Tim

A strategy for generating negative control pairs: version # 2

I describe another, more workable (and simpler) strategy for generating negative control pairs. Again, define the following variables.

- The number of negative control gRNAs N_{grna}. Label the NT gRNAs $1, 2, \ldots, N_{\text{grna}}$.
- The $N_{\text{cell}} \times N_{\text{gene}}$ matrix of gene expressions and the N_{cell}-dimensional gRNA-to-cell assignment vector.
- The number of pairs to generate N_{pairs}.
- The undercover group size $k \leq N_{\text{grna}}/2$.
- The minimum number of treatment cells N_{trt} and control cells N_{cntrl} needed for a pair to pass pairwise QC.

We proceed in several steps.

Step 1: Tabulate the number of of cells with nonzero expression for each individual NT gRNA and gene. First, we compute an $N_{\text{grna}} \times N_{\text{gene}}$ matrix M, where entry (i, j) is the number of cells containing NT gRNA i with nonzero expression of gene j. We easily can construct this matrix either in memory or out-of-core by summing over columns of the gene expression matrix.

Step 2: Determine if it is feasible to enumerate the possible undercover gRNA groups. We check the value of $N_{\text{possible-groups}} := \binom{N_{\text{gene}}}{k}$. If $N_{\text{possible-groups}}$ is a huge number (e.g, $\binom{100}{50} \approx 10^{30}$), then it is not possible to enumerate the possible undercover gRNA groups. If $N_{\text{possible-groups}}$ is small, by contrast (e.g., $\binom{9}{2} = 36$), then it is possible to enumerate the possible undercover gRNA groups. We check if $N_{\text{possible-groups}}$ exceeds some pre-defined threshold (e.g., 20,000), carrying out a different routine in either case. If $N_{\text{possible-groups}} \leq 20,000$, then we proceed to step 3a. Otherwise, we proceed to step 3b.

Step 3a: Enumerate the possible undercover gRNA groups. If

$N_{\text{possible-groups}} \leq 20,000$, we enumerate the possible undercover gRNA groups. We map each possible undercover gRNA group to a length-k vector of integers sorted in increasing order, where the integers represent individual NT gRNAs. For example, the undercover gRNA group containing NT gRNAs 2, 3, and 7 (arbitrarily labeled) would be mapped to the vector `[ 2, 3, 7 ]`. We then generate the entire set of $N_{\text{possible-groups}}$ length-k vectors containing integers in the range $\{1, \ldots, N_{\text{grna}}\}$. We store these vectors in an ordered list `x`. We also set $N_{\text{grna-groups}} = N_{\text{possible-groups}}$.

Step 3b: Sample a set of possible undercover gRNA groups. If $N_{\text{possible-groups}} > 20,000$, then we do not attempt to enumerate the entire set of possible undercover gRNA groups. Instead, we sample a set of undercover gRNA groups. We proceed as follows. First, we estimate the fraction of undercover gRNA-gene pairs (of group size k) that passes QC. We do this by pairing a randomly generated undercover gRNA group to a randomly selected gene and checking if that pair passes the pairwise QC threshold. We sample (with replacement) a large number (e.g., $5,000$) undercover gRNA-gene pairs in this way, producing an estimate $\hat{p}$ of the fraction of undercover gRNA-gene pairs that passes QC. We then set the number of gRNA groups to sample $N_{\text{grna-groups}}$ to

$$N_{\text{grna-groups}} = \frac{c \cdot N_{\text{pairs}}}{\hat{p} N_{\text{genes}}},$$

where $c > 1$ is a number that ensures we sample a *conservative* number of gRNA groups (i.e., more than we need). Finally, we sample $N_{\text{grna-groups}}$ gRNA groups by sampling from the set of length k vectors containing integers in the range $\{1, \ldots, N_{\text{grna}}\}$ via membership checking sampling.* We store these $N_{\text{grna-groups}}$ vectors in an ordered list `x`.

* To be more specific, we sample $N_{\text{grna-groups}}$ gRNA groups as follows. We initialize an empty set (implemented as a hash table) $\mathcal{D}$. We then construct a length-k sample from the set $\{1, \ldots, N_{\text{grna}}\}$ via Fisher-Yates sampling and sort the resulting vector. Finally, we check for inclusion of this vector in $\mathcal{D}$. If the vector already is in $\mathcal{D}$, we proceed to the next iteration. Otherwise, we add this vector to $\mathcal{D}$. We conclude this process when the number of elements in $\mathcal{D}$ is equal to $N_{\text{grna-groups}}$. In the rare case that $N_{\text{grna-groups}} \geq N_{\text{possible-groups}}$, we can construct the gRNA groups via enumerating over all combinations, as in step 3a.

Step 4: Sample without replacement from the set of undercover gRNA group-gene pairs. The final step is to sample a set of undercover gRNA group-gene pairs without replacement. Recall that step 3 yields a list `x` of undercover gRNA groups of length $N_{\text{grna-groups}}$. (This is true whether we have carried out step 3a or step 3b). There are thus $N_{\text{grna-group}} \cdot N_{\text{gene}}$ pairs that we could sample. We map each gRNA group-gene pair in this set of pairs to an integer in the set $\{1, \ldots, N_{\text{gene}} \cdot N_{\text{grna-group}}\}$. The map is defined as follows: for an integer $i \in \{1, \ldots, N_{\text{gene}} \cdot N_{\text{grna-group}}\}$, we carry out the integer division

$$\texttt{grna\_group\_idx} = i\%/\%N_{\text{gene}},$$

which defines a gRNA group index. Next, we compute the remainder of this division

$$\texttt{gene\_idx} = i\%\%N_{\text{gene}}$$

to compute a gene index. Through this map, sampling without replacement from the set of integers $\{1, \ldots, N_{\text{gene}} \cdot N_{\text{grna-group}}\}$ is identical to sampling without replacement from the set of undercover gRNA group-gene pairs.

If the number of pairs to sample N_{pairs} exceeds the number of pairs that we possibly could sample $N_{\text{gene}} \cdot N_{\text{grna-group}}$, then we iterate through the pairs one-by-one, checking if the pair passes pairwise-QC and, if so, adding it to the set pairs to return. If, on the other hand, the number of pairs to sample N_{pairs} is less than $N_{\text{gene}} \cdot N_{\text{grna-group}}$, we sample without replacement from the set $\{1, \ldots, N_{\text{gene}} \cdot N_{\text{grna-group}}\}$, discarding those pairs that do not pass QC. (We can implement this final sampling without replacement step via Fisher-Yates sampling or sparse Fisher-Yates sampling.)

Background on without replacement sampling See the preprint "Simple, Optimal Algorithms for Random Sampling without Replacement" (Ting, 2021) for descriptions of Fisher-Yates, sparse Fisher-Yates, and membership checking algorithms for without replacement sampling.