# Supplementary Material

## 1 Loss functions

In Section 2.1 of the main manuscript, we claim that ranking parameters using $p$-values is roughly equivalent to choosing a rank vector, $R$, to optimize

$$\mathbb{E}(L((\theta_1, ..., \theta_N), R)|X) = \mathbb{E}(\sum_{i \leq N} R_i I\{\theta_i = 0\}|X) \tag{1.1}$$

The equivalence is exact when there is a monotonic increasing relationship between the $p$-value for experiment $i$ and the posterior probability: $\mathbb{P}(\theta_i = 0|X)$. For the examples described in this manuscript, both the $p$-value and posterior probability are functions of the test statistic $Z$. In this case $p$-values will exactly optimize the following loss function provided that $\mathbb{P}(\theta_i = 0|X)$ is decreasing in the test statistic $|Z|$. While this is only guaranteed to be true for certain classes of priors (for example any normal prior with mean 0), it should be approximately true for all reasonable priors. The assertion that (1) is minimized by choosing ranks according to the posterior probabilities $\mathbb{P}(\theta_i \neq 0|X)$ can be deduced by applying the following result to the numbers $a_i = -\mathbb{P}(\theta_i \neq 0|X)$.

*Claim.* Consider a permutation, $\pi$, of $\{1, ..., n\}$ to be "consistent with the ranks of $a_1, \ldots, a_n$" if, for all $i$ and $j$, $\pi(i) < \pi(j) \implies a_i \leq a_j$. Then, given

numbers $a_1, ..., a_n$, the permutations $\pi$ of $\{1, ..., n\}$ that maximize the sum $\sum \pi(i)a_i$ are those that are consistent with the ranks of $a_1, \ldots, a_n$.

*Proof.* Let $\pi$ be any permutation not consistent with the ranks of $a$. We will show that we can find a permutation $\pi^{'}$ such that $\sum \pi^{'}(i)a_i > \sum \pi(i)a_i$. By our assumption about $\pi$, there exist $j$ and $k$ such that $\pi(j) < \pi(k)$ and $a_j > a_k$. Define $\pi^{'}$ by:

$$\begin{aligned} \pi^{'}(j) &= \pi(k) \\ \pi^{'}(k) &= \pi(j) \\ \pi^{'}(i) &= \pi(i) \text{ for } i \notin \{j, k\}. \end{aligned}$$

Then,

$$\begin{aligned} \textstyle\sum_i \pi^{'}(i)a_i - \sum_i \pi(i)a_i &= \textstyle\sum_{i \in \{j,k\}} (\pi^{'}(i) - \pi(i))a_i \\ &= (\pi^{'}(j) - \pi(j))a_j + (\pi^{'}(j) - \pi(k))a_k \\ &= (\pi(k) - \pi(j))(a_j - a_k) > 0. \end{aligned}$$

Thus any permutation, $\pi$, not consistent with the ranks of $a$ can be improved, so that all optimal permutations must be consistent with the ranks of $a$.

Conversely, to see that all permutations that are consistent with the ranks of $a$ are optimal, suppose two permutations, $\pi_1$ and $\pi_2$are both consistent with the ranks of $a$. We have for all $i$ and $j$

$$\pi_1(i) < \pi_1(j) \implies a_i \leq a_j \text{ and } \pi_2(i) < \pi_2(j) \implies a_i \leq a_j$$

Let $o_1 = \pi_1^{-1}$ and $o_2 = \pi_2^{-1}$ be the corresponding orderings. Then from the display above it follows that for all $i$ and $j$,

$$i < j \implies a_{o_1(i)} \leq a_{o_1(j)} \text{ and } i < j \implies a_{o_2(i)} \leq a_{o_2(j)},$$

that is, $a_{o_1(1)} \leq a_{o_1(2)} \leq ... \leq a_{o_1(n)}$ and $a_{o_2(1)} \leq a_{o_2(2)} \leq .... \leq a_{o_2(n)}$. So

we see that $a_{o_1(i)} = a_{o_2(i)}$ for all $i$, from which we get

$$\sum_i \pi_1(i) a_i = \sum_j j a_{o_1(j)} = \sum_j j a_{o_2(j)} = \sum_i \pi_2(i) a_i.$$

□

# 2 Simulations

## 2.1 RNA Sequencing

Here we give more background regarding the calculation of $S_j$, $m_i$ and $d_i$, the parameters represented in the negative binomial distribution from equation (10) of the main manuscript, and subsequent simulation of RNA-sequence data. For convenience, equation (1) is represented again below:

$$G_{ij} \sim \begin{array}{c} NB(S_j m_i, d_i)\, j \in \text{group 1} \\ NB(S_j 2^{\theta_i} m_i, d_i)\, j \in \text{group 2} \end{array}$$

where as detailed in the main manuscript, $G_{ij}$ represents the simulated negative binomial count for individual $j$, gene and the fixed parameters $m_i$, $S_j$, and $d_i$, representing scaled mean counts, relative library sizes and dispersions for gene $i$, library individual $j$.

- First, the Turing-Good procedure, [5] as implemented in [4] was used to estimate the proportion of DNA fragments originating from each gene in the Bottomly mouse expression data set [1], based on the matrix of RNA-Sequence counts. These proportions were produced separately for each of the 21 libraries and then averaged over the 21 libraries. We refer to the resulting estimated proportions as $m_1, ...., m_N$ were $N$=36,536. Note, however, that only 13,932 of these genes correspond to non-zero estimated proportions.

- Depending on the value for the percentage of differently expressed genes, $p_{DE}$, (either 5%, 10% or 20%), independent binary indicators for differential expression status were simulated for each of the $N$ genes, based on the appropriate bernoulli distribution.

- Log (base 2) fold change values, $\theta_i$, were then simulated for each gene

  - According to a $N(0, \sigma_{FC})$ distribution, where $\sigma_{FC}$ was set at either 0.5 or 1 when the gene was tagged as differentially expressed
  - Set at 0 when the gene was not differentially expressed.

- The total library size, $S_j$, for individual $j$ was randomly selected from the integers $(n_1, ...., n_{21})$ with $n_l$, $l = 1, 2, ..., 21$ representing the aggregate sum total of reads from the $l^{th}$ Bottomly sample.

- The mean count for gene $i$ individual $j$ was set at $\begin{matrix} S_j m_i & j = 1, ..., n \\ S_j 2^{\theta_i} m_i & j = n+1, ..., 2n. \end{matrix}$.

- The R package, Edge-R, [7] was used to estimate dispersion parameters: $d_i^*$ for each gene from the Bottomly mouse expression dataset. $d_i$ was then simulated using: $d_i = d_i^* e^{Z_i}$ where $Z_i \sim N(0, 1)$.

- Finally, the simulated count $G_{ij}$ is simulated from the negative binomial with mean $S_j 2^{\theta_i} m_i$ and dispersion parameter $d_i^*$.

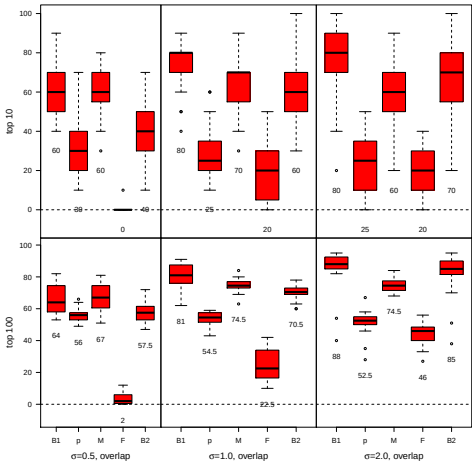
- To reduce the number of rows in the final dataset, we excluded non-informative genes, $i$, where $\sum_{i=1}^{2n} G_{ij} < 10$ in the final dataset.

## Simulation results:

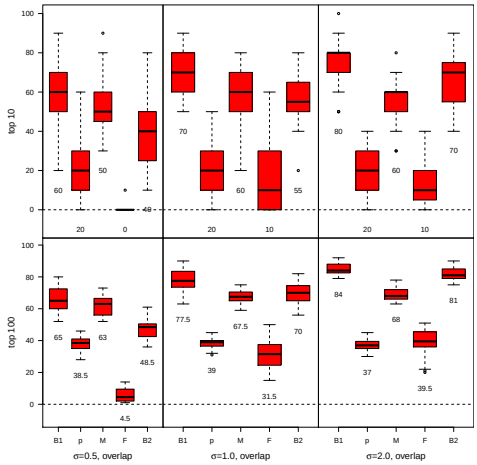
More extensive results to the RNA Sequence simulations are given below.

Figure 1: RNASeq simulations. Ranking performance measured via percentage overlap between the estimated and true ranks of the top K experiments. Simulations assumed mean counts estimated from [1], and simulated fold change from a log-normal distribution; Estimated fold changes and standard errors were calculated with DEseq2. Each boxplot represents 8 simulations. B1: Bayesian ranking as described in manscript, p: p-value using DEseq2, M: non-negative matrix factorization, F: FCROS, B2: Empirical Bayes using Smoothing by Roughening

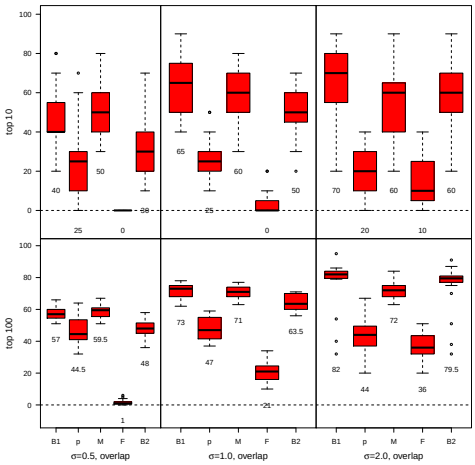
(a) $p_{DE}$=0.05




(b) $p_{DE}$=0.2




(c) n=10




(d) n=50

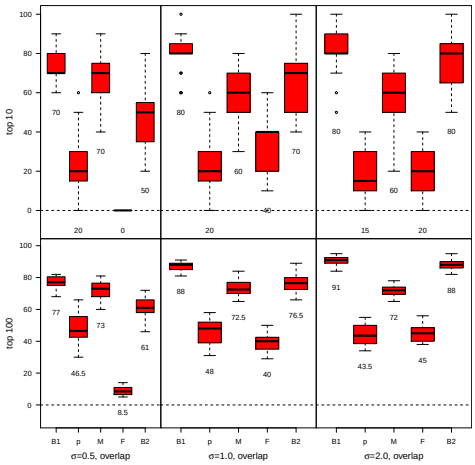
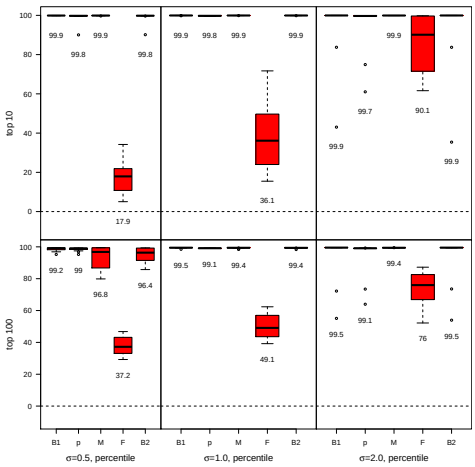
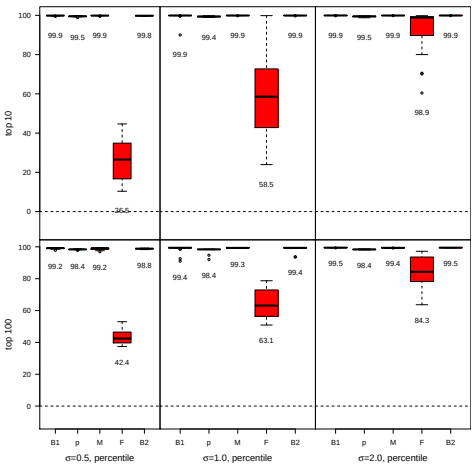


Figure 2: RNASeq simulations. Ranking performance measured via the percentile of set of parameters ranked within the top K in a particular simulation, within the list of true absolute value parameters. Simulations assumed mean counts estimated from [1], and simulated fold change from a log-normal distribution; Estimated fold changes and standard errors were calculated with DEseq2. Each boxplot represents 8 simulations. B1: Bayesian ranking as described in manscript, p: p-value using DEseq2, M: non-negative matrix factorization, F: FCROS, B2: Empirical Bayes using Smoothing by Roughening

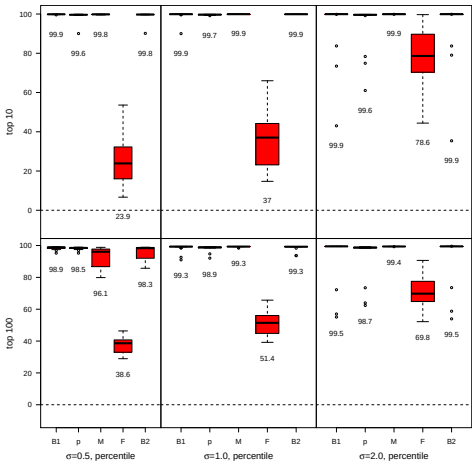
(a) $p_{DE}$=0.05




(b) $p_{DE}$=0.2




(c) n=10




(d) n=50

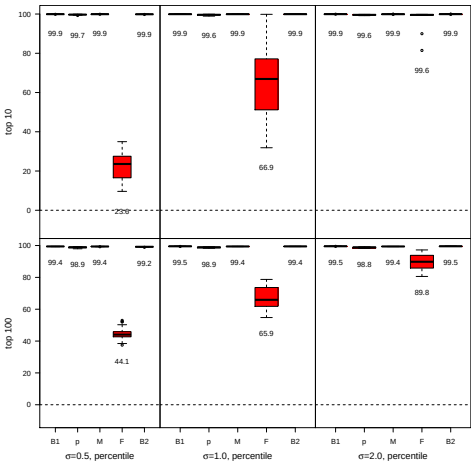
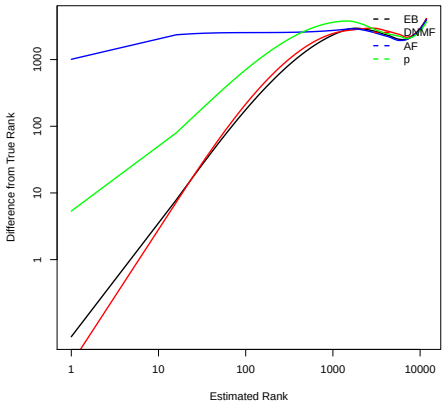
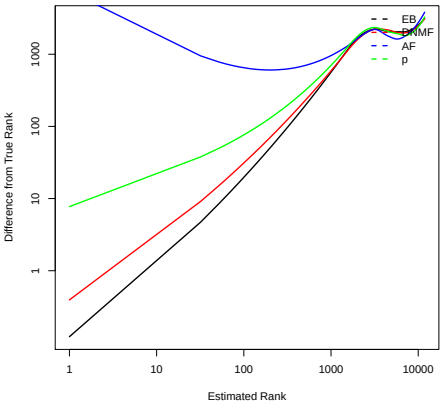


Figure 3: RNASeq simulations. Ranking performance measured via the diffrence between assigned ranks and true ranks. Simulations assumed mean counts estimated from [1], and simulated fold change from a log-normal distribution; Estimated fold changes and standard errors were calculated with DEseq2. Each line represents a loess smooth of 8 simulations. EB: Bayesian ranking as described in manscript, p: p-value using DEseq2, DNMF: non-negative matrix factorization, AF: fold change ranking based on FCROS

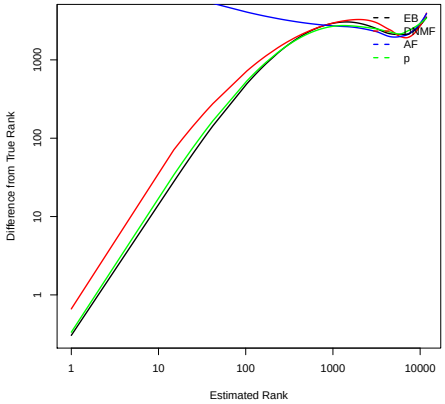
(a) $p_{DE}$=0.05



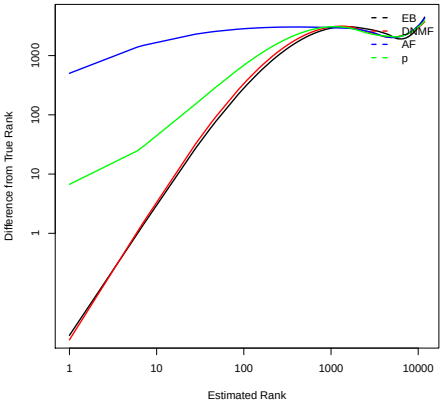

(b) $p_{DE}$=0.2




(c) n=10




(d) n=50



## Additional Simulations

Additional RNASeq simulations were carried out using the simulation process described above. The purpose of these additional simulations is three-fold:

1. To compare the default Empirical Bayes squared error loss ranking with other alternative loss functions and other possible ranking methods

2. To investigate whether pre-selection (that is only ranking the experiments that were significant at 5% using a Benjamini Hochberg FDR threshold) helped or hinders ranking.

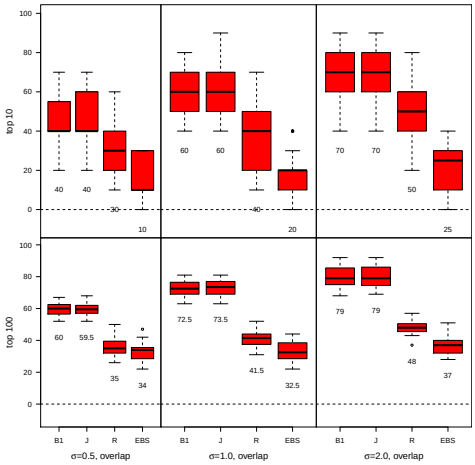
3. To investigate the effect of Monte Carlo error in estimating the posterior distributions of ranks on Ranking performance.

In more detail, Jewett et. al [9] considered a number of possible loss functions for ranking, extending the default squared error rank loss. Here we investigate the performance of choosing the rank vector that minimizes posterior 'position loss' (here the penalty for assigning rank i to position j as $(\theta_j - \theta_{(i)})^2$) in terms of the overlap beween the top $K$ assigned ranks and top $K$ true ranks. The Bayesian computation of this loss is difficult for large parameter sets, as it requires estimating: $E(\theta_j - \theta_{(i)}|X)^2$ for each combination of experiments $i, j \leq N$, that is $N(N-1)/2$ separate calcualtions . Our strategy to avoid this is to fit the Empirical Bayes prior using all the data-points, but only rank those that were significant at a 5% false discovery rate threshold. This also allows us to compare the ranking performance of the hybrid ranking scheme, that involves a pre-ranking selection step and only ranks the parameters that are selected to ranking all parameters. Note that all of the data is integrated when fitting the Empircal Bayes prior, and the overlaps are reported with top K parameters from the full parameter list (not from the sublist that meets the pre-selection threshold). We also compare our approach against the ranking achieved using the package 'R-values' [8] Ranking according to R-values is interesting as informally it attempts

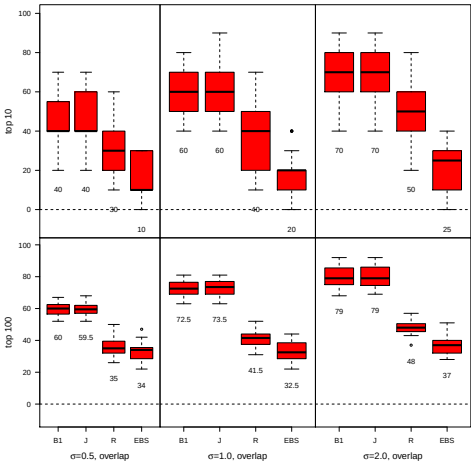
to maximize the overlap between the true and reported top units over all possible list sizes. We had difficulty using the R-values package with our MCMC output, so we ran instead on default settings (which fits a normal prior distribution for effect sizes, with estimated hyperparameters). Again, only parameters that are significant at 5% FDR are ranked. Finally, we compare our Bayesian ranking algorithm to posterior expected fold changes produced via EB-Seq [10]. Results indicate comparable ranking performance for squared error loss and the Jewett et. al. position loss optimized according to our fitted Empirical Bayes ranking model; both of which are superior to using r-values on default settings (which fits an alternative prior). To investigate the effect of Monte-Carlo error, we give results where posterior ranks are calculated using 1, 10, 100 and 1,000 simulations. Figures 4 shows that the effect of Monte Carlo error is practically negligible, even for 10 simulations from the posterior) when effect sizes are reasonably large ($\sigma = 0.5$, 1 or 2). In contrast 1,000 Monte Carlo simulations from the posterior are necessary to generate accurate ranking when the effect sizes are smaller (see $\sigma = .3$, note that not enough SNPs were FDR significant to compare Monte Carlo error at $\sigma = 0.2$). Note that more Monte Carlo simulations might be necessary to ensure accurate ranking if pre-selection is not used. Interestingly, comparing Figure 4(d) with Figure 1(c) (both with n=10), pre-selection (Figure 4(d)) seems to help slightly with identifying the top ranks, in addition to reducing the effect of Monte Carlo error in estimating posterior ranks when effect sizes are weak, but has a negative effect on selecting the top ranks when effect sizes are strong.

Figure 4: RNASeq simulations. Simulations, similar to Figure 2, except that only experiments that had statistically significant $\hat{\theta}$ at a 5% FDR are ranked. Ranking performance measured via percentage overlap between the estimated and true ranks of the top K experiments. Simulations assumed mean counts estimated from [1], and simulated fold change from a log-normal distribution; Estimated fold changes and standard errors were calculated with DEseq2. Each boxplot represents 8 simulations of 10 case RNA-Seq samples and 10-control RNA-Seq samples. The number of Monte Carlo simulations from the posterior for the Bayesian ranking algorithm is varied over the 4 boxplots. B1: Bayesian ranking as described in manscript, J: Bayesian ranking as described in manuscript, except that Jewett[9] position squared error loss used in ranking, R: r-values on default settings [8], EBS: ranking according to EBseq estimated posterior fold change [10]

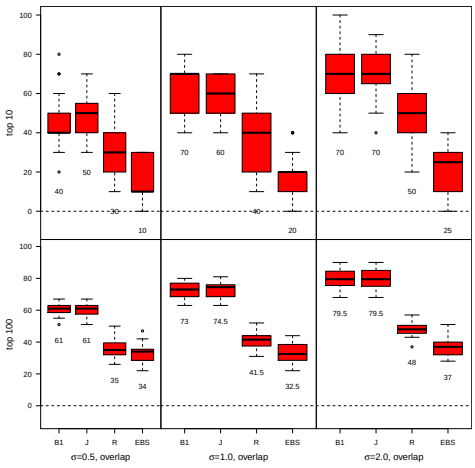
(a) nsim=1




(b) nsim=10




(c) nsim=100




(d) nsim=1,000

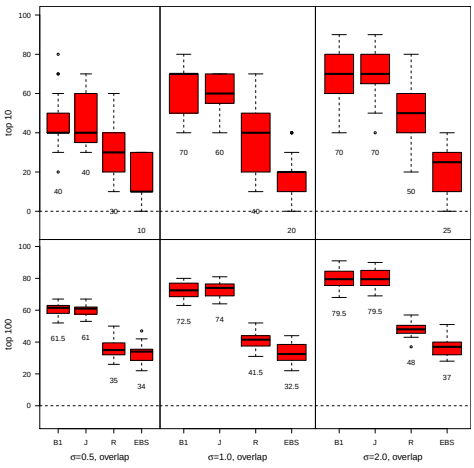
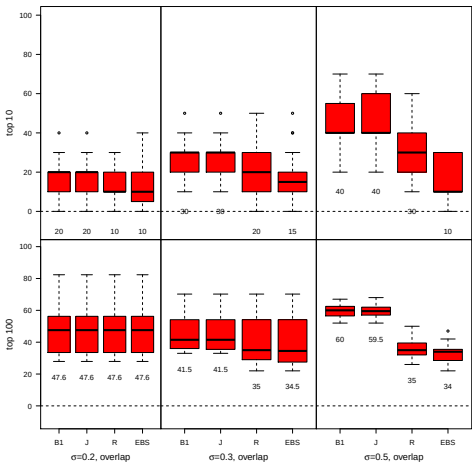
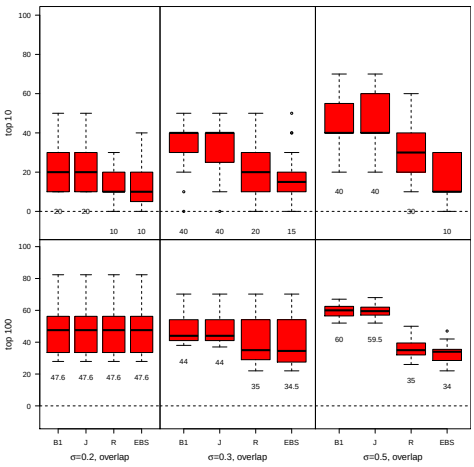


Figure 5: RNASeq simulations. Simulations, similar to Figure 2, except that only experiments that had statistically significant $\hat{\theta}$ at a 5% FDR are ranked. Ranking performance measured via percentage overlap between the estimated and true ranks of the top K experiments. Simulations assumed mean counts estimated from [1], and simulated fold change from a log-normal distribution; Estimated fold changes and standard errors were calculated with DEseq2. Each boxplot represents 8 simulations. The number of Monte Carlo simulations from the posterior for the Bayesian ranking algorithm is varied over the 4 boxplots. Here the true fold changes are on average smaller than the simulations in Figure 4; implying that the Monte Carlo error will be higher. B1: Bayesian ranking as described in manscript, J: Bayesian ranking as described in manuscript, except that Jewett position squared error loss used in ranking [9], R: r-values on default settings [8], EBS: ranking according to EBseq estimated posterior fold change [10]

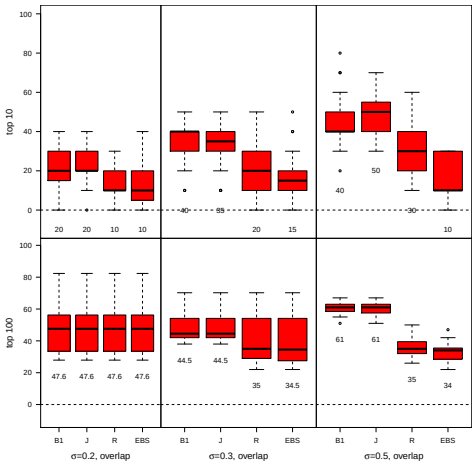
(a) nsim=1




(b) nsim=10




(c) nsim=100




(d) nsim=1,000

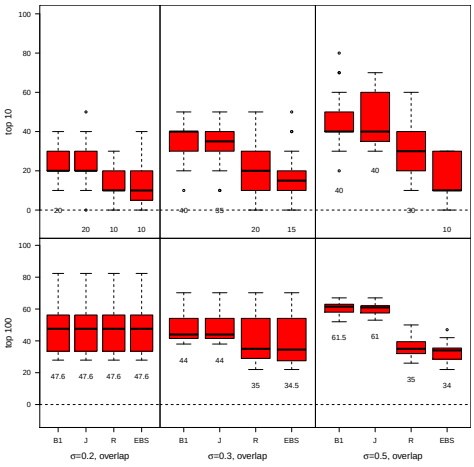
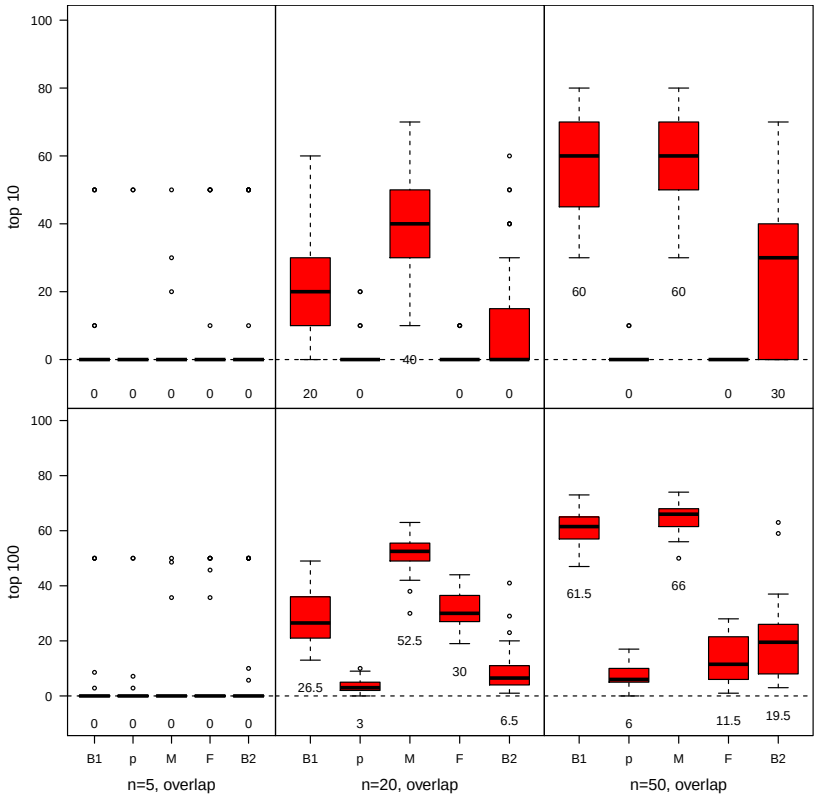


Figure 6: RNA-seq results on real data (rankings evaluated using Limma). B1: Bayesian ranking as described in manscript, p: p-value using Limma, M: non-negative matrix factorization, F: FCROS, B2: Empirical Bayes using Smoothing by Roughening)



## 2.2 Genome Wide Association Studies

As described in the main manuscript, summary Genome Wide Association results were downloaded for 4 study accession numbers, representing Crohn's disease (pha002847), Schizophrenia (pha002857), Multiple Sclerois (pha002861) and Parkinson's Disease (pha002868) , from the dbGAP database. The steps in the simulation of a dataset, for a particular simulation set up (involving a particular disease and either $n$ cases and controls) were as follows:

- The simulations described in the main manuscript assumed underlying true odds ratios for each SNP calculated by adjusting the raw odds ratios for the 'Winner's Curse' phenonemon, using an Empirical Bayes method described in [3]. The results (on the log-scale) are shown in Figure 2.1 below. Note that the blue distribution represents the log OR-distribution that was assumed for the simulations, whereas the red distribution (including individual raw log odds ratios marked by asterixes) shows the distribution of the raw log odds ratios before applying the method.

- Marginal SNP frequencies for each SNP included in the 4 files were found by matching rsid with a database of SNP frequencies for the hg18 genome, found from the UCSC Genome Browser.

- Next, SNP minor allele frequencies in cases and controls were estimated by using Bayes theorem, assuming an additve logistic model relating probability of disease to the number of copies of the SNP and a disease prevalence of 0.01. The log-ORs, representing the $\beta$- coefficient in these regressions, were calculated according to the shrinkage procedure above.

- Finally, sampled log-odds ratios and standard errors were independently calculated for each SNP from these minor allele frequencies,

Figure 7: Shrunken effect sizes used for simulation

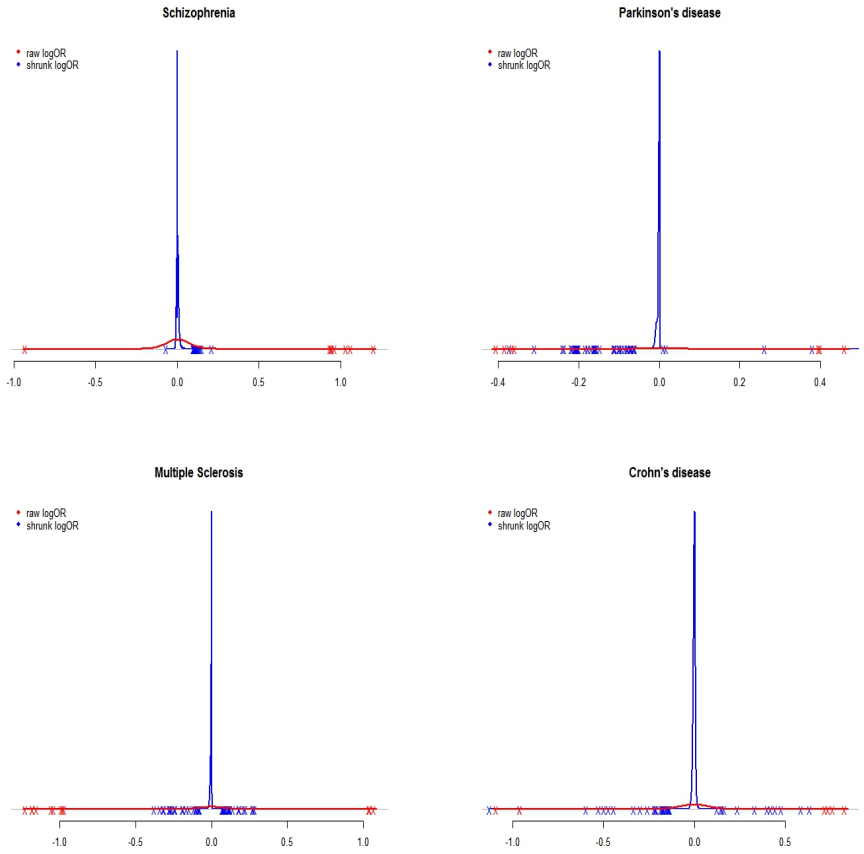


Figure 8: Schizophrenia

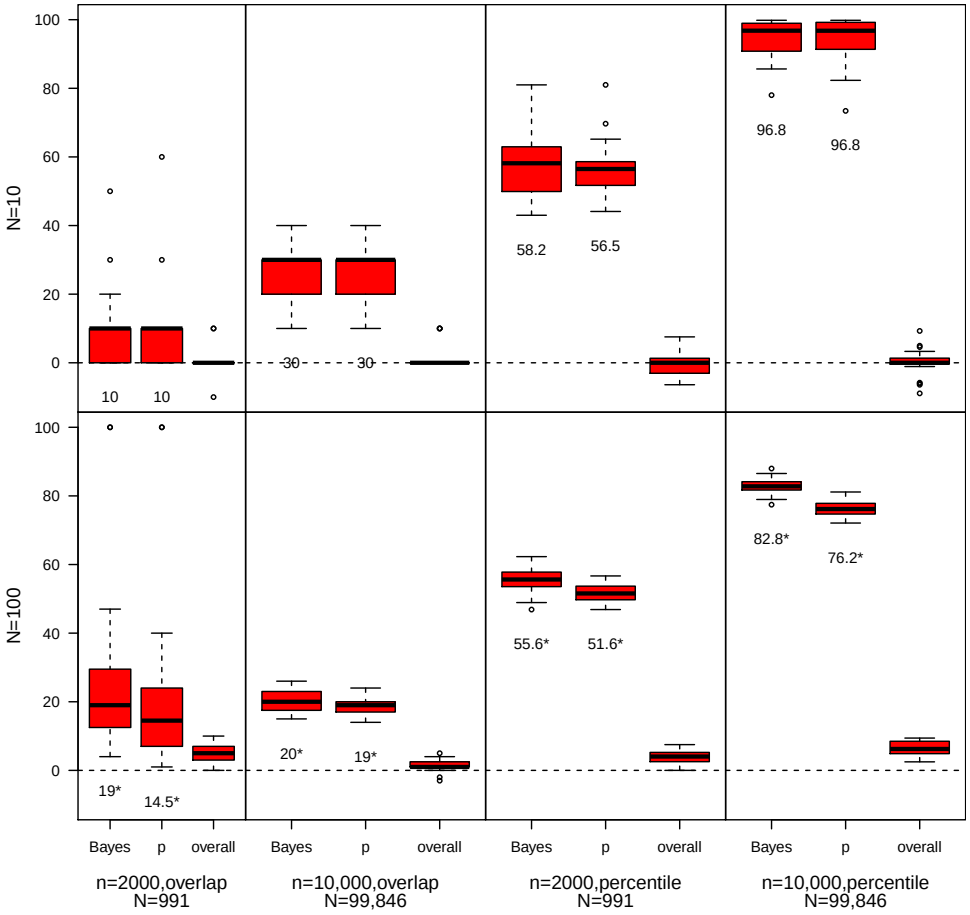


Figure 9: Crohn's Disease

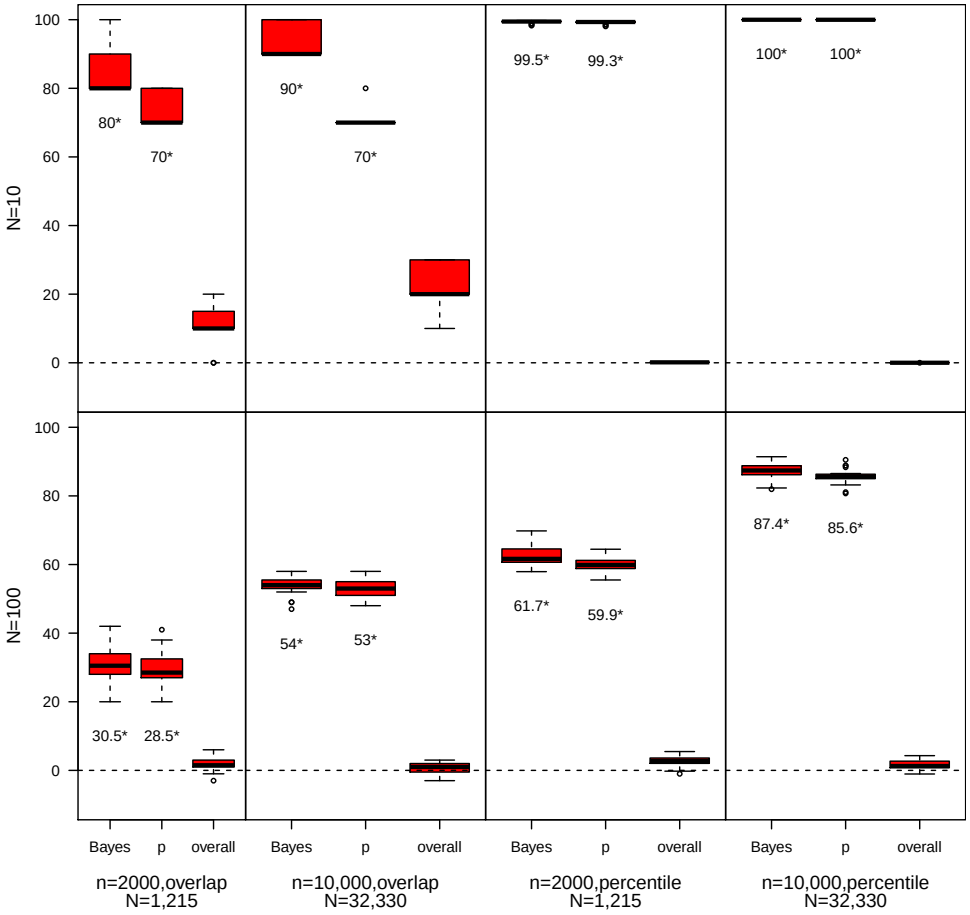


Figure 10: Parkinson's Disease

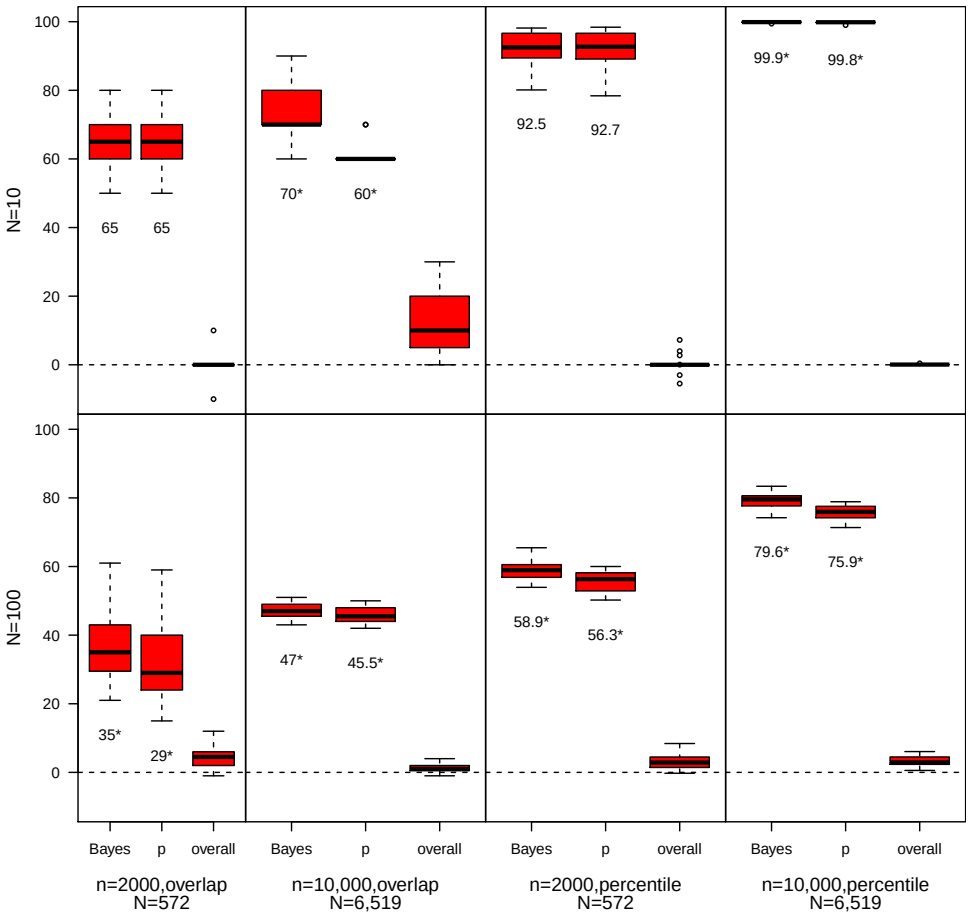


Figure 11: MS

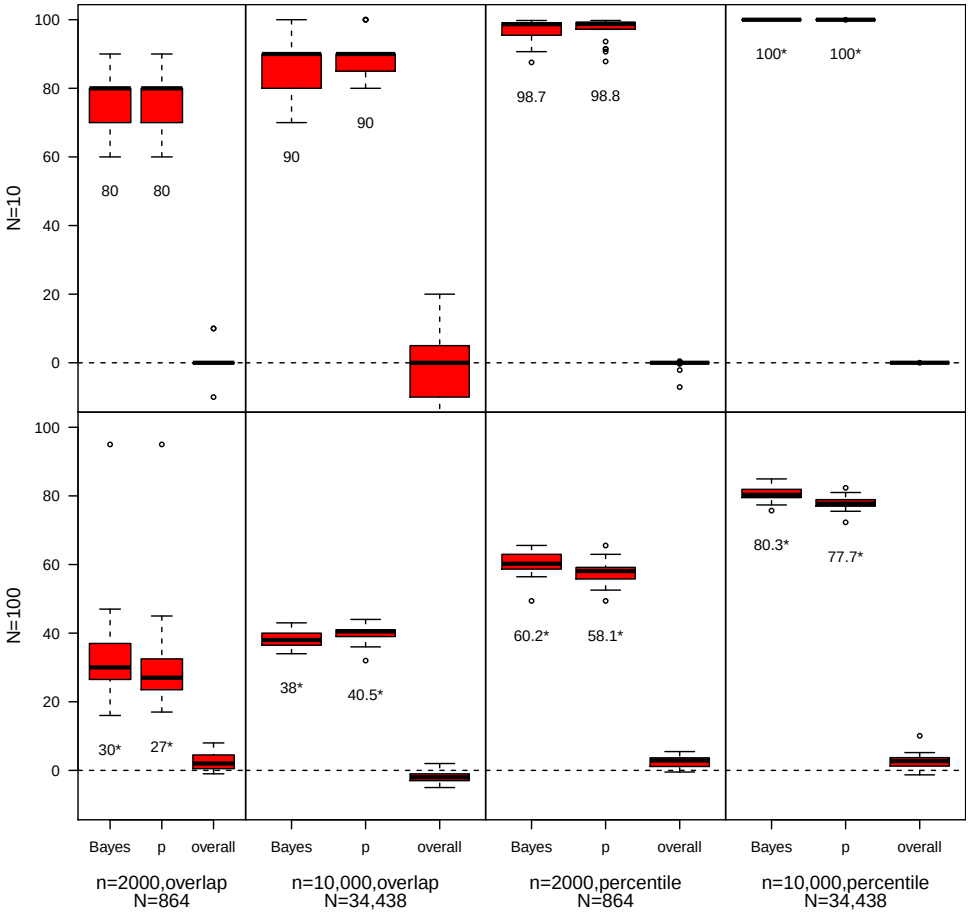

N=10
N=100
100
80
60
40
20
0
80
80
90
90
98.7
98.8
100*
100*
30*
27*
38*
40.5*
60.2*
58.1*
80.3*
77.7*
Bayes
p
overall
n=2000,overlap
N=864
n=10,000,overlap
N=34,438
n=2000,percentile
N=864
n=10,000,percentile
N=34,438

Figure 12: n=2000: weak signal, Bayesian ranking only on 5% FDR significant SNPs

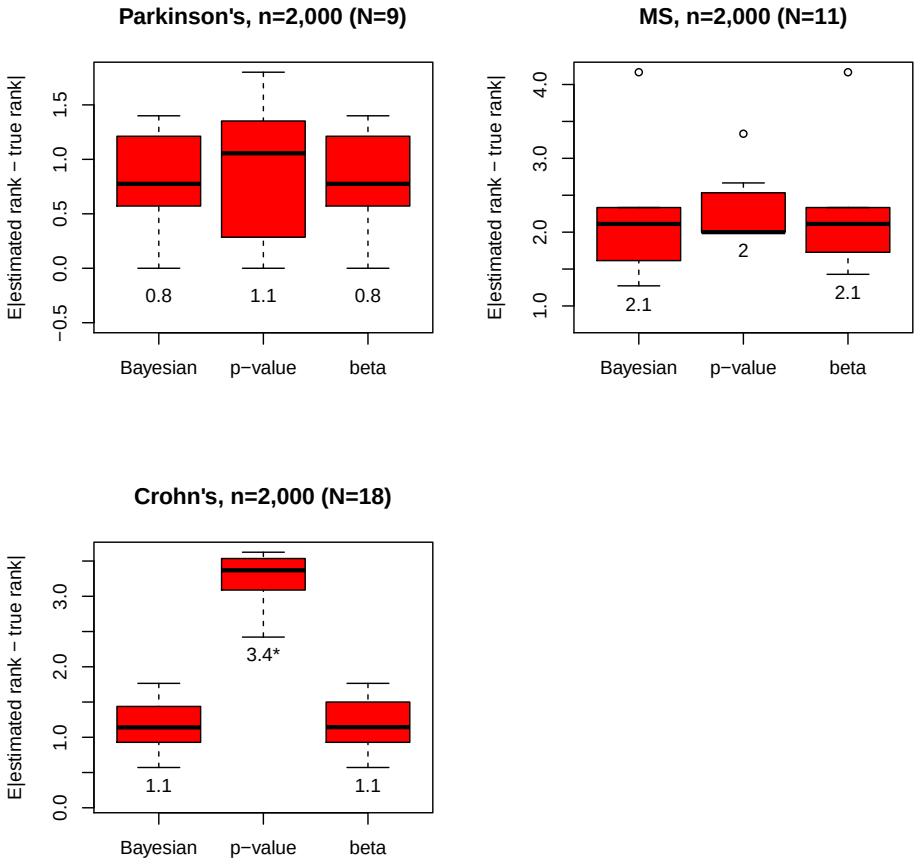
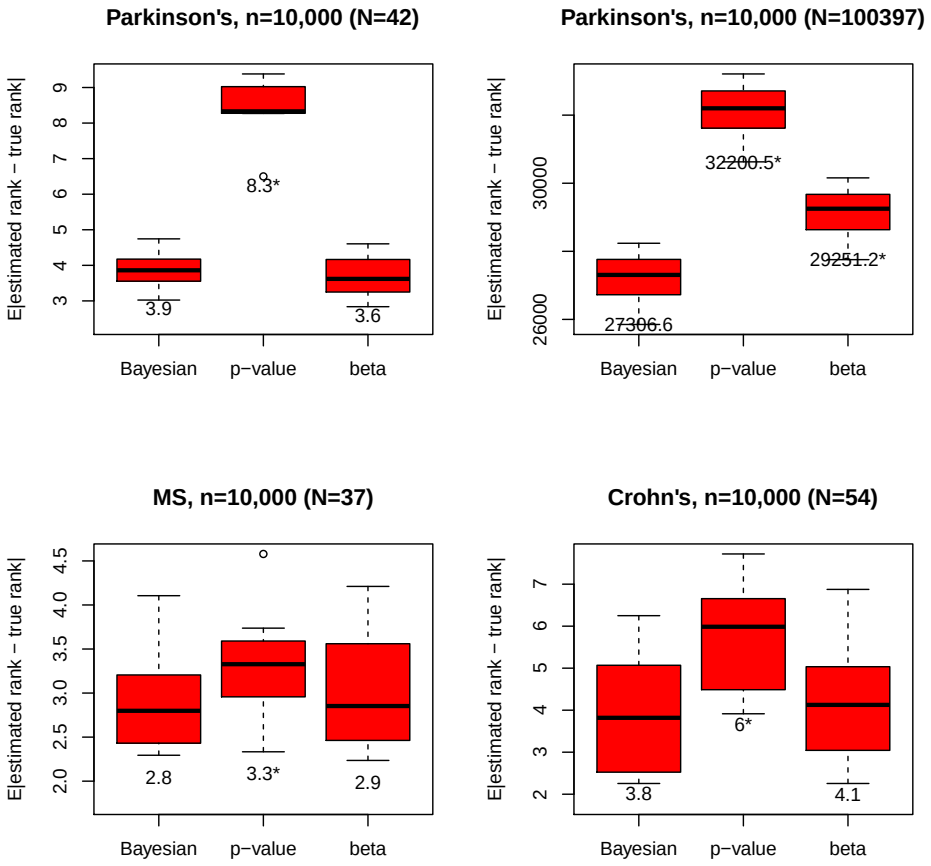


Figure 13: n=10,000: Bayesian ranking only on 5% FDR significant SNPs



# 3 Settings for EM algorithm

As discussed in the main manuscript, we have adapted the EM algorithm described in [6], to estimate the parameters: $\mathbf{m} = (m_0, ..., m_J)$,$\mathbf{s} = (s_0, ..., s_J)$ and $\pi = (\pi_0, .., \pi_J)$. We investigated the influence of a number of tuning parameters for the algorithm in a series of simulations. These tuning parameters were: the maximum number of non-null clusters that could be selected, $J_{max}$, which has a default of 4 in our package; a penalty term, $p$, so that the mixing probabilities on step $t$ of the algorithm, $\pi_0^t, ..., \pi_J^t$ are adjusted to:

$$\begin{aligned}
\pi_0^{t,adjust} &= \frac{1}{N + (N/p)}\{\sum_{i \leq N} P^{(t)}(\theta_i = 0|Z_i) + N/p\} = \frac{1}{1 + (1/p)}(\pi_0^t + 1/p) \\
\pi_j^{t,adjust} &= \frac{1}{1 + (1/p)}\pi_j^t;
\end{aligned}$$

(this is equivalent to adding a prior pseudo-count of $N/p$ to the standard EM estimate of the null probability, then rescaling -[6] uses $p = 5$ as a default in estimating FDR); finally, the effect of forcing the 2nd mixing component to have a mean of 0 and standard deviation of 10. The simulations investigated the ranking quality in terms of the overlap between the indexes for the true top $K$ effect sizes, $\theta_i$ , and the indexes according to the top $K$ estimated for $K \in \{10,\ 100\}$, when 10,000 standardized effect sizes: $\mu = \theta\ /SE(\hat{\theta})$ were simulated under the following mixture normal distributions:

$$\mu \sim \pi_0\delta\{0\} + \tfrac{(1-\pi_0)}{4}N(-3,1) + \tfrac{(1-\pi_0)}{4}N(-1,1) + \tfrac{(1-\pi_0)}{4}N(1,1) + \tfrac{(1-\pi_0)}{4}N(3,1),$$
$$\text{or: } \mu \sim \pi_0\delta\{0\} + (1-\pi_0)N(3,1),$$

depending on whether the number of components, $nc$, was 5 or 2. Other distributions were also investigated, with larger effect sizes and differing numbers of mixing components, but the results are excluded here for brevity. Standard errrors were simulated according to an exponential distribution with rate 1. A few observations regarding these simulations, the results of

which are summarized in supplementary tables 1 and 2 below, are as follows. First, fixing the mean and variance of the second cluster (results not shown) didn't have a great impact on the estimated overlap for the mixture-normal simulation scenarios investigated, provided flexibility was provided to estimate at least 5 clusters ($J_{max} = 4$); the results presented correspond to estimated rankings where the second component was allowed to freely vary. However, as mentioned in the main manuscript, a large variance component was helpful to detect rare strong associations in the context of GWAS data with sparse signals (as simulated in 2.3), so is included as a default in the algorithm. While a substantial Dirichlet penalization ($p = 5$) did improve estimation of the null mixing probability, using a much weaker penalty ($p = 5000$) as shown in Table 1 and 2, or potentially removing the penalty altogether, seem to give more favorable results in Bayesian ranking. The original motivation for the Dirichlet penalty in [6] was actually to allow shrinkage to an empirical null distribution, useful when the test statistics: $\hat{\theta}/SE(\hat{\theta})$ may not be $N(0,1)$ under the null hypothesis, as well as to increase stability of prior estimation in the context of FDR estimation. In the context of Bayesian ranking, these arguments are not as compelling since we are not fitting an empirical null distribution (the first mixing component is always $N(0,1)$), and are interested in ranking rather than effect estimation. The effect of $J_{max}$ corresponds to the extra benefit we achieve by ranking according to the mixture algorithm in [6] compared to estimating with a spike and slab prior). Interestingly, the overlap between the true ranking and estimated ranks seem to be reasonably robust, even when a spike and slab prior is used, indicating a rather limited benefit of fitting more complicated mixture normal distributions. Different combinations of the tuning parameters ($J_{max}$,$p$) were compared by looking at the maximum regret (the deviation from the best overlap over all tuning parameters and the actual overlap), maximized over all possible simulation parameters. ($J_{max} = 4, p = 5000$) was the tuning parameter set minimizing this maximum regret.

Table 1: Effect of tuning parameters on average %overlap in top 10 SNPs (Higher is better)

| | | **Simulation parameters** | | | | | |
|---|---|---|---|---|---|---|---|
| | | nc | 2 | | 5 | | **max-regret %** |
| **Tuning parameters** | | $\pi_0$ | 0.8 | 0.98 | 0.8 | 0.98 | |
| $J_{max}$ | $p$ | | | | | | |
| 1 | 5 | | 62(1.3) | 45(1.5) | 55(1.4) | 32(1.4) | 5 |
| 4 | 5 | | 60(1.3) | 45(1.7) | 55(1.3) | 33(1.3) | 4 |
| 1 | 5000 | | 60(1.2) | 45(1.4) | 55(1.5) | 35(1.4) | 3 |
| 4 | 5000 | | 60(1.2) | 43(1.3) | 55(1.5) | 37(1.4) | 2 |

Table 2: Effect of tuning parameters on average % overlap in top 100 SNPs (Higher is better)

| | | **Simulation parameters** | | | | | |
|---|---|---|---|---|---|---|---|
| | | nc | 2 | | 5 | | **max-regret %** |
| **Tuning parameters** | | $\pi_0$ | 0.8 | 0.98 | 0.8 | 0.98 | |
| $J_{max}$ | $p$ | | | | | | |
| 1 | 5 | | 65(0.4) | 33(0.5) | 58(0.4) | 27(0.4) | 10 |
| 4 | 5 | | 66(0.3) | 37(0.5) | 60(0.4) | 33(0.4) | 4 |
| 1 | 5000 | | 66(0.4) | 40(0.4) | 60(0.4) | 37(0.4) | 1 |
| 4 | 5000 | | 67(0.4) | 41(0.4) | 60(0.4) | 37(0.5) | 0 |

# References


[1] Daniel Bottomly, NA Walter, Jessica Ezzell Hunter, Priscila Darakjian, Sunita Kawane, Kari J Buck, Robert P Searles, Michael Mooney, Shannon K McWeeney, and Robert Hitzemann. Evaluating gene expression in c57bl/6j and dba/2j mouse striatum using rna-seq and microarrays. *PloS one*, 6(3):e17820, 2011.

[2] Angela N Brooks, Li Yang, Michael O Duff, Kasper D Hansen, Jung W Park, Sandrine Dudoit, Steven E Brenner, and Brenton R Graveley. Conservation of an rna regulatory map between drosophila and mammals. *Genome research*, 21(2):193–202, 2011.

[3] John P Ferguson, Judy H Cho, Can Yang, and Hongyu Zhao. Empirical bayes correction for the winner's curse in genetic association studies. *Genetic epidemiology*, 37(1):60–68, 2013.

[4] William A Gale and Geoffrey Sampson. Good-turing frequency estimation without tears*. *Journal of Quantitative Linguistics*, 2(3):217–237, 1995.

[5] Irving J Good. The population frequencies of species and the estimation of population parameters. *Biometrika*, 40(3-4):237–264, 1953.

[6] Omkar Muralidharan et al. An empirical bayes mixture method for effect size and false discovery rate estimation. *The Annals of Applied Statistics*, 4(1):422–438, 2010.

[7] Mark D Robinson, Davis J McCarthy, and Gordon K Smyth. edger: a bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, 26(1):139–140, 2010.

[8] Nicholas C Henderson and Michael A Newton. Making the cut: improved ranking and selection for large-scale inference. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 78(4):781–804, 2016.

[9] Patricia I Jewett, Li Zhu, Bin Huang, Eric J Feuer, and Ronald E Gangnon. Optimal bayesian point estimates and credible intervals for ranking with application to county health indices. *Statistical methods in medical research*, p. 0962280218790104, 2018.

[10] Ning Leng, John A Dawson, James A Thomson, Victor Ruotti, Anna I Rissman, Bart MG Smits, Jill D Haag, Michael N Gould, Ron M Stewart, and Christina Kendziorski. Ebseq: an empirical bayes hierarchical model for inference in rna-seq experiments. *Bioinformatics*, 29(8):1035–1043, 2013.