graphsim: An R package for simulating gene expression data from graph structures of biological pathways

S. Thomas Kelly *[1, 2] and Michael A. Black †[1]

[1]Department of Biochemistry, University of Otago, Dunedin, New Zealand
[2]Center of Integrative Medical Sciences, RIKEN, Yokohama, Japan

30th June 2020

Summary

Transcriptomic analysis is used to capture the molecular state of a cell or sample in many biological and medical applications. In addition to identifying alterations in activity at the level of individual genes, understanding changes in the gene networks that regulate fundamental biological mechanisms is also an important objective of molecular analysis. As a result, databases that describe biological pathways are increasingly uesad to assist with the interpretation of results from large-scale genomics studies. Incorporating information from biological pathways and gene regulatory networks into a genomic data analysis is a popular strategy, and there are many methods that provide this functionality for gene expression data. When developing or comparing such methods, it is important to gain an accurate assessment of their performance. Simulation-based validation studies are frequently used for this. This necessitates the use of simulated data that correctly accounts for pathway relationships and correlations. Here we present a versatile statistical framework to simulate correlated gene expression data from biological pathways, by sampling from a multivariate normal distribution derived from a graph structure. This procedure has been released as the `graphsim` R package on CRAN and GitHub (https://github.com/TomKellyGenetics/graphsim) and is compatible with any graph structure that can be described using the `igraph` package. This package allows the simulation of biological pathways from a graph structure based on a statistical model of gene expression.

Introduction: inference and modelling of biological networks

Network analysis of molecular biological pathways has the potential to lead to new insights into biology and medical genetics (Barabási and Oltvai 2004; Hu, Thomas, and Brunak 2016). Since gene expression profiles capture a consistent signature of the regulatory state of a cell (Perou et al. 2000; Ozsolak and Milos 2011; Svensson, Vento-Tormo, and Teichmann 2018), they can be used to analyse complex molecular states with genome-scale data. However, biological pathways are often analysed in a reductionist paradigm as amorphous sets of genes involved in particular functions, despite the fact that the relationships defined by pathway structure could further inform gene expression analyses. In many cases, the pathway relationships are well-defined, experimentally-validated, and are available in public databases (Croft et al. 2014). As a result, network analysis techniques can play an important role in furthering our understanding of biological pathways and aiding in the interpretation of genomics studies.

Gene networks provide insights into how cells are regulated, by mapping regulatory interactions between target genes and transcription factors, enhancers, and sites of epigenetic marks or chromatin structures (Barabási and Oltvai 2004; Yamaguchi et al. 2007). Inference using these regulatory interactions genomic

*tom.kelly@riken.jp
†mik.black@otago.ac.nz

analysis has the potential to radically expand the range of candidate biological pathways to be further explored, or to improve the accuracy of bioinformatics and functional genomic analysis. A number of methods have been developed to utilise timecourse gene expression data (Yamaguchi et al. 2007; Arner et al. 2015) using gene regulatory modules in state-space models and recursive vector autoregressive models (Hirose et al. 2008; Shimamura et al. 2009). Various approaches to gene regulation and networks at the genome-wide scale have led to novel biological insights (Arner et al. 2015; Komatsu et al. 2013), however, inference of regulatory networks has thus far primarily relied on experimental validation or resampling-based approaches to estimate the likelihood of specific network modules being predicted (Markowetz and Spang 2007; Hawe, Theis, and Heinig 2019).

Simulating datasets that account for pathway structure are of particular interest for benchmarking regulatory network inference techniques and methods being developed for genomics data containing complex biological interactions (Schaffter, Marbach, and Floreano 2011; Saelens et al. 2019). Dynamical models using differential equations have been employed, such as by GeneNetWeaver (Schaffter, Marbach, and Floreano 2011), to generate simulated datasets specifically for benchmarking gene regulatory network inference techniques. There is also renewed interest in modelling biological pathways and simulating data for benchmarking due to the emergence of single-cell genomics technologies and the growing number of bioinformatics techniques developed to use this data (Zappia, Phipson, and Oshlack 2017; Saelens et al. 2019). Packages such as 'splatter' (Zappia, Phipson, and Oshlack 2017), which uses the gamma-poisson distribution, have been developed to model single-cell data. SERGIO (Dibaeinia and Sinha 2019) and dyngen (Cannoodt et al. 2020) build on this by adding gene regulatory networks and multimodality respectively. These methods have been designed based on known deterministic relationships or synthetic reaction states, to which stochasticity is then added. However, it is computationally-intensive to model these reactions at scale or run many iterations for benchmarking. In some cases, it is only necessary to model the statistical variability and "noise" of RNA-Seq data in order to evaluate methods in the presence of multivariate correlation structures.

There is a need, therefore, for a systematic framework for statistical modelling and simulation of gene expression data derived from hypothetical, inferred or known gene networks. Here we present a package to achieve this, where samples from a multivariate normal distribution are used to generate normally-distributed log-expression data, with correlations between genes derived from the structure of the underlying pathway or gene regulatory network. This methodology enables simulation of expression profiles that approximate the log-transformed and normalised data from microarray studies, as well as, bulk or single-cell RNA-Seq experiments. This procedure has been released as the `graphsim` package to enable the generation of simulated gene expression datasets containing pathway relationships from a known underlying network. These simulated datasets can be used to evaluate various bioinformatics methodologies, including statistical and network inference procedures.

Methodology and software

Here we present a procedure to simulate gene expression data with correlation structure derived from a known graph structure. This procedure assumes that transcriptomic data have been generated and follow a log-normal distribution (i.e., $log(X_{ij}) \sim MVN(\mu, \Sigma)$, where μ and Σ are the mean vector and variance-covariance matrix respectively, for gene expression data derived from a biological pathway) after appropriate normalisation (Law et al. 2014; Li et al. 2015). Log-normality of gene expression matches the assumptions of the popular `limma` package (Matthew E R. et al. 2015), which is often used for the analysis of intensity-based data from gene expression microarray studies and count-based data from RNA-Seq experiments. This approach has also been applied for modelling UMI-based count data from single-cell RNA-Seq experiments in the `DESCEND` R package (Wang et al. 2018).

In order to simulate transcriptomic data, a pathway is first constructed as a graph structure, using the `igraph` R package (Csardi and Nepusz 2006), with the status of the edge relationships defined (i.e, whether they activate or inhibit downstream pathway members). This procedure uses a graph structure such as that

presented in Figure 1a. The graph can be defined by an adjacency matrix, A (with elements A_{ij}), where

$$A_{ij} = \begin{cases} 1 & \text{if genes } i \text{ and } j \text{ are adjacent} \\ 0 & \text{otherwise} \end{cases}$$

A matrix, R, with elements R_{ij}, is calculated based on distance (i.e., number of edges contained in the shortest path) between nodes, such that closer nodes are given more weight than more distant nodes, to define inter-node relationships. A geometrically-decreasing (relative) distance weighting is used to achieve this:

$$R_{ij} = \begin{cases} 1 & \text{if genes } i \text{ and } j \text{ are adjacent} \\ (\frac{1}{2})^{d_{ij}} & \text{if a path can be found between genes } i \text{ and } j \\ 0 & \text{if no path exists between genes } i \text{ and } j \end{cases}$$

where d_{ij} is the length of the shortest path (i.e., minimum number of edges traversed) between genes (nodes) i and j in graph G. Each more distant node is thus related by $\frac{1}{2}$ compared to the next nearest, as shown in Figure 2b. An arithmetically-decreasing (absolute) distance weighting is also supported in the `graphsim` R package which implements this procedure:

$$R_{ij} = \begin{cases} 1 & \text{if genes } i \text{ and } j \text{ are adjacent} \\ 1 - \frac{d_{ij}}{diam(G)} & \text{if a path can be found between genes } i \text{ and } j \\ 0 & \text{if no path exists between genes } i \text{ and } j \end{cases}$$

Assuming a unit variance for each gene, these values can be used to derive a Σ matrix:

$$\Sigma_{ij} = \begin{cases} 1 & \text{if } i = j \\ \rho R_{ij} & \text{otherwise} \end{cases}$$

where ρ is the correlation between adjacent nodes. Thus covariances between adjacent nodes are assigned by a correlation parameter (ρ) and the remaining off-diagonal values in the matrix are based on scaling these correlations by the geometrically weighted relationship matrix (or the nearest positive definite matrix for Σ with negative correlations).

Computing the nearest positive definite matrix is necessary to ensure that the variance-covariance matrix can be inverted when used as a parameter in multivariate normal simulations, particularly when negative correlations are included for inhibitions (as shown below). Matrices that cannot be inverted occur rarely with biologically plausible graph structures but this approach allows for the computation of a plausible correlation matrix when the given graph structure is incomplete or contains loops. When required, the nearest positive definite matrix is computed using the `nearPD` function of the `Matrix` R package (Bates and Maechler 2016) to perform Higham's algorithm (Higham 2002) on variance-covariance matrices. The `graphsim` package gives a warning when this occurs.

Illustrations

Generating a Graph Structure

The graph structure in Figure 1a was used to simulate correlated gene expression data by sampling from a multivariate normal distribution using the `mvtnorm` R package (Genz and Bretz 2009; Genz et al. 2016). The graph structure visualisation in Figure 1 was specifically developed for (directed) `igraph` objects in and is available in the `graphsim` package. The `plot_directed` function enables customisation of plot parameters for

each node or edge, and mixed (directed) edge types for indicating activation or inhibition. These inhibition links (which occur frequently in biological pathways) are demonstrated in Figure 1b.

A graph structure can be generated and plotted using the following commands in R:

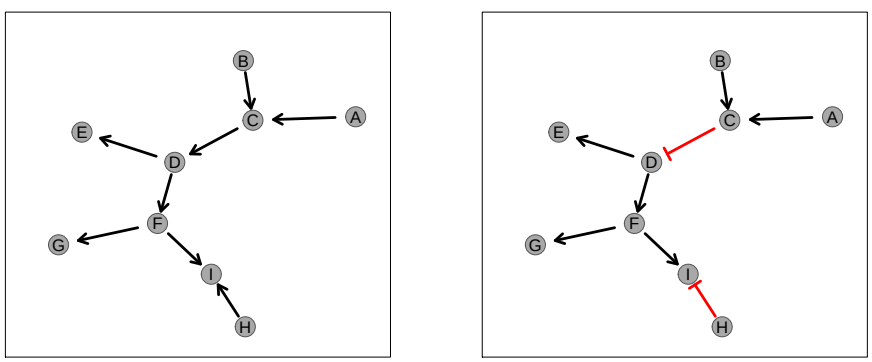
```
#install packages required (once per computer)
install.packages("graphsim")
```

```
#load required packages (once per R instance)
library("graphsim")
#load packages for examples
library("igraph"); library("gplots"); library("scales")
```

```
#generate graph structure
graph_edges <- rbind(c("A", "C"), c("B", "C"), c("C", "D"), c("D", "E"),
   c("D", "F"), c("F", "G"), c("F", "I"), c("H", "I"))
graph <- graph.edgelist(graph_edges, directed = TRUE)

#plot graph structure (Figure 1a)
plot_directed(graph, state ="activating", layout = layout.kamada.kawai,
  cex.node = 2, cex.arrow = 4, arrow_clip = 0.2)

#generate parameters for inhibitions for each edge in E(graph)
state <- c(1, 1, -1, 1, 1, 1, 1, -1)
#plot graph structure with inhibitions (Figure 1b)
plot_directed(graph, state=state, layout = layout.kamada.kawai,
  cex.node = 2, cex.arrow = 4, arrow_clip = 0.2)
```



(a) Activating pathway structure

(b) Inhibiting pathway structure

Figure 1: **Simulated graph structures**. A constructed graph structure used as an example to demonstrate the simulation procedure in Figures 2 and 3. Activating links are denoted by black arrows and inhibiting links by red edges.

Generating a Simulated Expression Dataset

The correlation parameter of $\rho = 0.8$ is used to demonstrate the inter-correlated datasets using a geometrically-generated relationship matrix (as used for the example in Figure 2c). This Σ matrix was then used to sample from a multivariate normal distribution such that each gene had a mean of 0, standard deviation 1, and covariance within the range $[0, 1]$ so that the off-diagonal elements of Σ represent correlations. This procedure

generated a simulated (continuous normally-distributed) log-expression profile for each node (Figure 2e) with a corresponding correlation structure (Figure 2d). The simulated correlation structure closely resembled the expected correlation structure (Σ in Figure 2c) even for the relatively modest sample size ($N = 100$) illustrated in Figure 2. Once a gene expression dataset comprising multiple pathways has been generated (as in Figure 2e), it can then be used to test procedures designed for analysis of empirical gene expression data (such as those generated by microarrays or RNA-Seq) that have been normalised on a log-scale.

The simulated dataset can be generated using the following code:

```
#plot relationship matrix
heatmap.2(make_distance_graph(graph, absolute = FALSE),
  scale = "none", trace = "none", col = colorpanel(50, "white", "red"),
  colsep = 1:length(V(graph)), rowsep = 1:length(V(graph)))

#plot sigma matrix
heatmap.2(make_sigma_mat_dist_graph(graph, cor = 0.8, absolute = FALSE),
  scale = "none", trace = "none", col = colorpanel(50, "white", "red"),
  colsep = 1:length(V(graph)), rowsep = 1:length(V(graph)))

#simulate data
expr <- generate_expression(100, graph, cor = 0.8, mean = 0,
  comm = FALSE, dist =TRUE, absolute = FALSE, state = state)
#plot simulated correlations
heatmap.2(cor(t(expr)), scale = "none", trace = "none",
          col = colorpanel(50, "white", "red"),
  colsep = 1:length(V(graph)), rowsep = 1:length(V(graph)))
#plot simulated expression data
heatmap.2(expr, scale = "none", trace = "none", col = bluered(50),
  colsep = 1:length(V(graph)), rowsep = 1:length(V(graph)), labCol = "")
```

The simulation procedure (Figure 2) can similarly be used for pathways containing inhibitory links (Figure 3) with several refinements. With the inhibitory links (Figure 3a), distances are calculated in the same manner as before (Figure 3b) with inhibitions accounted for by iteratively multiplying downstream nodes by -1 to form modules with negative correlations between them (Figures 3c and 3d). A multivariate normal distribution with these negative correlations can be sampled to generate simulated data (Figure 3e).

The following changes are needed to handle inhibitions:

```
#generate parameters for inhibitions
state <- c(1, 1, -1, 1, 1, 1, 1, -1)
plot_directed(graph, state=state, layout = layout.kamada.kawai,
  cex.node=2, cex.arrow=4, arrow_clip = 0.2)

#plot relationship matrix
heatmap.2(make_distance_graph(graph, absolute = FALSE),
  scale = "none", trace = "none", col = colorpanel(50, "white", "red"),
  colsep = 1:length(V(graph)), rowsep = 1:length(V(graph)))

#plot sigma matrix
heatmap.2(make_sigma_mat_dist_graph(graph, state, cor = 0.8, absolute = FALSE),
  scale = "none", trace = "none", col = colorpanel(50, "blue", "white", "red"),
  colsep = 1:length(V(graph)), rowsep = 1:length(V(graph)))

#simulated data
expr <- generate_expression(100, graph, state, cor = 0.8, mean = 0,
  comm = FALSE, dist =TRUE, absolute = FALSE)
```

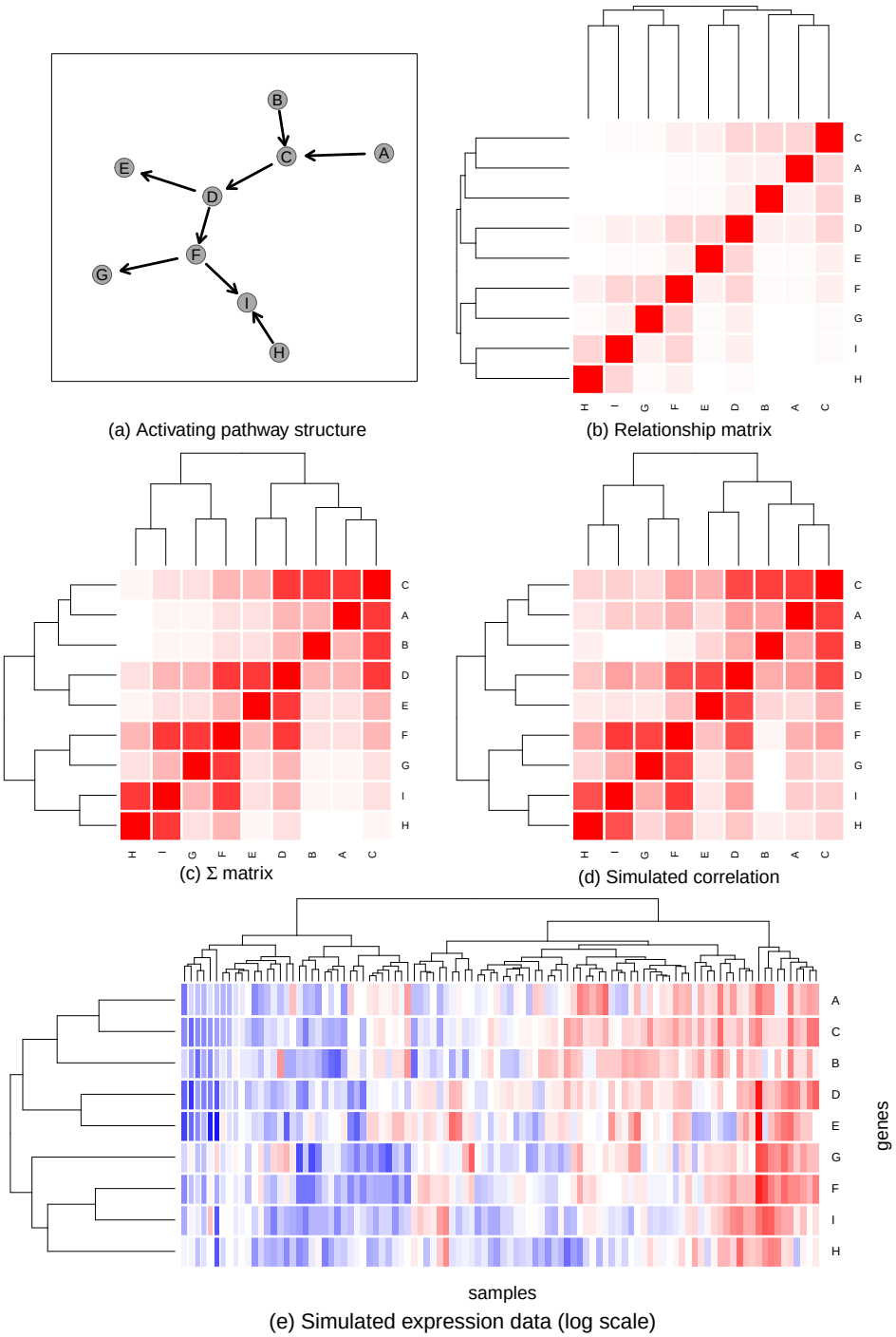


Figure 2: **Simulating expression from a graph structure**. An example of a graph structure (a) that has been used to derive a relationship matrix (b), Σ matrix (c) and correlation structure (d) from the relative distances between the nodes. Non-negative values are coloured white to red from 0 to 1 (e). The Σ matrix has been used to generate a simulated expression dataset of 100 samples (coloured blue to red from low to high) via sampling from the multivariate normal distribution. Here genes with closer relationships in the pathway structure show a higher correlation between simulated values.

```r
#plot simulated correlations
heatmap.2(cor(t(expr)), scale = "none", trace = "none",
          col = colorpanel(50, "blue", "white", "red"),
  colsep = 1:length(V(graph)), rowsep = 1:length(V(graph)))
#plot simulated expression data
heatmap.2(expr, scale = "none", trace = "none", col = bluered(50),
  colsep = 1:length(V(graph)), rowsep = 1:length(V(graph)), labCol = "")
```

The simulation procedure is also demonstrated here (Figure 4) on a pathway structure for a known biological pathway (Reactome pathway R-HSA-2173789): "TGF-β receptor signaling activates SMADs" (Figure 4a) derived from the Reactome database version 52 (Croft et al. 2014). Distances are calculated in the same manner as before (Figure 4b) producing blocks of correlated genes (Figures 4c and 4d). This shows that the multivariate normal distribution can be sampled to generate simulated data to represent expression with the complexity of a biological pathway (Figure 4e). Here *SMAD7* exhibits negative correlations with the other SMADs consistent with its functions as an "inhibitor SMAD" which competitively inhibits *SMAD4*.

We can import the graph structure into R as follows and run simulations as above:

```r
#import graph from data
graph <- identity(TGFBeta_Smad_graph)
#generate parameters for inhibitions
state <- E(graph)$state

plot_directed(graph, state = state, layout = layout.kamada.kawai,
  border.node=alpha("black", 0.75), fill.node="lightblue",
  col.arrow = c(alpha("navyblue", 0.25), alpha("red", 0.25))[state],
  cex.node = 1.5, cex.label = 0.8, cex.arrow = 2)

#plot relationship matrix
heatmap.2(make_distance_graph(graph, absolute = FALSE),
  scale = "none", trace = "none", col = colorpanel(50, "white", "red"),
  colsep = 1:length(V(graph)), rowsep = 1:length(V(graph)), labCol = "")

#plot sigma matrix
heatmap.2(make_sigma_mat_dist_graph(graph, state, cor = 0.8, absolute = FALSE),
  scale = "none", trace = "none", col = colorpanel(50, "blue", "white", "red"),
  colsep = 1:length(V(graph)), rowsep = 1:length(V(graph)), labCol = "")

#simulated data
expr <- generate_expression(100, graph, state, cor = 0.8,
  mean = 0,comm = FALSE, dist =TRUE, absolute = FALSE)
#plot simulated correlations
heatmap.2(cor(t(expr)), scale = "none", trace = "none",
          col = colorpanel(50, "blue", "white", "red"),
  colsep = 1:length(V(graph)), rowsep = 1:length(V(graph)), labCol = "")
#plot simulated expression data
heatmap.2(expr, scale = "none", trace = "none", col = bluered(50),
colsep = 1:length(V(graph)), rowsep = 1:length(V(graph)), labCol = "")
```

These simulated datasets can also be used for simulating gene expression data within a graph network to test genomic analysis techniques. Correlation structure can be included in datasets generated for testing whether true positive genes or samples can be detected in a sample with the background of complex pathway structure.

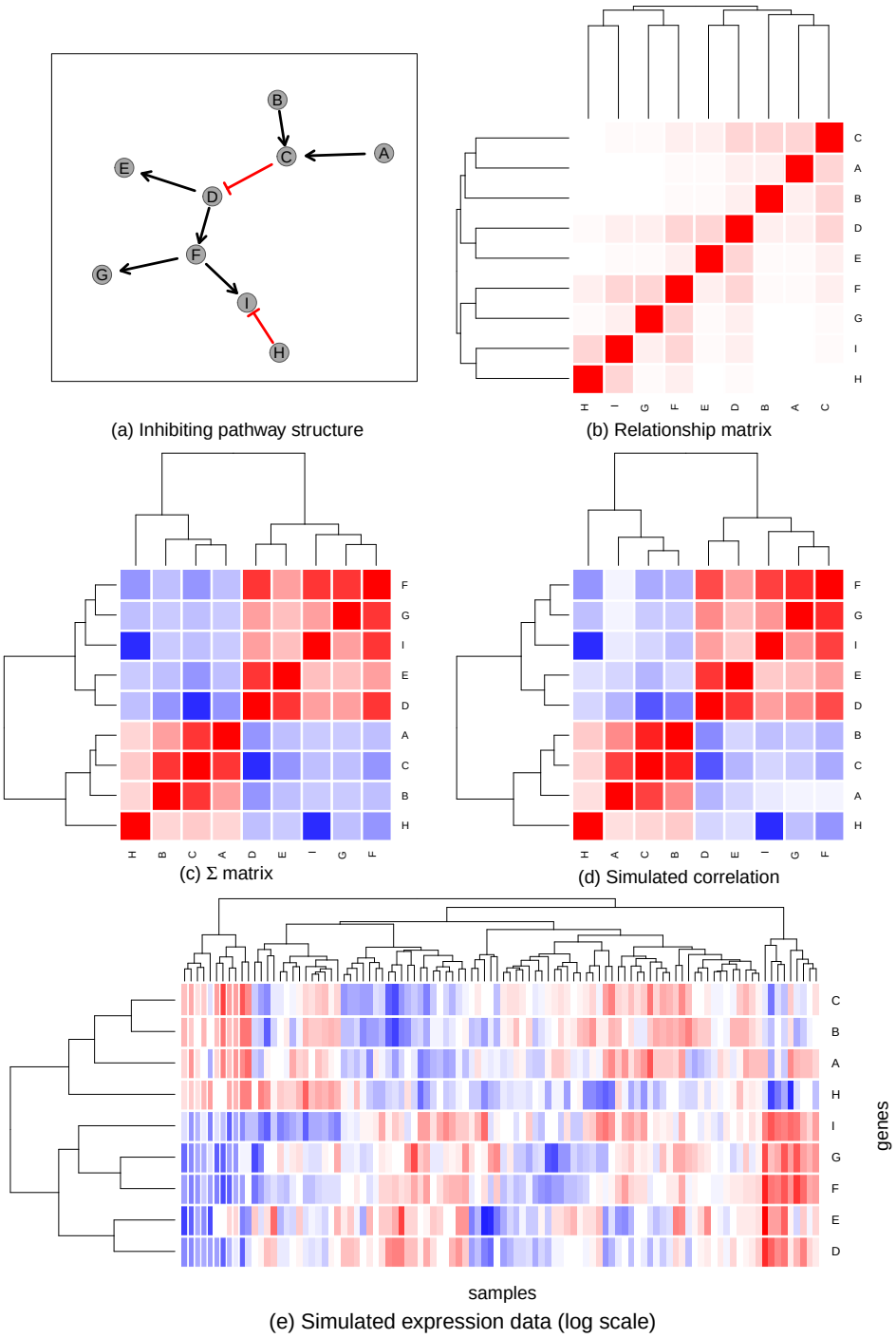


Figure 3: **Simulating expression from graph structure with inhibitions**. An example of a graph structure (a), that has been used to derive a relationship matrix (b), Σ matrix (c), and correlation structure (d), from the relative distances between the nodes (e). These values are coloured blue to red from -1 to 1. This has been used to generate a simulated expression dataset of 100 samples (coloured blue to red from low to high) via sampling from the multivariate normal distribution. Here the inhibitory relationships between genes are reflected in negatively correlated simulated values.

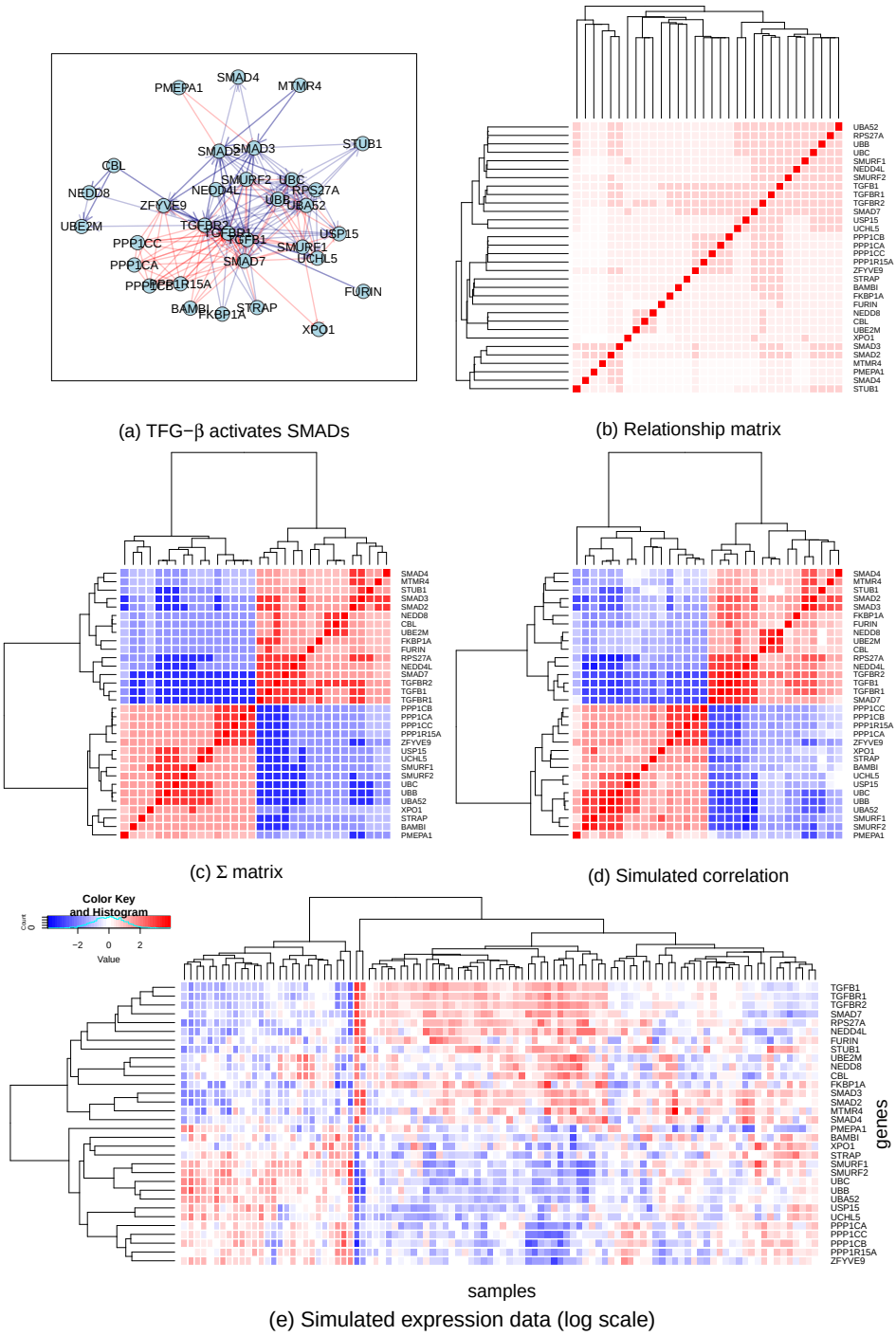


Figure 4: **Simulating expression from a biological pathway graph structure**. The graph structure (a) of a known biological pathway, "TGF-β receptor signaling activates SMADs" (R-HSA-2173789), was used to derive a relationship matrix (b), Σ matrix (c) and correlation structure (d) from the relative distances between the nodes. These values are coloured blue to red from -1 to 1 (e). This has been used to generate a simulated expression dataset of 100 samples (coloured blue to red from low to high) via sampling from the multivariate normal distribution. Here modules of genes with correlated expression can be clearly discerned.

Summary and discussion

Biological pathways are of fundamental importance to understanding molecular biology. In order to translate findings from genomics studies into real-world applications such as improved healthcare, the roles of genes must be studied in the context of molecular pathways. Here we present a statistical framework to simulate gene expression from biological pathways, and provide the `graphsim` package in R to generate these simulated datasets. This approach is versatile and can be fine-tuned for modelling existing biological pathways or for testing whether constructed pathways can be detected by other means. In particular, methods to infer biological pathways and gene regulatory networks from gene expression data can be tested on simulated datasets using this framework. The package also enables simulation of complex gene expression datasets to test how these pathways impact on statistical analysis of gene expression data using existing methods or novel statistical methods being developed for gene expression data analysis. This approach is intended to be applied to bulk gene expression data but could in principle be adapted to modelling single-cell or different modalities such as genome-wide epigenetic data.

Computational details

The results in this paper were obtained using R 4.0.2 with the `igraph` 1.2.5 `Matrix` 1.2-17, `matrixcalc` 1.0-3, and `mvtnorm` 1.1-1 packages. R itself and all dependent packages used are available from the Comprehensive Archive Network (CRAN) at https://CRAN.R-project.org. The `graphsim` 1.0.0 package presented can be installed from CRAN and the issues can be reported to the development version on GitHub (https://github.com/TomKellyGenetics/graphsim). This package is included in the `igraph.extensions` library on GitHub (https://github.com/TomKellyGenetics/igraph.extensions) which installs various tools for `igraph` analysis. This software is cross-platform and compatible with installations on Windows, Mac, and Linux operating systems. The package GitHub repository also contains vignettes with more information and examples on running functions released in the package. Updates to the package (`graphsim` 1.0.0) will be released on CRAN.

Complete examples of code needed to produce the figures in this paper are available in the Rmarkdown version in the package GitHub repository (https://github.com/TomKellyGenetics/graphsim/paper). Further details are available in the vignettes as well.

Acknowledgements

This package was developed as part of a PhD research project funded by the Postgraduate Tassell Scholarship in Cancer Research Scholarship awarded to STK. We thank members of the Laboratory of Professor Satoru Miyano at the University of Tokyo, Institute for Medical Science, Professor Seiya Imoto, Associate Professor Rui Yamaguchi, and Dr Paul Sheridan (Assistant Professor at Hirosaki University,CSO at Tupac Bio) for helpful discussions in this field. We also thank Professor Parry Guilford at the University of Otago, Professor Cristin Print at the University of Auckland, and Dr Erik Arner at the RIKEN Center for Integrative Medical Sciences for their excellent advice during this project.

Author Contributions

S.T.K. and M.A.B. conceived of the presented methodology. S.T.K. developed the theory and performed the computations. M.A.B. provided guidance throughout the project and gave feedback on the package. All authors discussed the package and contributed to the final manuscript.

References

Arner, E., Daub C.O., Vitting-Seerup K., Andersson R., Lilje B., F. Drabløs, A. Lennartsson, et al. 2015. "Transcribed enhancers lead waves of coordinated transcription in transitioning mammalian cells." *Science* 347 (6225): 1010–4.

Barabási, A. L., and Z. N. Oltvai. 2004. "Network Biology: Understanding the Cell's Functional Organization." *Nat Rev Genet* 5 (2): 101–13.

Bates, D., and M. Maechler. 2016. *Matrix: Sparse and Dense Matrix Classes and Methods*. https://CRAN.R-project.org/package=Matrix.

Cannoodt, R., Saelens W., Deconinck L., and Saeys Y. 2020. "Dyngen: A Multi-Modal Simulator for Spearheading New Single-Cell Omics Analyses." *bioRxiv*. Cold Spring Harbor Laboratory. https://doi.org/10.1101/2020.02.06.936971.

Croft, D., Mundo A.F., Haw R., Milacic M., Weiser J., Wu G., Caudy M., et al. 2014. "The Reactome pathway knowledgebase." Journal Article. *Nucleic Acids Res* 42 (database issue): D472–D477. https://doi.org/10.1093/nar/gkt1102.

Csardi, G., and T. Nepusz. 2006. "The Igraph Software Package for Complex Network Research." *InterJournal* Complex Systems: 1695. http://igraph.org.

Dibaeinia, P., and Sinha S. 2019. "A Single-Cell Expression Simulator Guided by Gene Regulatory Networks." *bioRxiv*. Cold Spring Harbor Laboratory. https://doi.org/10.1101/716811.

Genz, A., and F. Bretz. 2009. "Computation of Multivariate Normal and t Probabilities." In *Lecture Notes in Statistics*. Vol. 195. Heidelberg: Springer-Verlag.

Genz, A., Bretz F., Miwa T., Mi X., Leisch F., Scheipl F., and T. Hothorn. 2016. *Mvtnorm: Multivariate Normal and T Distributions*. http://CRAN.R-project.org/package=mvtnorm.

Hawe, J. S., Theis F. J., and M. Heinig. 2019. "Inferring Interaction Networks From Multi-Omics Data." *Front Genet* 10: 535.

Higham, N. J. 2002. "Computing the Nearest Correlation Matrix–a Problem from Finance." *IMA Journal of Numerical Analysis* 22 (3). Oxford University Press: 329–43. https://doi.org/10.1093/imanum/22.3.329.

Hirose, O., Yoshida R., Imoto S., Yamaguchi R., Higuchi T., Charnock-Jones D. S., Print C., and Miyano S. 2008. "Statistical Inference of Transcriptional Module-Based Gene Networks from Time Course Gene Expression Profiles by Using State Space Models." *Bioinformatics* 24 (7): 932–42. https://doi.org/10.1093/bioinformatics/btm639.

Hu, J. X., C. E. Thomas, and Brunak S. 2016. "Network biology concepts in complex disease comorbidities." *Nat. Rev. Genet.* 17 (10): 615–29.

Komatsu, M., T. Yoshimaru, T. Matsuo, K. Kiyotani, Y. Miyoshi, T. Tanahashi, K. Rokutan, et al. 2013. "Molecular features of triple negative breast cancer cells by genome-wide gene expression profiling analysis." *Int. J. Oncol.* 42 (2): 478–506.

Law, C. W., Chen Y., Shi W., and Smyth G. K. 2014. "voom: Precision weights unlock linear model analysis tools for RNA-seq read counts." *Genome Biol.* 15 (2). Springer Nature: R29. https://doi.org/10.1186/gb-2014-15-2-r29.

Li, P., Piao Y., Shon H. S., and Ryu K. H. 2015. "Comparing the normalization methods for the differential analysis of Illumina high-throughput RNA-Seq data." *BMC Bioinformatics* 16 (October): 347.

Markowetz, F., and Spang R. 2007. "Inferring cellular networks–a review." *BMC Bioinformatics* 8 Suppl 6 (September): S5.

Matthew E R., and Wu Phipson B., Hu Y., Law C.W., Shi W., and Smyth G.K. 2015. "limma Powers Differential Expression Analyses for RNA-Sequencing and Microarray Studies." *Nucleic Acids Research* 43 (7): e47.

Ozsolak, F., and Milos P. M. 2011. "RNA sequencing: advances, challenges and opportunities." *Nat. Rev. Genet.* 12 (2): 87–98.

Perou, C. M., Sørlie T., Eisen M. B., van de Rijn M., Jeffrey S. S., Rees C. A., Pollack J. R., et al. 2000. "Molecular portraits of human breast tumours." *Nature* 406 (6797): 747–52.

Saelens, W., Cannoodt R., Todorov H., and Saeys Y.. 2019. "A comparison of single-cell trajectory inference methods." *Nat. Biotechnol.* 37 (5): 547–54.

Schaffter, T.., Marbach D.., and Floreano D.. 2011. "GeneNetWeaver: in silico benchmark generation and performance profiling of network inference methods." *Bioinformatics* 27 (16): 2263–70.

Shimamura, T.,. Imoto S, Yamaguchi R., Fujita A., Nagasaki M., and Miyano S. 2009. "Recursive Regularization for Inferring Gene Networks from Time-Course Gene Expression Profiles." *BMC Systems Biology* 3 (1): 41. https://doi.org/10.1186/1752-0509-3-41.

Svensson, V., R. Vento-Tormo, and Teichmann S. A. 2018. "Exponential scaling of single-cell RNA-seq in the past decade." *Nat Protoc* 13 (4): 599–604.

Wang, J., Huang M., Torre E., Dueck H., Shaffer S., J.m Murray, Raj A., Li M., and Zhang N. R. 2018. "Gene expression distribution deconvolution in single-cell RNA sequencing." *Proc. Natl. Acad. Sci. U.S.A.* 115 (28): E6437–E6446.

Yamaguchi, R., Yoshida R., Imoto S., Higuchi T., and Miyano S. 2007. "Finding module-based gene networks with state-space models - Mining high-dimensional and short time-course gene expression data." *IEEE Signal Processing Magazine* 24 (1): 37–46.

Zappia, L., Phipson B., and Oshlack A. 2017. *Genome Biol.* 18 (1): 174.