Supplementary Material for "MOSim: Multi-Omics Simulation in R"

July 30, 2018

1 Description of the simulation algorithm

`MOSim` simulates multi-omic data sets with a flexible experimental design. To illustrate how the algorithm works, we chose a design of two time-series A and B with 4 time points each and with 3 replicates per time point and experimental group, i.e. 24 samples in total for each omic data type. We will simulate data for RNA-seq, DNase-seq, ChIP-seq, miRNA-seq, Methyl-seq and transcription factors. `MOSim` takes RNA-seq as the central omic data type because the goal is simulating the regulatory mechanisms driving gene expression. Thus, the first omic to be simulated is gene expression (RNA-seq counts).

1.1 Gene expression simulation

Let $\mathbf{x^0}$ be the vector containing the seed counts for RNA-seq for all genes (the STATegra default sample or a sample provided by the user), which has been previously adjusted to the specified sequencing depth. Genes are randomly classified into differentially expressed genes (*DEGs*) and non-differentially expressed genes (*nonDEGs*), where the percentage of DEGs is decided by the user. Genes in *nonDEG* class are labeled as "flat" for all the experimental groups. The DEG class is divided into the following subclasses for each experimental group (Figure S1): FL (flat), CI (continuous induction), CR (continuous repression), TI (transitory induction) and TR (transitory repression). Again, the percentage of DEGs in each subclass is determined by the user. Therefore, when having two experimental groups as in this example, DEG genes can be labeled with the different combinations of profiles (in groups A-B): FL-FL, FL-TR, TI-CR, and so on. Next we describe the steps followed to simulate the gene expression matrix.

In the case of designs not considering time series, users can also choose the percentage of up and down regulated genes and the algorithm will select them randomly. The definition of up or down regualations takes the first experimental group as the reference. When more than two experimental grups are to be simulated, a gene is considered to be up (or down) regulated if presenting an

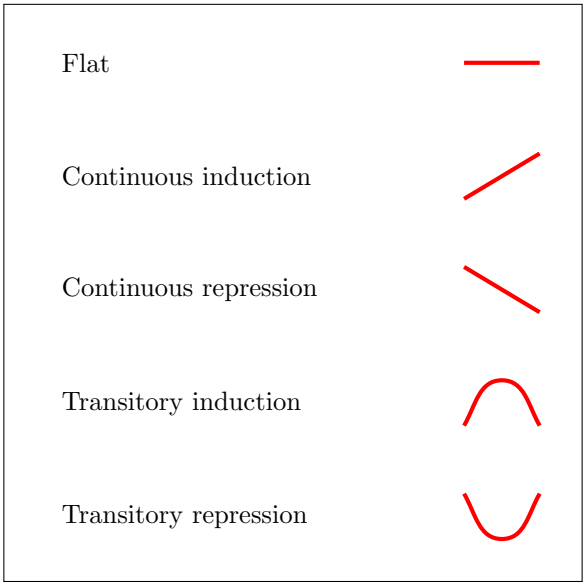

Figure S1: Time profiles representation.

increase (or decrease) in expression in at least one of the conditions with regard to the first one. `MOSim` increases (or decreases) the expression in the seed sample to simulate the change for up (or down) regulated genes in any of the experimental conditions as described below for flat genes in time course designs.

1.1.1 Seed count data for each condition A and B

We duplicate $\mathbf{x^0}$ as many times as the number of experimental groups, two in this case: $\mathbf{x^{0,A}}$ and $\mathbf{x^{0,B}}$. The values for these two vectors are the same, with an only exception: values for genes belonging to FL subclass in DEG class are modified so there is a change in expression between both groups. For a gene g in this subclass, $\mathbf{x_g^{0,B}}$ is replaced by a random integer between $x_g^{0,A} + 10 + P_{80}$ and $x_g^{0,A} + 10 + P_{99}$ if the gene is up-regulated in condition B, where P_{80} and P_{99} are the 80th and 99th percentile of $\mathbf{x^{0,A}}$ after removing values equal to 0, respectively. If the gene is down-regulated in condition B, the range is set to $x_g^{0,A} - 10 - P_{99}$ and $x_g^{0,A} - 10 - P_{80}$. In both cases, the values are restricted to be within the limits $\min\{\mathbf{x^{0,A}}\}$ and $\max\{\mathbf{x^{0,A}}\}$.

When adjusting $\mathbf{x^{0,A}}$ and $\mathbf{x^{0,B}}$ to the user specified depth, the value of every gene g in non-DE group is modified to be the same for all groups, so it is set to the mean value $(x_g^{0,A} + x_g^{0,B})/2$.

1.1.2 Auxiliary random vector r

For the next step, which is generating the time series, we need to create an auxiliary vector **r** that will be used to assure that a changing profile is different enough from a flat profile. Initially, **r** is randomly generated by sampling from $\mathbf{x^0}$, and both of them have the same length. Let P_{30} be the 30th percentile of $\mathbf{x^0}$ excluding zero values. For each gene g that is non-flat in all experimental groups, if $\max\{x_g^0, r_g\} < P_{30}$, r_g is swapped with other value r_j, being gene j randomly chosen from genes in *nonDEG* class whose expression is above P_{30}, so the original data distribution is preserved. When there are not enough values to swap, the remaining ones are taken from a normal distribution of mean P_{30} and standard deviation 1.

We also need to assure that the variation across time for DEG is neither too low or too great so we modify r_g again to force that the difference between x_g^0 and r_g meets some minimum and maximum restrictions. Let d be the difference between x_g^0 and r_g in absolute number, and P_{90} the 90th percentile of $\mathbf{x^0}$ excluding zero values. For each gene g in DEG excluding FL-FL subclass, r_g is modified according to the following scenarios:

- When d is greater than P_{90} and r_g is greater than x_g^0 then $r_g = r_g - (d - P_{90})$
- When d is greater than P_{90} and r_g is lower than x_g^0 then $r_g = r_g + (d - P_{90})$
- When d is lower than P_{30} and r_g is greater than x_g^0 then $r_g = r_g + (P_{30} - d)$
- When d is lower than P_{30} and r_g is lower than x_g^0 then $r_g = r_g - (P_{30} - d)$

1.1.3 Generating the time series for each group

Let us take group A as an example on how to generate the time series (the procedure will be analogous for group B). Let $\mathbf{x^{t,A}}$ be the vector for time t ($t = t_1, t_2, t_3, t_4$). For a gene g with FL profile in group A, for all time points, we define $x_g^{t,A} = (x_g^{t,A} + r_g)/2 + N(0, 0.3)$. The only exception is for genes in DEG with FL subclass in both groups, for which $x_g^{t,A} = x_g^{t,A} + N(0, 0.3)$. For a gene g in the remaining classes the time points are generated as follows:

1. Transform $t = t_1, t_2, t_3, t_4$ into $t^* = 0, 1, 2, 3$, and define $T = \max\{t^*\}$, i.e. $T = 3$ in this case.
2. Let $f(t^*)$ be a function of t^* that takes values in the interval $[0, 1]$ for induction profiles and in $[-1, 0]$ for repression profiles and that is defined as follows:
   (a) $f(t^*) = a_1 + b_1 t$, for CI profile
   (b) $f(t^*) = -a_1 - b_1 t$, for CR profile
   (c) $f(t^*) = a_2 + b_2 t + c_2 t^2$, for TI profile
   (d) $f(t^*) = -a_2 - b_2 t - c_2 t^2$, for TR profile

For continuous profiles, we need $f(0) = 0$, so $a_1 = 0$. Since we also need the maximum (or minimum) value of $f(t^*)$ to be 1 (or -1), $b_1 = 1/T$ necessarily. For transitory profiles, we randomly select the time point t_{max} (or t_{min}) where the maximum (or minimum) is to be reached from the interval $[T*0.25, T*0.75]$, so the maximum (or minimum) is within the central 50% of the time line. Hence, $f'(t_{max})$ (or $f'(t_{min})$) must be 0, and we need $f(t_{max})$ (or $f(t_{min})$) to be 1 (or -1). We also need the minimum (or maximum) $f(t^*)$ value to be 0. To meet all these requirements, the coefficients must take the following values:

- If $t_{max} \geq T/2:$ $\quad a_2 = 0$; $b_2 = 2/t_{max}$; and $c_2 = -1/t_{max}^2$.
- If $t_{max} < T/2:$ $\quad a_2 = 1 + c_2 t_{max}^2 = 1 - t_{max}^2/(T - t_{max})^2$; $b_2 = -2c_2 t_{max} = 2t_{max}/(T - t_{max})^2$; and $c_2 = -1/(T - t_{max})^2$.

3. We define $M_g = max(x_g^0, r_g)$ and $m_g = \min\{x_g^0, r_g\}$, and then we randomly choose integers p_g^R and p_g^I from the intervals $[(M_g + m_g)/2; M_g]$ and $[m_g; (M_g + m_g)/2]$, respectively.

4. The final expression values for each time point are:

   (a) $x_g^{t,A} = p_g^R + (p_g^R - m_g) * f(t^*) + N(0, 0.3)$, for CR and TR classes

   (b) $x_g^{t,A} = p_g^I + (M_g - p_g^I) * f(t^*) + N(0, 0.3)$, for CI and TI classes

1.1.4 Simulating replicates.

The replicates for each gene, time point and condition are generated from a negative binomial (NB) distribution with mean μ and variance σ^2.

Let $x_g^{t,G}$ be the count value of a gene g in group G (A or B) and at time t. The NB mean is set to $\mu_g = \max\{0.1, x_g^{t,G}\}$. To model the dependence between the NB mean and variance, we analyzed several data sets from different omics and experiments, and observed a linear relationship between log-transformed values of means and variances ($R^2 > 0.95$ for all models): $\sigma_g^2 = 10^a * (\mu_g + 1)^b - 1$. To assure a minimum variance we really used $\sigma_g^2 = \max\{0.03; 10^a * (\mu_g + 1)^b - 1\}$. A regression model was applied to estimate coefficients a and b and these estimations were used as default values for each omic that can be changed by users to increase or decrease the default variability.

Once the mean and variance for the NB are set, the replicates are randomly generated from this distribution, and the generated counts are adjusted to the desired sequencing depth.

Due to the NB variability, the average of the replicates $\overline{x}_g^{t,G}$ can be very different from the original means $x_g^{t,G}$ provided, and this can lead to a great variation across time for genes that are supposed to be constant (FL class). To avoid it, we generate a new value $z_g^{t,G}$ per time point and group for each gene g in FL class from the following normal distribution $N(x_g^{t,G}, 0.025 * x_g^{t,G})$. Let d be the difference between $\overline{x}_g^{t,G}$ and $z_g^{t,G}$. The value for each replicate i obtained from the NB distribution ($x_g^{i,t,G}$) is replaced by $x_g^{i,t,G} - d$. As this is done for

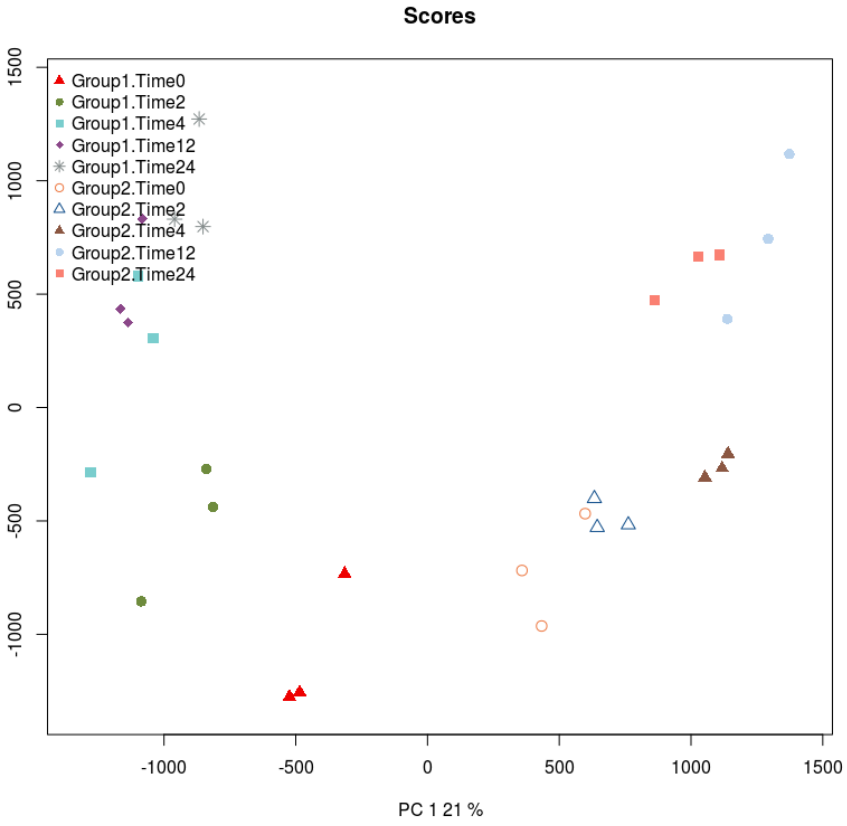

Figure S2: PCA score plot for simulated gene expression data.

each time point, we force the final values of FL genes to be around a similar value for all the time points.

Figure S2 shows a Principal Component Analysis (PCA) score plot on the gene expression data simulated to illustrate how the two time series, time points and replicates behave.

1.2 Simulation of regulatory omics (except methylation and transcription factors)

The other omics supported by the algorithm act as gene expression regulators. This is why an association list linking each omic feature to the corresponding RNA-seq gene identifier is needed. The procedures to simulate methylation data or to obtain TF regulations are different, and will be explained in the next section.

Let $\mathbf{y^0}$ be the vector containing the seed counts for a given regulatory omic

(STATegra default sample or sample provided by the user). All feature IDs must be included in the association list (default list from STATegra project or provided by the user).

`MOSim` considers three types of regulation: activation (A), repression (R) and no regulation or no effect (NE). The steps followed by the algorithm to generate each omic data type and the regulation of each DEG are described next.

1.2.1 Selection of regulators with regulatory effect

Users can choose the percentage of regulators with effect for each omic (otherwise the default values are applied). According to this percentage, the algorithm randomly selects $numE$ regulators from the regulators in the association list. The rest of associations corresponding to regulators in the association list with no effect are labeled as NE.

1.2.2 Initial assignation of regulation type to each association

For regulators with effect, we randomly assign a type of effect (A or R) to each of the regulated genes given by the association list according to the probabilities set by the user. The only exception is that the associations regulator-nonDE gene are flagged with NE tag.

1.2.3 Adjusting the initially assigned regulation type

Adjusting the initially assigned regulation type is sometimes necessary when the regulator regulates more than one DEG. In that case, we analyze the different classes of DEGs in both experimental groups (FL-FL, TR-FL, CI-CR, etc.), select the one with the highest percentage of genes affected by the regulator (no matter if A or R), and name it as "majority class". Genes in the remaining classes are all labeled with NE, except if they have an equal or opposite temporal profile to the previously selected class in any of the experimental groups, and are not FL. We name them as "equal" or "opposite" class, respectively.

We assign the most frequent effect (R or A) to all genes in the "majority class" and for the two experimental groups. In case of ties, we randomly choose R or A by using the initial probabilities. We assign the same or opposite effect to genes in equal or opposite classes in the corresponding experimental groups, and NE effect to the rest of groups.

1.2.4 Assigning profiles to regulators

The profile of a regulator depends on the profile of the genes it regulates and on the type of the effect of the regulator. A regulator with an activator effect for a given experimental group will have the same profile than the genes in its "majority" class for that group, or the unique regulated gene when corresponding. A regulator with a repressor effect will have the opposite profile to the genes in its "majority" class, or the unique regulated gene when corresponding. The opposite profile to FL is FL, to CI is CR (and vice versa), and to TI is TR

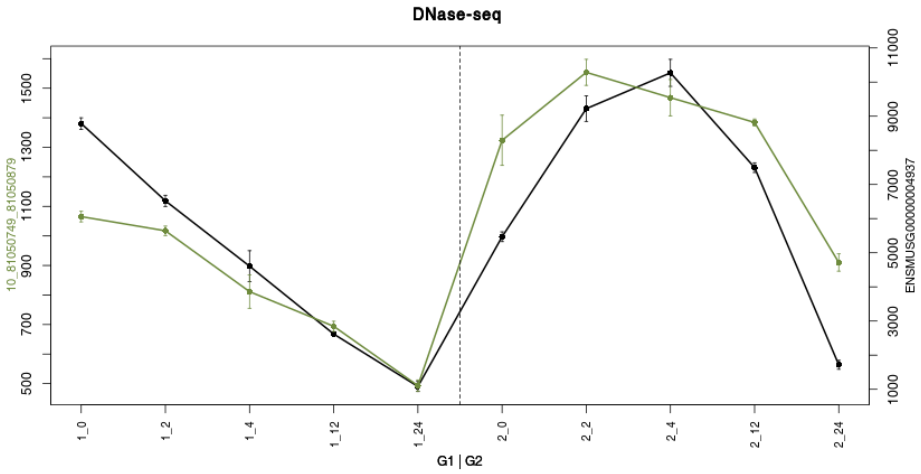

Figure S3: Gene expression and DNase-seq temporal profiles for a random gene.

(and vice versa). If the regulator has NE effect on all its regulated genes, the regulator profile is set to FL.

Regulators of FL-FL genes in DEG class are also assigned a FL-FL profile with different average value for each experimental group. The up-regulated condition will be the same for both the gene and the regulator when the effect is activation. If the effect is repression, the up(down)-regulated condition/s for the gene will be the down(up)-regulated condition/s for the regulator. The new values are generated using the same procedure described for gene expression.

1.2.5 Generating time series and replicates for each group.

Once the profiles have been assigned to each regulator, the time series and replicates generation follows the same procedure described for gene expression. Figure S3 shows an example of a DEG (in black) and the regulation given by the chromatin accessibility of the associated genomic region (in green). In this case, the gene has a continuous repression profile in the first condition and a transitory induction profile in the second condition. The DNase-seq region acts as activator in both conditions and hence has the same type of profile than the gene. A miRNA regulation is illustrated in Figure S4, where the miRNA has a repressor effect and therefore has an opposite temporal profile to the gene.

1.3 Generation of TF expression data

As TFs are genes, their expression is not simulated from scratch but taken from the simulated gene expression data. The TF-target gene association table must be provided (it is available in the package if STATegra default data are used), and `MOSim` classifies as TFs all the genes included in the column "TF" of

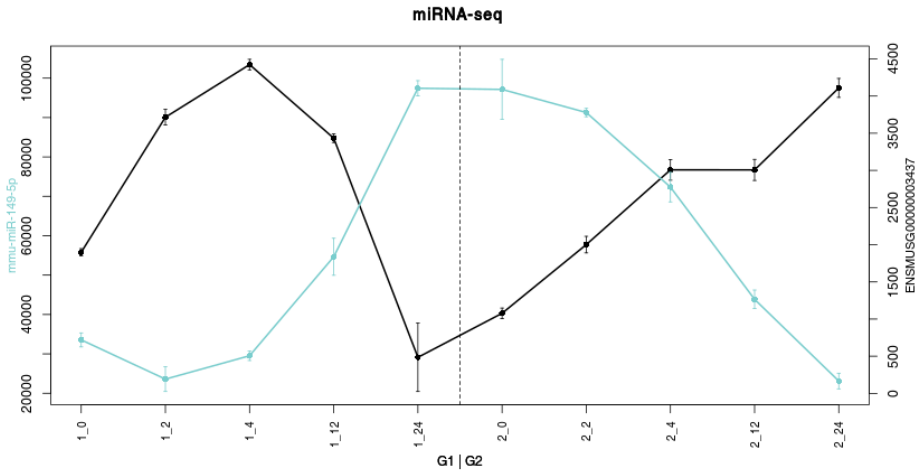

Figure S4: Gene expression and miRNA-seq temporal profiles for a random gene.

such table. Users decide the percentage of DE TFs so, during gene expression simulation, DE genes are selected in such a way that they meet the requirement of the number of DE TFs.

All the associations of non-DE TFs with genes are flagged with NE tag. For each DE TF, we search for the target genes in the association table. If the target gene is DE and the TF and gene have the same profile in a given condition, that condition is labeled with A. If TF and gene have opposite profiles, the condition is labeled with R. NE label is applied in the rest of cases.

1.4 Simulation of methylation data

Unlike the rest of omics, Methyl-seq simulation does not require a vector containing the seed methylation values for every CpG site. `MOSim` just needs the chromosomal position of the CpG sites to be simulated and their association to genes. Again, both files can be taken from STATegra default data or provided by the user.

The algorithm used to simulate bisulfite sequencing methylation data was adapted from the WGBSSuite tool described in Rackham *et al.* (2015). The original simulation procedure can be divided in 4 major steps: simulation of CpG locations, simulation of methylation status for each CpG site, simulation of read coverage per site and, finally, simulation of methylated/un-methylated/transit read counts using a binomial (or truncated negative binomial) distribution.

In the first step, WGBSSuite simulates CpG locations for a single chromosome assuming 4 possible situations: CpG islands -or regions with a high frequency of CpG sites-, CpG deserts and transition states between them in

both directions. As commented before, `MOSim` does not simulate CpG locations but take them from the provided sample so this step can be skipped. Moreover, `MOSim` allows for the simulation of CpG sites in different chromosomes.

In the second step, the methylation status of CpG sites is modeled based on the distance between them. Nearby sites, like those belonging to a CpG island, desert or transition states, are forced to keep the same methylation status, and hence are considered as blocks. `MOSim` takes into account such blocks when calculating the simulation settings. All CpG sites in the same block will share hence the same profile and regulatory effect. The profile for each CpG block is determined as in the rest of regulatory omics, and depends on the profile of the regulated gene and on the type of effect (activation or repression).

Once the initial methylation values are generated for each experimental group, the procedure described in the previous sections is applied with some adjustments to consider blocks, and to take into account that methylation values vary from 0 to 1. The original WGBSSuite algorithm only contemplates the possibility of simulating 2 groups so it had to be adapted to cover other experimental designs. The probability of success in the binomial distribution used to generate the counts in each group is modulated in order to mimic the changes between two conditions. The WGBSSuite algorithm was adapted to simulate the temporal profiles. The replicates are obtained from the binomial distribution, instead of the negative binomial distribution applied for the rest of omics.

The generated data can be returned as β values (proportion of methylated/unmethylated reads) or as M values, with $M_i = log_2(\gamma_i/1 - \gamma_i)$ where $\gamma_i = \min\{\max\{\beta_i, threshold\}, 1 - threshold\}$ for every CpG site i, with threshold taking the value 0.01.

References

Rackham, O. J. L., Dellaportas, P., Petretto, E., and Bottolo, L. (2015). Wgbssuite: simulating whole-genome bisulphite sequencing data and benchmarking differential dna methylation analysis tools. *Bioinformatics*, **31**(14), 2371–2373.