. Sequencing depths were generated and outliers were added in the same manner as described in Section 3.1.

Table 3 and Table 4 display the results for Poisson log-normal generated data, without and with outliers, respectively. Due to the additional randomness in the first step in this data generation method (z_{1j} and z_{2j} are independent), the correlation actually displayed in the simulated data will be less than ρ. For this reason we report the results for slightly higher ρ values ($\rho = \pm 0.4, \pm 0.6, \pm 0.8$).

We observe that iCC outperforms DIV and VST in all comparisons. While the AUCs of all methods are lower than what were observed for the negative binomial based simulations, DIV and VST are more heavily affected. When outliers are not included, the performance of DIV and VST is much worse than that of iCC. When outliers are included, the difference becomes less pronounced, with DIV

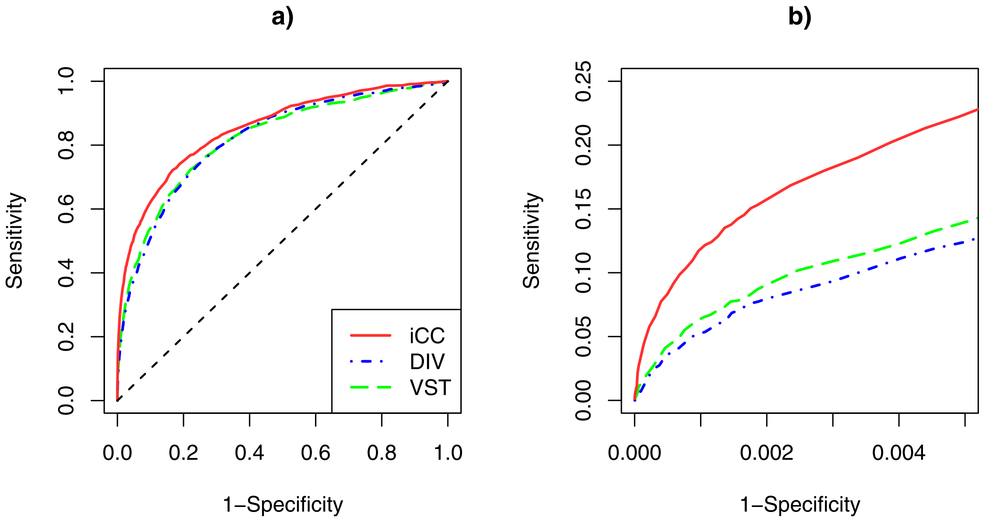

Figure 1. ROC curves of three different methods on the E. coli data. a) The complete ROC curves. Results of different methods are shown by different line types. The black dash line is the $y = x$ line. We see that the red curve is consistently the highest. b) The starting part of the ROC curves. This is usually the part of interest to biologists. We see that the red curve is much higher than the other two curves.

and VST's performance improving and iCC's performance declining slightly. However, iCC still significantly outperforms both. Hence, while the distribution assumption does impact the performance of iCC, it still performs significantly better overall.

4. APPLICATION TO A REAL DATASET

To test the performance of iCC on real data, we use real RNA-Seq data for the organism *Escherichia coli* from a K-12 strain. We collected data from nine different studies ([28, 27, 36, 30, 16, 11, 29, 10, 15]), which includes 50 experiments. The raw reads are downloaded from NCBI's Sequence Read Archive ([41]), mapped to the Ensembl *E. coli* K-12 genome using TopHat ([21], version 2.0.8), and the number of uniquely mapped reads mapped to each gene are counted by using HTSeq ([7]). Fifteen experiments have less than one million of uniquely mapped reads and are removed from further analysis. The 35 experiments remaining in our study contain 1,271,699 to 53,486,413 uniquely mapped reads. Sequencing depths for each experiment were approximated by using total counts from each experiment.

To evaluate the performance of each method, we compare the computed gene co-expression pairs with the most current version (8.5) of the transcriptional regulatory network for *E. coli* available on RegulonDB ([37]). RegulonDB is the primary database for transcriptional regulation in *E. coli*, manually curated from scientific publications. The network consists of transcriptional units, which are genes that, due to their position within the DNA of *E. coli*, all express at once. We treat this network as the ground truth, and gene pairs that are connected in this experiment-based network as truly correlated, and use it as a comparison for our inferred correlations from the 35 experiments.

Genes with the highest expression were then chosen from the set of genes that also appear in the RegulonDB network. Mean expression was calculated by dividing counts by the sequencing depth for each experiment, and then finding the mean for each gene across all experiments. A relative interquartile range (IQR) was calculated by finding the IQR for each gene across all experiments and dividing by mean expression. All genes with both a mean expression of 2×10^{-5} or greater and relative IQR of 0.5 or greater were kept for analysis, resulting in 881 genes.

VST, DIV and iCC were then applied to each possible gene pairing, using the robust methods described in Section 2.4. Thresholds were then determined for each method's adjacency matrix to create GCNs of equal size for comparison by sorting the estimated correlations.

By comparing with the RegulonDB, we plot the ROC curve (on the left panel of Figure 1) for different cutoffs of the absolute values of correlation coefficients. We can see that iCC has the uniformly highest ROC curve among the three methods. The AUCs are 0.8524 for iCC, 0.8198 for VST and 0.8195 for DIV. So overall, iCC performs the best. In RegulonDB, correlated gene pairs are only a very small proportion (0.0163%) of all gene pairs, which means that there are many more true negatives than true positives; so the starting part of the ROC curve, which corresponds to high specificity, is often the critical part. From the right panel of Figure 1, which only plots this part, we see that iCC is about two times as sensitive as the other two methods, which is a huge difference. This indicates that iCC is much more efficient in correctly identifying the correlated gene pairs.

5. CONCLUSION AND DISCUSSIONS

The mainstream techniques that measure transcriptome expression have shifted from microarrays to RNA-Seq. The count nature of RNA-Seq and the difference in sequencing depth makes the regular Pearson's correlation coefficient not directly applicable. Although one can normalize count data to make them look like microarray data, this approach can be inefficient. We propose to use iCC in this case, and we robustify it so that it works for data with outliers. Our iCC seems to provide superior estimation of the correlation of RNA-Seq data, particularly when outliers are present. This is observed in both simulated data and a real dataset.

We have compared the performance of iCC with two transformation-based methods, DIV and VST. Two other frequently used normalizing methods for count data are the Anscombe transform [8], where $x'_{ij} \leftarrow 2\sqrt{x_{ij}/d_j + 3/8}$, and log transform, $x'_{ij} \leftarrow \log(x_{ij}/d_j + 1)$. However, as most count values for RNA-Seq data are large in comparison to the 3/8 adjustment by the Anscombe transformation, the difference in performance between VST and Anscombe will be minimal in practice. On the other hand, the log transformation generates negatively skewed data, over-correcting the positive skewness in the Poisson or negative binomial distribution. These have been confirmed on simulation data (not shown): we observe that the Anscombe transformation gives almost identical AUCs to VST in all of our simulation scenarios, and the log transformation often gives the lowest AUCs among all methods.

Another way to overcome the limitation of Pearson's correlation coefficient would be the use of a nonparametric method to estimate the correlation, such as Spearman's rank correlation coefficient or Kendall's τ. We did perform additional simulations to compare their performance with iCC (data not shown). In simulations with no outliers (in both negative-binomial data and Poisson loglinear data), iCC and the two nonparametric methods performed very similarly for simulated high levels of expression; for low levels of expression, however, iCC outperformed both nonparametric methods significantly. This suggests that the nonparametric methods are less efficient in tough situations. When outliers were included, the performance of iCC and both nonparametric methods were slightly effected, but a similar difference in performance was still observed, suggesting that iCC handles outliers as efficiently as the nonparametric methods.

Many algorithms for network construction rely on accurate calculation of the correlation coefficient. For example, WGCNA ([23]), a widely used package for correlation-based network analysis, estimates the correlation as the very first step, and all following steps use this estimated correlation. Our iCC provides a uniform way for calculating the correlation for different types of data, in a robust manner. Given a model of the data and the estimated parameters of the model, iCC can be calculated, using exactly the same steps regardless of the model of the data. One does not need to find any *ad hoc* method for transforming the data, or explore on identifying the best transformation. iCC also provides a uniform way of detecting the outliers, as the detection is on p_{ij}, whose distribution is (roughly) independent of the distribution of the data.

CONFLICT OF INTEREST STATEMENT

None Declared.

ACKNOWLEDGEMENTS

J. Li is supported by the startup grant from the University of Notre Dame. We thank two anonymous reviewers, a guest editor, and the editor for their thoughtful comments and constructive suggestions that greatly improved the manuscript.

Received 8 February 2015

REFERENCES

[1] AC'T Hoen, Peter, Friedländer, Marc R, Almlöf, Jonas, Sammeth, Michael, Pulyakhina, Irina, Anvar, Seyed Yahya, Laros, Jeroen FJ, Buermans, Henk P J, Karlberg, Olof, Brännvall, Mathias and others. *Reproducibility of high-throughput mRNA and small RNA sequencing across laboratories*, Nature Biotechnology (2013).

[2] Agresti, Alan. *Categorical Data Analysis*, 2nd ed., Wiley, New York (2002). MR1914507

[3] Allen, Genevera I and Liu, Zhandong. *A log-linear graphical model for inferring genetic networks from high-throughput sequencing data*, Bioinformatics and Biomedicine (BIBM), 2012 IEEE International Conference, 1–6 (2012).

[4] Allen, Genevera I and Liu, Zhandong. *A Local Poisson Graphical Model for Inferring Networks From Sequencing Data*, IEEE (2013).

[5] Allen, Jeffrey D, Xie, Yang, Chen, Min, Girard, Luc and Xiao, Guanghua. *Comparing Statistical Methods for Constructing Large Scale Gene Networks*, PLoS ONE 7(1),(2012).

[6] Anders, S and Huber, W. *Differential expression analysis for sequence count data*, Genome Biology 11(1), R106 (2010).

[7] Anders, Simon, Pyl, Paul Theodor and Huber, Wolfgang. *HTSeq–A Python framework to work with high-throughput sequencing data*, bioRxiv (2014).

[8] Anscombe, Francis J. *The transformation of Poisson, binomial and negative-binomial data*, Biometrika 246–254 (1948). MR0028556

[9] Basso, Katia, Margolin, Adam A, Stolovitzky, Gustavo, Klein, Ulf, Dallas-Favera, Riccardo and Califano, Andrea. *Reverse engineering of regulatory networks in human B cells*, Nature Genetics 37(4), 2382–390 (2005).

[10] BioProject PRJNA224571, BioProject. *Escherichia coli K-12 Transcriptome or Gene Expression*, NCBI (2013). Unpublished raw data

[11] BioProject PRJNA75069, BioProject. *Comparison of non-coding transcripts from E. coli and Salmonella grown under similar conditions*, NCBI (2011). Unpublished raw data

[12] Bulmer, M G. *On fitting the Poisson lognormal distribution to species-abundance data*, Biometrics 101-110 (1974).

[13] Cario, Marne C and Nelson, Barry L. *Modeling and generating random vectors with arbitrary marginal distributions and correlation matrix*, Tech. rep., Citeseer (1997).

[14] Coleman, Jacob, Replogle, Joseph, Chandler, Gabriel and Hardin, Johanna. *Resistant Sparse Multiple Canonical Correlation*, arXiv preprint arXiv:1410.3355 (2014).

[15] Dr. Aswin Sai Narain Laboratory, SRX367603. *ChIP-seq investigation of H-NS of Escherichia coli growth in the presence and absence of an antibiotic “Bicyclomycin”, a potential inhibitor of transcription termination factor rho*, NCBI (2014). Unpublished raw data

[16] Fleckenstein, J M, and Qadri, F and Rasko, D R. *Comparative genome analysis of enterotoxigenic E. coli strains isolates from infections of different clinical severity*, NCBI (2014). Unpublished raw data

[17] Gallopin, Mélina, Rau, Andrea and Jaffrézic, Florence. *A Hierarchical Poisson Log-Normal Model for Network Inference from RNA Sequencing Data*, PloS one 8(10), (2013).

[18] Giorgi, Federico M, Del Fabbro, Cristian and Licausi, Francesco. *Comparative study of RNA-seq-and Microarray-derived coexpression networks in Arabidopsis thaliana*, Bioinformatics 29(6), 717–724 (2013).

[19] Hong, Shengjun, Chen, Xiangning, Jin, Li and Xiong, Momiao. *Canonical correlation analysis for RNA-seq co-expression networks*, Nucleic Acids Research 41(08), (2013).

[20] Iancu, Ovidiu D, Kawane, Sunita, Bottomly, Daniel, Searles, Robert, Hitzemann, Robert and McWeeney, Shannon. *Utilizing RNA-Seq data for de novo coexpression network inference*, Bioinformatics 28(12), 1592–1597 (2012).

[21] Kim, Daehwan, Pertea, Geo, Trapnell, Cole, Pimentel, Harold, Kelley, Ryan and Salzberg, Steven L. *TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions*, Genome Biology 14(4), R36 (2013).

[22] Kim, Hyeongmin, Lee, Taeheon, Park, WonCheoul, Lee, Jin Woo, Kim, Jaemin, Lee, Bo-Young, Ahn, Hyeonju, Moon, Sunjin, Cho, Seoae, Do, Kyoung-Tag and others. *Peeling Back the Evolutionary Layers of Molecular Mechanisms Responsive to Exercise-Stress in the Skeletal Muscle of the Racing Horse*, DNA research 20(3), 287–298 (2013).

[23] Langfelder, Peter and Horvath, Steve. *WGCNA: an R package for weighted correlation network analysis*, BMC Bioinformatics 9(1), 559 (2008).

[24] Lee, Homin K, Hsu, Amy K, Sajdak, Jon, Qin, Jie and Pavlidis, Paul. *Coexpression analysis of human genes across many microarray data sets*, Genome Research 14(6), 1085–1094 (2004).

[25] Li, J and Tibshirani, R. *Finding consistent patterns: a nonparametric approach for identifying differential expression in RNA-Seq data*, Statistical Methods in Medical Research 22(5), 519–536 (2013). MR3190673

[26] Li, J, Witten, D M, Johnstone, I and Tibshirani, R. *Normalization, testing, and false discovery rate estimation for RNA-sequencing data*, Biostatistics 13(3), 523–538 (2012).

[27] Li, Shan, Dong, Xia and Su, Zhengchang. *Directional RNA-seq reveals highly complex condition-dependent transcriptomes in E. coli K12 through accurate full-length transcripts assembling*, BMC Genomics 14(1), 520 (2013).

[28] Liu, Wei, Dong, Shi Lei, Xu, Fei, Wang, Xue Qin, Withers, T Ryan, Hongwei, D Yu and Wang, Xin. *Effect of Intracellular Expression of Antimicrobial Peptide LL-37 on Growth of Escherichia coli Strain TOP10 under Aerobic and Anaerobic Conditions*, Antimicrobial agents and chemotherapy 57(10), 4707–4716 (2013).

[29] Max F Perutz Laboratories, SRX326842. *Wildtype Input RNA from E.coli*, NCBI (2013). Unpublished raw data

[30] McClure, Ryan, Balasubramanian, Divya, Sun, Yan, Bobrovskyy, Maksym, Sumby, Paul, Genco, Caroline A, Vanderpool, Carin K and Tjaden, Brian. *Computational analysis of bacterial RNA-Seq data*, Nucleic Acids Research 41(14), (2013).

[31] Niaki, Seyed Taghi Akhavan and Abbasi, Babak. *Generating correlation matrices for normal random vectors in NORTA algorithm using artificial neural networks*, Journal of Uncertain Systems 2(3), 192–201 (2008).

[32] Niavarani, M R and Smith, Alan J R. *Modeling and Generating Multi-Variate-Attribute Random Vectors Using a New Simulation Method Combined with NORTA Algorithm*, Journal of Uncertain Systems 7(2), 83–91 (2013).

[33] Persson, Staffan, Wei, Hairong, Milne, Jennifer, Page, Grier P and Somerville, Christopher R. *Identification of genes required for cellulose synthesis by regression analysis of public microarray data sets*, Proceedings of the National Academy of Sciences of the United States of America 102(24), 8633–8638 (2005).

[34] R Core Team. *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria (2014). URL http://www.R-project.org/

[35] Robinson, M D, McCarthy, D J and Smyth, G K. *edgeR: a Bioconductor package for differential expression analysis of digital gene expression data*, Bioinformatics 26(1), 139–40 (2010).

[36] Sahl, Jason W and Rasko, David A. *Analysis of global transcriptional profiles of enterotoxigenic Escherichia coli isolate E24377A*, Infection and Immunity 80(3), 1232–1242 (2012).

[37] Salgado, Heladia, Peralta-Gil, Martin, Gama-Castro, Socorro, Santos-Zavaleta, Alberto, Muñiz-Rascado, Luis, García-Sotelo, Jair S, Weiss, Verena, Solano-Lira, Hilda, Martínez-Flores, Irma, Medina-Rivera, Alejandra and others. *RegulonDB v8. 0: omics data sets, evolutionary conservation, regulatory phrases, cross-validated gold standards and more*, Nucleic Acids Research 41(D1), D203–213 (2013).

[38] Schäfer, Juliane and Strimmer, Korbinian. *An empirical Bayes approach to inferring large-scale gene association networks*, Bioinformatics 21(6), 754–764 (2005).

[39] Stuart, Joshua M, Segal, Eran, Koller, Daphne and Kim, Stuart K. *A gene-coexpression network for global discovery of conserved genetic modules*, Science 302(5643), 249–255 (2003).

[40] Venables, W N and Ripley, B D. *Modern Applied Statistics with S*, Springer, 4th ed. (2002).

[41] Wheeler, David L, Barrett, Tanya, Benson, Dennis A, Bryant, Stephen H, Canese, Kathi, Chetvernin, Vyacheslav, Church, Deanna M, DiCuccio, Michael, Edgar, Ron, Federhen, Scott, others. *Database resources of the national center for biotechnology information*, Nucleic Acids Research 35(suppl 1), D5–D12 (2007).

[42] Wolfe, Cecily, Kohane, Isaac and Butte, Atul. *Systematic survey reveals general applicability of “guilt-by-association” within gene coexpression networks*, BMC Bioinformatics 6(1), 227 (2005).

Alicia T. Specht
153 Hurley Hall, Notre Dame
IN 46556
USA
E-mail address: aspecht2@nd.edu

Jun Li
153 Hurley Hall, Notre Dame
IN 46556
USA
E-mail address: jun.li@nd.edu