Supporting Information for: Denser sampling may be more effective than repeat experiments for high throughput time series studies

Emre Sefer, Michael Kleyman, and Ziv-Bar Joseph

Computational Biology Department, School of Computer Science, Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh, PA 15213, USA
{zivbj}@cs.cmu.edu

Supporting code and datasets: www.sb.cs.cmu.edu/repeats

Supporting Methods

Estimating Eq. 4 for a step function Repeating the pairwise comparison in Eq. 5 for all points in S_i and M_i returns the folllowing $i-1$ and $T-i$ distributions to be satisfied respectively:

$$p(\sum_{a=1}^{j}\sum_{z=1}^{n_r} d_{a:z} \leq n_r \frac{j}{2}), \quad j \in 1, \dots, i-1 \tag{9}$$

$$p(\sum_{a=i}^{j-1}\sum_{z=1}^{n_r} d_{a:z} \geq n_r \frac{j-i}{2}), \quad j \in i+1, \dots, T \tag{10}$$

$d_{a:z}$ terms in Eq. 9 and Eq. 10 are independent of each other, so the probability of selecting t_i in Eq. 4 can be separated into two integrals as in:

$$p(s_r = t_i^r | c_r = 1, s_g, c_g, \sigma^2) = p\big(\mathcal{L}(t_i, 1) > \mathcal{L}(t_j, 1), \forall j \neq i\big) = p\big(\mathcal{L}(t_i, 1) > \mathcal{L}(t_j, 1), \forall j \in S_i\big)\, p\big(\mathcal{L}(t_i, 1) > \mathcal{L}(t_j, 1), \forall j \in M_i\big) \tag{11}$$

where $p\big(\mathcal{L}(t_i, 1) > \mathcal{L}(t_j, 1), \forall j \in S_i\big)$ is the probability of the likelihood defined by t_i being higher than the likelihood of all other points that are smaller than t_i. $d_{a:z}$ variables for each time point t_a in S_i have acyclic dependencies between them, $d_{a:z}$ variables depend only on the variables of time points between t_a and t_{i-1}. Due to the existence of this ordering between variables, $p\big(\mathcal{L}(t_i, 1) > \mathcal{L}(t_j, 1), \forall j \in S_i\big)$ can be expressed by the following nested integral:

$$p(\mathcal{L}(t_i, 1) > \mathcal{L}(t_j, 1), \forall j \in S_i) = \int_{-\infty}^{\frac{n_r}{2}} p(\hat{d}_{i-1} | m_{i-1}^{i}, \sigma_{i-1}^{i}) \int_{-\infty}^{n_r - \hat{d}_{i-1}} p(\hat{d}_{i-2} | m_{i-2}^{i-1}, \sigma_{i-2}^{i-1}) \dots \int_{-\infty}^{n_r \frac{i-1}{2} - \sum_{t=2}^{i-1} \hat{d}_t} p(\hat{d}_1 | m_1^2, \sigma_1^2)\, d_{\hat{d}_1} \dots d_{\hat{d}_{i-2}} d_{\hat{d}_{i-1}} \tag{12}$$

where $\hat{d}_{i-1} = \sum_{z=1}^{n_r} d_{i-1:z}$ is a variable for summation of all repeats for the $i-1$'th time point. Each $\hat{d}_j$ is distributed gaussian with mean $m_j^{j+1} = n_r \sum_{m=j,\, t_m \geq s_g}^{j} 1$ and standard deviation $\sigma_j^{j+1} = \sigma\sqrt{n_r}$. The gaussians are independent of each other over interval $[-\infty, \frac{n_r}{2}]$, so this becomes:

$$p\big(\mathcal{L}(t_i, 1) > \mathcal{L}(t_j, 1), \forall j \in S_i\big) = A + \int_{-\infty}^{\frac{n_r}{2}} p(\hat{d}_{i-1}) \int_{\frac{n_r}{2}}^{n_r - \hat{d}_{i-1}} p(\hat{d}_{i-2}) \dots \int_{\frac{n_r}{2}}^{n_r \frac{i-1}{2} - \sum_{t=2}^{i-1} \hat{d}_t} p(\hat{d}_1)\, d_{\hat{d}_1} \dots d_{\hat{d}_{i-2}} d_{\hat{d}_{i-1}} \tag{13}$$

where $A = \prod_{j \in S_i} \Phi(\frac{n_r}{2}, m_j^{j+1}, \sigma_j^{j+1})$. Eq. 13 can be efficiently estimated by Gaussian quadrature or by MCMC [1]. We use similar derivation to estimate $p\big(\mathcal{L}(t_i, 1) > \mathcal{L}(t_j, 1), \forall j \in M_i\big)$. For large T^d, exact estimation of nested multidimensional integral in Eq. 11 can be complicated so we instead estimate its upper and lower bounds as below.

Estimating upper and lower bounds Exact estimation of nested multidimensional integral in Eq. 11 can be complicated for large T^d. In this case, we can rather estimate its lower and upper bounds quite efficiently. $\prod_{j \neq i} p(\mathcal{L}(t_i, 1) > \mathcal{L}(t_j, 1))$ gives a lower bound estimate since these pairwise terms are not originally independent. We can estimate an upper bound of $p\big(\mathcal{L}(t_i, 1) > \mathcal{L}(t_j, 1), \forall j \in S_i\big)$ as follows: Let $D(n)$ be upper bound of the integral defined only by the topmost n equations in (9). By approximating the multi-dimensional integral symmetrically, upper bound can be estimated recursively by:

$$D(n+1) = D(n)\Big(1 - \frac{1}{n+1}(1 - A_{i-n-1:i-n})\Big) \tag{14}$$

with base case $D(1) = A_{i-1:i} = \Phi(\frac{n_r}{2}, m_{i-1}^{i}, \sigma_{i-1}^{i})$, and its upper bound is given by $D(i-1) = \frac{\prod_{j=1}^{i-1}(A_{j:j+1}+i-1)}{(i-2)!}$. Similarly, upper bound of $p\big(\mathcal{L}(t_i, 1) > \mathcal{L}(t_j, 1), \forall j \in M_i\big)$ can be estimated by:

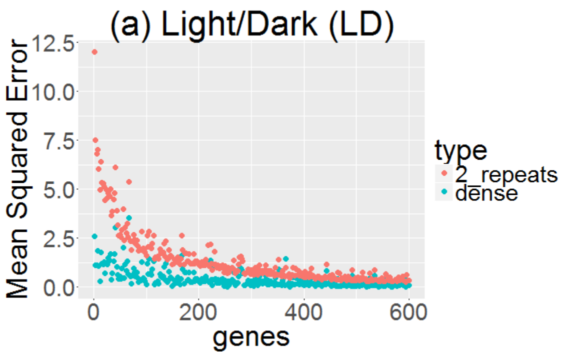
$$U(n+1) = U(n)\Big(1 - \frac{A_{i+n:i+n+1}}{n+2-i}\Big) \tag{15}$$

where $U(n)$ is upper bound of the integral defined only by the topmost $n-i+1$ equations in (10), and base case is $U(i) = 1 - A_{i:i+1} = 1 - \Phi(\frac{n_r}{2}, m_i^{i+1}, \sigma_i^{i+1})$. Solution of this recursion is $U(T-1) = \frac{\prod_{j=i}^{T-1}(j-i+1-A_{j:j+1})}{(T-i)!}$. Let I^d be the vector points in T^d ordered by their absolute distance from s_g. Once upper bound of Eq. 11 is estimated, we can estimate the corresponding lower bound of $E(f_{\text{mis}})$ by Algorithm 1. Upper bound of $E(f_{\text{mis}})$ can be estimated similarly by the same algorithm where we use lower bound of Eq. 11 instead of its upper bound estimation in Lines $5-9$.

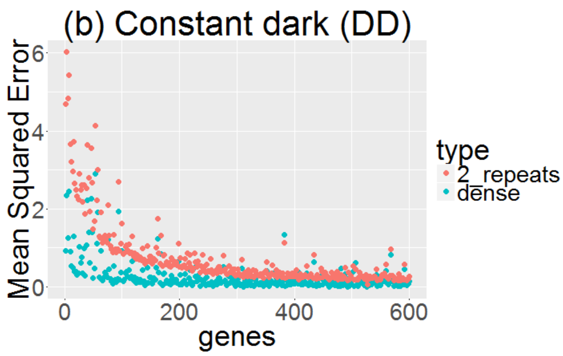
Algorithm 1 Table 1 related to Methods: An algorithm for computing a lower bound for $E(f_{\text{mis}})$.

1: $r = 1$, $d = 0$ $\{r$ is the remaining probability mass, d is the expected distance$\}$
2: Let I be an ordering of the points in T w.r.t. their distance from s_g
3: **while** $I \neq \emptyset$ **do**
4: $\quad t_i \leftarrow$ first point in I; $I = I \setminus t_i$
5: $\quad lb_i = 1$
6: $\quad$ **for** $t_j \in I$ **do**
7: $\quad\quad c_j = P(\mathcal{L}(t_i, 1) > \mathcal{L}(t_j, 1))$
8: $\quad\quad lb_i = lb_i c_j$
9: $\quad$ **end for**
10: $\quad d = d + r lb_i(|t_i - s_g|)$
11: $\quad r = r(1 - lb_i)$
12: **end while**
13: return d

Supporting Figure



(a) Light/Dark (LD)



(b) Constant dark (DD)

Figure S1, related to Figure 4: Using piecewise linear curves to compare sampling strategies over all genes exhibiting circadian and diel rhythms. Genes sorted by absolute MSE difference between *Dense* and *Repeat*$_2$ when using 8 experiments over LD and DD data respectively. Similar to Figure 7 which performs the same comparison using splines we see that *Dense* generally greatly outperforms *Repeat*on this data.

Data S1. Software Source Code, Related to Experimental Procedures

References

1. Press, W.H.: Numerical recipes 3rd edition: The art of scientific computing. Cambridge university press (2007)