QuickMMCTest – Quick Multiple Monte Carlo Testing

Axel Gandy and Georg Hahn

Imperial College London
Department of Mathematics, London SW7 2AZ, U.K.

1 Reporting test results in two different ways

This section compares QuickMMCTest using two methods to compute final decisions on all hypotheses. These are p-value estimates computed with a pseudo-count $\hat{p}_i = (S_i + 1)/(k_i + 1)$, where $S_i$ and $k_i$ are as defined in QuickMMCTest, as well as classifications based on empirical rejection probabilities $r_i/R$ (compared to a cutoff of 0.5, see Section 2.2 in the main article).

We repeat the simulation study presented in Section 4.2 (main article) for a low effort using the fixed p-value distribution of Section 4.1 in the main article.

Table 1 shows simulation results. Using empirical rejection probabilities instead of pseudo-count estimates in connection with QuickMMCTest to classify hypotheses never yields an increase in the number of switched classifications. For most testing procedures, the decrease observed is considerable.

More importantly, using discrete p-value estimates with a pseudo-count makes QuickMMCTest susceptible to potentially not being able to record any rejections for other procedures or datasets. This especially applies to the computation of weights in QuickMMCTest using alternative approaches relying on p-value estimates computed with a pseudo-count. This is due to the fact that in the first iterations of any algorithm, the number of samples drawn for all hypotheses is typically not very high. It is thus very likely that the lower bound on each p-value lies above the testing threshold.

At high effort, QuickMMCTest yields equally precise results for both p-value estimates and empirical rejection probabilities (table not shown).

2 Power comparison of the two QuickMMCTest variants

We repeat the power comparisons of Section 4.4 of the main article with the aim to compare the power of the two QuickMMCTest variant using point estimates and empirical rejection probabilities (see Section 2.2, main article) The setup used in this section is the same as in Section 4.1 of the main article.

Figure 1 shows the same comparison as Figure 2 of the main article for the two QuickMMCTest variants. From the figure one can see that empirical rejection probabilities lead to a higher power.
Table 1: Reporting testing results in two different ways. Average numbers of switched classification numbers (average numbers of erroneously rejected hypotheses in brackets) for common multiple testing procedures applied at a low effort ($K = 1000m$). Constant testing threshold 0.1.

|                          | QuickMMCTest | Bonferroni (1936) | 64.5 (1) | 43.8 (2.6) |
|--------------------------|--------------|-------------------|----------|------------|
|                          | p.c.         | Simes (1986)      | 2 (0.9)  | 2 (0.9)    |
|                          | e.r.p.       | Hochberg (1988)   | 64.2 (1) | 43.4 (2.5) |
|                          |              | Benjamini and Hochberg (1995) | 2 (1) | 2 (1) |
|                          |              | Benjamini and Yekutieli (2001) | 15 (2.8) | 14.5 (3.3) |
|                          |              | Sidak (1967)      | 74.4 (0.6) | 36.3 (2.9) |
|                          |              | Holm (1979)       | 76.5 (0.5) | 39.5 (3.2) |

p.c., pseudo-count; e.r.p., empirical rejection probabilities.

Figure 1: Average per-pair power against number of samples per hypothesis. Comparison of QuickMMCTest with empirical rejection probabilities and p-value point estimates (computed with a pseudo-count in both numerator and denominator) for the Bonferroni (1936) correction (left) and the Simes (1986) procedure (right). Log-scale on the x-axis.
than point estimates at low computational effort (numbers of samples per hypothesis). For larger numbers of samples per hypothesis, both variants perform equally.

This result is confirmed by Figure 2: Similarly to Figure 3 of the main article, we compare the two QuickMMCTest variants (with empirical rejection probabilities and with pseudo-count point estimates) for the Bonferroni (1936) correction (with “power” defined as the per-pair power) and for the Benjamini and Hochberg (1995) procedure (with “power” defined as one minus the average false non-discovery proportion). As before, empirical rejection probabilities lead to a higher power at low computational effort. As the number of samples drawn per hypothesis increases, both variants perform comparably.

### 3 Dependence of QuickMMCTest on the number of updates

This section investigates the dependence of QuickMMCTest on the parameter $n_{\text{max}}$ controlling the number of iterations and thus the number of posterior updates.

QuickMMCTest was run on the p-values fixed in Section 4.1 (main article) using $R = 1000$, the Benjamini and Hochberg (1995) procedure at threshold $\alpha^* = 0.1$ and a total effort of $K = 1000m$, where $m = 5000$ is the number of hypotheses. The parameter $n_{\text{max}}$ was varied.

Table 2: Number of updates $n_{\text{max}}$ against number of switched classifications for QuickMMCTest. Simulation setting as in Section 4.1 (main article).

| $n_{\text{max}}$ | 1      | 2      | 5      | 10     | 20     | 50     | 100    | 200    |
|------------------|--------|--------|--------|--------|--------|--------|--------|--------|
| switched classifications | 31.735 | 10.594 | 3.448  | 1.967  | 1.689  | 1.628  | 1.571  | 1.572  |
Table 3: Numbers of switched classifications for \texttt{QuickMMCTest} as a function of both the total number of samples $K$ (as a multiple of the number of hypotheses $m = 5000$) and the number of updates $n_{\text{max}}$. Simulation setting as in Section 4.1 (main article).

| $n_{\text{max}}$ | 10m | 100m | 500m | 1000m |
|------------------|-----|------|------|-------|
| 1                | 373.0 | 373.0 | 51.6 | 31.8 |
| 2                | 373.0 | 76.0 | 28.5 | 10.9 |
| 5                | 373.0 | 47.4 | 6.7  | 3.6  |
| 10               | 373.0 | 36.6 | 4.4  | 2.1  |

Table 2 shows simulation results. As expected, the number of switched classifications decreases first with an increasing number of updates due to the more accurate computation of the weights. This effect dominates until, for even larger values of $n_{\text{max}}$ than displayed in Table 2, the available total number of samples for all hypotheses per iteration is so low that hypotheses with very low weights essentially do not receive any samples any more per iteration, thus increasing again their susceptibility to being erroneously classified. Running \texttt{QuickMMCTest} with more updates also considerably increases its runtime, therefore using $n_{\text{max}} = 10$ or $n_{\text{max}} = 20$ seems to yield a reasonable trade-off between speed and accuracy. The choice $n_{\text{max}} = 10$ is used as a default choice in Section 4 of the main article.

Table 3 extends this comparison further. The simulation setting is again the same one as described above (Section 4.1 of the main article). All results are based on 1000 repetitions. To investigate whether it is possible to obtain a “bias scenario” in which a particularly “bad” set of initial samples influences the weights in such a way as to give biased results in \texttt{QuickMMCTest}, Table 3 shows the number of switched classifications for very low total numbers of samples $K$ and low iteration numbers $n_{\text{max}}$. The low values of $n_{\text{max}}$ are used to keep \texttt{QuickMMCTest} from recovering from a biased initial sample during the remaining iterations. Table 3 shows that for very low sample sizes, \texttt{QuickMMCTest} is not able to reject any hypothesis. This phenomenon is explained in Section 6 of the Supplementary Material. For increasing samples sizes, the number of switched classifications decreases as expected. However, and not surprisingly, spending all effort in the first iteration yields the poorest results. Importantly, performing at least one update (that is, two iterations), considerably improves performance, and higher numbers of iterations yield increasingly better results.

However, the choice of $n_{\text{max}}$ should not be too big: If $n_{\text{max}}$ is taken too large, the number $\Delta = K/n_{\text{max}}$ of samples drawn per iteration will be too small, and it will be unlikely that hypotheses with very low weights receive new samples in line 13 of the \texttt{QuickMMCTest} algorithm. Ideally, the integers $K$ and $n_{\text{max}}$ should be chosen such that at least $n_{\text{max}} = 10$ (even better, around $n_{\text{max}} = 20$ to $n_{\text{max}} = 50$) iterations are performed, and such that per iteration, the number of available samples $\Delta = K/n_{\text{max}}$ is a multiple of $m$. 

4
Table 4: Parameter $R$ against numbers of switched classification numbers for QuickMMCTest.

| $R$  | 10   | 100  | 1000 | 10000 |
|------|------|------|------|-------|
| switch classifications | 4.629 | 2.314 | 2.052 | 2.022 |

Figure 3: Allocation of sample weights of QuickMMCTest for simulated data ($m = 5000$ hypotheses) with proportion $\pi = 0.9$ from the null. Sample weights are normed to lie in the interval $[0, 1]$. Multiple testing with the Benjamini and Hochberg (1995) procedure. Generated p-values in sorted order (solid curve), threshold of the Benjamini and Hochberg (1995) procedure (dashed line) and allocation of sample weights of QuickMMCTest (dotted line). Left: Entire range of hypotheses. Right: Zoomed-in range around the last rejected hypothesis.

4  Dependence of QuickMMCTest on the parameter $R$

Using the same set-up as in Section 3, we also evaluate the influence of the parameter $R$ on the QuickMMCTest algorithm while keeping the default choice of $n_{\text{max}} = 10$ (Section 4.1 in the main article) fixed. The parameter $R$ controls the accuracy with which weights are computed per iteration.

Results are shown in Table 4. Not surprisingly, the test results returned by QuickMMCTest become more and more accurate with an increasing accuracy of the weights. In order to not spend too much time on the computation of weights, using $R = 1000$ as a default choice (Section 4.1 in the main article) seems to yield a reasonable trade-off between speed and accuracy.

5  Allocation of weights of QuickMMCTest in a sample run

Figure 3 shows how QuickMMCTest allocates samples to all hypotheses. We generated $m = 5000$ p-values as in Section 4.1 of the main article. QuickMMCTest was run using the parameter choices made for Section 4.1 of the main article and its allocation of samples was recorded. The allocation upon termination of QuickMMCTest (dotted line) totalling 10000$m$ samples was normed to lie in the interval $[0, 1]$ and plotted alongside the generated p-values (solid line) and the threshold of the Benjamini and Hochberg (1995) procedure (dashed line).

Figure 3 shows that QuickMMCTest heavily concentrates its samples around the threshold.
6 Zero rejected hypotheses occur at a low effort

The naive method draws a constant number of samples per hypothesis and computes p-value estimates $\hat{p}_i = (e_i + 1)/(s + 1)$ as proposed in Davison and Hinkley (1997), where $e_i$ denotes the number of exceedances observed for hypothesis $H_{0i}$ among $s$ samples.

Assuming full knowledge of all p-values, let $\tau_0$ be the proportion of non-rejected hypotheses observed when applying the multiple testing procedure to the p-values. In order to observe the correct number of rejections with approximated p-values, the $(1 - \tau_0)m$th ordered p-value estimate has to be the last one lying below the critical value at its rank in the sorted sequence of estimates.

For the Hochberg (1988) procedure applied at $\alpha^* = 0.1$, a constant number of 87 switched classifications were observed for the naive method at both a low and a high effort in Table 1 of the main article. Indeed, applying the Hochberg (1988) procedure to the fixed p-values leads to 87 rejections and 4913 non-rejections, thus $\tau_0 = 4913/5000 = 0.9826$. The critical value of the Hochberg (1988) procedure at the 87th p-value is approximately $2 \cdot 10^{-5}$.

Estimated p-values with a pseudo-count are bounded from below by $L = 1/(s + 1)$, where $s$ is the number of samples. Even for $s = 10000$ (high effort), $L > 2 \cdot 10^{-5}$ and thus all p-value estimates lie above the (correct) rejection area. Although theoretically it would be possible that another (possibly underestimated) p-value lies below a critical value again at a higher rank (as most critical values are non-decreasing), this rarely happens as hypotheses having higher ranked p-values are typically non-rejected, hence their p-values (and estimates) are expected to be (much) larger than their corresponding critical values. Thus in most cases, the fact that $L > 2 \cdot 10^{-5}$ actually leads to all hypotheses being consistently non-rejected in each run, independently of the actual number of exceedances observed. It is hence impossible to record any rejection.

This phenomenon disappears in three cases. First, using more samples $s$ to estimate each p-value will decrease the lower bound $L$. Alternatively, datasets with less hypotheses from the null and thus lower $\tau_0$ lead to a higher threshold at the $(1 - \tau_0)m$th p-value. Third, estimating p-values without a pseudo-count leads to estimates which attain a value of zero and hence allow one to observe rejections for any number of samples $s$, although using such estimates does not guarantee appropriate error control (Section 2.2 of the main article).

7 Simulation study at a variable testing threshold

The simulation study comparing the naive approach to QuickMMCTest (Section 4.2 in the main article) was also carried out at a variable testing threshold. Apart from the testing threshold, the set-up we used is identical to the one of Section 4.2 (main article).

We corrected the threshold using an estimate of the proportion $\pi_0$ of true null hypotheses of Pounds and Cheng (2006), employed in various real data studies (Han and Dalal, 2012; Lu et al., 2011; Jupiter et al., 2010; Cheng, 2009). The corrected testing threshold is given by $\alpha(p^*) = \alpha^*/\hat{\pi}_0(p^*)$, where $\alpha^* = 0.1$ is an uncorrected threshold and $\hat{\pi}_0(p) = \min (1, 2/m \sum_{i=1}^m p_i)$ is the Pounds and Cheng (2006) estimate of $\pi_0$.

The naive method is used with a variable testing threshold by computing usual p-value
Table 5: Average number of switched classifications (average numbers of switched rejections in brackets) for the naive method compared to \texttt{QuickMMCTest} (Alg. 1) for common multiple testing procedures. Testing threshold of Pounds and Cheng (2006) at $\alpha^* = 0.1$.

| Method                        | low effort ($s = 1000$) | high effort ($s = 10000$) |
|-------------------------------|--------------------------|-----------------------------|
|                               | naive                    | Alg. 1                      | naive                        | Alg. 1                        |
| Bonferroni (1936)              | 90 (0)                   | 40.4 (2.5)                  | 90 (0)                       | 3.5 (1.4)                     |
| Simes (1986)                  | 30.7 (10.1)              | 1.9 (0.9)                   | 8.9 (4.4)                    | 0 (0)                         |
| Hochberg (1988)               | 90 (0)                   | 39.9 (2.5)                  | 90 (0)                       | 3.5 (1.6)                     |
| Benjamini and Hochberg (1995) | 30.4 (9.9)               | 1.9 (0.8)                   | 9.1 (4.5)                    | 0 (0)                         |
| Benjamini and Yekutieli (2001)| 168 (0)                  | 14 (3.8)                    | 22.5 (7)                     | 1.7 (1.2)                     |
| Sidak (1967)                  | 94 (0)                   | 31 (2.4)                    | 94 (0)                       | 3.1 (0.5)                     |
| Holm (1979)                   | 91 (0)                   | 33.3 (3)                    | 91 (0)                       | 3.4 (1.6)                     |

estimates $\hat{p}$ with a pseudo-count (Section 4.2 in the main article) and by applying the multiple testing procedure to these estimates using the plug-in threshold $\alpha(\hat{p})$.

Simulation results are given in Table 5. The naive method either observes no rejections or performs poorly at a low effort when being applied to common multiple testing procedures. Similarly to Section 4.2 in the main article, \texttt{QuickMMCTest} achieves meaningful results for all procedures at a low effort. Its test results only contain few erroneously rejected hypotheses.

The naive method still fails to record any rejections for some procedures at a high effort. On the contrary, \texttt{QuickMMCTest} yields a multiple fold decrease in the number of switched classifications compared to the naive method and only few erroneously rejected hypotheses.

8 Comparison to common methods at a variable threshold

The parameters of the methods used in Section 4.3 of the main article were as follows:

1. The naive method was run with $s = 1000$ samples per hypothesis at low effort and $s = 10000$ samples at high effort. Estimates were computed using a pseudo-count (Davison and Hinkley, 1997) as $\hat{p}_i = (e_i + 1)/(s + 1)$, where $e_i$ is the number of exceedances observed for $H_{0i}$ among $s$ samples drawn. A decision on all hypotheses is obtained (at a variable testing threshold) by evaluating the multiple testing procedure on the estimates $\hat{p} = (\hat{p}_1, \ldots, \hat{p}_m)$ using the plug-in threshold $\alpha(\hat{p})$.

2. In order to match the overall effort, the algorithm of Besag and Clifford (1991) was run by sequentially drawing one sample at a time for each hypothesis until either $h = 20$ exceedances (as proposed by the authors) were observed (in which case this hypothesis was excluded from receiving further samples) or the total effort was reached. P-values were computed with a pseudo-count in the numerator as proposed by the authors.

3. The algorithm of Guo and Peddada (2008) was implemented using a geometric sequence of sample numbers $B_0 \leq B_1 \leq \ldots B_N$, where $N = 9$ and the geometric increase was computed in order to match the overall effort, as well as Clopper and Pearson (1934)
Table 6: Average number of switched classifications (average numbers of switched rejections in brackets) for common methods compared to QuickMMCTest using the Bonferroni (1936) correction. Testing threshold of Pounds and Cheng (2006) at \( \alpha^* = 0.1 \).

| Method                        | low effort | high effort |
|-------------------------------|------------|-------------|
| Naive method                  | 90 (0)     | 90 (0)      |
| Besag and Clifford (1991)     | 90 (0)     | 5.8 (1.5)   |
| Guo and Peddada (2008)        | 90 (0)     | 6.1 (1.1)   |
| Sandve et al. (2011)          | 90 (0)     | 20.1 (1.3)  |
| Jiang and Salzman (2012)      | 90 (0)     | 17.4 (2.7)  |
| Gandy and Hahn (2014)         | 90 (0)     | 8.8 (1.1)   |
| QuickMMCTest                  | 40.2 (2.3) | 3.6 (1.4)   |

confidence intervals as proposed by the authors. P-value estimates were computed with a pseudo-count in both the numerator and denominator.

4. The MCFDR algorithm of Sandve et al. (2011) is the only method whose effort cannot easily be matched to a pre-specified total effort. We therefore tune the only parameter \( h \) of MCFDR to meet the upper bound for the effort: for \( h = 65 \), the effort of MCFDR roughly equals \( K = 1000m \), and for \( h = 650 \), the effort roughly equals \( K = 10000m \), where \( m = 5000 \) is the number of hypotheses. In order to ensure a fair comparison, MCFDR is stopped on reaching \( K \) samples and QuickMMCTest is then applied using at most the number of samples MCFDR spent instead of \( K = 1000m \) or \( K = 10000m \). P-value estimates were computed with a pseudo-count in both the numerator and denominator.

5. The method of Jiang and Salzman (2012) was run with parameters \( a = 10 \) and \( \delta = 0.01 \) as proposed by the authors in their simulation study. One new sample is drawn per hypothesis in each iteration and p-values were computed as proposed in the original article.

6. The MMCTest algorithm was run as suggested by Gandy and Hahn (2014). We used 10 iterations and a geometric increase \( a \) in MMCTest (see the original article) equal to the value used in the implemention of the algorithm of Guo and Peddada (2008) in order to exactly match the overall effort. P-value estimates were computed with a pseudo-count in both the numerator and denominator.

We repeat the simulation study given in Section 4.3 of the main article using the Bonferroni (1936) correction in connection with the variable testing threshold of Pounds and Cheng (2006). Similarly to Section 7 we use the p-values fixed in Section 4.1 (main article). Low and high effort were defined as in Section 4.3 of the main article.

The comparison of commonly used methods to QuickMMCTest in Table 6 confirms the picture already observed at a constant threshold (Section 4.3 in the main article). Due to the low threshold of the Bonferroni (1936) correction, all methods except for QuickMMCTest fail to compute meaningful test results at a low effort.

At a high effort, the naive method is again unable to observe any rejections and most other methods yield results with reasonable accuracy. The methods of Besag and Clifford (1991) and Guo and Peddada (2008) perform especially well and are only outperformed by QuickMMCTest.
Table 7: Average number of switched classifications (average numbers of switched rejections in brackets) for common methods compared to QuickMMCTest using the Benjamini and Hochberg (1995) procedure. Testing threshold of Pounds and Cheng (2006) at $\alpha^* = 0.1$.

| Method                     | low effort | high effort |
|----------------------------|------------|-------------|
| Naive method               | 32.1 (9.6) | 9 (3.4)     |
| Besag and Clifford (1991)  | 18.4 (7.5) | 18.8 (7.5)  |
| Guo and Peddada (2008)     | 4.4 (2)    | 0.2 (0.2)   |
| Sandve et al. (2011)       | 10.5 (4.2) | 2.7 (1.3)   |
| Jiang and Salzman (2012)   | 13.7 (5.1) | 3.6 (1.6)   |
| Guo and Peddada (2008)     | 4.4 (2)    | 0.2 (0.2)   |
| QuickMMCTest               | 2.4 (1.2)  | 0.2 (0.1)   |

When using the Benjamini and Hochberg (1995) procedure, the picture observed in Table 7 again resembles the one observed in the main article: For the Benjamini and Hochberg (1995) procedure, all methods are able to observe rejections and thus to compute meaningful test results for both a low and a high effort. As in the main article, the method of Guo and Peddada (2008) and QuickMMCTest perform comparably well and considerably better than the other methods. The precise numbers in Table 7 are almost the same as the ones in Table 3 of the main article.

References

Benjamini, Y. and Hochberg, Y. (1995). Controlling the false discovery rate: A practical and powerful approach to multiple testing. *J Roy Statist Soc Ser B*, 57(1):289–300.

Benjamini, Y. and Yekutieli, D. (2001). The control of the false discovery rate in multiple testing under dependency. *Ann Stat*, 29(4):1165–1188.

Besag, J. and Clifford, P. (1991). Sequential Monte Carlo p-values. *Biometrika*, 78(2):301–304.

Bonferroni, C. (1936). Teoria statistica delle classi e calcolo delle probabilità. *Pubblicazioni del R Istituto Superiore di Scienze Economiche e Commerciali di Firenze*, 8:3–62.

Cheng, C. (2009). Internal validation inferences of significant genomic features in genome-wide screening. *Comput Stat Data An*, 53:788–800.

Clopper, C. and Pearson, E. (1934). The use of confidence or fiducial limits illustrated in the case of the binomial. *Biometrika*, 26(4):404–413.

Davison, A. and Hinkley, D. (1997). *Bootstrap Methods and Their Application*. Cambridge University Press.

Gandy, A. and Hahn, G. (2014). MMCTest – A Safe Algorithm for Implementing Multiple Monte Carlo Tests. *Scand J Stat*, 41(4):1083–1101.

Guo, W. and Peddada, S. (2008). Adaptive choice of the number of bootstrap samples in large scale multiple testing. *Stat Appl Genet Mol Biol.*, 7(1):1–16.

Han, B. and Dalal, S. (2012). A Bernstein-type estimator for decreasing density with application to p-value adjustments. *Comput Stat Data An*, 56:427–437.
Hochberg, Y. (1988). A sharper Bonferroni procedure for multiple tests of significance. *Biometrika*, 75(4):800–802.

Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scand J Stat*, 6(2):65–70.

Jiang, H. and Salzman, J. (2012). Statistical properties of an early stopping rule for resampling-based multiple testing. *Biometrika*, 99(4):973–980.

Jupiter, D., Sahutoglu, J., and VanBuren, V. (2010). TreeHugger: A new test for enrichment of gene ontology terms. *INFORMS Journal on Computing*, 22(2):210–221.

Lu, H.-F., Dong, H.-T., Sun, C.-B., Qing, D.-J., Li, N., Wu, Z.-K., Wang, Z.-Q., and Li, Y.-Z. (2011). The panorama of physiological responses and gene expression of whole plant of maize inbred line yq7-96 at the three-leaf stage under water deficit and re-watering. *Theor Appl Genet*, 123:943–958.

Pounds, S. and Cheng, C. (2006). Robust estimation of the false discovery rate. *Bioinformatics*, 22(16):1979–1987.

Sandve, G., Ferkingstad, E., and Nygård, S. (2011). Sequential Monte Carlo multiple testing. *Bioinformatics*, 27(23):3235–3241.

Sidak, Z. (1967). Rectangular confidence regions for the means of multivariate normal distributions. *J Amer Statist Assoc*, 62(318):626–633.

Simes, R. (1986). An improved Bonferroni procedure for multiple tests of significance. *Biometrika*, 73(3):751–754.