

The covariance shift (C-SHIFT) algorithm for normalizing biological data

Evgenia Chunikhina^{*†}, Paul Logan[‡], Yevgeniy Kovchegov[§], Anatoly Yambartsev^{*}, Debashis Mondal[‡], Andrey Morgun[¶]

^{*}IME, University of São Paulo, São Paulo, Brazil Email: chunikhe.vazhno@gmail.com

[†]Computer Science, Willamette University, Salem, OR, USA

[‡]Statistics Department, Oregon State University, Corvallis, OR, USA

[§]Department of Mathematics, Oregon State University, Corvallis, OR, USA

[¶]College of Pharmacy, Oregon State University, Corvallis, OR, USA

Abstract—Omics technologies are powerful tools for analyzing patterns in gene expression data for thousands of genes. Due to a number of systematic variations in experiments, the raw gene expression data is often obfuscated by undesirable technical noises. Various normalization techniques were designed in an attempt to remove these non-biological errors prior to any statistical analysis. One of the reasons for normalizing data is the need for recovering the covariance matrix used in gene network analysis. In this paper, we introduce a novel normalization technique, called the covariance shift (C-SHIFT) method. This normalization algorithm uses optimization techniques together with the blessing of dimensionality philosophy and energy minimization hypothesis for covariance matrix recovery under additive noise (in biology, known as the bias). Thus, it is perfectly suited for the analysis of logarithmic gene expression data. Numerical experiments on synthetic data demonstrate the method's advantage over the classical normalization techniques. Namely, the comparison is made with rank, quantile, cyclic LOESS (locally estimated scatterplot smoothing), and MAD (median absolute deviation) normalization methods.

Gene expression analysis plays an important role in genomic research. Several omics technologies such as RNAseq and microarrays allow for the collection of massive amounts of simultaneous measurements of gene expression levels of thousands to tens of thousands of genes. Analyzing different patterns of gene expressions helps to gain insight into complex biological phenomena such as development, aging, onset and progression of diseases, and cellular response/reaction to drugs/treatments. Although new technologies are constantly developing, it is well known that all of them generate some technical noise which affects the measured gene expression levels [8, 20]. To extract accurate biological information it becomes necessary to normalize

the data to filter out/compensate for these non-biological noises/errors. Normalization is a crucial pre-processing step in the gene expression data analysis. The gene expression data will vary significantly after different normalization methods. Thus, the results of further data analysis (e.g. gene expression network) will be critically dependent on a choice of a normalization technique. A variety of normalization procedures have been used on gene expression data sets. See [3, 4, 12, 13, 15, 17, 19] and reference therein for a review and comparison of current normalization strategies. In this paper we develop a novel normalization technique, called the covariance shift (C-SHIFT) method, and compare it to the following well known normalization methods used in large scale data analysis: rank, quantile, cyclic LOESS (locally estimated scatterplot smoothing), and MAD (median absolute deviation). See [1, 4, 14, 15] and references therein for more details on the above listed normalization methods.

Consider a situation where the gene expression data is subjected to multiplicative noise (aka bias). Specifically, let $X_n^{(i)}$ be the true gene expression, where subscript index n stands for the n -th gene in the network and the superscript index i stands for the i -th measurement. The observed gene expression, denoted by $\tilde{X}_n^{(i)}$, is different from $X_n^{(i)}$ due to all gene expressions in the i -th measurement being distorted by i.i.d. multiplicative noise variable $W^{(i)}$, i.e.,

$$\tilde{X}_n^{(i)} = W^{(i)} X_n^{(i)}. \quad (1)$$

Here, both the observed and the true gene expressions are positive, i.e., $X_n^{(i)} > 0$ and $W^{(i)} > 0$. The random variables $X_n^{(i)}$ are independent of the variable $W^{(i)}$.

In biology, the multiplicative noise $W^{(i)}$ is referred to as the bias. The bias is prompted by random

events causing an error in the measurement of the total amount of RNA. Such random events are often related to different levels of tissue preservation in different sample that leads to variability of RNA degradation. Consequently, this leads to an RNA detection problem. Additionally, there are other technical reasons for an error in the measurement of the total amount of RNA in a given sample that may lead to a bias in (1). All other noise (e.g. misreading parts of RNA) goes into the variable $X_n^{(i)}$.

The multiplicative noise in (1) implies the corresponding additive noise (bias) in the logarithmic gene expression data:

$$\tilde{Y}_n^{(i)} = Y_n^{(i)} + V^{(i)}, \quad (2)$$

where we let $\tilde{Y}_n^{(i)} := \log \tilde{X}_n^{(i)}$, $Y_n^{(i)} := \log X_n^{(i)}$, and $V^{(i)} := \log W^{(i)}$.

The bias, whether multiplicative as in (1) or additive as in (2), causes the correlations to be shifted away from -1 . Indeed, since

$$\begin{aligned} Cov(\tilde{X}_n, \tilde{X}_m) &= E[W^2]Cov(X_n, X_m) \\ &\quad + Var(W)E[X_n]E[X_m] \end{aligned} \quad (3)$$

the correlation of observed gene expressions

$$\begin{aligned} corr(\tilde{X}_n, \tilde{X}_m) &= \frac{Cov(X_n, X_m) + \nu E[X_n]E[X_m]}{\sqrt{(Var(X_n) + \nu(E[X_n])^2)(Var(X_m) + \nu(E[X_m])^2)}}, \end{aligned} \quad (4)$$

where $\nu := \frac{Var(W)}{E[W^2]} > 0$. Notice that if $Cov(X_n, X_m)$ is negative, by adding positive multiples of $\nu > 0$ in the numerator and the denominator as in (4), we arrive at $corr(\tilde{X}_n, \tilde{X}_m) > corr(X_n, X_m)$. In other words, all negative correlations $corr(X_n, X_m)$ will either turn into positive, or negative of smaller magnitude in the observed variables \tilde{X}_n . While the multiplicative bias $W^{(i)} \approx 1$ (similarly, the additive bias $V^{(i)} \approx 0$) may not appear critical, they are known to cause significant problems in gene correlation structure analyses. Specifically, this phenomenon is known to cause the disappearance of the large magnitude negative correlations in the observed biological data, \tilde{X}_n , which hampers the ability to perform certain types of statistical data analysis, such as the false discovery rate (FDR) method.

The same is observed for the logarithmic data (2). Similarly to (3), the independent additive noise in (2) implies an increase of covariance,

$$Cov(\tilde{Y}_n, \tilde{Y}_m) = Cov(Y_n, Y_m) + \omega, \quad (5)$$

where $\omega = Var(V) > 0$. Consequently, the correlations in the logarithmic data are

$$corr(\tilde{Y}_n, \tilde{Y}_m) = \frac{Cov(Y_n, Y_m) + \omega}{\sqrt{(Var(Y_n) + \omega)(Var(Y_m) + \omega)}}. \quad (6)$$

Here too, if $Cov(Y_n, Y_m)$ is negative, by adding $\omega > 0$ in the numerator and the denominator, we obtain

$$corr(\tilde{Y}_n, \tilde{Y}_m) > corr(Y_n, Y_m).$$

Thus, the phenomenon of disappearance of the large magnitude negative correlations also applies to the logarithmic data \tilde{Y}_n .

Denote by \widehat{Cov} the empirical covariances taken over N subjects for each of $\binom{M}{2}$ pairs of genes. Similarly, let \widehat{Var} denote the empirical variance. Then, equation (2) yields the observed empirical covariance

$$\widehat{Cov}(\tilde{Y}_n, \tilde{Y}_m) = \widehat{Cov}(Y_n, Y_m) - \hat{a}_n - \hat{a}_m + \hat{\omega} \quad (7)$$

for all pairs of gene indices n and m , where $\hat{a}_n = -\widehat{Cov}(Y_n, V)$ for all $n = 1, \dots, M$, and $\hat{\omega} = \widehat{Var}(V) > 0$. As is often the case, $\hat{\omega}$ can be very large relative to the values of \hat{a}_n , causing the disappearance of the large magnitude negative correlations in empirical data.

The goal of the covariance shift (C-SHIFT) normalization method introduced here is the recovery of the true empirical covariances $\widehat{Cov}(Y_n, Y_m)$ and the respective true empirical correlations in the case of the logarithmic gene expression data or any other situations with additive noise as in (2).

Let $\tilde{C} = (\widehat{Cov}(\tilde{Y}_n, \tilde{Y}_m))_{n,m}$ be the empirical covariance matrix of the observed data $\tilde{Y}_n^{(i)}$, and let $C = (\widehat{Cov}(Y_n, Y_m))_{n,m}$ be the empirical covariance matrix of the cleaned data $Y_n^{(i)}$ (i.e., the true empirical covariance) that we desire to recover. Formula (7) rewritten in the matrix form states

$$C = \tilde{C} + \hat{a}\mathbf{1}^T + \mathbf{1}^T\hat{a} - \hat{\omega}\mathbf{1}\mathbf{1}^T, \quad (8)$$

where $\hat{a} = (\hat{a}_1, \dots, \hat{a}_M)^T$, and $\mathbf{1}$ denotes the column vector of 1's, hence $\mathbf{1}\mathbf{1}^T$ is a square matrix of 1's.

Our goal here is to estimate \hat{a} and $\hat{\omega}$ in (8), and thus recover the true empirical covariance matrix C . This will be done in Section I. We assume large dimension M . There will be two cases.

Case I: If $\det(\tilde{C}) = 0$ (e.g. $N < M$), we make a small perturbation of the diagonal entries of \tilde{C} (the variances) resulting in a new covariance matrix being positive definite whose smallest eigenvalue is still

very close to zero. Next, we use energy minimization to estimate \hat{a}_n and $\hat{\omega}$ in (8).

Case II: If \tilde{C} is positive definite, our approach exploits the phenomenon sometimes referred to as the *curse of dimensionality* [2, 16] and sometimes as the *blessing of dimensionality* [5, 7, 10], postulating that in higher dimensions almost all data points are located near extrema (i.e., in the outer shell)*. In other words, for large M , we anticipate the smallest eigenvalue of C to be near zero. As a rigorous bound, we observe that if some of the correlations $\text{corr}(Y_n, Y_m)$ are located in $[-1, \delta - 1]$ interval, then the smallest eigenvalue of C is located within $[0, \delta \min \widehat{\text{Var}}(Y_n)]$ interval. Thus, as in Case I, under the blessing of dimensionality assumption, we again use energy minimization for estimating \hat{a}_n and $\hat{\omega}$.

The problem of improving the existing and developing new normalization methods is very important for scientists working with biological data. The fact that normalization alters the data-correlation structure was stated in Saccenti [19]. Besides [19] gives a comprehensive overview of normalization methods. In Bolstad *et al.* [4] the authors compare three complete data normalization methods (cyclic loess, contrast based method, and quantile) that make use of data from all arrays in an experiment to two methods that make use of a baseline array. The comparison was done on two publicly available datasets with the results favoring the complete data methods. For more on the normalization methods, see [1, 6, 8, 9, 15, 18, 21].

I. THEORETICAL DERIVATIONS

Proposition 1. *Suppose \mathcal{M} is a symmetric positive definite square matrix. Then,*

$$v^* := \max \{v : \mathcal{M} - v \mathbf{1}\mathbf{1}^T \text{ is positive semidefinite} \} \\ = \frac{1}{\mathbf{1}^T \mathcal{M}^{-1} \mathbf{1}}.$$

Proof. Observe that

$$x^T (\mathcal{M} - v \mathbf{1}\mathbf{1}^T) x = x^T \mathcal{M} x - v \left(\sum x_i \right)^2 \geq 0$$

for all $x \in \mathbb{R}^M$ if and only if $v \leq v^*$, where v^* minimizes $x^T \mathcal{M} x$ under the condition $\sum x_i = \text{Const}$. Next, applying the Lagrange multipliers method, we obtain $2\mathcal{M}x = \lambda \mathbf{1}$, and therefore,

$$v^* = \frac{x^T \mathcal{M} x}{\left(\sum x_i \right)^2} = \frac{\frac{\lambda}{2} x^T \mathbf{1}}{\left(\sum x_i \right)^2} = \frac{\lambda/2}{\mathbf{1}^T x} = \frac{1}{\mathbf{1}^T \mathcal{M}^{-1} \mathbf{1}}$$

*In this paper we will refer to the phenomenon as the blessing of dimensionality rather than the curse of dimensionality.

as $x = \frac{\lambda}{2} \mathcal{M}^{-1} \mathbf{1}$. \square

Suppose the empirical covariance matrix \tilde{C} is positive definite, i.e., \tilde{C} is of full rank. Consider values of a column vector $\alpha = (\alpha_1, \dots, \alpha_M)^T$ such that $\tilde{C} + \alpha \mathbf{1}^T + \mathbf{1}^T \alpha$ is positive definite. Then, by Prop. 1,

$$C_\alpha := \tilde{C} + \alpha \mathbf{1}^T + \mathbf{1}^T \alpha - v(\alpha) \mathbf{1}\mathbf{1}^T \quad (9)$$

is positive semidefinite with $\det(C_\alpha) = 0$, where we let

$$v(\alpha) := \frac{1}{\mathbf{1}^T (\tilde{C} + \alpha \mathbf{1}^T + \mathbf{1}^T \alpha)^{-1} \mathbf{1}}. \quad (10)$$

Next, recall the quantities \hat{a} and $\hat{\omega}$ in (8). If \tilde{C} is rank deficient, we perturb its diagonal entries by adding small positive (random or deterministic) values, and if \tilde{C} is positive definite, we assume the blessing of dimensionality phenomenon holds. Thus, in either case, we work under assumption that \tilde{C} is positive definite with its smallest eigenvalue located near zero. Then, Prop. 1 implies $\hat{\omega} \approx v(\hat{a})$, where $v(\alpha)$ is as defined in (10). Therefore, letting $\alpha = \hat{a}$ in (9), we will have $C_{\hat{a}}$ approximating C .

Finally, for all $X \in \mathbb{R}^{M \times N}$, let $\|X\|_F$ denote the Frobenius norm of X and let $\mathcal{E}(X) = \frac{1}{2} \|X\|_F^2$ be the energy function. Our next assumption states that \hat{a} can be estimated by the minimizer α^* of the energy function $\mathcal{E}(C_\alpha)$, i.e., we estimate \hat{a} by

$$\alpha^* = \text{argmin} \|C_\alpha\|_F.$$

The assumption is additionally justified by the observation that a random adjustment of the covariance via an additive noise (bias) as in (7) will result in an energy increase, i.e., $\mathcal{E}(\tilde{C}) > \mathcal{E}(C)$. Hence, we use C_{α^*} to approximate $C_{\hat{a}}$ and the desired true empirical covariance matrix C .

Lemma 1. *Suppose the empirical covariance matrix \tilde{C} is of full rank, and the quantities C_α and $v(\alpha)$ are as in (9) and (10). Then, the gradient of the Frobenius norm squared is given by*

$$\frac{1}{4} \nabla \|C_\alpha\|_F^2 = M\alpha \\ + [M^2 v^2(\alpha) - c v(\alpha) - 2M a v(\alpha)] A_\alpha^{-1} \mathbf{1} \\ + \tilde{C} \mathbf{1} + [a - M v(\alpha)] \mathbf{1}, \quad (11)$$

where $\|\cdot\|_F$ denotes the Frobenius norm, and we let

$$A_\alpha := \tilde{C} + \alpha \mathbf{1}^T + \mathbf{1}^T \alpha,$$

$$c := \mathbf{1}^T \tilde{C} \mathbf{1} \text{ and } a := \sum_{i=1}^M \alpha_i.$$

Proof. By (9), we have

$$\begin{aligned} \|C_\alpha\|_F^2 &= \|\tilde{C}\|_F^2 + 2M \sum_{i=1}^M \alpha_i^2 + M^2 v^2(\alpha) + 4 \left(\mathbf{1}^T \tilde{C} \alpha \right) \\ &\quad + 2a^2 - 2c v(\alpha) - 4Ma v(\alpha) \end{aligned} \quad (12)$$

Notice that

$$\begin{aligned} \frac{\partial}{\partial \alpha_i} A_\alpha &= \bar{e}_i \mathbf{1}^T + \mathbf{1} \bar{e}_i^T \quad \text{and} \\ \frac{\partial}{\partial \alpha_i} A_\alpha^{-1} &= -A_\alpha^{-1} (\bar{e}_i \mathbf{1}^T + \mathbf{1} \bar{e}_i^T) A_\alpha^{-1}, \end{aligned} \quad (13)$$

where \bar{e}_i is the i -th coordinate vector. Therefore, we have

$$\begin{aligned} \frac{\partial}{\partial \alpha_i} v(\alpha) &= v^2(\alpha) \mathbf{1}^T A_\alpha^{-1} (\bar{e}_i \mathbf{1}^T + \mathbf{1} \bar{e}_i^T) A_\alpha^{-1} \mathbf{1} \\ &= 2v(\alpha) \mathbf{1}^T A_\alpha^{-1} \bar{e}_i \end{aligned} \quad (14)$$

implying

$$\nabla v(\alpha) = 2v(\alpha) A_\alpha^{-1} \mathbf{1}. \quad (15)$$

Next, the gradient $\nabla \|C_\alpha\|_F^2$ in (11) is found via the equations (12) and (15). \square

First, observe that C_α is invariant under the addition of multiples of $\mathbf{1}$. Thus, without loss of generality, we restrict the domain to a hyperplane $a = \text{Const}$. Next, observe that $\mathbf{1}^T \nabla \|C_\alpha\|_F^2 = 0$ in (11). Thus, in the gradient descent method, the value of a remains constant, i.e., throughout the algorithm, vector α remains on the same hyperplane $a = \text{Const}$.

Lemma 2. *Suppose the empirical covariance matrix \tilde{C} is of full rank, and the quantities C_α , $v(\alpha)$, and A_α are as in (9), (10), and (12) respectively. Then, the Hessian of $\|C_\alpha\|_F^2$, denoted by $H_\alpha := \text{Hess}(\|C_\alpha\|_F^2)$ is expressed as follows*

$$\begin{aligned} \frac{1}{4} H_\alpha &= MI + \mathbf{1} \mathbf{1}^T - 2M v(\alpha) (A_\alpha^{-1} \mathbf{1} \mathbf{1}^T + \mathbf{1} \mathbf{1}^T A_\alpha^{-1}) \\ &\quad + (3M^2 v(\alpha) - c - 2Ma) v(\alpha) A_\alpha^{-1} \mathbf{1} \mathbf{1}^T A_\alpha^{-1} \\ &\quad - (M^2 v(\alpha) - c - 2Ma) A_\alpha^{-1}, \end{aligned} \quad (16)$$

where I is the identity matrix, $c = \mathbf{1}^T \tilde{C} \mathbf{1}$, and $a = \sum_{i=1}^M \alpha_i$.

Proof. By (11), we have

$$\begin{aligned} \frac{1}{4} H_\alpha &= \frac{1}{4} \nabla (\nabla \|C_\alpha\|_F^2)^T \\ &= M \nabla \alpha^T \\ &\quad + \left(\nabla (M^2 v^2(\alpha) - c v(\alpha) - 2Ma v(\alpha)) \right) \mathbf{1}^T A_\alpha^{-1} \\ &\quad + (M^2 v^2(\alpha) - c v(\alpha) - 2Ma v(\alpha)) \nabla \mathbf{1}^T A_\alpha^{-1} \\ &\quad + \nabla \mathbf{1}^T (a - M v(\alpha)), \end{aligned} \quad (17)$$

where $\nabla = \left(\frac{\partial}{\partial \alpha_1}, \dots, \frac{\partial}{\partial \alpha_M} \right)^T$ was used as the column vector of the partial derivative operators. The summation parts in (17) are calculated as follows. First,

$$M \nabla \alpha^T = MI. \quad (18)$$

Next, (15) implies

$$\begin{aligned} \nabla (M^2 v^2(\alpha) - c v(\alpha) - 2Ma v(\alpha)) \\ = 2(2M^2 v(\alpha) - c - 2Ma) v(\alpha) A_\alpha^{-1} \mathbf{1} - 2M v(\alpha) \mathbf{1}. \end{aligned} \quad (19)$$

Equation (13) implies

$$\begin{aligned} \nabla \mathbf{1}^T A_\alpha^{-1} &= \sum_{i=1}^M \bar{e}_i \mathbf{1}^T \frac{\partial}{\partial \alpha_i} A_\alpha^{-1} \\ &= - \sum_{i=1}^M \bar{e}_i \mathbf{1}^T A_\alpha^{-1} (\bar{e}_i \mathbf{1}^T + \mathbf{1} \bar{e}_i^T) A_\alpha^{-1} \\ &= - \sum_{i=1}^M (\bar{e}_i^T A_\alpha^{-1} \mathbf{1}) \bar{e}_i \mathbf{1}^T A_\alpha^{-1} \\ &\quad - (\mathbf{1}^T A_\alpha^{-1} \mathbf{1}) \sum_{i=1}^M \bar{e}_i \bar{e}_i^T A_\alpha^{-1} \\ &= -A_\alpha^{-1} \mathbf{1} \mathbf{1}^T A_\alpha^{-1} - (\mathbf{1}^T A_\alpha^{-1} \mathbf{1}) A_\alpha^{-1} \\ &= -A_\alpha^{-1} \mathbf{1} \mathbf{1}^T A_\alpha^{-1} - \frac{1}{v(\alpha)} A_\alpha^{-1}. \end{aligned} \quad (20)$$

Finally, (15) is used to derive

$$\nabla \mathbf{1}^T (a - M v(\alpha)) = \mathbf{1} \mathbf{1}^T - 2M v(\alpha) A_\alpha^{-1} \mathbf{1} \mathbf{1}^T. \quad (21)$$

Combining together equations (18)-(21) and substituting them into (17) we obtain (16). \square

Theorem 1. *Suppose the empirical covariance matrix \tilde{C} is of full rank, and the quantities C_α and $v(\alpha)$ are as in (9) and (10). Then, the Frobenius norm squared $\|C_\alpha\|_F^2$ is convex, i.e.,*

$$\Delta \|C_\alpha\|_F^2 \geq 0 \quad \forall \alpha. \quad (22)$$

Proof. We will use the notations of this section such as $c := \mathbf{1}^T \tilde{C} \mathbf{1}$ and $a := \sum_{i=1}^M \alpha_i$. Without loss of generality we consider α on the hyperplane $a = 0$.

Here, $A_\alpha = \tilde{C} + \alpha \mathbf{1}^T + \mathbf{1}^T \alpha$ is a positive definite symmetric matrix with eigenvalues

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_M > 0$$

counted with respect to algebraic multiplicity, and let $\{v_i\}_{i=1, \dots, M}$ be the corresponding orthonormal basis of eigenvectors.

Equation (16) implies

$$\begin{aligned} \frac{1}{4}\Delta\|C_\alpha\|_F^2 &= \frac{1}{4}\text{Tr}(H_\alpha) \\ &= M^2\left(1 - v(\alpha)\text{Tr}(A_\alpha^{-1})\right) \\ &\quad + c\left(\text{Tr}(A_\alpha^{-1}) - v(\alpha)\mathbf{1}^T A_\alpha^{-2}\mathbf{1}\right) \\ &\quad + 3M\left(Mv^2(\alpha)\mathbf{1}^T A_\alpha^{-2}\mathbf{1} - 1\right). \end{aligned} \quad (23)$$

The Laplacian in (23) is shown to be strictly positive in the following three steps. First, by the Cauchy-Bunyakovsky-Schwarz inequality, we have

$$\begin{aligned} Mv^2(\alpha)\mathbf{1}^T A_\alpha^{-2}\mathbf{1} - 1 \\ = v^2(\alpha)\left(\|\mathbf{1}\|_2\|A_\alpha^{-1}\mathbf{1}\|_2^2 - (\mathbf{1}^T A_\alpha^{-1}\mathbf{1})^2\right) \geq 0. \end{aligned} \quad (24)$$

Next, observe that for $M \geq 2$,

$$Mx + (1-x)^2 \geq 1 \quad \forall x \in [0, 1].$$

Thus, for a given probability mass function $\{p_k\}_{k=1,\dots,M}$ such that $p_k < 1$ for all k , and a given index $i \in \{1, \dots, M\}$, Jensen's inequality implies

$$\begin{aligned} Mp_i + \left(\sum_{j:j \neq i} \lambda_j^{-1} p_j\right) \left(\sum_{j:j \neq i} \lambda_j p_j\right) \\ = Mp_i + (1-p_i)^2 \left(\sum_{j:j \neq i} \lambda_j^{-1} q_j\right) \left(\sum_{j:j \neq i} \lambda_j q_j\right) \\ \geq Mp_i + (1-p_i)^2 \geq 1 \end{aligned} \quad (25)$$

where we let $q_j = \frac{p_j}{1-p_i}$ for all $j \neq i$. Summing over all i in (25), we obtain,

$$\begin{aligned} \sum_i \lambda_i^{-1} p_i + \frac{1}{M} \sum_i \lambda_i^{-1} \left(\sum_{j:j \neq i} \lambda_j^{-1} p_j\right) \left(\sum_{j:j \neq i} \lambda_j p_j\right) \\ \geq \frac{1}{M} \sum_i \lambda_i^{-1}. \end{aligned} \quad (26)$$

Eqn. (26) implies

$$\begin{aligned} \sum_i \lambda_i^{-1} p_i + \frac{1}{M} \sum_i \lambda_i^{-1} p_i \left(\sum_{j:j \neq i} \lambda_j^{-1}\right) \left(\sum_k \lambda_k p_k\right) \\ \geq \frac{1}{M} \sum_i \lambda_i^{-1}. \end{aligned} \quad (27)$$

which rewrites as

$$\begin{aligned} \sum_i \lambda_i^{-1} p_i + \frac{1}{M} \left(\sum_i \lambda_i^{-1} p_i\right) \left(\sum_j \lambda_j^{-1}\right) \left(\sum_k \lambda_k p_k\right) \\ \geq \frac{1}{M} \left(\sum_i \lambda_i^{-2} p_i\right) \left(\sum_k \lambda_k p_k\right) + \frac{1}{M} \sum_i \lambda_i^{-1}. \end{aligned} \quad (28)$$

Finally, we let $p_i = \frac{1}{M}(\mathbf{1}^T v_i)^2$ and substitute the following expressions into (28):

$$\begin{aligned} \sum_i \lambda_i p_i &= \frac{1}{M} \mathbf{1}^T A_\alpha \mathbf{1} = \frac{1}{M} \mathbf{1}^T \tilde{C} \mathbf{1} = \frac{c}{M} \quad \text{as } a = 0, \\ \sum_i \lambda_i^{-1} p_i &= \frac{1}{M} \mathbf{1}^T A_\alpha^{-1} \mathbf{1} = \frac{1}{M v(\alpha)}, \end{aligned}$$

$$\sum_i \lambda_i^{-1} = \text{Tr}(A_\alpha^{-1}), \quad \text{and} \quad \sum_i \lambda_i^{-2} p_i = \frac{1}{M} \mathbf{1}^T A_\alpha^{-2} \mathbf{1}.$$

Consequently, (28) rewrites as

$$\begin{aligned} M^2\left(1 - v(\alpha)\text{Tr}(A_\alpha^{-1})\right) \\ + c\left(\text{Tr}(A_\alpha^{-1}) - v(\alpha)\mathbf{1}^T A_\alpha^{-2}\mathbf{1}\right) \geq 0. \end{aligned} \quad (29)$$

Substituting (24) and (29) into (23), we then obtain $\Delta\|C_\alpha\|_F^2 \geq 0$. \square

II. C-SHIFT ALGORITHM AND EXPERIMENTS

In this section we provide the C-SHIFT algorithm and evaluate its performance on synthetic data sets. Moreover, we compare the C-SHIFT algorithm with the well-known and frequently used normalization methods: Quantile, Rank, LOESS, and Median absolute deviation (MAD). Our empirical results demonstrate that the C-SHIFT algorithm outperforms other methods.

A. C-SHIFT algorithm

The pseudocode for the C-SHIFT algorithm is given in Algorithm 1. Note that the algorithm takes into account both cases: when \tilde{C} has full rank and when \tilde{C} is rank deficient (i.e., \tilde{C} is positive semi-definite but not positive definite). When \tilde{C} is rank deficient the rank of $\tilde{C} + \alpha\mathbf{1}^T + \mathbf{1}^T\alpha$ may exceed the rank \tilde{C} by no more than 2, and therefore may also be rank deficient. Therefore, to make \tilde{C} a full rank we add to it a diagonal matrix $\text{diag}(f)$, where f is a vector of i.i.d. random variables from $\text{Unif}[0, 1]$.

To find the optimal $\alpha^* = \arg \min_\alpha \|C_\alpha\|_F^2$, we use gradient and Hessian, provided in equations (11) and (16), in the trust-region algorithm to minimize $\|C_\alpha\|_F^2$.

B. Numerical experiments

In this section we conduct experiments on two synthetic datasets that we generate using random covariance method (RCM) and cascade method. We start by describing both methods.

1) Data generation:

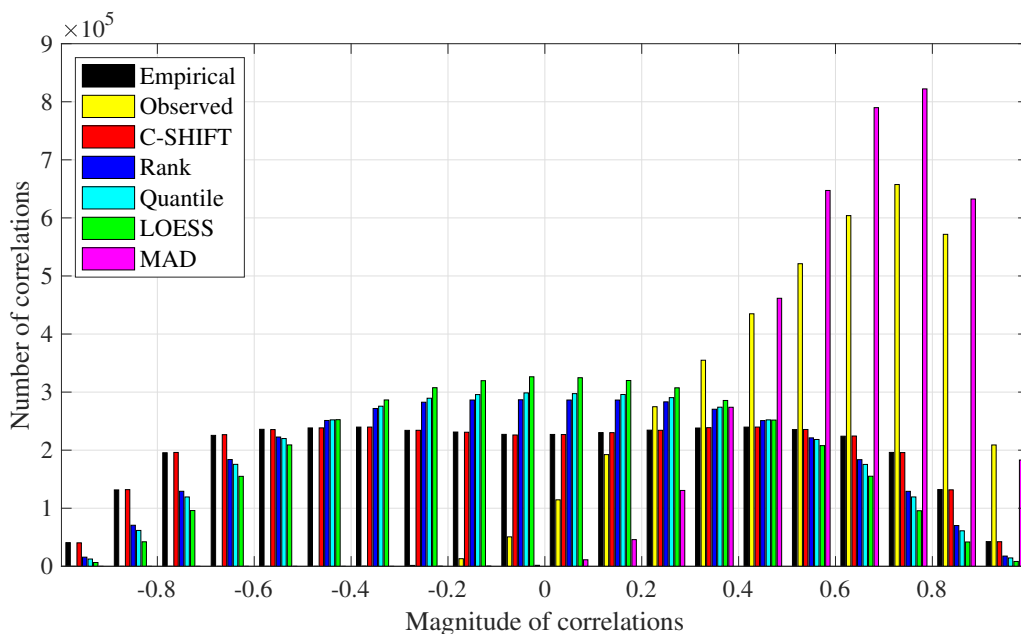


Fig. 1. A bar graph of correlations for the RCM data set. On the x-axis we display the range of correlations, partitioned into intervals of length 0.1. The height of each bar describes the number of correlations that belong to the corresponding interval. Bars of different colors correspond to different correlation matrices, indicated in the legend.

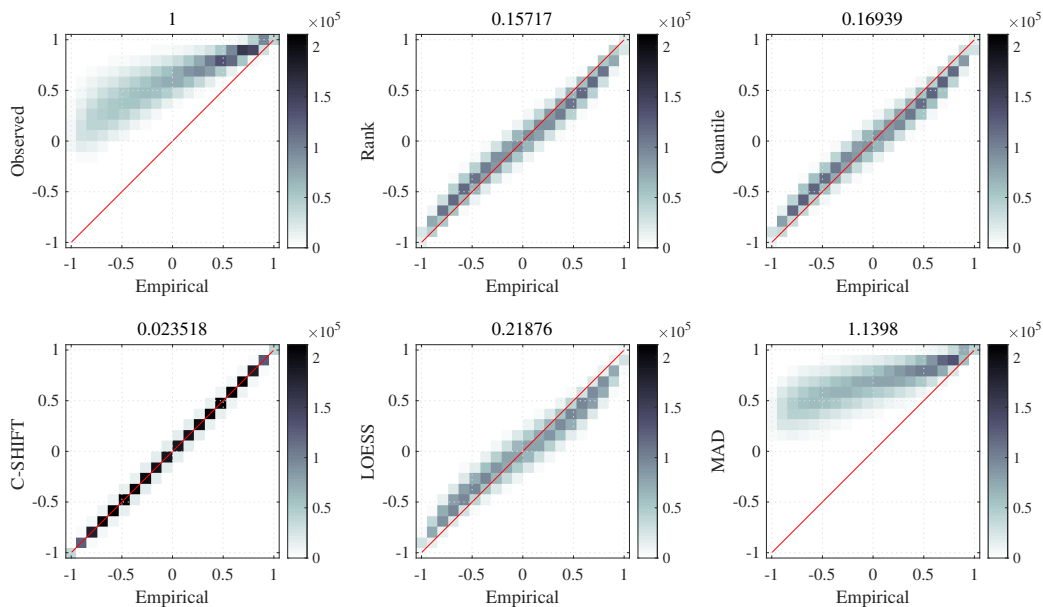


Fig. 2. The heat maps for the RCM data set. Each heat map illustrates the transformation of the true empirical correlations $corr(Y_n, Y_m)$ (horizontal axis) after adding bias and applying the corresponding normalization method. In the top left plot the vertical axis represents the observed correlations $corr(\tilde{Y}_n, \tilde{Y}_m)$. In the remaining five heat maps, the vertical coordinates represent the correlations after normalization. Going clockwise, these five heat maps are Rank, Quantile, MAD, LOESS, and C-SHIFT. The number on top of each heat map indicates the relative leftover error after normalization. Smaller numbers indicate better recovery performance.

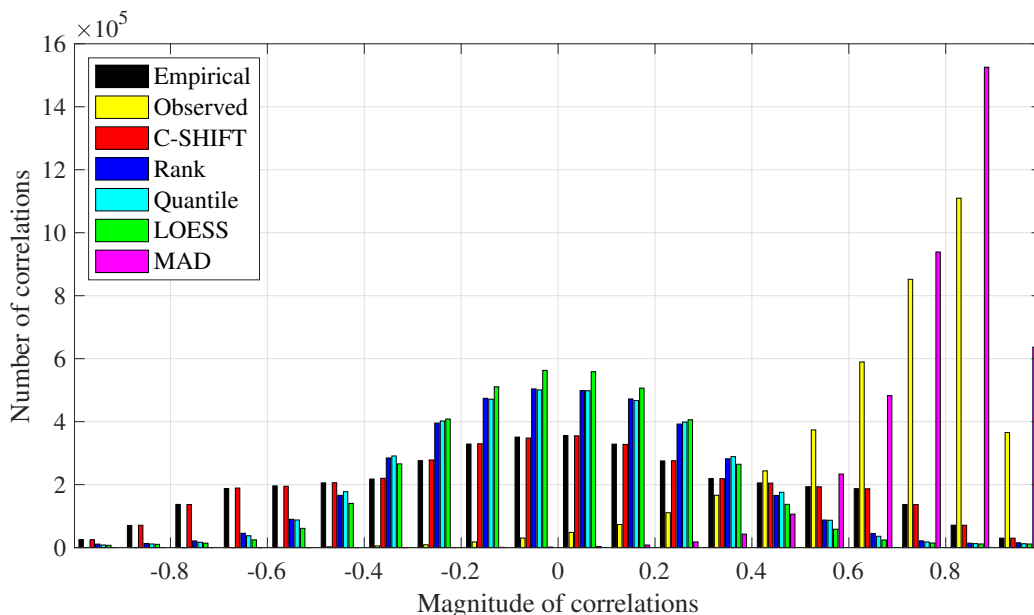


Fig. 3. A bar graph of correlations for the Cascade data set. On the x-axis we display the range of correlations, partitioned into intervals of length 0.1. The height of each bar describes the number of correlations that belong to the corresponding interval. Bars of different colors correspond to different correlation matrices, indicated in the legend.

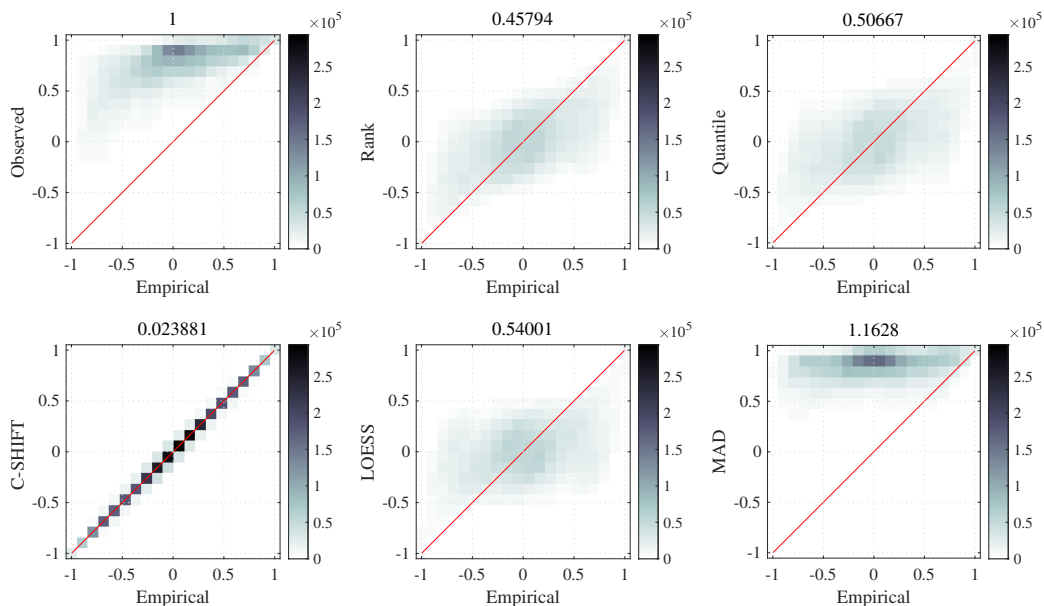


Fig. 4. The heat maps for the Cascade data set. Each heat map illustrates the transformation of the true empirical correlations $corr(Y_n, Y_m)$ (horizontal axis) after adding bias and applying the corresponding normalization method. In the top left plot the vertical axis represents the observed correlations after normalization $corr(\tilde{Y}_n, \tilde{Y}_m)$. In the remaining five heat maps, the vertical coordinates represent the correlations after normalization. Going clockwise, these five heat maps are Rank, Quantile, MAD, LOESS, and C-SHIFT. The darker the color, the higher the density. The number on top of each heat map indicates the relative leftover error after normalization. Smaller numbers indicate better recovery performance.

Input: observed covariance matrix \tilde{C}
Output: recovered empirical covariance matrix C
if \tilde{C} is rank deficient **then**
 $f \leftarrow$ i.i.d. $\text{Unif}[0,1]$
 $\tilde{C} \leftarrow \tilde{C} + \text{diag}(f)$
end if
 $v(\alpha) \leftarrow \left[\mathbf{1}^T (\tilde{C} + \alpha \mathbf{1} \mathbf{1}^T + \mathbf{1} \mathbf{1}^T \alpha)^{-1} \mathbf{1} \right]^{-1}$
 $C_\alpha \leftarrow \tilde{C} + \alpha \mathbf{1} \mathbf{1}^T + \mathbf{1} \mathbf{1}^T \alpha - v(\alpha) \mathbf{1} \mathbf{1}^T$
 $\alpha^* \leftarrow \arg \min_\alpha \|C_\alpha\|_F^2$
 $C \leftarrow C_{\alpha^*}$
if \tilde{C} is rank deficient **then**
 $C \leftarrow C - \text{diag}(f)$
end if
return C

Algorithm 1: C-SHIFT

a) *Random covariance method (RCM):* We generate a synthetic data set with $M = 2000$ genes and $N = 50$ measurements (samples) using RCM. For that we first generate an auxiliary matrix $H \in \mathbb{R}^{M \times m}$ ($m = 2$) whose entries are independent random variables, uniformly distributed over the interval $I = [-10, 10]$. Next, we sample a diagonal matrix $D \in \mathbb{R}^{M \times M}$ with diagonal entries being i.i.d. exponential random variables with parameter $\lambda_D = 30$. We let $\Sigma = HH^T + D$ be the population (parameter) covariance matrix. Then we generate the true empirical logarithmic data $Y^{(i)} = (Y_n^{(i)}) \sim \mathcal{N}(0, \Sigma)$ for each $i = 1, \dots, N$. Finally, we set the observed logarithmic data be $\tilde{Y}_n^{(i)} = Y_n^{(i)} + V^{(i)}$, where vector $V^{(i)}$ are $\mathcal{N}(-0.01, 100)$ random variables.

b) *Cascade method:* The *cascade* datasets were generated according by a directed acyclic weighted network $G = (V, E)$ aka directed acyclic graph (DAG). The graph was randomly generated via a recurrent *cascade* model. The parent-offspring relation is represented by the direction of edges $E = \{(u, v)\}$ of the graph G , i.e., u is the parent vertex and v is its offspring. For any vertex v let $pa(v)$ be the set of its parents, $pa(v) = \{u \in V : (u, v) \in E\}$. Next, for each edge $(u, v) \in E$ an independent random weight c_{uv} is assigned, with c.d.f.

$$pU_{[a_-, b_-]}(x) + (1 - p)U_{[a_+, b_+]}(x),$$

where the parameters $a_- < b_- \leq 0$, $0 \leq a_+ < b_+$, and $p \in (0, 1)$ are fixed, and $U_A(x)$ denotes the uniform c.d.f. on an interval A . We generated a random weighted DAG with the nodes $v \in V$ representing the genes. The random variables $\{Y_v\}_{v \in V}$ representing the logarithmic gene expressions are generated

as a noisy multiplicative cascade via the following structural linear recursive equations:

$$Y_v = \sum_{u \in pa(v)} c_{uv} Y_u + \varepsilon_v,$$

where the recursion begins with $Y_0 = y_0$, and proceeds from generation to generation. The noise variables $(\varepsilon_v, v \in V)$ are i.i.d. $\mathcal{N}(0, \sigma^2)$, sampled independently from the random weights c_{uv} . For simulation of $(Y_v, v \in V)$ the following values of parameters were chosen:

p	$[a_-, b_-]$	$[a_+, b_+]$	σ^2	y_0	$ V $
1/3	$[-1.2, -0.5]$	$[0.5, 1.3]$	1	4.5	2000

2) *Simulation results:* We generate two data sets (RCM and Cascade) using the methods described in section II-B1. Each data set consists of a matrix with the empirical data $(Y_n^{(i)}) \in \mathbb{R}^{M \times N}$ and a matrix with the observed data $(\tilde{Y}_n^{(i)}) \in \mathbb{R}^{M \times N}$. In both, RCM and Cascade data sets, we let $M = 2000$ genes and $N = 50$ measurements (samples). For each data set, we normalize the covariance matrix \tilde{C} , obtained from the observed data, by using C-SHIFT, Rank, Quantile, LOESS, and MAD methods. We compare the performance of the algorithms using the results presented in Figures 1-4.

In Figures 1 and 3 we depict the bar graphs of correlations for RCM and Cascade data sets, respectively. On the x-axis we display the range of correlations, partitioned into the intervals of length 0.1. The height of each bar represents the number of correlations that belong to the corresponding interval. Bars of different colors correspond to different correlation matrices, indicated in the legend. In particular, black and yellow bars correspond to the correlation matrices of empirical and observed data, respectively. As we can see in both data sets, the correlations of the observed data (yellow) are shifted away from -1 so that there are no large magnitude negative correlations. The aim of the normalization algorithms is to shift the correlations back into correct positions, i.e., ideally, the correlations of the normalized data should match the empirical correlations. Note that for both data sets, the C-SHIFT method correctly recovers the number of correlations in each interval: the red bars almost perfectly match the black bars. In contrast, other normalization methods could not recover the correct numbers of correlations, especially for the correlations of larger magnitudes. Specifically, Rank, Quantile and LOESS normalization techniques tend to shift correlations mostly to the center of the bar plot, each forming a bell shape. Predictably, the MAD method has the worst performance in corre-

lation recovery. Finally, among the other three normalization techniques (Quantile, Rank, and LOESS), the latter method has the poorest performance.

Figures 2 and 4 contain six heat maps each, for RCM and Cascade data sets, respectively. Each heat map illustrates the transformation of the true empirical correlations $\text{corr}(Y_n, Y_m)$ (horizontal axis) after adding bias and applying the corresponding normalization method. We consider 2,001,000 correlations corresponding to all pairs of genes. For each point, representing a pair of genes (n, m) , the horizontal coordinate equals the true empirical correlation $\text{corr}(Y_n, Y_m)$ in all six plots. The vertical coordinate in the top left heat map is the correlation in the observed data, $\text{corr}(\tilde{Y}_n, \tilde{Y}_m)$. Importantly, it shows the shift of correlations rightward in the observed data. In the remaining five heat maps, the vertical coordinates represent the correlations after normalization. Going clockwise, these five heat maps are Rank, Quantile, MAD, LOESS, and C-SHIFT. The darker the color, the higher the density. Notice that the heat map for C-SHIFT is almost perfectly diagonal, which demonstrates how well C-SHIFT recovers the correlations. Thus, in addition to correctly recovering the right numbers of correlations in each interval (which was demonstrated in Figures 1 and 3), the proposed C-SHIFT algorithm also returns (shifts back) the correlations to the correct margins. Hence, the heat map is a diagonal line. The number on top of each heat map indicates the relative leftover error after normalization, i.e., the ℓ^2 -norm of the vector of differences between the horizontal and vertical coordinates, scaled by the Frobenius norm of the difference between the empirical and the observed correlation matrices. Thus, the left top heat map is assigned the value 1, and for each normalization method, the smaller the number the better it recovers the empirical correlation matrix. Any such number smaller than one is an improvement. The number for C-SHIFT is by far the smallest in each data set (0.023518 and 0.023881), while in the case of MAD normalization, the corresponding number even exceeds 1.

III. DISCUSSION

In systems biology, the gene co-expression networks (GCN) are reconstructed from the correlations between the genes. GCN recovery relies on removing the bias with a normalization method, and thus improving the estimation of correlations between the pairs of genes. However, the standard normalization techniques such as Rank, Quantile, LOESS, and MAD are known to be insufficient at

recovering the true empirical correlations while the C-SHIFT algorithm is specifically designed to recover the true empirical correlations. The multiple experiments with synthetic data sets demonstrate the algorithm's superior performance in comparison to the standard normalization techniques.

Importantly, we notice that the C-SHIFT algorithm corrects the positive shift of covariances (and correlations) observed when $\hat{\omega} = \widehat{\text{Var}}(V)$ is larger than $\hat{\alpha}_n = -\widehat{\text{Cov}}(Y_n, V)$ ($n = 1, \dots, M$) in (7). Hence, the independence of V from Y_n assumption can be replaced with a weaker assumption stating that $\text{Cov}(Y_n, V) \ll \text{Var}(V)$. This will be explored in a follow-up publication.

An alternative version of the C-SHIFT algorithm is based on trace minimization approach instead of energy minimization. In this alternative C-SHIFT algorithm, the positive semi-definite matrix C_{α^*} with

$$\alpha^* = \text{argmin Tr}(C_{\alpha})$$

is used to approximate the true empirical covariance matrix C . The analogs of Lemmas 1 and 2 and the convexity result in Theorem 1 are also established for $\text{Tr}(C_{\alpha})$ in the trace minimization approach. See [11]. Empirically it appears that this alternative approach produces the same α^* as the original C-SHIFT algorithm based on energy minimization as presented in this paper, and therefore it recovers the empirical covariance C with the same accuracy. Thus, the alternative, trace minimizing C-SHIFT algorithm can be used instead of Algorithm 1. This approach will be analyzed in a follow-up paper.

Finally, the C-SHIFT algorithm was deposited on GitHub at <https://github.com/prlogan/C-SHIFT>

ACKNOWLEDGMENTS

This research is supported by the FAPESP awards 2018/14952-7 and 2018/07826-5, and by the NSF award DMS-1412557.

REFERENCES

- [1] Dhammika Amaratunga and Javier Cabrera. Analysis of data from viral dna microchips. *Journal of the American Statistical Association*, 96(456):1161–1170, 2001.
- [2] Francis Bach. Breaking the curse of dimensionality with convex neural networks. *The Journal of Machine Learning Research*, 18(1):629–681, 2017.
- [3] Martin Bilban, Lukas K Buehler, Steven Head, Gernot Desoye, and Vito Quaranta. Normalizing dna microarray data. *Current Issues in Molecular Biology*, 4:57–64, 2002.

- [4] Benjamin M Bolstad, Rafael A Irizarry, Magnus Åstrand, and Terence P. Speed. A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics*, 19(2):185–193, 2003.
- [5] David L. Donoho. High-dimensional data analysis: The curses and blessings of dimensionality. In *AMS conference on Mathematical Challenges of the 21st Century*. Citeseer, 2000.
- [6] Jianqing Fan, Heng Peng, and Tao Huang. Semilinear high-dimensional model for normalization of microarray data: a theoretical analysis and partial consistency. *Journal of the American Statistical Association*, 100(471):781–796, 2005.
- [7] Alexander N Gorban and Ivan Yu Tyukin. Blessing of dimensionality: mathematical foundations of the statistical physics of data. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 376(2118):20170237, 2018.
- [8] Alexander J Hartemink, David K Gifford, Tommi S Jaakkola, and Richard A Young. Maximum-likelihood estimation of optimal scaling factors for expression array normalization. In *Microarrays: Optical Technologies and Informatics*, volume 4266, pages 132–140. International Society for Optics and Photonics, 2001.
- [9] Jianhua Hu and Xuming He. Enhanced quantile normalization of microarray data to reduce loss of information in gene expression profiles. *Biometrics*, 63(1):50–59, 2007.
- [10] Paul C Kainen. Utilizing geometric anomalies of high dimension: When complexity makes computation easier. In *Computer Intensive Methods in Control and Signal Processing*, pages 283–294. Springer, 1997.
- [11] Paul Logan. *C-SHIFT, Quantile Theory, and Assessing Monotonicity*. PhD thesis, Oregon State University, 2020.
- [12] Taesung Park, Sung-Gon Yi, Sung-Hyun Kang, SeungYeoun Lee, Yong-Sung Lee, and Richard Simon. Evaluation of normalization methods for microarray data. *BMC Bioinformatics*, 4(1):33, 2003.
- [13] Sylvain Pradervand, Johann Weber, Jérôme Thomas, Manuel Bueno, Pratyaksha Wirapati, Karine Lefort, G Paolo Dotto, and Keith Harshman. Impact of normalization on mirna microarray expression profiling. *RNA*, 15(3):493–501, 2009.
- [14] Xing Qiu, Hulin Wu, and Rui Hu. The impact of quantile and rank normalization procedures on the testing power of gene differential expression analysis. *BMC Bioinformatics*, 14(1):124, 2013.
- [15] John Quackenbush. Microarray data normalization and transformation. *Nature Genetics*, 32(4):496–501, 2002.
- [16] Miloš Radovanović, Alexandros Nanopoulos, and Mirjana Ivanović. Hubs in space: Popular nearest neighbors in high-dimensional data. *Journal of Machine Learning Research*, 11(Sep):2487–2531, 2010.
- [17] Youlan Rao, Yoonkyung Lee, David Jarjoura, Amy S Ruppert, Chang-gong Liu, Jason C Hsu, and John P Hagan. A comparison of normalization techniques for microRNA microarray data. *Statistical applications in genetics and molecular biology*, 7(1), 2008.
- [18] Cavan Reilly, Changchun Wang, and Mark Rutherford. A method for normalizing microarrays using genes that are not differentially expressed. *Journal of the American Statistical Association*, 98(464):868–878, 2003.
- [19] Edoardo Saccenti. Correlation patterns in experimental data are affected by normalization procedures: consequences for data analysis and network inference. *Journal of Proteome Research*, 16(2):619–634, 2017.
- [20] Andreas Scherer. *Batch effects and noise in microarray experiments: sources and solutions*. John Wiley & Sons, 2009.
- [21] Gordon K Smyth and Terry Speed. Normalization of cDNA microarray data. *Methods*, 31(4):265–273, 2003.