Published in final edited form as:
Commun Stat Simul Comput. 2012 January 1; 41(1): 89–98. doi:10.1080/03610918.2011.579368.

A Simple Distribution-Free Algorithm for Generating Simulated High-Dimensional Correlated Data with an Autoregressive Structure

ANDRES AZUERO[1], **DAVID T. REDDEN**[2], **HEMANT K. TIWARI**[2], **SENAIT G. ASMELLASH**[3], and **CHANDRIKA J. PIYATHILAKE**[4]

[1]Department of Community Health Outcomes and Systems, University of Alabama at Birmingham, Birmingham, Alabama, USA

[2]Department of Biostatistics, University of Alabama at Birmingham, Birmingham, Alabama, USA

[3]Department of Surgery, University of Alabama at Birmingham, Birmingham, Alabama, USA

[4]Department of Nutrition Sciences, University of Alabama at Birmingham, Birmingham, Alabama, USA

Abstract

A distribution-free method to generate high-dimensional sequences of dependent variables with an autoregressive structure is presented. The quantile or fractile correlation (i.e., the moment correlation of the quantiles) is used as measure of dependence among a set of contiguous variables. The proposed algorithm breaks the sequence in small parts and avoids having to define one large correlation matrix for the entire high-dimensional sequence of variables. Simulations based on proteomics data are presented. Results suggest that negligible or no loss of fractile correlation occurs by splitting the generation of a sequence into small parts.

Keywords

Distribution-free methods; Fractile correlations; High-dimensional data; Simulation

1. Introduction

High-dimensional data, i.e. datasets whose dimensions or number of variables are in the tens or hundreds of thousands, are inherent in many modern applications such as biologic and biomedical data (Li and Ronghui, 2008), telecommunications (Emdad, 2008), the world-wide-web (Kogan, 2007), consumer behavior data (Naik et al., 2008), and consumer financial history (Diwakar and Vaidya, 2009), among others.

Advances in knowledge and technology in the last 20 years have brought applications with radically larger numbers of variables. For instance, in genetics, a “Genome-wide Association Study (GWAS)” aims to detect associations between a medical condition and more than a million genetic markers (Pearson and Manolio, 2008). In addition, the advent of high-throughput technology for next-generation sequencing will produce unprecedented size

Address correspondence to Andres Azuero, Department of Community Health Outcomes and Systems, University of Alabama at Birmingham, NB 1019G, 1530 3rd Ave. S., Birmingham, AL 35294-1210, USA; andreo@uab.edu.

Mathematics Subject Classification 00A72; 68U20; 62P99; 62H20; 15A99.

of marker data ranging from 10s to 100s of millions of genetic variants. Not only is dealing with such large numbers of variables a complex task, but also the number of observations or cases is radically smaller than the number of variables, precluding the use of traditional statistical methods, such as regression models, in order to make inferences on the whole set of variables simultaneously. For biological data, such as those arising from genomics or proteomics, an additional complication is that contiguous variables might not be nicely normally distributed or even continuous, and are expected to be correlated or in some form of dependency. Multiple testing of correlated variables may result in correlated test statistics, which may affect the experiment-wise error rates, if corrections for multiple testing that assume independence are used (Kim and van de Wiel, 2008).

The purpose of this article is to present a distribution-free, simple, and easily implementable algorithm, to generate high-dimensional simulated correlated data without having to define a single correlation matrix for the entire dataset. Although high-dimensional data can be obtained from public databases, such data might not be practical for use in developing statistical methods, since the "truth" (e.g., the set of biological markers truly associated with a medical condition) is not known. Obviously, the advantage of simulated data is that the "truth" is known a-priori and can be compared to the results of a proposed statistical method. The algorithm proposed in this article is based on a series of methods developed to simulate non-Gaussian processes using Spearman's rank correlations (Phoon et al., 2004) and quantile or fractile correlations (Iman and Conover, 1982; Fackler, 1991) as measures of dependence. Rank and fractile correlations are invariant to monotonic transformations of the data. In contrast, the more common Pearson product-moment correlation is only invariant to location and scale transformations.

2. Methods

A common method to generate a relatively small number k of correlated random standard normal variables, with n observations or cases for each variable, given a symmetric moment correlation matrix $\mathbf{C}_{k\times k}$, consists of finding a matrix $\mathbf{D}_{k\times k}$ such that $\mathbf{D}^{\mathrm{T}}\mathbf{D} = \mathbf{C}$, where $\mathbf{D}$ is calculated by a singular value decomposition or a Cholesky decomposition (for positive definite $\mathbf{C}$). Then, after generating a matrix of uncorrelated random standard normal variables $\mathbf{R}_{n\times k}$, the matrix $\mathbf{(RD)}_{n\times k}$ yields a matrix of k standard normal variables with n observations, having the specified moment correlation structure among its k columns. Early references for this method date back to Moonan (1957) and Scheuer and Stoller (1962).

In order to extend this method to non-normal variables, we consider the fact that for Uniform(0,1) variables the moment correlation is equal to the rank correlation as well as to the fractile correlation. In this approach the initial step is to select a fractile correlation matrix $\mathbf{F}_{k\times k}$ and then transform this matrix to a moment correlation matrix $\mathbf{C}_{k\times k}$ by $c_{ij}=2\sin\left(\frac{\pi}{6}f_{ij}\right)$. This transformation, derived by Karl Pearson in 1907, applies only to normally distributed variables (Pearson, 1907; Hotelling and Pabst, 1936). Next, the matrix $\mathbf{D}$ is calculated using a singular value or a Cholesky decomposition. Then, after generating a matrix of uncorrelated random normal standard variables $\mathbf{R}_{n\times k}$ and calculating the matrix of correlated standard normal variables $\mathbf{(RD)}_{n\times k}$, a matrix $\mathbf{U}_{n\times k}$ of Uniform(0,1) variables is obtained by applying a probability integral transformation to **RD**, i.e., applying the standard normal cumulative distribution function (cdf) to each of the elements of **RD**. This matrix **U** has the specified fractile correlation structure among its k columns. Next, the rows of **U** can be transformed from Uniform(0,1) into different distributions by inverse cdf transformations. Since the fractile correlation is invariant to monotone transformations, as long as the inverse cdf transformation is monotone, the fractile correlation is not affected

and therefore the new variables will retain the initial quantile correlation structure $\mathbf{F}_{k\times k}$, regardless of their final distribution function.

The aforementioned Choleski decomposition is a matrix factorization for symmetric positive definite matrices (Bock, 1998). It results in an upper triangular matrix and lower triangular matrix, which is the transpose of the upper triangular matrix. Consider the Cholesky decomposition $\mathbf{C} = \mathbf{D}^{\mathrm{T}}\mathbf{D}$, where $\mathbf{C}_{k\times k}$ is a symmetric positive definite matrix, $\mathbf{D}^{\mathrm{T}}$ is the lower triangular matrix, and $\mathbf{D}$ is the upper triangular matrix. Since $\mathbf{D}^{\mathrm{T}}$ and $\mathbf{D}$ are triangular matrices, one of the features of this decomposition is that the element in row 1 column 1 of $\mathbf{C}$, c_{11}, is factored into $\sqrt{c_{11}}$, that is:

$$\begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1k} \\ c_{21} & c_{22} & \cdots & c_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ c_{k1} & c_{k2} & \cdots & c_{kk} \end{bmatrix} = \begin{bmatrix} d_{11} & 0 & \cdots & 0 \\ d_{21} & d_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ d_{k1} & d_{k2} & \cdots & d_{kk} \end{bmatrix} \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1k} \\ 0 & d_{22} & \cdots & d_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & c_{kk} \end{bmatrix}$$

and since $c_{11} = d_{11} \times d_{11} + 0 \times 0 + \ldots + 0 \times 0 = (d_{11})^2$, then $\sqrt{c_{11}} = d_{11}$.

If $\mathbf{C}$ is a symmetric positive definite correlation matrix then all the diagonal elements of $\mathbf{C}$ are equal to 1 and since $c_{11} = 1$ then $d_{11} = \sqrt{1} = 1$. Thus the first column of $\mathbf{D}$ is made of $d_{11} = 1$ and the remaining elements of this column are equal to zero. The consequence of this feature when generating random variables is that if $\mathbf{C} = \mathbf{D}^{\mathrm{T}}\mathbf{D}$ is a Cholesky decomposition then after generating a matrix of uncorrelated random normal standard variables $\mathbf{R}_{n\times k}$, the matrix $\mathbf{(RD)}_{n\times k}$ yields a matrix of k standard normal variables with n observations, having the specified moment correlation structure among its k columns, and the first column of $\mathbf{(RD)}_{n\times k}$ is equal to the first column of $\mathbf{R}_{n\times k}$. Based on this consideration, we propose an algorithm that generates a long sequence of random Uniform(0,1) variables with an approximate autoregressive structure ($\rho > 0$). The proposed algorithm breaks the sequence in small parts and avoids having to define one large fractile correlation matrix $\mathbf{F}$ for the whole sequence. Then, this sequence of correlated Uniform(0,1) variables may be used to generate variables with different distributions by means of inverse CDF transformations. In short, the objective is to generate a matrix of K columns and n rows of Uniform(0,1) variables ($K \gg n$), where the K columns have approximately an autoregressive AR(1) structure. The proposed algorithm is as follows:

1. For a small k such that k is a divisor of K, generate a matrix $\mathbf{R}_{n\times k}$ of k random standard normal variables (columns) with n rows. Each column will be independent of the other columns.

2. Input the desired k by k autoregressive fractile correlation matrix $\mathbf{F}_{k\times k}$ (must be symmetric, positive, definite).

3. Transform the fractile correlation matrix $\mathbf{F}_{k\times k}$ into a moment correlation matrix $\mathbf{C}_{k\times k}$ by $c_{ij} = 2\sin\left(\frac{\pi}{6} f_{ij}\right)$.

4. Calculate $\mathbf{D}_{k\times k}$, the upper triangular Choleski decomposition matrix of the moment correlation matrix $\mathbf{C}$, where $\mathbf{C} = \mathbf{D}^{\mathrm{T}}\mathbf{D}$.

5. Post-multiply the matrix of independent standard normal variables $\mathbf{R}_{n\times k}$ by the upper triangular Cholesky decomposition matrix $\mathbf{D}$, i.e., $(\mathbf{RD})_{N\times k} = \mathbf{R}^{(1)}_{n\times k}$. The transformed set of standard normal variables will have the desired moment correlation structure, yet the first column remains unchanged, i.e., column 1 of $\mathbf{R}^{(1)}_{n\times k}$ is equal to column 1 of $\mathbf{R}_{n\times k}$.

To generate the next set:

6. Generate another matrix of k independent random standard normal variables (columns) with n rows.

7. Take the last column (column k) of the previous set $\mathbf{R}^{(1)}_{n\times k}$ and make it the first column of the new set, resulting in a matrix of n rows by $j = k + 1$ columns, $\mathbf{S}_{n\times j}$.

8. Input the j by j autoregressive fractile correlation matrix $\mathbf{F}_{j\times j}$.

9. Transform the fractile correlation matrix $\mathbf{F}_{j\times j}$ into a moment correlation matrix $\mathbf{C}_{j\times j}$.

10. Calculate the upper triangular Choleski decomposition matrix $\mathbf{D}_{j\times j}$ of the moment correlation matrix $\mathbf{C}$.

11. Post-multiply the matrix of independent standard normal variables $\mathbf{S}_{n\times j}$ by the upper triangular Cholesky decomposition matrix $\mathbf{D}$, i.e., $\mathbf{SD}_{n\times j}$ The transformed set of standard normal variables will have the desired moment correlation structure, yet the first column remains unchanged, which is the last column (column k) of the previous set $\mathbf{R}^{(1)}_{n\times k}$.

12. Remove the first column of the new correlated set $\mathbf{(SD)}_{n\times j}$ (which is the same last column of the previous set $\mathbf{R}^{(1)}_{n\times k}$), resulting in a second n by k matrix $\mathbf{R}^{(2)}_{n\times k}$.

13. Join both n by k correlated sets, resulting in a n by $2k$ matrix of correlated standard normal variables $\mathbf{Q}_{n\times 2k} = \left[\mathbf{R}^{(1)}_{n\times k} | \mathbf{R}^{(2)}_{n\times k}\right]$ having an approximate autoregressive correlation structure.

To generate a long sequence of dimension K:

14. Repeat steps 6–13 as needed in order to generate the desired number of correlated standard normal variables $\mathbf{Q}_{n\times K} = \left[\mathbf{R}^{(1)}_{n\times k} | \mathbf{R}^{(2)}_{n\times k} | \ldots | \mathbf{R}^{(m)}_{n\times k}\right]$, where $m \times k = K$.

15. Transform the correlated standard normal variables $\mathbf{Q}$ into correlated Uniform(0,1) columns using the probability integral transformation, i.e., $\mathbf{U} = \phi\mathbf{Q}$, where ϕ is the standard normal cdf.

16. The Uniform(0,1) columns of $\mathbf{U}$ will have approximately the desired autoregressive fractile correlation structure, and may be used to generate variables with different distributions by means of inverse cdf transformations. As long as the inverse cdf transformations are monotone, the fractile correlation structure is maintained.

3. Simulations

Data collected from a small pilot project in proteomics were used to design the first of two simulation experiments presented here. The data consist of mass spectra generated from urine samples collected from 251 women using a complex laboratory technique called Matrix-assisted laser desorption/ionization (MALDI) time-of-flight (TOF) mass spectrometry (MS). The mass spectra data were collected over a mass-to-charge ratio (m/z) range of 2,000 Daltons to 20,000 Daltons using a Bruker Ultraflex III (Bruker Daltonics, Billerica, MA) MALDI TOF mass spectrometer and preprocessed with SpecAlign (Wong et al., 2005) Preprocessing of the MALDI spectra consisted of baseline subtraction, total ion current normalization, de-noising, peak-picking, and alignment as described in Norris et al. (2007), and generated a total of 171 peaks. Thus, at 171 different m/z values, relative peak intensities were measured on each participant's urine sample. The final dataset was composed of 171 variables, corresponding to the neighboring m/z peaks, and 251 observations or cases, corresponding to the study participants. The average distance between

the neighboring m/z peaks used was 107 Daltons (SD 474 Daltons). A feature of protein intensity data, measured in normalized total ion current (TIC), is that intensity values are non-negative. Further, descriptive statistics revealed that the observed intensity data were right skewed. The average skewness coefficient computed over each of the 171 m/z peaks was estimated at 3.05 (SD = 2.28; Range = [0.13, 11.6]). In addition, the intensity data appeared to be non-normally distributed. In 165m/z peaks, the null hypothesis of normality for the intensity data, using Lilliefor's or Kolmogorov-Smirnov tests (Gibbons and Chakraborti, 2003), was rejected at the 0.001 significance level. Although several theoretical distributions can be used to model non-negative right-skewed data, a Dagum distribution appeared to provide one of the best fits for the observed intensity data, according to the results from a distribution-fitting routine implemented in the software package EasyFit v5.0 (MathWave Technologies, 2008). Using the parameterization for the Dagum distribution implemented in the R package VGAM (The R Foundation for Statistical Computing, 2009), the maximum likelihood estimates for the Dagum parameters were $a = 1.713$, $b = 620.28$, and p.arg = 0.73012. Figure 1 shows a histogram of the observed intensity data and fitted Dagum density function. Maximum likelihood estimates of Dagum parameters were also computed for the intensity observations within each of the 171 m/z peaks separately. Alternatively, since the more common Gamma distribution can also be used to model right-skewed data, Gamma distribution parameters were also estimated for each of the 171 m/z peaks, resulting in 171 sets of Dagum and Gamma parameters to be used later in the simulations. Finally, the intensity data appeared to be correlated. The average rank correlation for intensity values between two subsequent m/z peaks was 0.38 (SD = 0.27, IQR = [0.18, 0.54], Range = [−0.34, 0.99]). With these preliminary parameters, we used our proposed algorithm to generate two simulated datasets of $n = 100$ rows by $K = 100{,}000$ correlated columns of right-skewed data, starting from the same dataset of correlated Uniform(0,1) variables, which results from step 15 of our proposed algorithm. In the first dataset, the data were generated under a Dagum ($a = 1.713$, $b = 620.28$, and p.arg = 0.73012) distribution; in the second dataset each column had 0.5 probability of being generated under a Dagum or alternatively under a Gamma distribution, with parameters selected randomly from the set of 171 parameters computed from the sample data. The purpose of generating the right-skewed data under different distributional assumptions was to examine if distributional assumptions made any difference in terms of the resulting rank correlation structure. The initial correlated Uniform(0,1) columns were generated under a fractile autoregressive structure with correlation between contiguous columns $\rho = 0\ 4$. Thus, the desired fractile correlation matrix for any randomly selected set of, say, five contiguous variables in the dataset is as follows:

$$\begin{bmatrix} 1 & 0.4 & 0.16 & 0.064 & 0.026 \\ 0.4 & 1 & 0.4 & 0.16 & 0.064 \\ 0.16 & 0.4 & 1 & 0.4 & 0.16 \\ 0.064 & 0.16 & 0.4 & 1 & 0.4 \\ 0.026 & 0.064 & 0.16 & 0.4 & 1 \end{bmatrix}$$

Note that there is no restriction on the size of the matrix chosen to evaluate the correlation values, such as the above 5 × 5 matrix, and the number columns used to generate data. To show that correlation is preserved when the number of columns used to generate data is less than the number of columns of the matrix chosen to evaluate the correlation values, we chose the number of columns to be generated at each step, k, to be 4. So in this simulation experiment, the total number of small sequences to generate and then join was equal to $K/k = 25{,}000$ for a total of $K = 100{,}000$ columns. We then randomly selected 10,000 sequences of five contiguous columns and tabulated the observed fractile correlations among the columns in the selected sequences. The fractile correlation structure was evaluated on the

Uniform(0,1) variables resulting from step 15 of the algorithm. Then, for comparison we also evaluated the rank correlation structure among the simulated Dagum and Dagum/Gamma variables. The simple R code written for this simulation is available from the authors upon request.

Also, in order to determine whether a relevant decrease in observed fractile correlation, if any, was caused by splitting the generation of a sequence of variables, we conducted a second simulation experiment where, beginning with the same set of five random standard normal variables (each with 100 observations), a sequence of five correlated Uniform(0,1) variables, with a fractile autoregressive structure ($\rho = 0.8$), was generated by two methods. In the first method, correlation on the five variables was induced by a single correlation matrix factored by a singular value decomposition. In the second method, the sequence of five variables was split into two sequences. First, correlation was induced on the first three contiguous variables. Then, similar to our proposed algorithm, the third and last column of this initial sequence is used as the first column of a second correlated sequence of three columns. As in our proposed algorithm, the correlation structure was induced by a Cholesky decomposition of a correlation matrix, so that the first column of the second sequence remains unaltered. Next, the initial three-column sequence was joined to the last two columns of the second sequence. We conducted the experiment 10,000 times and tabulated the observed correlations among the columns, for the sets of Uniform(0,1) variables generated by each of these two methods.

Table 1 shows the results from the first simulation experiment. The table provides descriptive statistics for the cells in fractile and rank correlation matrices of 10,000 randomly selected sets of five contiguous variables taken from: (1) the initial generated dataset of 100,000 Uniform(0,1) variables and used to evaluate the observed fractile correlations; (2) right-skewed data generated under a Dagum ($a = 1.713$, $b = 620.28$, and p.arg = 0.73012) distribution; and (3) right-skewed data where each column had 0.5 probability of being generated under Dagum or alternatively Gamma distributions. The distribution for the observed fractile correlations appeared unbiased with respect to and symmetric around the target values. The conclusions are similar for the rank correlations, except for a minimal decrease in correlation value, compared to the fractile correlation. The rank correlations were identical, regardless of distributional assumptions.

Table 2 shows the results from the second simulation experiment. The table provides descriptive statistics for the cells in correlation matrices of 10,000 sets of five correlated Uniform(0,1) variables generated by two methods. Method 1: a single fractile correlation was used to induce correlation on the set. Method 2: the sequence was split in two parts, similar to our proposed algorithm. In this case, the cells of interest are [1, 5], [2, 5], [1, 4], and [2, 4], since with Method 2 correlation between the variables in these cells was not induced directly but through the third variable in the sequence. For these four cells, the distributions for the observed correlations are ostensibly similar between Methods 1 and 2. These results suggest that minimal or negligible loss correlation, if any, may occur by splitting the generation of a sequence in smaller parts.

4. Concluding Remarks

The simulations conducted for testing the proposed algorithm resulted, on the average, in the targeted fractile correlation values. Also, our results suggest that there is no decrease in correlation values when generating a sequence with our proposed algorithm, as long as each of the parts used to build the long sequence are of small dimension (e.g., the number of columns ≤5 for each part). In fact, in the results of our second simulation experiment, the correlations resulting from simulating a short sequence of five variables with a single

correlation matrix are, for all practical purposes, indistinguishable from the correlations observed when the sequence was generated in two parts.

Limitations of the algorithm include the requirement of a positive definite (or semidefinite) correlation matrix, since it is the type of matrix that can be factored with a Cholesky decomposition. Further, the algorithm is restricted to autoregressive structures. Although in the examples we used the same value of the autoregressive parameter for the entire sequence, the value of the parameter need not be the same for the entire sequence. The value of the parameter can be modified as needed for sections of the sequence, in order to improve the simulation of real-life systems. Likewise, since one of the end-products of the algorithm is a sequence of Uniform(0,1) variables, the algorithm can also be used to simulate binary or multinomial data under dependency, by assigning ranges on the interval (0,1) to each category. However, if binary or multinomial data with fewer than 5 categories are generated, some attenuation of the observed correlations should be expected.

Acknowledgments

This publication was supported by Grant Number (U54 CA118948-01) from the National Cancer Institute.

References

Bock, R. The Data Analysis Briefbook. Springer; New York: 1998.

Diwakar, H.; Vaidya, A. Data quality for decision support—the Indian banking scenario. In: Chan, C.; Chawla, S.; Sadiq, S.; Zhou, X., editors. Data Quality and High-Dimensional Data Analysis. Proceedings of the DASFAA 2008 Workshops; Hackensack, NJ: World Scientific Publishing Company; 2009. p. 60-77.

Emdad, F. High Dimensional Data Analysis: Overview, Analysis, and Applications. VDM Verlag; Saarbrücken, Germany: 2008.

Fackler P. Modeling interdependence: An apporach to simulation and elicitation. American Journal of Agricultural Economics. 1991; 73(4):1091–1097.

Gibbons, J.; Chakraborti, S. Nonparametric Statistical Inference. 4th ed. Marcel Dekker, Inc.; New York: 2003.

Hotelling H, Pabst M. Rank correlation and tests of significance involving no assumption of normality. Annals of Mathematical Statistics. 1936; 7(1):29–43.

Iman R, Conover W. A distribution-free approach to inducing rank correlation among input variables. Communications in Statistics-Simulation and Computation. 1982; 11(3):311–334.

Kim K, van de Wiel M. Effects of dependence in high-dimensional multiple testing. BMC Bioinformatics. 2008; 9(114)

Kogan, J. Introduction to Clustering Large and High-Dimensional Data. Cambridge University Press; New York: 2007.

Li, X.; Ronghui, X. High-Dimensional Data Analysis in Cancer Research. Springer-Verlag; New York: 2008.

Moonan W. Linear transformation to a set of stochastically dependent normal variables. Journal of the American Statistical Association. 1957; 52(278):247–252.

Naik P, Wedel M, Bacon L, Bodapati A, Bradlow E, Kamakura W, Kreulen J, Lenk P, Madigan D, Montgomery A. Challenges and opportunities in high-dimensional choice data analyses. Marketing Letters. 2008; 19:201–213.

Norris J, Cornett D, Mobley J, Andersson M, Seeley E, Chaurand P, Caprioli R. Processing MALDI mass spectra to improve mass spectral direct tissue analysis. International Journal of Mass Spectrometry. 2007; 260:212–221. [PubMed: 17541451]

Pearson A, Manolio T. How to interpret a genome-wide association study. Journal of the American Medical Association. 2008; 299(11):1335–1344. [PubMed: 18349094]

Pearson, K. On Further Methods of Determining Correlation. Dulau and Co.; London: 1907.

Phoon K, Queck S, Huang H. Simulation of non-Gaussian processes using fractile correlation. Probabilistic Engineering Mechanics. 2004; 19:287–292.

Scheuer E, Stoller D. On the generation of normal random vectors. Technometrics. 1962; 4(2):278–281.

Wong J, Cagney G, Cartwright H. SpecAlign-processing and alignment of mass spectra datasets. Bioinformatics. 2005; 21:2088–2090. [PubMed: 15691857]

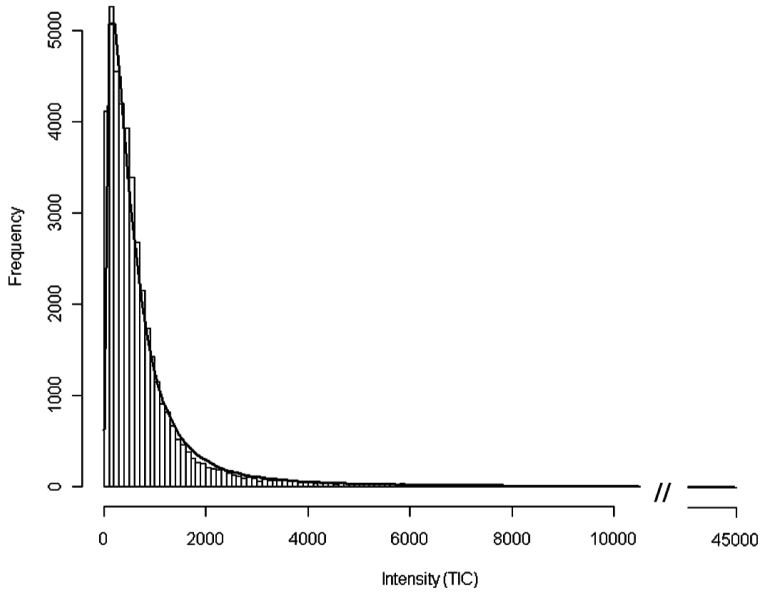

Figure 1.
Histogram of the observed intensity data measured in total ion current (TIC) and fitted Dagum.

Table 1

Descriptive statistics for the cells in correlation matrices of 10,000 randomly selected sets of five contiguous variables taken from (1) initial generated dataset of 100,000 Uniform(0,1) variables; (2) variables generated under a Dagum (a = 1.713, b = 620.28, and p.arg = 0.73012) distribution; and (3) each variable had 0.5 probability of being generated under a Dagum or alternatively under a Gamma distribution. Diagonal cells (equal to 1) and symmetric cells are omitted

Cell	Target value	Type	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	Std. Dev.
1,2	0.400	Fractile[1]	0.047	0.343	0.401	0.398	0.458	0.714	0.086
		Rank[2]	0.063	0.339	0.398	0.396	0.456	0.712	0.087
		Rank[3]	0.063	0.339	0.398	0.396	0.456	0.712	0.087
1,3	0.160	Fractile[1]	−0.238	0.092	0.163	0.160	0.228	0.506	0.098
		Rank[2]	−0.253	0.091	0.162	0.159	0.228	0.505	0.099
		Rank[3]	−0.253	0.091	0.162	0.159	0.228	0.505	0.099
1,4	0.064	Fractile[1]	−0.341	−0.003	0.065	0.065	0.132	0.458	0.099
		Rank[2]	−0.315	−0.004	0.065	0.064	0.132	0.426	0.099
		Rank[3]	−0.315	−0.004	0.065	0.064	0.132	0.426	0.099
1,5	0.026	Fractile[1]	−0.426	−0.043	0.024	0.025	0.094	0.355	0.100
		Rank[2]	−0.404	−0.043	0.024	0.025	0.094	0.358	0.100
		Rank[3]	−0.404	−0.043	0.024	0.025	0.094	0.358	0.100
2,3	0.400	Fractile[1]	0.010	0.339	0.400	0.396	0.456	0.689	0.087
		Rank[2]	0.009	0.335	0.396	0.393	0.454	0.695	0.088
		Rank[3]	0.009	0.335	0.396	0.393	0.454	0.695	0.088
2,4	0.160	Fractile[1]	−0.198	0.093	0.161	0.160	0.227	0.591	0.099
		Rank[2]	−0.202	0.092	0.159	0.158	0.226	0.590	0.100
		Rank[3]	−0.202	0.092	0.159	0.158	0.226	0.590	0.100
2,5	0.064	Fractile[1]	−0.283	−0.003	0.066	0.065	0.133	0.458	0.099
		Rank[2]	−0.284	−0.003	0.065	0.065	0.133	0.426	0.100
		Rank[3]	−0.284	−0.003	0.065	0.065	0.133	0.426	0.100
3,4	0.400	Fractile[1]	0.007	0.342	0.401	0.398	0.460	0.677	0.086

Cell	Target value	Type	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	Std. Dev.
3,5	0.160	Rank[2]	−0.012	0.338	0.399	0.396	0.457	0.672	0.087
		Rank[3]	−0.012	0.338	0.399	0.396	0.457	0.672	0.087
		Fractile[1]	−0.265	0.096	0.162	0.160	0.227	0.519	0.098
		Rank[2]	−0.268	0.096	0.160	0.160	0.226	0.516	0.097
		Rank[3]	−0.268	0.096	0.160	0.160	0.226	0.516	0.097
4,5	0.400	Fractile[1]	0.017	0.342	0.402	0.399	0.460	0.690	0.086
		Rank[2]	0.018	0.339	0.400	0.397	0.458	0.686	0.087
		Rank[3]	0.018	0.339	0.400	0.397	0.458	0.686	0.087

[1] Evaluated on Uniform(0,1) variables.

[2] Evaluated on variables generated under a Dagum distribution.

[3] Evaluated on variables generated under Dagum or Gamma distributions.

Table 2

Descriptive statistics for the cells in correlation matrices of 10,000 sets of 5 correlated variables generated by two methods. Method 1 (M1): a single fractile correlation was used to induce correlation on the set. Method 2 (M2): the sequence was split in two parts. The cells of interest are [1,5], [2,5], [1,4], and [2,4]. In M2, correlation between the variables in these four cells is not induced directly but through the third variable in the sequence

Cell	Target value	Method	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	Std. Dev.
1,2	0.8	1	0.5608	0.7752	0.8021	0.7994	0.8267	0.9107	0.0389
		2	0.5878	0.774	0.8015	0.7985	0.8263	0.9136	0.0389
1,3	0.64	1	0.3345	0.5988	0.6423	0.6393	0.6831	0.8373	0.0624
		2	0.3528	0.5987	0.6421	0.6385	0.6823	0.8212	0.0624
1,4	0.51	1	0.1555	0.4605	0.5143	0.5108	0.5637	0.7617	0.0764
		2	0.1583	0.4679	0.5188	0.5163	0.5689	0.7852	0.0764
1,5	0.41	1	0.0442	0.3529	0.4111	0.4081	0.4674	0.6908	0.0855
		2	0.0017	0.3614	0.4196	0.4162	0.4744	0.6665	0.0855
2,3	0.8	1	0.6216	0.7743	0.8027	0.7992	0.8273	0.9250	0.0391
		2	0.6040	0.7748	0.8018	0.7991	0.8264	0.9099	0.0391
2,4	0.64	1	0.3589	0.5988	0.6418	0.6387	0.6821	0.8355	0.0621
		2	0.3992	0.6047	0.6470	0.6433	0.6862	0.8257	0.0621
2,5	0.51	1	0.2029	0.4607	0.5139	0.5107	0.5635	0.7572	0.0762
		2	0.1794	0.4692	0.5200	0.5173	0.5689	0.7476	0.0762
3,4	0.8	1	0.5684	0.7752	0.8016	0.7992	0.8266	0.9093	0.0390
		2	0.6153	0.7756	0.8022	0.7993	0.8266	0.9098	0.0390
3,5	0.64	1	0.3598	0.5987	0.6423	0.6385	0.6822	0.8225	0.0621
		2	0.3707	0.6008	0.6428	0.6399	0.6824	0.8289	0.0621
4,5	0.8	1	0.6063	0.7745	0.8021	0.7995	0.8271	0.9156	0.0388
		2	0.5529	0.7755	0.8019	0.7992	0.8262	0.9169	0.0388

Commun Stat Simul Comput. Author manuscript; available in PMC 2012 January 1.