Empirical QC on Replogle data

Tim

2024-03-13

In this writeup I investigate the question of pairwise QC on the Replogle rd7 data. First, I condition on `n_nonzero_trt` and `n_nonzero_cntrl` to see if there is evidence of inflation due to selection bias. Next, I implement the Poisson-based pairwise QC, retaining pairs for which the two-sided pilot alternative p-value is sufficiently small. Finally, I check to see whether implementing this alternative pairwise QC strategy helps to resolve miscalibration.

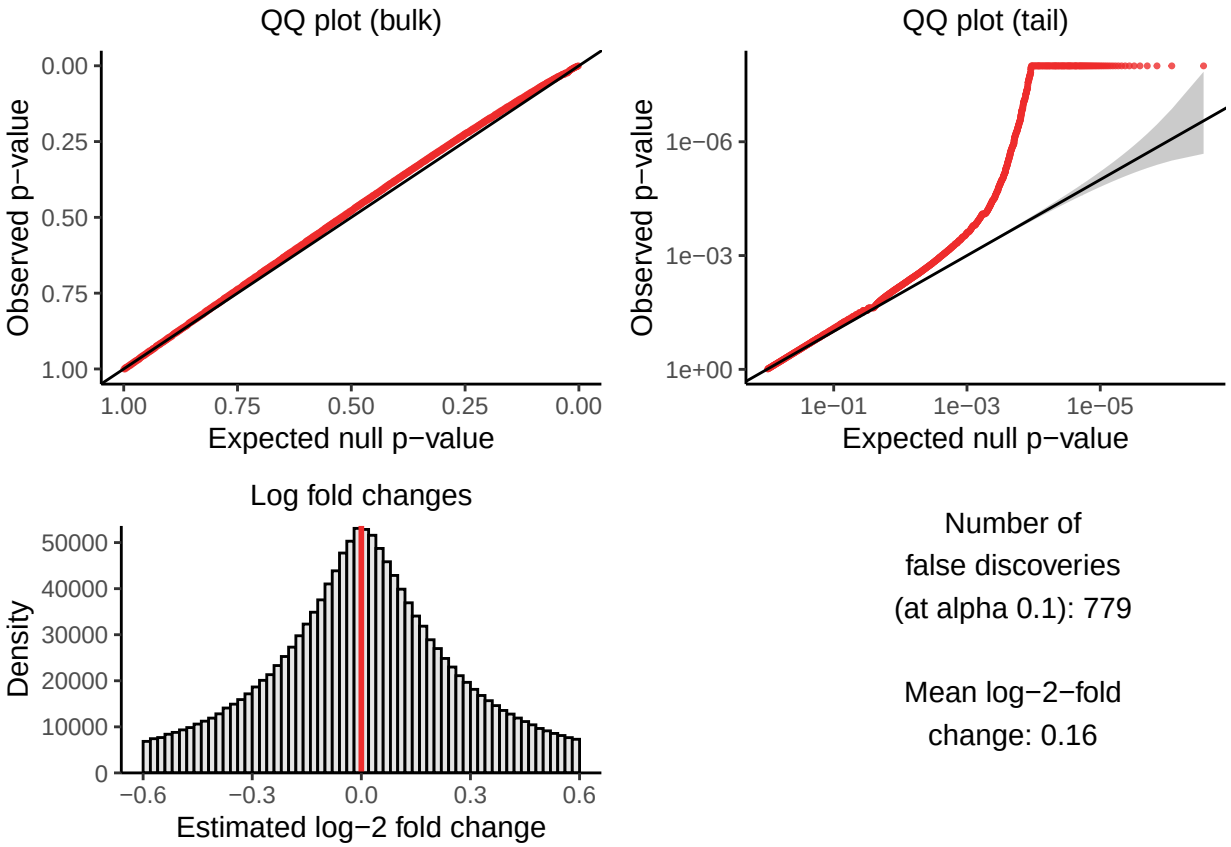
Takeaway: the more sophisticated QC strategy **does not** seem to improve calibration.

Filtering on `n_nonzero_trt` and `n_nonzero_cntrl`

I applied `sceptre` to analyze the negative control pairs of the rd7 dataset. I restricted my attention to pairs for which `n_nonzero_trt` and `n_nonzero_cntrl` are greater than or equal to 1. (We might consider relaxing this restriction in a subsequent analysis.)

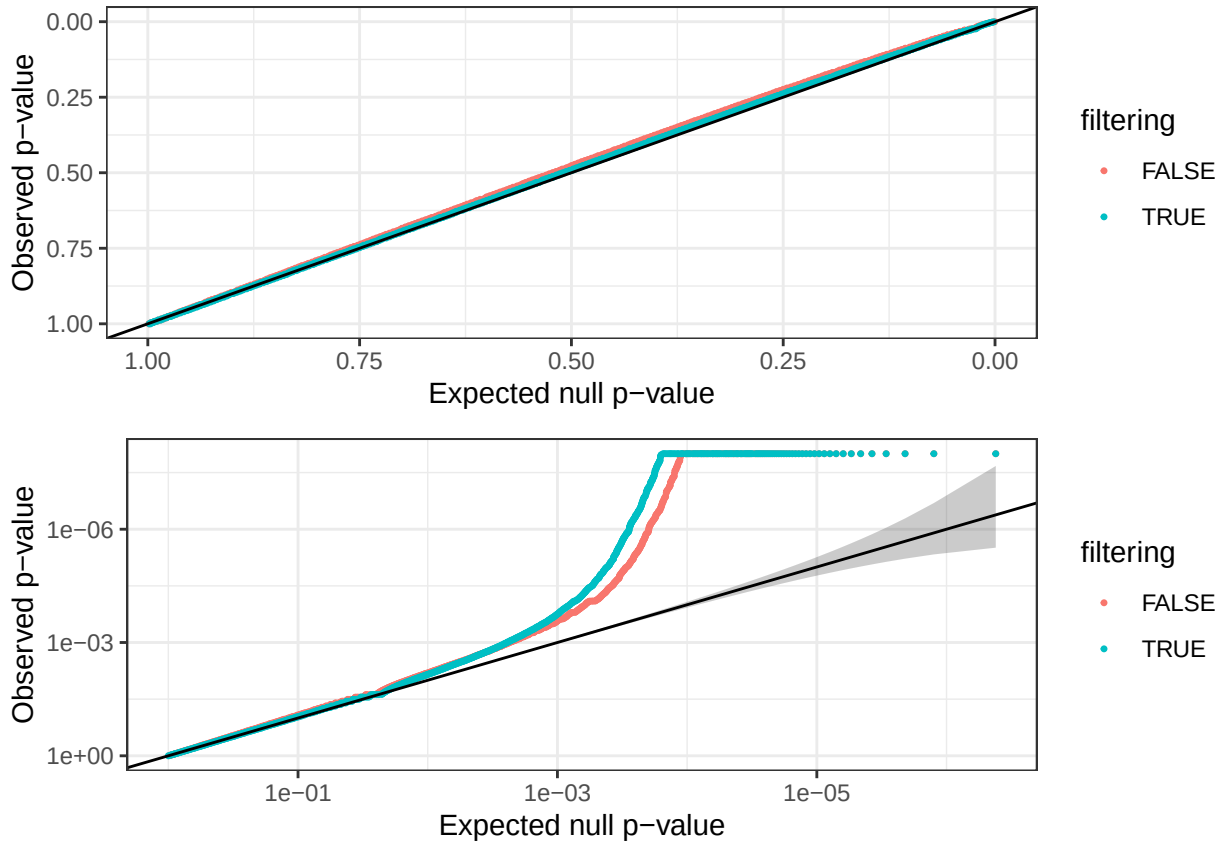
```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.0     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
## [conflicted] Will prefer dplyr::filter over any other package.
```

I start by rendering the standard calibration check plot.



Clearly, `sceptre` is pretty severely miscalibrated on the rd7 dataset.

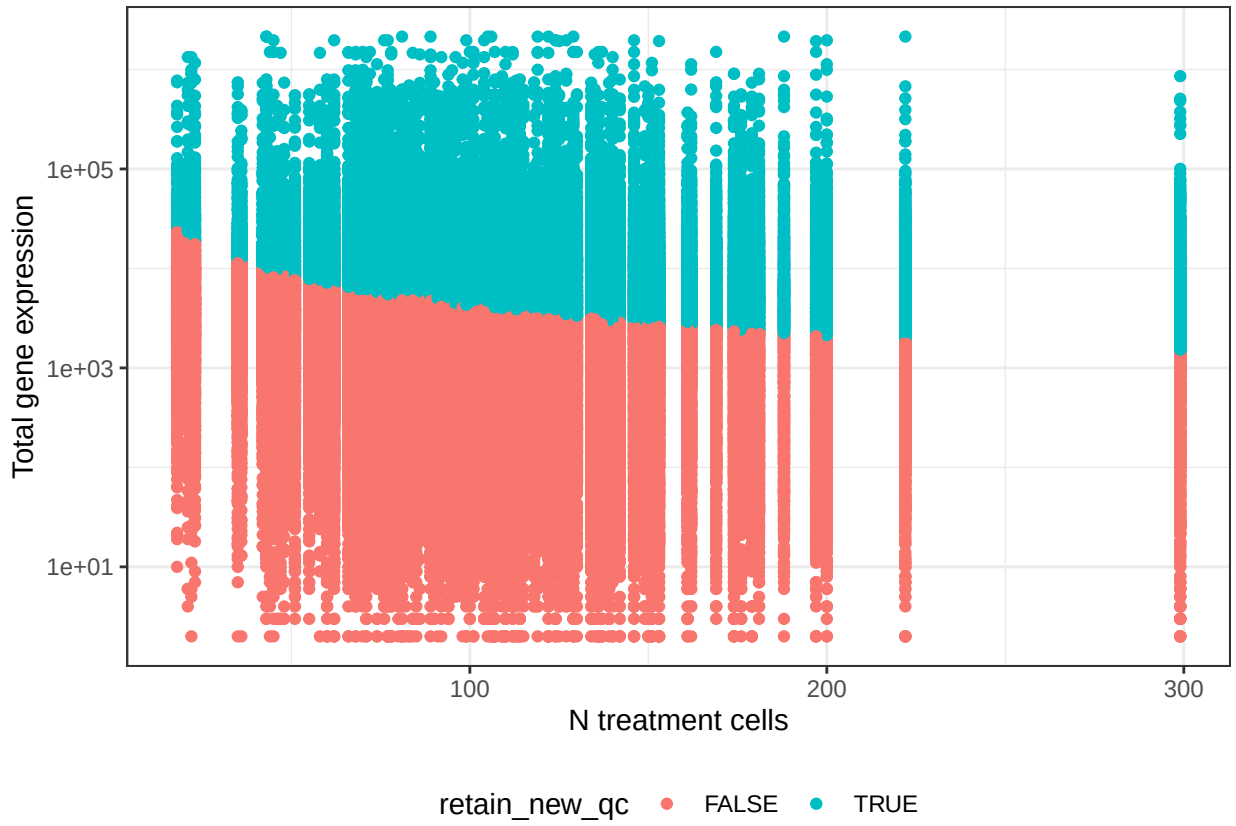
Below, I plot two sets of pairs: the entire set of pairs, and the pairs for which `n_nonzero_trt` and `n_nonzero_cntrl` are greater than or equal to 7. I downsample the former set of pairs such that the former set of pairs and the latter set of pairs are equal in number.



The untransformed (resp., transformed) QQ plot is shown on the top (resp., bottom). The two sets of pairs seem to exhibit roughly equal calibration. Thus, we do not see immediate evidence that filtering on `n_nonzero_trt` and `n_nonzero_cntrl` introduces selection bias to the rd7 dataset.

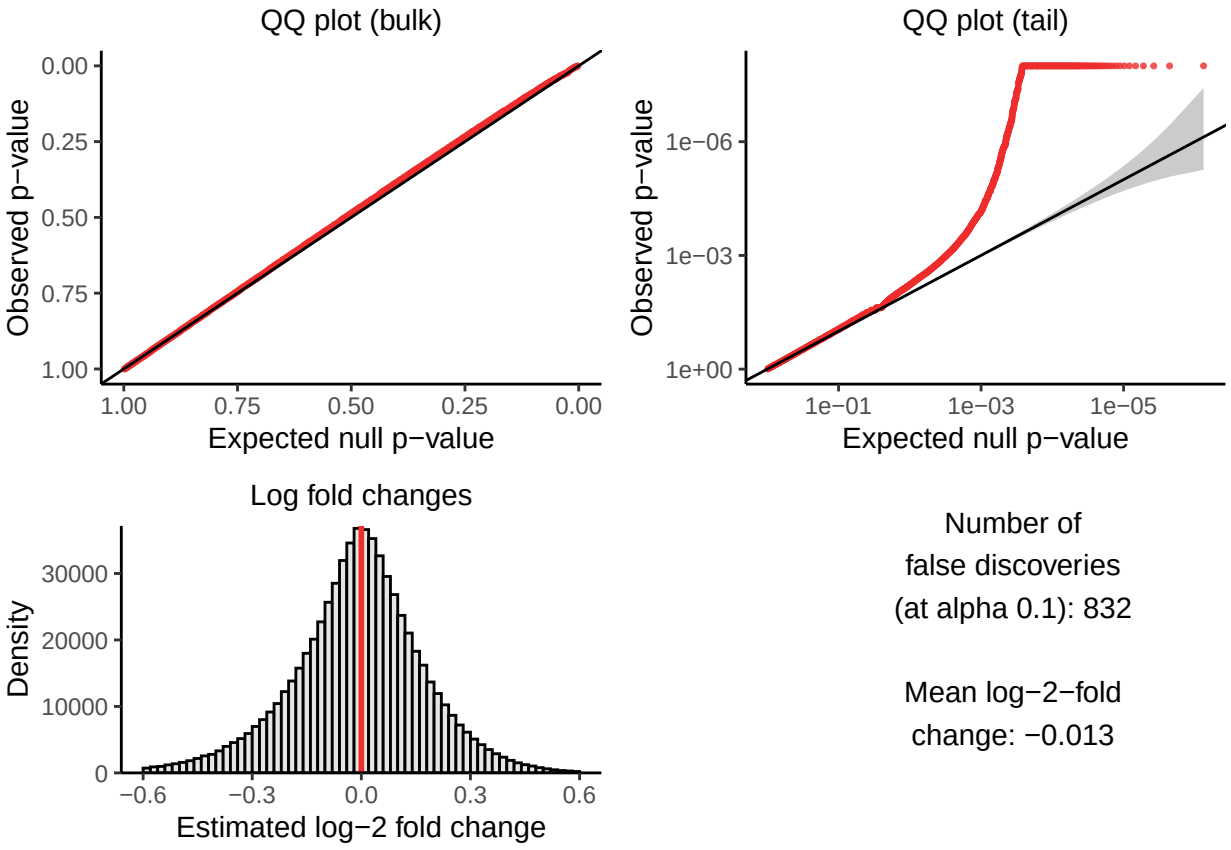
Implementing the Poisson-based pairwise QC strategy

I implement the Poisson-based pairwise QC strategy on the Replogle negative control pairs. To this end I compute the pilot, two-tailed alternative p-value p_{pilot} for each pair, setting the hypothesized fold change to 1/2 and 2 for the left- and right-tailed p-value, respectively. I retain pairs for which p_{pilot} is less than 0.005. Below, I plot the outcome of the pairwise QC. Each point represents a pair, with t (total UMI count) on the y-axis and n (number of treatment cells) on the x-axis.



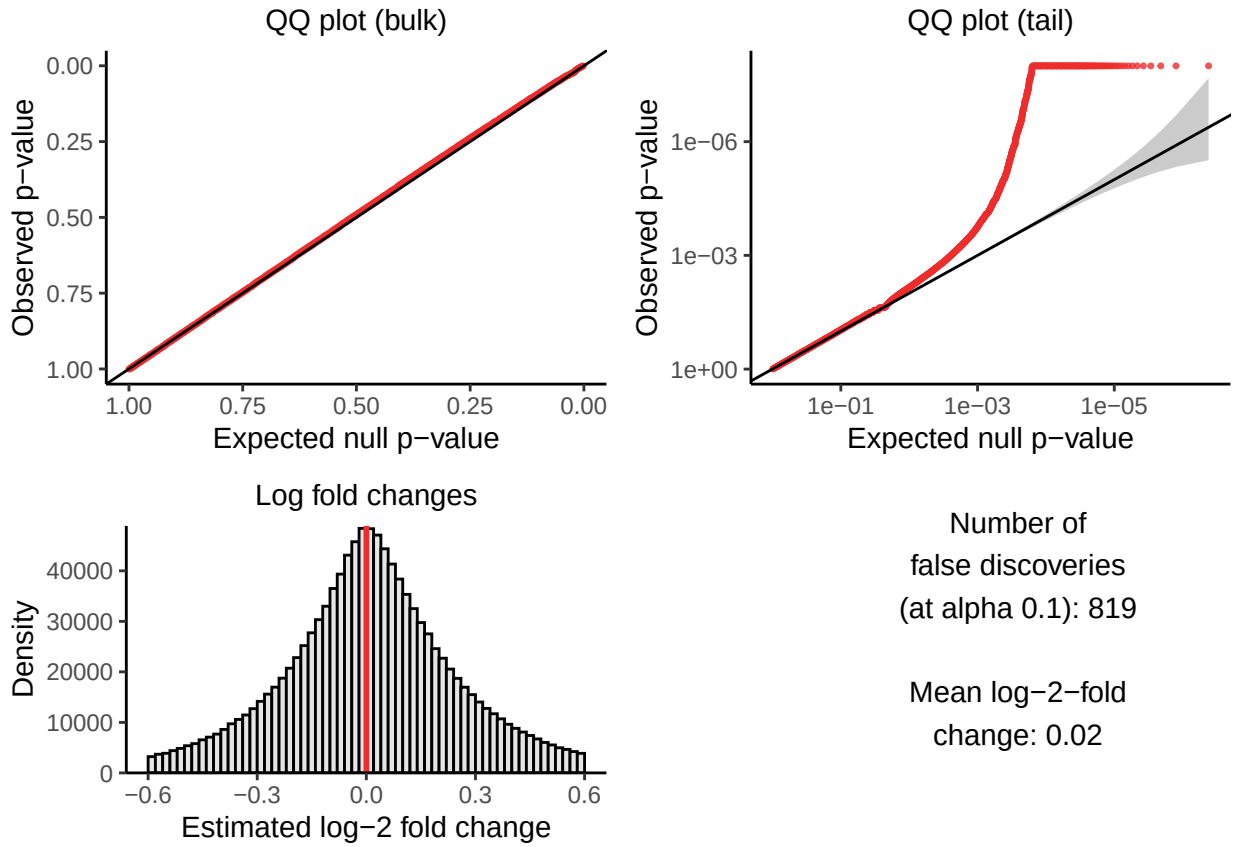
The Poisson-based pairwise QC seems reasonable. We retain pairs for which t and n are sufficiently large.

Next, I plot the calibration results for the pairs that pass the Poisson-based pairwise QC. Unfortauntely, the miscalibration remains severe.



Comparing to the original pairwise QC strategy

Finally, I implement the original pairwise QC strategy, filtering for pairs for which `n_nonzero_trt >= 7` and `n_nonzero_cntrl >= 7`. I plot the calibration results for these pairs.



Perforance across the two pairwise QC strategies seems to be broadly similar. In conclusion, the more sophisticated pairwise QC **does not** seem to improve calibration.

Why are we not seeing an improvement? It is instructive to look at the top pairs across the two QC strategies. The top ten pairs under the new QC strategy are as follows.

```
new_qc_res |>
  select(response_id, grna_target, p_value, n_nonzero_trt, n_nonzero_cntrl) |>
  arrange(p_value) |>
  slice(1:10)
```

```
##         response_id              grna_target       p_value n_nonzero_trt
##              <char>                   <char>         <num>         <int>
##  1: ENSG00000139278   non-targeting_vector_1 1.000000e-250            26
##  2: ENSG00000163584 non-targeting_vector_106 1.000000e-250           299
##  3: ENSG00000116251 non-targeting_vector_106 3.743232e-245           238
##  4: ENSG00000198712  non-targeting_vector_71 1.771863e-149           109
##  5: ENSG00000198727  non-targeting_vector_71 3.061768e-120           108
##  6: ENSG00000159335  non-targeting_vector_49  7.721336e-84            73
##  7: ENSG00000198763  non-targeting_vector_71  2.788435e-78           109
##  8: ENSG00000198804  non-targeting_vector_71  4.891580e-78           109
##  9: ENSG00000147403 non-targeting_vector_106  5.937366e-77           298
## 10: ENSG00000104904   non-targeting_vector_7  1.289635e-67            31
##     n_nonzero_cntrl
##               <int>
##  1:           12465
```

```
##  2:           12813
##  3:           13021
##  4:           13221
##  5:           13219
##  6:           13121
##  7:           13218
##  8:           13222
##  9:           13026
## 10:           13262
```

The top ten pairs under the original QC strategy are below.

```
orig_qc_res |>
  select(response_id, grna_target, p_value, n_nonzero_trt, n_nonzero_cntrl) |>
  arrange(p_value) |>
  slice(1:10)
```

```
##         response_id              grna_target       p_value n_nonzero_trt
##              <char>                   <char>         <num>         <int>
##  1: ENSG00000139278   non-targeting_vector_1 1.000000e-250            26
##  2: ENSG00000163584 non-targeting_vector_106 1.000000e-250           299
##  3: ENSG00000116251 non-targeting_vector_106 3.743232e-245           238
##  4: ENSG00000198712  non-targeting_vector_71 1.771863e-149           109
##  5: ENSG00000198727  non-targeting_vector_71 3.061768e-120           108
##  6: ENSG00000159335  non-targeting_vector_49  7.721336e-84            73
##  7: ENSG00000198763  non-targeting_vector_71  2.788435e-78           109
##  8: ENSG00000198804  non-targeting_vector_71  4.891580e-78           109
##  9: ENSG00000147403 non-targeting_vector_106  5.937366e-77           298
## 10: ENSG00000104904   non-targeting_vector_7  1.289635e-67            31
##     n_nonzero_cntrl
##               <int>
##  1:           12465
##  2:           12813
##  3:           13021
##  4:           13221
##  5:           13219
##  6:           13121
##  7:           13218
##  8:           13222
##  9:           13026
## 10:           13262
```

These sets of pairs coincide exactly. Moreover, each of these pairs has a large effective sample size and thus surives *both* QC filters.

In conclusion our suboptimal pairwise QC strategy does **not** explain the miscalibration we observe on the rd7 dataset.

Addendum: number of negative control pairs in each set

The number of negative control pairs in each set of negative control pairs is as follows.

```r
# unfiltered set
nrow(calib_res_0)
```

```
## [1] 1776981
```

```r
# original QC
nrow(orig_qc_res)
```

```
## [1] 1198459
```

```r
# Poisson-based QC
nrow(new_qc_res)
```

```
## [1] 663981
```