Moderated estimation of fold change and dispersion for RNA-Seq data with DESeq2

Michael I Love [1,2,3], Wolfgang Huber [1], Simon Anders [1,*]

[1] Genome Biology Unit, European Molecular Biology Laboratory, Heidelberg, Germany
[2] Department of Biostatistics and Computational Biology, Dana Farber Cancer Institute and Department of Biostatistics, Harvard School of Public Health, Boston, MA, USA
[3] Department of Computational Molecular Biology, Max Planck Institute for Molecular Genetics, Berlin, Germany
[*] Corresponding author (email: sanders@fs.tum.de)

19 February 2014

Abstract: In comparative high-throughput sequencing assays, a fundamental task is the analysis of count data, such as read counts per gene in RNA-Seq data, for evidence of systematic changes across experimental conditions. Small replicate numbers, discreteness, large dynamic range and the presence of outliers require a suitable statistical approach. We present *DESeq2*, a method for differential analysis of count data. *DESeq2* uses shrinkage estimation for dispersions and fold changes to improve stability and interpretability of the estimates. This enables a more quantitative analysis focused on the strength rather than the mere presence of differential expression and facilitates downstream tasks such as gene ranking and visualization. *DESeq2* is available as an R/Bioconductor package.

Background

The rapid adoption of high-throughput sequencing (HTS) technologies for genomic studies has resulted in a need for statistical methods to assess quantitative differences between experiments. An important task here is the analysis of RNA-Seq data with the aim of finding genes that are differentially expressed across groups of samples. This task is general: methods for it are typically also applicable for other comparative HTS assays, including ChIP-Seq, 4C, HiC, or counts of observed taxa in metagenomic studies.

Besides the need to account for the specifics of count data, such as non-Normality and a dependence of the variance on the mean, a core challenge is the small number of samples of typical HTS experiments – often as few as two or three replicates per condition. Inferential methods that treat each gene separately suffer here from lack of power, due to the high uncertainty of within-group variance estimates. In high-throughput assays, this can be overcome by pooling information across genes; specifically, by exploiting assumptions about the similarity of the variances of different genes measured in the same experiment [1].

Many methods for differential expression analysis of RNA-Seq data perform such information sharing across genes for variance (or, equivalently, dispersion) estimation. The *edgeR* method [2, 3] moderates the dispersion estimate for each gene toward a common estimate across all genes, or toward a local estimate from genes with similar expression strength, using a weighted conditional likelihood. Our *DESeq* method [4] detects and corrects dispersion estimates which are too low through modeling of the dependence of the dispersion on the average expression strength over all samples. *DSS* [5] uses a Bayesian approach to provide an estimate for the dispersion aimed at fully capturing the heterogeneity of dispersion across

samples. *BaySeq* [6] and *ShrinkBayes* [7] estimate priors for a Bayesian model over all genes, and then provide posterior probabilities or false discovery rates for the case of differential expression.

The most common approach to comparative analysis of transcriptomics data is to test the null hypothesis that the logarithmic fold change (LFC) between treatment and control for a gene's expression is exactly zero, i.e., that the gene is not at all affected by the treatment. Often the goal of a differential analysis is a list of genes passing multiple-test adjustment, ranked by p-value. However, small changes, even if statistically highly significant, might not be the most interesting candidates for further investigation. Ranking by fold-change, on the other hand, is complicated by the noisiness of LFC estimates for genes with low counts. Furthermore, the number of genes called significantly differentially expressed depends as much on the sample size and other aspects of experimental design as it does on the biology of the experiment – and well-powered experiments often generate an overwhelmingly long list of "hits" [8]. We therefore developed a statistical framework to facilitate gene ranking and visualization based on stable estimation of effect sizes (LFCs), as well as testing of differential expression with respect to user-defined thresholds of biological significance.

Here we present *DESeq2*, an update to the *DESeq* methodology [4]. *DESeq2* integrates recent methodological advances with novel features to facilitate a more quantitative analysis of comparative RNA-Seq data, including shrinkage of dispersion and fold change estimates. We demonstrate the advantages of *DESeq2*'s new features by describing a number of applications possible with shrunken fold changes and their estimates of standard error, including improved gene ranking and visualization, hypothesis tests above and below a threshold, and the "regularized logarithm" transformation for quality assessment and clustering of overdispersed count data. We furthermore compare *DESeq2*'s statistical power with existing tools on real datasets, revealing that our methodology has high sensitivity and precision, while effectively controlling the false positive rate. *DESeq2* is available as an R/Bioconductor package [9] at `http://www.bioconductor.org/`.

Results and discussion

Model and normalization

The starting point of a *DESeq2* analysis is a count matrix K with one row for each gene i and one column for each sample j, the matrix entries K_{ij} indicating the number of sequencing reads that have been unambiguously mapped to a gene in a sample. Note that although we refer in this paper to counts of reads in genes, the methods presented here can be applied as well to other kinds of HTS count data. For each gene, we fit a generalized linear model (GLM) [10] as follows.

We model read counts K_{ij} as following a Negative Binomial distribution with mean μ_{ij} and dispersion α_i. The mean is taken as a quantity q_{ij}, proportional to the concentration of cDNA fragments from the gene in the sample, scaled by a normalization factor s_{ij}, i.e., $\mu_{ij} = s_{ij} q_{ij}$. For many applications, the same constant s_j can be used for all genes in a sample, which then accounts for differences in sequencing depth between samples. To estimate these *size factors*, the *DESeq2* package offers the median-of-ratios method already used in *DESeq* [4]. However, it can be advantageous to calculate gene-specific normalization factors s_{ij} to account for further sources of technical biases such as GC content, gene length or the like, using published methods [11, 12], and these can be supplied as well.

We use GLMs with logarithmic link, $\log_2 q_{ij} = \sum_r x_{jr} \beta_{ir}$, with design matrix elements x_{jr} and coefficients β_{ir}. In the simplest case of a comparison between two groups, such as treated and control samples, the design matrix elements indicate whether a sample j is treated or not, and the GLM fit returns coefficients indicating the base line expression of the gene for control samples and the $\log_2$ fold change between treatment and control. The use of linear models, however,

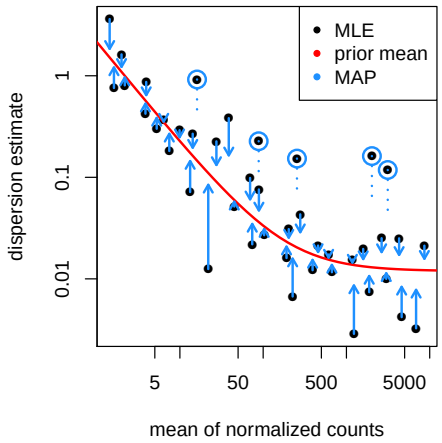


Figure 1: **Shrinkage estimation of dispersion.** Plot of dispersion estimates over the average expression strength for a 3 vs 4 sample comparison in the Bottomly et al. [13] dataset: First, gene-wise maximum likelihood estimates (MLE) are obtained using only the respective gene's data (black dots). Then, a curve (red) is fit to the MLEs to capture the overall trend of dispersion-mean dependence. This fit is used as a prior mean for a second estimation round, which results in the final maximum *a posteriori* (MAP) estimates of dispersion (arrow heads). This can be understood as a shrinkage (along the blue arrows) of the noisy gene-wise estimates towards the consensus represented by the red line. The black points circled in blue are detected as dispersion outliers and not shrunk toward the prior (shrinkage would follow the dotted line). For clarity, only a subset of genes is shown, which is enriched for dispersion outliers.

provides the flexibility to also analyze more complex designs, as is often useful in genomic studies [14].

Empirical Bayes shrinkage for dispersion estimation

Within-group variability, i.e., the variability between replicates, is modeled by the dispersion parameter α_i, which describes the variance of counts via $\mathrm{Var}\, K_{ij} = \mu_{ij} + \alpha_i \mu_{ij}^2$. Accurate estimation of the dispersion parameter α_i is critical for any statistical inference of differential expression. For studies with large sample sizes this is usually not a problem. For controlled experiments, however, sample sizes tend to be smaller (experimental designs with as little as two or three replicates are common and reasonable), resulting in highly variable dispersion estimates for each gene. If used directly, these noisy estimates would compromise the accuracy of differential expression testing.

One sensible solution is to share information across genes. In *DESeq2*, we assume that genes of similar average expression strength have similar dispersion. We here explain the concepts of our approach using as example a dataset by Bottomly et al. [13] with RNA-Seq data of mice of two different strains. For the mathematical details, see Methods.

We first treat each gene separately and estimate "gene-wise" dispersion estimates (using maximum likelihood), which rely only on the data of each individual gene (black dots in Figure 1). Next, we fit a curve to capture the dependence of these estimates on average expression strength (red line in the figure). This provides an accurate estimate for the expected dispersion value for genes of a given expression strength but cannot represent deviations of individual genes from this overall trend. We then shrink the gene-wise dispersion estimates toward the values predicted by the curve to obtain final dispersion values (blue arrow heads). We use an empirical Bayes approach (Methods), which lets the strength of shrinkage depend (i) on an estimate of how close true dispersion values tend to be to the

fit and (ii) on the degrees of freedom: as the sample size increases, the shrinkage decreases in strength, and eventually becomes negligible. Our approach therefore accounts for gene-specific variation to the extent that the data provides this information, while the fitted curve aids estimation and testing in less information-rich settings. We note that our approach is similar to the one used by *DSS* [5].

Note that a number of genes with gene-wise dispersion estimates below the curve have their final estimates raised substantially. The shrinkage procedure thereby helps avoid potential false positives which can result from underestimates of dispersion. If, on the other hand, an individual gene's dispersion is far above the distribution of the gene-wise dispersion estimates of other genes, then the shrinkage would lead to a greatly reduced final estimate of dispersion. We reasoned that in many cases, the reason for extraordinarily high dispersion of a gene is that it does not obey our modeling assumptions; some genes may show much higher variability than others for biological reasons, even though they have the same average expression levels. In these cases, inference based on the greatly reduced, shrunken dispersion estimates could lead to undesirable false positive calls. *DESeq2* handles these cases by using the gene-wise estimate instead of the shrunken estimate when the former is more than 2 residual standard deviations above the curve.

Empirical Bayes shrinkage for fold-change estimation

A common difficulty in the analysis of HTS data is the strong variance of logarithmic fold change estimates (LFCs) for genes with low read count. Figure 2A demonstrates this issue using again the dataset by Bottomly et al. [13] with RNA-Seq data from 10 and 11 samples from mice of two different strains. As visualized by the MA-plot in the top left panel, weakly expressed genes seem to show much stronger differences between the strains than strongly expressed genes. This phenomenon, seen in most HTS datasets, is a direct consequence of the fact that one is dealing with *count* data, in which ratios are inherently more noisy when counts are low. This heteroskedasticity (variance of LFCs depending on mean count) considerably complicates downstream analysis and data interpretation, as it makes effect sizes difficult to compare across the dynamic range of the data.

We propose to shrink LFC estimates toward zero in a manner such that shrinkage is stronger when the available information for a gene is low, which may be because counts are low, dispersion is high, or there are few degrees of freedom. We again employ an empirical Bayes procedure: we first perform ordinary GLM fits to obtain maximum-likelihood estimates (MLE) for the LFCs and then fit a zero-centered Normal distribution to the observed distribution of MLEs over all genes. This distribution is used as a prior on LFCs in a second round of GLM fits, and the maximum of the posterior estimates (MAP) are kept as final estimates of the LFCs. Furthermore, standard errors for these estimates are reported, which is derived from the posterior's curvature at its maximum. (See Methods for details.)

The resulting MAP LFCs are now biased toward zero in a manner that removes the problem of "exaggerated" LFCs for low counts. As Figure 2B shows, the strongest LFCs are no longer exhibited by genes with weakest expression. Rather, the estimates are more evenly spread around zero, and for very weakly expressed genes (less than one read per sample on average), LFCs hardly deviate from zero, reflecting that accurate LFC estimates are not possible here. The strength of shrinkage does not depend simply on the mean count, but rather on the amount of information (as indicated by the observed Fisher information, see Methods) available for the fold change estimation. Two genes with equal expression strength but different dispersions will experience different amount of shrinkage (Figure 2C-D). The shrinkage of LFC estimates can be described as a "bias-variance trade-off" [15]: for genes with little information for LFC estimation, a reduction of the strong variance is "bought" at the cost of accepting a certain bias towards zero, and this can result in an overall reduction in mean squared error, e.g., when comparing to LFC estimates from a new dataset. Genes with high information for LFC

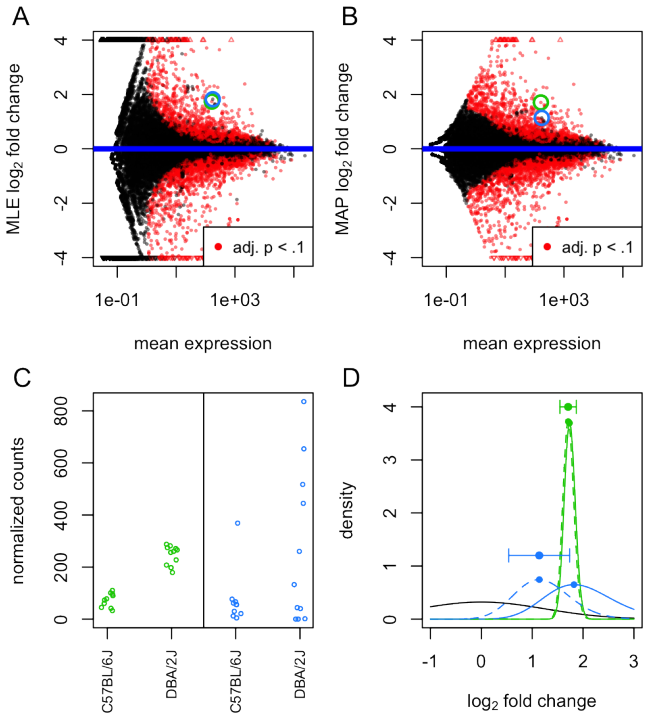


Figure 2: **Effect of shrinkage on logarithmic fold change estimates.** Plots of the (A) maximum likelihood estimate (MLE, i.e., no shrinkage) and (B) maximum *a posteriori* (MAP) estimate (i.e., with shrinkage) for the logarithmic fold changes attributable to mouse strain, over the average expression strength for a 10 vs 11 sample comparison of the Bottomly et al. [13] dataset. Small triangles at the top and bottom of the plots indicate points that would fall outside of the plotting window. Two genes with similar mean count and MLE logarithmic fold change are highlighted in green and blue. (C) The counts (normalized by size factor s_j) for these genes reveal low dispersion for the gene in green and high dispersion for the gene in blue. (D) Density plots of the likelihoods (solid lines, scaled to integrate to 1) and the posteriors (dashed lines) for the green and blue gene and of the prior (solid black line): due to the higher dispersion of the blue gene, its likelihood is wider and less peaked (indicating less information), and the prior has more influence on its posterior than in the case of the green gene. The stronger curvature of the green posterior at its maximum translates to a smaller reported standard error for the MAP LFC estimate (horizontal error bar).

estimation will have, in our approach, LFCs with both low bias and low variance. Furthermore, as the degrees of freedom increase, and the experiment provides more information for LFC estimation, the shrunken estimates will converge to the unshrunken estimates. We note that Bayesian efforts toward moderating fold changes for RNA-Seq include hierarchical models [7] and the "generalized fold change" using a posterior distribution of logarithmic fold changes [16].

The MAP LFCs offer a more reproducible quantification of transcriptional differences than MLE LFCs. To demonstrate this, we split the Bottomly et al. samples equally into two groups, I and II, such that each group contains a balanced split of the strains, simulating a scenario where an experiment (samples in group I) is performed, analyzed and reported, and then independently replicated (samples in group II). Within each group, we estimated LFCs between the strains and compared between group I and II, using the MLE LFCs (Figure 3A) and using the MAP LFCs (Figure 3B). Because the shrinkage moves large LFCs that are not well supported by the data toward zero, the agreement between the two independent sample groups increases considerably.

This makes shrunken LFCs also very suitable for ranking genes, e.g. to prior-

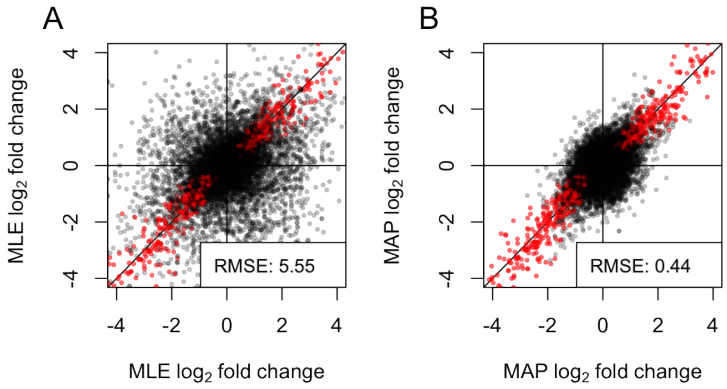


Figure 3: **Stability of logarithmic fold changes.** *DESeq2* is run on equally split halves of the data of Bottomly et al. [13], and the logarithmic fold changes from the halves are plotted against each other, (A) showing MLEs, i.e., without LFC shrinkage, (B) showing MAP estimates, i.e., with shrinkage. Points in the top left and bottom right quadrant indicate genes with a change of sign of logarithmic fold change. Red points indicate genes with adjusted p-value less than 0.1. The legend displays the root mean squared error of the estimates in group I to those in group II.

itize them for detailed follow-up experiments. For example, if we sort the genes in the two sample groups of Figure 3 by unshrunken LFC estimates, and consider the 100 genes with the strongest up- or down-regulation in group I, we find only 21 of these again among the top 100 up- or down-regulated genes in group II. However, if we rank the genes by shrunken LFC estimates, the overlap improves to 81 of 100 genes (Supplemental Figure S1).

A simpler, often used method is to add a fixed number ("pseudocount") to all counts before forming ratios. However, this requires the choice of a tuning parameter and only helps with those high-uncertainty LFCs that are due to low counts. The information-based approach of *DESeq2* offers a more comprehensive solution (Supplemental Figure S2).

Hypothesis tests for differential expression

After GLMs are fit for each gene, one may test for each model coefficient whether it differs significantly from zero. To this end, *DESeq2* reports standard error for each LFC, estimated from the curvature of the coefficient's posterior (dashed lines in Figure 2D) at its maximum. For significance testing, *DESeq2* uses Wald tests: the shrunken estimate of LFC is divided by its standard error, resulting in a z statistic which can be compared to a standard normal. (See Methods for details.) The Wald test allows testing of individual coefficients, or contrasts of coefficients, without the need to fit a reduced model as with the likelihood ratio test, though the likelihood ratio test is also available as an option in *DESeq2*. The Wald test p-values from the subset of genes that pass an independent filtering step, described in the next section, are adjusted for multiple testing using the procedure of Benjamini and Hochberg [17].

Automatic independent filtering

Due to the large number of tests performed in the analysis of RNA-Seq and other genome-wide experiments, the p-values from the gene-wise tests are typically further transformed by multiple testing adjustments. A popular objective is control or estimation of the false discovery rate (FDR). Multiple testing adjustment is associated with a loss of power, in the sense that the false discovery rate for a set of genes is higher than the individual p-values of these genes. However, the

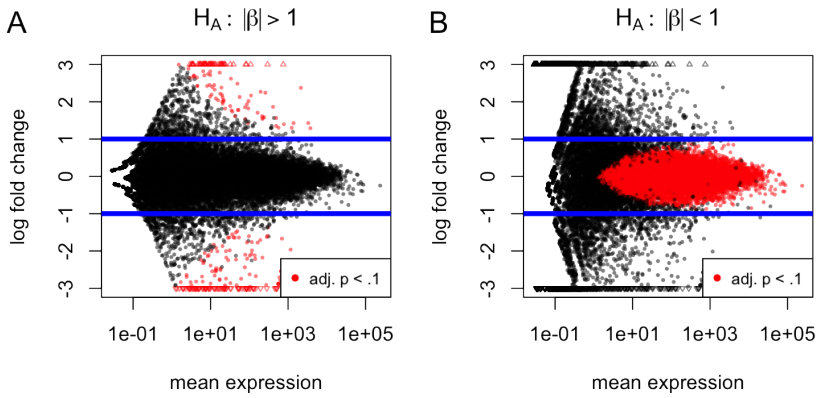


Figure 4: **Hypothesis testing involving non-zero thresholds.** Shown are MA-plots for a 10 vs 11 comparison using the Bottomly et al. [13] dataset, with highlighted points indicating low adjusted p-values. The alternate hypotheses are that logarithmic fold changes are (A) greater than 1 in absolute value or (B) less than 1 in absolute value.

loss can be reduced if genes are omitted from the testing that have little or no chance of being detected as differentially expressed, provided that the criterion for omission is independent of the test statistic under the null [18]. *DESeq2* uses the average expression strength of each gene, across all samples provided, as its filter criterion, and it omits all genes with mean normalized counts below a filtering threshold from multiple testing adjustment. *DESeq2* by default will choose a threshold that maximizes the number of genes found at a user-specified target FDR. In Figure 2A-B, genes with significant LFC at an FDR of 10% are depicted in red.

Depending on the distribution of mean normalized counts for each gene, the resulting increase in power can be substantial, sometimes rescuing a set of genes with adjusted p-values below a given threshold in an experiment which otherwise would have had no such genes passing multiple test adjustment.

Hypothesis tests with thresholds on effect size

Specifying minimum effect size.

Most approaches to testing for differential expression, including the default approach of *DESeq2*, test against the null hypothesis of *zero* logarithmic fold change. Hence, once some genes are genuinely affected by the difference in experimental treatment, this null hypothesis implies that the gene under consideration is *perfectly* decoupled from the affected genes. Due to the high interconnectedness of cells' regulatory networks, it seem reasonable to argue that a change in one gene's expression might indirectly influence nearly all other genes, although many of them so indirectly and hence weakly that the change caused is small. Nevertheless, with sufficient sample size, even genes with a very small, but non-zero logarithmic fold change will eventually be detected as differentially expressed. A change should therefore be of sufficient magnitude to be considered *biologically significant*. For small scale experiments, statistical significance is often a much stricter requirement than biological significance, thereby relieving the researcher from the need to decide on a threshold for biological significance.

For well-powered experiments, however, a statistical test against the conventional null hypothesis of zero logarithmic fold change may report genes with statistically significant changes that are so weak in effect strength that they could be considered irrelevant or distracting. A common procedure is to disregard genes whose estimated logarithmic fold change β_{ir} is below some threshold, $|\beta_{ir}| \leq \theta$. However, this approach loses the benefit of an easily interpretable false discovery

rate, as the p-value and adjusted p-value still correspond to the test of *zero* logarithmic fold change. It is therefore desirable to include the threshold into the statistical testing procedure directly, i.e., not to filter post-hoc on a reported fold-change *estimate*, but rather to statistically evaluate whether there is sufficient evidence that the logarithmic fold change is above the chosen threshold.

DESeq2 offers tests for composite null hypotheses of the form $|\beta_{ir}| \leq \theta$. (See Methods for details.) Figure 4A demonstrates how such a thresholded test gives rise to a curved decision boundary: to reach significance, the estimated LFC has to exceed the specified threshold by an amount that depends on the available information. Such approaches to generate gene lists that satisfy both statistical and biological significance criteria have been previously discussed for both microarray and sequencing data [19, 20].

Specifying maximum effect size.

Sometimes, a researcher is interested in finding genes that are only very weakly affected by the treatment or experimental condition. This amounts to a setting similar to the one just discussed, but the roles of null and alternative hypotheses are swapped. This is because, again, the question "which genes are not differentially expressed?" is hard to answer definitively, while a more tractable question is: "for which genes is there evidence that the effect of the treatment was only weak?" Here, one needs to quantify the meaning of *weak* for the biological question at hand by choosing a suitable threshold θ for the LFC. For such analyses, *DESeq2* offers a test of the composite null hypothesis $|\beta_{ir}| \geq \theta$, which will report genes as significant for which there is evidence that their LFC is weaker than θ. Figure 4B shows the outcome of such a test. For genes with very low read count, even an estimate of zero LFC is not significant, as the large uncertainty of the estimate does not allow us to exclude that the gene may in truth be more than weakly affected by the experimental condition. Note the lack of LFC shrinkage: To find genes with weak differential expression, *DESeq2* requires that the LFC shrinkage has been disabled. This is because the zero-centered prior used for LFC shrinkage embodies a *prior* belief that LFCs tend to be small, and hence is inappropriate here.

Detection of count outliers

Parametric methods for detecting differential expression can have gene-wise estimates of logarithmic fold change overly influenced by individual count outliers that do not fit the distributional assumptions of the model [21]. An example of such an outlier would be a gene with single-digit counts for all samples, except one sample with a count in the thousands. As the aim of differential expression analysis is typically to find *consistently* up- or down-regulated genes, it is useful to consider diagnostics for detecting individual observations which overly influence the logarithmic fold change estimate and p-value for a gene. A standard outlier diagnostic is Cook's distance [22], which is defined within each gene for each sample as the scaled distance that the coefficient vector, $\vec{\beta}_i$, of a linear or generalized linear model would move if the sample were removed and the model refit.

DESeq2 therefore flags, for each gene, those samples which have a Cook's distance greater than the 0.99 quantile of the $F(p, m-p)$ distribution, where p is the number of model parameters including the intercept, and m is the number of samples. The use of the F distribution is motivated by the heuristic reasoning that removing a single sample should not move the vector $\vec{\beta}_i$ outside of a 99% confidence region around $\vec{\beta}_i$ fit using all the samples [22]. However, if there are 2 or fewer replicates for a condition, these samples do not contribute to outlier detection, as there are insufficient replicates to determine outlier status.

How should one deal with flagged outliers? In an experiment with many replicates, discarding the outlier and proceeding with the remaining data might make most use of the available data. In a small experiment with few samples,

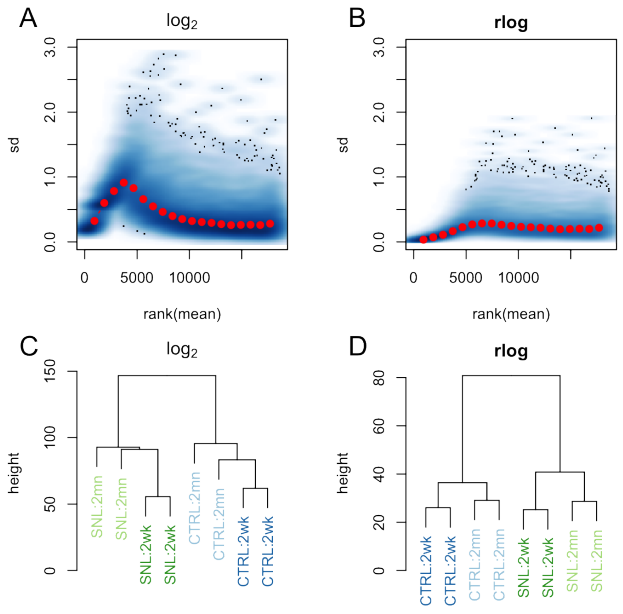


Figure 5: **Variance stabilization and clustering after transformation.** Two transformations are applied to the counts of the Hammer et al. [23] dataset: the logarithm of normalized counts plus a pseudocount, i. e. $f(K_{ij}) = \log_2(K_{ij}/s_j + 1)$, and the rlog. The gene-wise standard deviation of transformed values is variable across the range of the mean of counts using the logarithm (A), while relatively stable using the rlog (B). A hierarchical clustering on Euclidean distances and complete linkage using the rlog (D) transformed data clusters the samples into the groups defined by treatment and time, while using the logarithm transformed counts (C) produces a more ambiguous result.

however, the presence of an outlier may impair inference regarding the affected gene, and merely ignoring the outlier may even be considered data cherry-picking – and therefore, it may be more prudent to exclude the whole gene from downstream analysis.

Hence, *DESeq2* offers two possible responses to flagged outliers. By default, outliers in conditions with 6 or fewer replicates cause the whole gene to be flagged and removed from subsequent analysis, including p-value adjustment for multiple testing. For conditions that contain 7 or more replicates, *DESeq2* replaces the outlier counts with an imputed value, namely the trimmed mean over all samples, scaled by the size factor, and then re-estimates the dispersion, logarithmic fold changes and p-values for these genes. As the outlier is replaced with the value predicted by the null hypothesis of no differential expression, this is a more conservative choice than simply omitting the outlier. When there are many degrees of freedom, the second approach avoids discarding genes which might contain true differential expression.

Supplementary Figure S3 displays the outlier replacement procedure for a single gene in a 7 by 7 comparison of the Bottomly et al. [13] dataset. While the original fitted means are heavily influenced by a single sample with a large count, the corrected logarithmic fold changes provide a better fit to the majority of the samples.

Regularized logarithm transformation

The *DESeq2* framework includes shrinkage of logarithmic fold changes for count data informed by the variance-mean relationship across all genes. Along these lines, we propose a "regularized logarithm" transformation (rlog), which behaves

similarly to a $\log_2$ transformation for genes with high counts, while shrinking together the values for different samples for genes with low counts. It therefore avoids a commonly observed consequence of the standard logarithm transformation, which spreads apart data points for genes with low counts, where random noise is likely to dominate any biologically meaningful signal. The rlog transformation then helps to stabilize the variance of counts across samples, which would otherwise strongly depend on the mean counts, facilitating visualization and clustering of samples.

The rlog transformation is calculated by fitting for each gene a GLM with a base-line expression and, for each sample, shrunken logarithmic fold changes with respect to the base-line, using the same empirical Bayes procedure as before (Methods). Here, however, the sample covariate information is not used, in order to treat all samples equally. The rlog transformation accounts for variation in sequencing depth across samples as it represents the logarithm of q_{ij} after accounting for the size factors s_{ij}. This is in contrast to the variance stabilizing transformation (VST) introduced in DESeq [4]: while the VST for counts is also effective at stabilizing variance, it does not directly take into account differences in size factors, and in datasets with large variation in sequencing depth (say, dynamic range of size factors > 4) we observed undesirable artifacts in the performance of the VST. A disadvantage of the rlog transformation with respect to the VST is, however, that the ranking of genes within a sample will change if neighboring genes undergo shrinkage of different strength. As with the VST, for typical RNA-Seq datasets (with asymptotic dispersion values less than 0.1), the value of $\mathrm{rlog}(K_{ij})$ for large counts is approximately equal to $\log_2(K_{ij})$. Both the rlog transformation and the VST are provided in the *DESeq2* package.

We demonstrate the use of the rlog transformation on the RNA-Seq dataset of Hammer et al. [23], wherein RNA was sequenced from the dorsal root ganglion of rats which had undergone spinal nerve ligation and controls, at 2 weeks and at 2 months after the ligation. The count matrix for this dataset was downloaded from the ReCount online resource [24]. This dataset offers more subtle differences between different conditions than the Bottomly et al. [13] dataset. Figure 5 provides diagnostic plots of the normalized counts under the ordinary logarithm plus one pseudocount and the rlog transformation, showing that the rlog both stabilizes the variance through the range of the mean of counts and helps to find meaningful patterns in the data. The rlog transformation is therefore more appropriate than the usual logarithm for visualization and machine learning applications such as clustering or classification, where otherwise the high variance of logarithm-transformed low counts might overly contribute to the Euclidean distances between samples.

Gene-level analysis

DESeq2 performs analysis on counts of reads which can be uniquely assigned to genes, while a number of other algorithms [25, 26] perform differential analysis on a probabilistic assignment of reads to transcripts. Not attempting to deconvolve the total read count for a gene into a probabilistic assignment to transcripts might result in false positives of differential expression from a change in proportion of isoforms with different lengths, and even in a wrong sign of LFCs if expression fold changes are small compared to differences in length of alternatively used isoforms. However, in our benchmark, discussed in the following section, we found that disparate sign of LFC from count-based and probabilistic-assignment-based methods was rare for genes found to be differential expressed by either method (Supplemental Figure S4). Furthermore, if estimates for average transcript length are available for the conditions, these can be incorporated into the *DESeq2* framework as gene- and sample-specific normalization factors. In addition, the statistical methodology developed here is extendable to perform isoform-specific analysis, either through generalized linear modeling at the exon level with a gene-specific mean [27] or through counting alternative splicing events

through the use of splice graphs [28].

Benchmarks

Benchmark criteria

To compare the performance of *DESeq2* to other state-of-the-art tools, we benchmarked a number of algorithms. We chose to use real RNA-Seq data rather than simulated data, because we feel that while simulation is useful to verify how well an algorithm behaves with idealized, theoretical data, and can identify potential problems that already manifest themselves at that level, simulation does not inform us how well the theory fits reality. However, with real RNA-Seq data there is the complication of not knowing directly or fully the underlying truth. Acknowledging this complication, we considered three performance metrics for differential expression calling: false positive rate (ratio of false positives out of all non-differentially-expressed genes, equal to one minus the specificity), sensitivity (ratio of true positives out of all differentially expressed genes), and precision (ratio of true positives out of genes called differentially expressed).

We can obtain meaningful estimates of specificity from looking at datasets where we believe all genes fall under the null hypothesis of no differential expression [29]. Sensitivity and precision are more difficult to estimate, as they require independent knowledge of those genes that are differentially expressed. To circumvent this problem, we used experimental reproducibility on independent samples (but in the same dataset) as a proxy. We used a dataset with large numbers of replicates in both of two groups, where we expect that truly differentially expressed genes exist. We repeatedly split this dataset into an evaluation set and a larger verification set, and compared the calls from the evaluation set with the calls from the verification set which were taken as "truth". It is important to keep in mind that the calls from the verification set are only an approximation of the true differential state, and the approximation error has a systematic and a stochastic component. The stochastic component vanishes with large enough sample size for the verification set. For the systematic errors, our benchmark assumes that these affect all algorithms equally and therefore do not change the ranking of the algorithms.

The performance of *DESeq2* was benchmarked against the following other algorithms for differential expression at the gene-level: the Negative Binomial based approaches *DESeq (old)* [4], *edgeR* [30], and *DSS* [5], the *voom* normalization method followed by linear modeling using the *limma* package [31], the *SAMseq* permutation method of the *samr* package [21], and the *Cuffdiff 2* [25] method of the Cufflinks suite. For version numbers of the software used, see Supplementary Table S3. For all algorithms, the p-values from genes with non-zero sum of read counts across samples were adjusted using the Benjamini-Hochberg procedure [17].

False positive rate

To evaluate the false positive rate of the algorithms, we tested the amount of differential expression calling by generating mock comparisons from a dataset with many samples and no known condition dividing the individuals represented by the samples. We downloaded the RNA-Seq data of Pickrell et al. [32] on lymphoblastoid cell lines derived from unrelated Nigerian individuals. We chose a set of 26 RNA-Seq samples of the same read length (46 base pairs) and of different male individuals. We randomly drew without replacement 10 samples from the set to perform a comparison of 5 against 5, and this process was repeated 30 times. We estimated the false positive rate using a small critical value, as the genes passing a strict p-value threshold are those likely to pass multiple test adjustment. Namely, we estimated the probability to generate a p-value less than 0.01 as the sum of p-values less than 0.01 divided by the total number of tests excluding genes with zero sum of read counts across samples. The results over the 30 replications

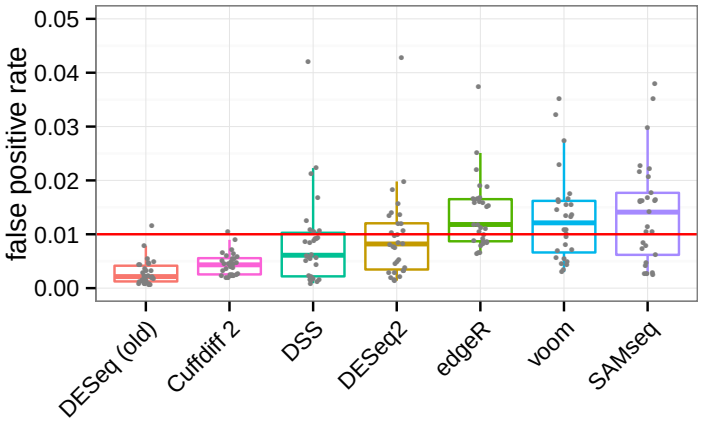


Figure 6: **Benchmark of false positive calling.** Shown are estimates of P(p-value < 0.01) under the null hypothesis. The number of p-values less than 0.01 divided by the total number of tests, from randomly selected comparisons of 5 vs 5 samples from the Pickrell et al. [32] dataset, with no known condition dividing the samples. Type-I error control requires that the tool does not substantially exceed the nominal value of 0.01 (red line). 3 values cropped on the vertical axis were 2 outliers for the *voom* algorithm and 1 for the *SAMseq* algorithm.

are summarized in Figure 6, indicating that all algorithms generally control the number of false positives. *DESeq (old)* and *Cuffdiff 2* appear overly conservative in this analysis, not using up their type-I error "budget".

Sensitivity

To obtain an impression of the sensitivity of the algorithms, we considered the Bottomly et al. [13] dataset, which contains 10 and 11 replicates of two different, genetically homogeneous mice strains. This allowed for a split of 3 vs 3 for the evaluation set and 7 vs 8 for the verification set, which were balanced across the 3 experimental batches. Random splits were replicated 30 times. Batch information was not provided to the *DESeq (old)*, *DESeq2*, *DSS*, *edgeR*, and *voom* algorithms, which can accomodate complex experimental designs, in order to have comparable calls across all algorithms.

We rotated though each algorithm in order to determine the calls of the verification set. Against a given algorithm's verification set calls, we tested the evaluation set calls for every algorithm. We used this approach rather than a consensus-based method, as we did not want to favor or disfavor any particular algorithm or group of algorithms. Defining E as the set of genes with adjusted p-value less than 0.1 in the evaluation set, and V as the set of genes with adjusted p-value less than 0.1 in the verification set, the sensitivity was estimated as $|E \cap V|/|V|$. Figure 7 displays the estimates of sensitivity for each algorithm-algorithm pair, where the different panels designate which algorithm was chosen for the verification set.

The ranking of algorithms is generally consistent regardless of which algorithm is chosen to determine calls in the verification set. *DESeq2* has comparable sensitivity to *edgeR* and *voom* though less than *DSS*. The median sensitivity estimates were typically between 0.2 and 0.4 for all algorithms. That all algorithms have relatively low median sensitivity can be explained by the small sample size of the evaluation set and the fact that increasing the sample size in the verification set increases power. It is expected that the permutation-based *SAMseq* method rarely produced adjusted p-value less than 0.1 in the evaluation set, because the 3 vs 3 comparison does not enable enough permutations.

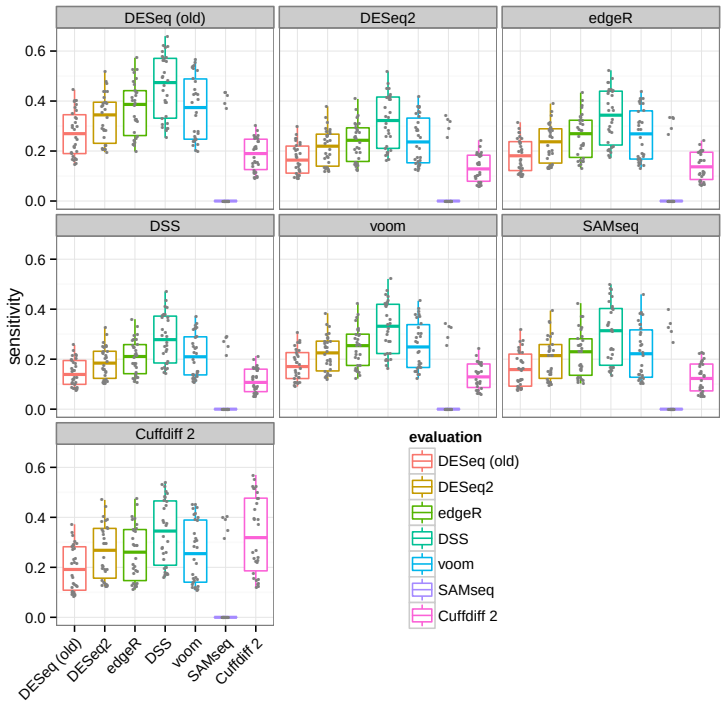


Figure 7: **Sensitivity estimated from experimental reproducibility.** Each algorithm's sensitivity in the evaluation set (boxplots) is evaluated using the calls of each other algorithm in the verification set (panels with grey label). Sensitivity is estimated as the ratio of genes with adjusted p-value < 0.1 in the evaluation set (positives) within the set of genes with adjusted p-value < 0.1 in the verification set (true).

Precision

Another important consideration from the perspective of an investigator is the precision, or ratio of true positives in the set of genes which pass the adjusted p-value threshold. This can also be reported as $1 - \text{FDR}$, the false discovery rate. Using the previously defined sets, E for evaluation set and V for verification set, the precision was estimated as $|E \cap V|/|E|$. The estimates of precision are displayed in Figure 8, where we can see that *DESeq2* often has the second highest median precision, behind *DESeq (old)*. We can also see that algorithms which had higher median sensitivity, e.g., *DSS*, are generally associated here with lower median precision. The rankings differed significantly when *Cuffdiff 2* was used to determine the verification set calls. This is likely due to the additional steps *Cuffdiff 2* performs to deconvolve changes in isoform-level abundance from gene-level abundance, apparently at the cost of lowered precision when compared against its own calls.

The absolute number of calls for the evaluation and verification sets can be seen in Supplemental Figures S5 and S6, which mostly matches the order seen in the sensitivity plot of Figure 7. Supplemental Figures S7 and S8 provide heatmaps and clustering based on the Jaccard index of calls for one replicate of the evaluation and verification sets, indicating a large overlap of calls across the different algorithms.

In summary, the benchmarking tests showed that *DESeq2* effectively controlled type-I error, maintaining a median false positive rate below the chosen critical value in a mock comparison of groups of samples randomly chosen from a larger pool. In addition *DESeq2* achieved a balance of both high sensitivity and high precision when experimental reproducibility in a verification set was

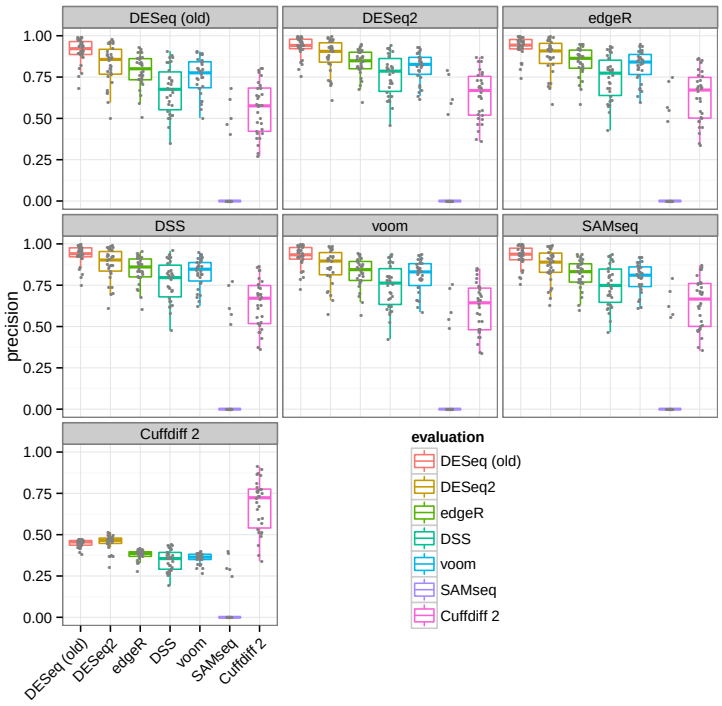


Figure 8: **Precision estimated from experimental reproducibility.** Each algorithm's precision in the evaluation set (boxplots) is evaluated using the calls of each other algorithm in the verification set (panels with grey label). Precision is estimated as the ratio of genes with adjusted p-value < 0.1 in the verification set (true) within the set of genes with adjusted p-value < 0.1 in the evaluation set (positives).

considered.

Conclusion

DESeq2 offers a comprehensive and general solution for gene-level analysis of RNA-Seq data. The use of shrinkage estimators substantially improves the stability and reproducibility of analysis results compared to maximum-likelihood based solutions. The use of empirical Bayes priors provides automatic control of the amount of shrinkage based on the amount of information for the estimated quantity available in the data. This allows *DESeq2* to offer consistant performance over a large range of dataset types and makes it applicable for small studies with few replicates as well as for large observational studies. *DESeq2*'s heuristics for outlier detection help to recognize genes for which the modeling assumptions are unsuitable and so avoids type-I errors caused by these. The embedding of these strategies in the framework of generalized linear models enables the treatment of both simple and complex designs.

A critical advance is the shrinkage estimator for fold changes, which offers a sound and statistically well-founded solution to the practically relevant problem of comparing fold change across the wide dynamic range of RNA-Seq experiments. This is of value for many downstream analysis tasks, including the ranking of genes for follow-up studies, visualization of changes in heat maps and analysis with machine-learning or ordination techniques such as principal-component analysis and clustering using Euclidean distance, which require homoskedastic input data.

DESeq2 hence offers to practitioners a wide set of features with state-of-the-art inferential power. Its use cases are not limited to RNA-Seq data or other

transcriptomics assays; rather, many kinds of high-throughput count data can be used. Other areas for which *DESeq* or *DESeq2* or have been used include ChIP-Seq assays (e.g., the *DiffBind* package [33, 34]), barcode-based assays (e.g., [35]) and metagenomics data (e.g., [36]). Finally, the *DESeq2* package is well integrated in the Bioconductor infrastructure [9] and comes with extensive documentation, including a vignette that demonstrates a complete analysis step by step and discusses advanced use cases.

Methods

A summary of the notation used in the following section is provided in Supplemental Table S1.

Model and normalization

The read count K_{ij} for gene i in sample j is described with a generalized linear model (GLM) of the Negative Binomial family with logarithmic link:

$$K_{ij} \sim NB(\text{mean} = \mu_{ij}, \text{dispersion} = \alpha_i) \tag{1}$$

$$\mu_{ij} = s_{ij} q_{ij}$$

$$\log q_{ij} = \sum_r x_{jr} \beta_{ir}. \tag{2}$$

For notational simplicity, the equations here use the natural logarithm as the link function, though the *DESeq2* software reports estimated model coefficients and their estimated standard errors on the $\log_2$ scale.

By default, the normalization constants s_{ij} are considered constant within a sample, $s_{ij} = s_j$, and are estimated with the median-of-ratios method previously described and used in *DESeq* [4] and *DEXSeq* [27]:

$$s_j = \underset{i:\, K_i^{\mathrm{R}} \neq 0}{\text{median}} \frac{K_{ij}}{K_i^{\mathrm{R}}} \quad \text{with} \quad K_i^{\mathrm{R}} = \left(\prod_{j=1}^{m} K_{ij} \right)^{1/m}.$$

Alternatively, the user can supply normalization constants s_{ij} calculated using other methods (e.g., using *cqn* [11] or *EDASeq* [12]), which may differ from gene to gene.

Expanded design matrices

For consistency with our software's documentation, in the following text we will use the terminology of the *R* statistical language. In linear modeling, a categorical variable or *factor* can take on two or more values or *levels*. In standard design matrices, one of the values is chosen as a reference value or *base level* and absorbed into the intercept. In standard GLMs, the choice of base level does not influence the values of contrasts (LFCs). This, however, is no longer the case in our approach using ridge-regression-like shrinkage on the coefficients (described below), when factors with more than two levels are present in the design matrix, because the base level will not undergo shrinkage while the other levels do.

To recover the desirable symmetry between all levels, *DESeq2* uses *expanded design matrices* which include an indicator variable for *each* level of each factor, in addition to an intercept column (i.e., none of the levels is absorbed into the intercept). While such a design matrix no longer has full rank, a unique solution exists because the zero-centered prior distribution (see below) provides regularization. For dispersion estimation and for estimating the width of the LFC prior, standard design matrices are used.

Contrasts

Contrasts between levels and standard errors of such contrasts can be calculated as they would in the standard design matrix case, i.e., using:

$$\beta_i^c = \vec{c}^t \vec{\beta}_i \tag{3}$$

$$\mathrm{SE}(\beta_i^c) = \sqrt{\vec{c}^t \Sigma_i \vec{c}}, \tag{4}$$

where $\vec{c}$ represents a numeric contrast, e.g., 1 and -1 specifying the numerator and denominator of a simple two level contrast, and $\Sigma_i = \mathrm{Cov}(\vec{\beta}_i)$, defined below.

Estimation of dispersions

We assume the dispersion parameter α_i follows a log-Normal prior distribution that is centered around a trend that depends on the gene's mean normalized read count:

$$\log \alpha_i \sim N(\log \alpha_{\mathrm{tr}}(\bar{\mu}_i), \sigma_{\mathrm{d}}^2). \tag{5}$$

Here, α_{tr} is a function of the gene's mean normalized count, $\bar{\mu}_i = \frac{1}{m}\sum_j (K_{ij}/s_{ij})$. It describes the mean-dependent expectation of the prior. σ_{d} is the width of the prior, a hyperparameter describing how much the individual genes' true dispersions scatter around the trend. For the trend function, we use the same parametrization as we used for *DEXSeq* [27], namely,

$$\alpha_{\mathrm{tr}}(\bar{\mu}) = \frac{a_1}{\bar{\mu}} + \alpha_0. \tag{6}$$

We get final dispersion estimates from this model in three steps, which implement a computationally fast approximation to a full empirical Bayes treatment. We first use the count data for each gene separately to get preliminary gene-wise dispersion estimates α_i^{gw} by maximum likelihood estimation. Then, we fit the dispersion trend α_{tr}. Finally, we combine the likelihood with the trended prior to get maximum *a posteriori* (MAP) values as final dispersion estimates. Details for the three steps follow.

Gene-wise dispersion estimates. To get a gene-wise dispersion estimate for a gene i, we start by fitting a Negative Binomial GLM without logarithmic fold change prior for the design matrix X to the gene's count data. This GLM uses a rough method of moments estimate of dispersion, based on the within-group variances and means. The initial GLM is necessary to obtain an initial set of fitted values, $\hat{\mu}_{ij}^0$. We then maximize the Cox-Reid adjusted likelihood of the dispersion, conditioned on the fitted values $\hat{\mu}_{ij}^0$ from the initial fit, to obtain the gene-wise estimate α_i^{gw}, i.e.,

$$\alpha_i^{\mathrm{gw}} = \arg\max_{\alpha} \ell_{\mathrm{CR}}(\alpha; \vec{\mu}_{i\cdot}^0, \vec{K}_{i\cdot})$$

with

$$\ell_{\mathrm{CR}}(\alpha; \vec{\mu}, \vec{K}) = \ell(\alpha) - \frac{1}{2}\log(\det(X^t W X)) \tag{7}$$

$$\ell(\alpha) = \sum_j \log f_{\mathrm{NB}}(K_j; \mu_j, \alpha),$$

where $f_{\mathrm{NB}}(k; \mu, \alpha)$ is the probability mass function of the Negative Binomial distribution with mean μ and dispersion α, and the second term provides the Cox-Reid bias adjustment [37]. This adjustment, first used in the context of dispersion estimation for SAGE data [38] and then for HTS data [3] in *edgeR*, corrects for the negative bias of dispersion estimates from using the maximum likelihood estimates (MLE) for the fitted values $\hat{\mu}_{ij}^0$ (analogous to Bessel's correction in the usual sample variance formula; for details, see [39, Section 10.6]). It is formed from the Fisher information for the fitted values, which is here calculated as $\det(X^t W X)$,

where W is the diagonal weight matrix from the standard iteratively re-weighted least squares (IRLS) algorithm. As the GLM's link function is $g(\mu) = \log(\mu)$ and its variance function is $V(\mu; \alpha) = \mu + \alpha\mu^2$, the elements of the diagonal matrix W_i are given by:

$$w_{jj} = \frac{1}{g'(\mu_j)^2 V(\mu_j)} = \frac{1}{1/\mu_j + \alpha}.$$

The optimization in Equation (7) is performed on the scale of $\log \alpha$ using a backtracking line search with proposals accepted which satisfy Armijo conditions [40].

Dispersion trend. A parametric curve of the form (6) is fit by regressing the gene-wise dispersion estimates α_i^{gw} onto the means of the normalized counts, $\bar{\mu}_i$.

Under model (5), the distribution of the residuals of such a fit is a hierarchical mixture of the log-Normal distribution postulated as prior and the sampling distribution of the dispersion estimator. For the latter, one may reasonably assume a roughly χ^2-like distribution. Assuming that the sampling variance dominates, at least in the few-sample case, it seems prudent not to use ordinary least square regression but rather gamma-family GLM regression. Furthermore, dispersion outliers might skew the fit and hence, a scheme to exclude such outliers is warranted.

The hyperparameters a_1 and α_0 of (6) are obtained by iteratively fitting a gamma-family GLM. At each iteration, genes with ratio of dispersion to fitted value outside the range $[10^{-4}, 15]$ are left out until the sum of squared logarithmic fold changes of the new coefficients over the old coefficients is less than 10^{-6} (same approach as in *DEXSeq* [27]).

Dispersion prior. As also observed by Wu et al. [5], a log-Normal prior fits the observed dispersion distribution better for typical RNA-Seq than a conjugate prior would. We solve the computational difficulty of working with a non-conjugate prior using the following argument: the logarithmic residuals from the trend fit, $\log \alpha_i^{\mathrm{gw}} - \log \alpha_{\mathrm{tr}}(\bar{\mu}_i)$, arise from two contributions, namely the scatter of the true logarithmic dispersions around the trend, given by the prior with variance σ_{d}^2, and the sampling distribution of the logarithm of the dispersion estimator, with variance σ_{lde}^2. Due to its similarity to a variance estimator, it is reasonable to expect the sampling distribution of a dispersion estimator to be approximately a scaled χ^2 distribution with $m - p$ degrees of freedom, with m the number of samples and p the number of coefficients. The variance of the logarithm of a χ^2-distributed random variable is given [41] by the trigamma function ψ_1,

$$\mathrm{Var} \log X^2 = \psi_1(f/2) \quad \text{for} \quad X^2 \sim \chi_f^2.$$

Therefore, $\sigma_{\mathrm{lde}}^2 \approx \psi_1((m-p)/2)$, i.e., the sampling variance of the logarithm of a variance or dispersion estimator is approximately constant across genes and depends only on the degrees of freedom of the model.

Supplementary Table S2 compares this approximation for the variance of logarithmic dispersion estimates with the variance of logarithmic Cox-Reid adjusted dispersion estimates for simulated Negative Binomial data, over a combination of different sample sizes, number of parameters and dispersion values used to create the simulated data. The approximation is close to the sample variance for various typical values of m, p and α.

Therefore, the prior variance σ_{d}^2 is obtained by subtracting the expected sampling variance from an estimate of the variance of the logarithmic residuals, s_{lr}^2:

$$\sigma_{\mathrm{d}}^2 = \min\{s_{\mathrm{lr}}^2 - \psi_1((m-p)/2),\ 0.25\}.$$

The prior variance σ_{d}^2 is thresholded at a minimal value of 0.25 so that the dispersion estimates are not shrunk entirely to $\alpha_{\mathrm{tr}}(\bar{\mu}_i)$ in the case that the variance of the logarithmic residuals is less than the expected sampling variance.

In order to avoid an inflation of σ_d^2 due to dispersion outliers (i.e., genes not well captured by this prior; see below), we use a robust estimator for the standard deviation s_lr of the logarithmic residuals,

$$s_\mathrm{lr} = \underset{i}{\mathrm{mad}} \left(\log \alpha_i^\mathrm{gw} - \log \alpha_\mathrm{tr}(\bar{\mu}_i) \right), \tag{8}$$

where mad stands for the median absolute deviation, multiplied as usual by the scaling factor $1/\Phi^{-1}(3/4)$.

Three or less residuals degrees of freedom. When there are 3 or less residual degrees of freedom (number of samples minus number of parameters to estimate), the estimation of the prior variance σ_d^2 using the observed variance of logarithmic residuals s_lr^2 tends to underestimate σ_d^2. In this case, we instead estimate the prior variance through simulation. We match the distribution of logarithmic residuals to a density of simulated logarithmic residuals. These are the logarithm of χ^2_{m-p}-distributed random variables added to $N(0, \sigma_\mathrm{d}^2)$ random variables to account for the spread due to the prior. The simulated distribution is shifted by $-\log(m-p)$ to account for the scaling of the χ^2 distribution. We repeat the simulation over a grid of values for σ_d^2, and select the value which minimizes the Kullback-Leibler divergence from the observed density of logarithmic residuals to the simulated density.

Final dispersion estimate. We form a logarithmic posterior for the dispersion from the Cox-Reid adjusted logarithmic likelihood (7) and the logarithmic prior (5) and use its maximum (i.e., the maximum a posteriori, MAP, value) as final estimate of the dispersion:

$$\begin{aligned} \alpha_i^\mathrm{MAP} &= \arg\max_\alpha \left(\ell_\mathrm{CR}(\alpha; \vec{\mu}_{i\cdot}^0, \vec{K}_{i\cdot}) + p(\alpha) \right) \\ p(\alpha) &= \frac{-\left(\log \alpha - \log \alpha_\mathrm{tr}(\bar{\mu}_i)\right)^2}{2\sigma_\mathrm{d}^2}. \end{aligned} \tag{9}$$

Again, a backtracking line search is used to perform the optimization.

Dispersion outliers. For some genes, the gene-wise estimate α_i^gw can be so far above the prior expectation $\alpha_\mathrm{tr}(\bar{\mu}_i)$ that it would be unreasonable to assume that the prior is suitable for the gene. If the dispersion estimate for such genes is down-moderated toward the fitted trend, this might lead to false positives. Therefore, we use the heuristic of considering a gene as a "dispersion outlier", if the residual from the trend fit is more than two standard deviations of logarithmic residuals, s_lr (see Equation (8)), above the fit:

$$\log \alpha_i^\mathrm{gw} > \log \alpha_\mathrm{tr}(\bar{\mu}_i) + 2 s_\mathrm{lr}.$$

Such genes are flagged and the gene-wise estimate α_i^gw is not shrunk toward the trended prior mean. Instead of the MAP value α_i^MAP, we use the gene-wise estimate α_i^gw as a final dispersion value in the subsequent steps. In addition, the iterative fitting procedure for the parametric dispersion trend described above avoids that these dispersion outliers influence the prior mean.

Shrinkage estimation of logarithmic fold changes

To incorporate empirical Bayes shrinkage of logarithmic fold changes, we postulate a zero-centered Normal prior for the coefficients β_{ir} of model (2) that represent logarithmic fold changes (i.e., typically, all coefficients except for the intercept β_{i0}):

$$\beta_{ir} \sim N(0, \sigma_r^2). \tag{10}$$

As was observed with differential expression analysis using microarrays, genes with low intensity values tend to suffer from a small signal to noise ratio. Alternative estimators can be found which are more stable than the standard calculation of fold change as the ratio of average observed values for each condition [42, 43, 44]. *DESeq2*'s approach can be seen as an extension of these approaches for stable estimation of gene expression fold changes to count data.

Empirical prior estimate. To obtain values for the empirical prior widths σ_r for the model coefficients, we again approximate a full empirical Bayes approach, as with the estimation of dispersion prior, though here we do not subtract the expected sampling variance from the observed variance of maximum likelihood estimates. The estimation of the logarithmic fold change prior width is calculated as follows. We use the standard iteratively reweighted least squares (IRLS) algorithm [10] for each gene's model (1,2) to get maximum likelihood estimates for the coefficients $\beta_{ir}^{\mathrm{MLE}}$. We then fit, for each column r of the design matrix (except for the intercept) a zero-centered Normal to the empirical distribution of MLE fold change estimates $\{\beta_{ir}^{\mathrm{MLE}}\}_r$.

To make the fit robust against outliers with very high absolute LFC values, we use quantile matching: the width σ_r is chosen such that the $(1-p)$ empirical quantile of the absolute value of the observed LFCs $\{|\beta_{ir}^{\mathrm{MLE}}|\}_r$ matches the $(1-p/2)$ theoretical quantile of the prior, $N(0,\sigma_r^2)$, where p is set by default to 0.05. If we write the theoretical upper quantile of a Normal distribution as $Q_N(1-p)$ and the empirical upper quantile of the MLE LFCs as $Q_{|\beta_r|}(1-p)$, then the prior width is calculated as:

$$\sigma_r = \frac{Q_{|\beta_r|}(1-p)}{Q_{\mathcal{N}}(1-p/2)}.$$

To ensure that the prior width σ_r will be independent of the choice of base level, the estimates from the quantile matching procedure are averaged for each factor over all possible contrasts of factor levels. When determining the empirical upper quantile, extreme LFC values with $|\beta_{ir}^{\mathrm{MLE}}| > \log(1024)$ are excluded.

Final estimate of log fold changes. The logarithmic posterior for the vector, $\vec{\beta}_i$, of model coefficients β_{ir} for gene i is the sum of the logarithmic likelihood of the GLM (2) and the logarithm of the prior density (10), and its maximum yields the final MAP coefficient estimates:

$$\vec{\beta}_i = \arg\max_{\vec{\beta}} \left(\sum_j \log f_{\mathrm{NB}}\left(K_{ij}; \mu_j(\vec{\beta}), \alpha_i\right) + p(\vec{\beta}) \right),$$

where

$$\mu_j(\vec{\beta}) = s_{ij} e^{\sum_r x_{jr}\beta_r}, \quad p(\vec{\beta}) = \sum_r \frac{-\beta_r^2}{2\sigma_r^2},$$

and α_i is the final dispersion estimate for gene i, i.e., $\alpha_i = \alpha_i^{\mathrm{MAP}}$, except for dispersion outliers, where $\alpha_i = \alpha_i^{\mathrm{gw}}$.

The term $p(\vec{\beta})$, i.e., the logarithm of the Normal prior, can be read as a ridge penalty term, and therefore, we perform the optimization using the *iteratively reweighted ridge regression algorithm* [45], also known as *weighted updates* [46]. Specifically, the updates for a given gene are of the form:

$$\vec{\beta} \leftarrow (X^t W X + \vec{\lambda} I)^{-1} X^t W \vec{z},$$

with $\lambda_r = 1/\sigma_r^2$ and

$$z_j = \log\frac{\mu_j}{s_j} + \frac{K_j - \mu_j}{\mu_j},$$

where the current fitted values $\mu_j = s_j e^{\sum_r x_{jr}\beta_r}$ are formed from the current estimates $\vec{\beta}$ in each iteration.

Fisher information. The effect of the zero-centered Normal prior can be understood as to shrink the MAP LFC estimates based on the amount of information the experiment provides for this coefficient, and we briefly elaborate on this here. Specifically, for a given gene i, the shrinkage for an LFC β_{ir} depends on the *observed Fisher information*, given by

$$\mathcal{I}_m(\hat{\beta}_{ir}) = -\left[\frac{\partial^2}{\partial \beta_{ir}^2} \ell(\vec{\beta}_i; \vec{K}_i, \alpha_i)\right]_{\beta_{ir}=\hat{\beta}_{ir}},$$

where $\ell(\vec{\beta}_i; \vec{K}_i, \alpha_i)$ is the logarithm of the likelihood, and partial derivatives are taken with respect to LFC β_{ir}. For a Negative Binomial GLM, the observed Fisher information, or peakedness of the logarithm of the profile likelihood, is influenced by a number of factors including the degrees of freedom, the estimated mean counts μ_{ij}, and the gene's dispersion estimate α_i. The prior exerts its influence on the MAP estimate when the density of the likelihood and the prior are multiplied to calculate the posterior. Genes with low estimated mean values μ_{ij} or high dispersion estimates α_i have flatter profile likelihoods, as do datasets with few residual degrees of freedom, and therefore in these cases the zero-centered prior pulls the MAP estimate from a high-uncertainty MLE closer toward zero.

Wald test

The Wald test compares the beta estimate β_{ir} divided by its estimated standard error $\mathrm{SE}(\beta_{ir})$ to a standard Normal distribution. The estimated standard errors are taken from the diagonal of the estimated covariance matrix, Σ_i, for the coefficients, i.e., $\mathrm{SE}(\beta_{ir}) = \Sigma_{i,rr}$. Contrasts of coefficients are tested similarly by forming a Wald statistics using (3) and (4). We use the following formula for the coefficient covariance matrix for a generalized linear model with Normal prior on coefficients [45, 47]:

$$\Sigma_i = \mathrm{Cov}(\vec{\beta}_i) = (X^t W X + \vec{\lambda} I)^{-1} (X^t W X)(X^t W X + \vec{\lambda} I)^{-1}.$$

The tail integrals of the standard Normal distribution are multiplied by 2 to achieve a two-tailed test. The Wald test p-values from the subset of genes which pass the independent filtering step are adjusted for multiple testing using the procedure of Benjamini and Hochberg [17].

Composite null hypotheses

DESeq2 also offers to test composite null hypotheses of the form $\mathcal{H}_0 : |\beta_{ir}| \leq \theta$ in order to find genes whose LFC significantly exceeds a threshold $\theta > 0$. The composite null hypothesis is replaced by two simple null hypotheses: $\mathcal{H}_{0a} : \beta_{ir} = \theta$ and $\mathcal{H}_{0b} : \beta_{ir} = -\theta$. Two-tailed p-values are generated by integrating a Normal distribution centered on θ with standard deviation $\mathrm{SE}(\beta_{ir})$ from $|\beta_{ir}|$ toward ∞. The value of the integral is then multiplied by 2 and thresholded at 1. This procedure controls type-I error even when $\beta_{ir} = \pm\theta$, and is equivalent to the standard *DESeq2* p-value when $\theta = 0$.

Conversely, when searching for genes whose absolute LFC is significantly below a threshold, i.e., when testing the null hypothesis $\mathcal{H}_0 : |\beta_{ir}| \geq \theta$, the p-value is constructed as the maximum of two one-sided tests of the simple null hypotheses: $\mathcal{H}_{0a} : \beta_{ir} = \theta$ and $\mathcal{H}_{0b} : \beta_{ir} = -\theta$. The one-sided p-values are generated by integrating a Normal distribution centered on θ with standard deviation $\mathrm{SE}(\beta_{ir})$ from β_{ir} toward $-\infty$, and integrating a Normal distribution centered on $-\theta$ with standard deviation $\mathrm{SE}(\beta_{ir})$ from β_{ir} toward ∞.

Note that while a zero-centered prior on LFCs is consistent with testing the null hypothesis of small LFCs, it should not be used when testing the null hypothesis of large LFCs, because the prior would then favor the alternative hypothesis. *DESeq2* requires that no prior has been used when testing the null hypothesis of large LFCs, so that the data alone must provide evidence against the null.

Interactions

Two exceptions to the default *DESeq2* LFC estimation steps are used in the case of experimental designs with interaction terms. First, when any interaction terms are included in the design, the LFC prior width for main effect terms is not estimated from the data, but set to a wide value ($\sigma_r^2 = 1000$). This ensures that shrinkage of main effect terms will not result in false positive calls of significance for interactions. Second, when interaction terms are included and all factors have two levels, then standard design matrices are used rather than expanded model matrices, such that only a single term is used to test the null hypothesis that a combination of two effects is merely additive in the logarithmic scale.

Regularized logarithm

The rlog transformation is calculated as follows. The experimental design matrix X is substituted with a design matrix with an indicator variable for every sample in addition to an intercept column. A model as described in Equations (1,2) is fit with a zero-centered Normal prior on the non-intercept terms and using the fitted dispersion values $\alpha_{\mathrm{tr}}(\bar{\mu})$, which capture the overall variance-mean dependence of the dataset. The true experimental design matrix X is then only used in estimating the variance-mean trend over all genes. For the purpose of unsupervised analyses, for instance sample quality assessment, it can be desirable that the experimental design have no influence on the transformation, and hence *DESeq2* by default completely ignores the design matrix and re-estimates the dispersions treating all samples as replicates, i.e., *blind* dispersion estimation. The rlog transformed values are the fitted values,

$$\mathrm{rlog}(K_{ij}) \equiv \log_2 q_{ij} = \beta_{i0} + \beta_{ij},$$

where β_{ij} is the shrunken LFC for the j-th sample. The variance of the prior is set using a similar approach as taken with differential expression, by matching a zero-centered Normal distribution to observed LFCs. First a matrix of logarithmic fold changes is calculated by taking the logarithm (base 2) of the normalized counts plus a pseudocount of $\frac{1}{2}$ for each sample divided by the mean of normalized counts plus a pseudocount of $\frac{1}{2}$. The pseudocount ensures that all genes will have finite log ratio and therefore contribute to the calculation of the prior. This matrix of LFCs then represents the common-scale logarithmic ratio of each sample to the fitted value using only an intercept. The prior variance is found by matching the 95% quantile of a zero-centered Normal distribution to the 90% quantile of the values in the logarithmic fold change matrix.

Cook's distance for outlier detection

The maximum likelihood estimate of $\vec{\beta}_i$ is used for calculating Cook's distance. Considering a gene i and sample j, Cook's distance for generalized linear models is given by [48]:

$$D_{ij} = \frac{R_{ij}^2}{\tau p} \frac{h_{jj}}{(1-h_{jj})^2},$$

where R_{ij} is the Pearson residual of sample j, τ is an overdispersion parameter (in the Negative Binomial GLM, τ is set to 1), p is the number of parameters including the intercept, and h_{jj} is the j-th diagonal element of the hat matrix H:

$$H = W^{1/2} X (X^t W X)^{-1} X^t W^{1/2}.$$

Pearson residuals R_{ij} are calculated as

$$R_{ij} = \frac{(K_{ij} - \mu_{ij})}{\sqrt{V(\mu_{ij})}},$$

where μ_{ij} is estimated by the Negative Binomial GLM without the logarithmic fold change prior, and using the variance function $V(\mu) = \mu + \alpha\mu^2$. A method of

moments estimate α_i^{rob} which provides robustness against outliers is used here, estimating the variance using the median absolute deviation:

$$\alpha_i^{\text{rob}} = \max\left(\frac{s_{i,\text{rob}}^2 - \bar{\mu}_i}{\bar{\mu}_i^2}, 0\right),$$

with

$$s_{i,\text{rob}}^2 = \left(\underset{j}{\text{mad}}(K_{ij}/s_{ij})\right)^2$$

where, again, the mad operator includes the usual scaling factor of $1/\Phi^{-1}(3/4)$.

R/Bioconductor package

DESeq2 is implemented as a package for the R statistical environment as available as part of the Bioconductor project [9] at `http://www.bioconductor.org`. The count matrix and metadata including the gene model and sample information are stored in an S4 class derived from the SummarizedExperiment class of the *GenomicRanges* package [49]. SummarizedExperiment objects containing count matrices can be easily generated using the *summarizeOverlaps* function of the *GenomicAlignments* package [50]. This workflow automatically stores the gene model as metadata and additionally other information such as the genome and gene set versions. Other methods to obtain count matrices include the *htseq-count* script [51] and the Bioconductor packages *easyRNASeq* [52] and *featureCount* [53].

The *DESeq2* package comes with a detailed vignette working through a number of example differential expression analyses on real datasets, and the use of the rlog transformation for quality assessment and visualization. A single function, `DESeq`, is used to run the default analysis, while lower-level functions are also available for advanced users.

Reproducible code

Sweave vignettes for reproducing all figures and tables in this paper, including data objects for the experiments mentioned, and code for aligning reads and for benchmarking, can be found in a package *DESeq2paper*, available at `http://www-huber.embl.de/DESeq2paper/`.

List of abbreviations

FDR	false discovery rate
GLM	generalized linear model
HTS	high-throughput sequencing
LFC	logarithmic fold change
MAP	maximum *a posteriori*
MLE	maximum likelihood estimate
SE	standard error
VST	variance-stabilizing transformation

Competing interests

The authors declare that they have no competing interests.

Acknowledgements

The authors thank all users of DESeq and DESeq2 who provided valuable feedback. We thank Judith Zaugg for helpful comments on the manuscript. ML acknowledges funding via a stipend from the International Max Planck Research School for Computational Biology and Scientific Computing and a grant from

the National Institutes of Health. WH and SA acknowledge funding from the European Union's 7th Framework Programme (Health) via Project *Radiant*.

References

[1] Lönnstedt, I., Speed, T.: Replicated microarray data. Stat Sinica, 31–46 (2002)

[2] Robinson, M.D., Smyth, G.K.: Moderated statistical tests for assessing differences in tag abundance. Bioinformatics **23**(21), 2881–2887 (2007)

[3] McCarthy, D.J., Chen, Y., Smyth, G.K.: Differential expression analysis of multifactor RNA-seq experiments with respect to biological variation. Nucleic Acids Res **40**(10), 4288–4297 (2012)

[4] Anders, S., Huber, W.: Differential expression analysis for sequence count data. Genome Biol **11**(10), 106 (2010)

[5] Wu, H., Wang, C., Wu, Z.: A new shrinkage estimator for dispersion improves differential expression detection in RNA-seq data. Biostatistics **14**(2), 232–243 (2013)

[6] Hardcastle, T., Kelly, K.: baySeq: Empirical bayesian methods for identifying differential expression in sequence count data. BMC Bioinformatics **11**(1), 422 (2010)

[7] Van De Wiel, M.A., Leday, G.G.R., Pardo, L., Rue, H., Van Der Vaart, A.W., Van Wieringen, W.N.: Bayesian analysis of RNA sequencing data by estimating multiple shrinkage priors. Biostatistics **14**(1), 113–128 (2013)

[8] Boer, J.M., Huber, W.K., Sültmann, H., Wilmer, F., von Heydebreck, A., Haas, S., Korn, B., Gunawan, B., Vente, A., Füzesi, L., Vingron, M., Poustka, A.: Identification and classification of differentially expressed genes in renal cell carcinoma by expression profiling on a global human 31,500-element cDNA array. Genome Res **11**(11), 1861–1870 (2001)

[9] Gentleman, R.C., Carey, V.J., Bates, D.M., Bolstad, B., Dettling, M., Dudoit, S., Ellis, B., Gautier, L., Ge, Y., Gentry, J., Hornik, K., Hothorn, T., Huber, W., Iacus, S., Irizarry, R., Leisch, F., Li, C., Maechler, M., Rossini, A.J., Sawitzki, G., Smith, C., Smyth, G., Tierney, L., Yang, J.Y., Zhang, J.: Bioconductor: open software development for computational biology and bioinformatics. Genome Biol **5**(10), 80–16 (2004)

[10] McCullagh, P., Nelder, J.A.: Generalized Linear Models, Second edition edn. Monographs on Statistics & Applied Probability. Chapman & Hall/CRC, London, United Kingdom (1989)

[11] Hansen, K.D., Irizarry, R.A., Wu, Z.: Removing technical variability in RNA-seq data using conditional quantile normalization. Biostatistics **13**(2), 204–216 (2012)

[12] Risso, D., Schwartz, K., Sherlock, G., Dudoit, S.: GC-content normalization for RNA-seq data. BMC Bioinformatics **12**(1), 480 (2011)

[13] Bottomly, D., Walter, N.A.R., Hunter, J.E., Darakjian, P., Kawane, S., Buck, K.J., Searles, R.P., Mooney, M., McWeeney, S.K., Hitzemann, R.: Evaluating gene expression in C57BL/6J and DBA/2J mouse striatum using RNA-seq and microarrays. PLoS ONE **6**(3), 17820 (2011)

[14] Smyth, G.K.: Linear models and empirical bayes methods for assessing differential expression in microarray experiments. Stat Appl Genet Mol Biol **3**(1) (2004)

[15] Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd edn. Springer, New York City, USA (2009)

[16] Feng, J., Meyer, C.A., Wang, Q., Liu, J.S., Liu, X.S., Zhang, Y.: GFOLD: a generalized fold change for ranking differentially expressed genes from RNA-seq data. Bioinformatics **28**(21), 2782–2788 (2012)

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

[18] Bourgon, R., Gentleman, R., Huber, W.: Independent filtering increases detection power for high-throughput experiments. Proc Natl Acad Sci U S A **107**(21), 9546–9551 (2010)

[19] McCarthy, D.J., Smyth, G.K.: Testing significance relative to a fold-change threshold is a TREAT. Bioinformatics **25**(6), 765–771 (2009)

[20] Bi, Y., Davuluri, R.: NPEBseq: nonparametric empirical bayesian-based procedure for differential expression analysis of RNA-seq data. BMC Bioinformatics **14**(1), 262 (2013)

[21] Li, J., Tibshirani, R.: Finding consistent patterns: A nonparametric approach for identifying differential expression in RNA-seq data. Stat Methods Med Res **22**(5), 519–536 (2011)

[22] Cook, R.D.: Detection of influential observation in linear regression. Technometrics **19**(1), 15–18 (1977)

[23] Hammer, P., Banck, M.S., Amberg, R., Wang, C., Petznick, G., Luo, S., Khrebtukova, I., Schroth, G.P., Beyerlein, P., Beutler, A.S.: mRNA-seq with agnostic splice site discovery for nervous system transcriptomics tested in chronic pain. Genome Res **20**(6), 847–860 (2010)

[24] Frazee, A., Langmead, B., Leek, J.: ReCount: A multi-experiment resource of analysis-ready RNA-seq gene count datasets. BMC Bioinformatics **12**(1), 449 (2011)

[25] Trapnell, C., Hendrickson, D.G., Sauvageau, M., Goff, L., Rinn, J.L., Pachter, L.: Differential analysis of gene regulation at transcript resolution with RNA-seq. Nat Biotechnol **31**(1), 46–53 (2012)

[26] Glaus, P., Honkela, A., Rattray, M.: Identifying differentially expressed transcripts from RNA-seq data with biological variation. Bioinformatics **28**(13), 1721–1728 (2012)

[27] Anders, S., Reyes, A., Huber, W.: Detecting differential usage of exons from RNA-seq data. Genome Res **22**(10), 2008–2017 (2012)

[28] Sammeth, M.: Complete alternative splicing events are bubbles in splicing graphs. J Comput Biol **16**(8), 1117–1140 (2009)

[29] Irizarry, R.A., Wu, Z., Jaffee, H.A.: Comparison of affymetrix GeneChip expression measures. Bioinformatics **22**(7), 789–794 (2006)

[30] Robinson, M.D., McCarthy, D.J., Smyth, G.K.: edgeR: a bioconductor package for differential expression analysis of digital gene expression data. Bioinformatics **26**(1), 139–140 (2009)

[31] Law, C.W., Chen, Y., Shi, W., Smyth, G.K.: Voom: precision weights unlock linear model analysis tools for RNA-seq read counts. Genome Biol **15**(2), 29 (2014)

[32] Pickrell, J.K., Marioni, J.C., Pai, A.A., Degner, J.F., Engelhardt, B.E., Nkadori, E., Veyrieras, J.-B., Stephens, M., Gilad, Y., Pritchard, J.K.: Understanding mechanisms underlying human gene expression variation with RNA sequencing. Nature **464**(7289), 768–772 (2010)

[33] Stark, R., Brown, G.: DiffBind: differential binding analysis of ChIP-Seq peak data. Bioconductor package, available from http://www.bioconductor.org (2013)

[34] Ross-Innes, C.S., Stark, R., Teschendorff, A.E., Holmes, K.A., Ali, H.R., Dunning, M.J., Brown, G.D., Gojis, O., Ellis, I.O., Green, A.R., Ali, S., Chin, S.-F., Palmieri, C., Caldas, C., Carroll, J.S.: Differential oestrogen receptor binding is associated with clinical outcome in breast cancer. Nature **481**(7381), 389–393 (2012)

[35] Robinson, D.G., Chen, W., Storey, J.D., Gresham, D.: Design and analysis of bar-seq experiments. G3 (Bethesda) **4**(1), 11–18 (2013)

[36] McMurdie, P.J., Holmes, S.: Waste Not, Want Not: Why Rarefying Microbiome Data is Inadmissible (2013). 1310.0424. http://arxiv.org/abs/1310.0424

[37] Cox, D.R., Reid, N.: Parameter orthogonality and approximate conditional inference. J R Stat Soc Ser B Methodol **49**(1), 1–39 (1987)

[38] Robinson, M.D., Smyth, G.K.: Small-sample estimation of negative binomial dispersion, with applications to SAGE data. Biostatistics **9**(2), 321–332 (2007)

[39] Pawitan, Y.: In All Likelihood: Statistical Modelling and Inference Using Likelihood, 1st edn. Oxford University Press, New York City, USA (2001)

[40] Armijo, L.: Minimization of functions having lipschitz continuous first partial derivatives. Pac J Math **16**(1), 1–3 (1966)

[41] Abramowitz, M., Stegun, I.: Handbook of Mathematical Functions, 1st edn. Dover books on mathematics. Dover Publications, USA (1965)

[42] Newton, M., Kendziorski, C., Richmond, C., Blattner, F., Tsui, K.: On differential variability of expression ratios: Improving statistical inference about gene expression changes from microarray data. J Comput Biol **8**(1), 37–52 (2001)

[43] Huber, W., von Heydebreck, A., Sültmann, H., Poustka, A., Vingron, M.: Variance stabilization applied to microarray data calibration and to the quantification of differential expression. Bioinformatics **18**(suppl 1), 96–104 (2002)

[44] Durbin, B.P., Hardin, J.S., Hawkins, D.M., Rocke, D.M.: A variance-stabilizing transformation for gene-expression microarray data. Bioinformatics **18**(suppl 1), 105–110 (2002)

[45] Park, M.Y.: Generalized linear models with regularization. PhD thesis, Stanford University, Department of Statistics Sequoia Hall 390 Serra Mall Stanford University Stanford, CA (2006)

[46] Friedman, J., Hastie, T., Tibshirani, R.: Regularization paths for generalized linear models via coordinate descent. J Stat Softw **33**(1), 1–22 (2010)

[47] Cule, E., Vineis, P., De Iorio, M.: Significance testing in ridge regression for genetic data. BMC Bioinformatics **12**(1), 372 (2011)

[48] Cook, R.D., Weisberg, S.: Residuals and Influence in Regression, 1st edn. Chapman and Hall/CRC, New York, USA (1982)

[49] Lawrence, M., Huber, W., Pagès, H., Aboyoun, P., Carlson, M., Gentleman, R., Morgan, M.T., Carey, V.J.: Software for computing and annotating genomic ranges. PLoS Comput Biol **9**(8), 1003118 (2013)

[50] Pagès, H., Obenchain, V., Morgan, M.: GenomicAlignments: Representation and manipulation of short genomic alignments. Bioconductor package, available from `http://www.bioconductor.org` (2013)

[51] Anders, S., Pyl, P.T., Huber, W.: HTSeq: Analysing high-throughput sequencing data with Python. Software, available from `http://www-huber.embl.de/HTSeq` (2011)

[52] Delhomme, N., Padioleau, I., Furlong, E.E., Steinmetz, L.M.: easyRNASeq: a bioconductor package for processing RNA-seq data. Bioinformatics **28**(19), 2532–2533 (2012)

[53] Liao, Y., Smyth, G.K., Shi, W.: featureCounts: an efficient general purpose program for assigning sequence reads to genomic features. Bioinformatics, 656 (2013)

[54] Kim, D., Pertea, G., Trapnell, C., Pimentel, H., Kelley, R., Salzberg, S.: TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions. Genome Biol **14**(4), 36 (2013)

[55] Irizarry, R.A., Warren, D., Spencer, F., Kim, I.F., Biswal, S., Frank, B.C., Gabrielson, E., Garcia, J.G.N., Geoghegan, J., Germino, G., Griffin, C., Hilmer, S.C., Hoffman, E., Jedlicka, A.E., Kawasaki, E., Martínez-Murillo, F., Morsberger, L., Lee, H., Petersen, D., Quackenbush, J., Scott, A., Wilson, M., Yang, Y., Ye, S.Q., Yu, W.: Multiple-laboratory comparison of microarray platforms. Nat Methods **2**(5), 345–350 (2005)

Supplement

Supplemental methods

Read alignment for the Bottomly et al. and Pickrell et al. datasets

Reads were aligned using the TopHat2 aligner [54], and assigned to genes using the *summarizeOverlaps* function of the *GenomicRanges* package [49]. The SRA fastq files of the Pickrell et al. [32] dataset were aligned to the Homo Sapiens reference sequence GRCh37 downloaded in March 2013 from Illumina iGenomes. Reads were counted in the genes defined by the Ensembl GTF file, release 70, contained in the Illumina iGenome. The SRA fastq files of the Bottomly et al. [13] dataset were aligned to the Mus Musculus reference sequence NCBIM37 downloaded in March 2013 from Illumina iGenomes. Reads were counted in the genes defined by the Ensembl GTF file, release 66, contained in the Illumina iGenome.

Benchmarking code

The code used to run the count-based algorithms is contained in the file `/inst/script/runScripts.R` in the *DESeq2paper* package (available at `http://www-huber.embl.de/DESeq2paper`). The code which ran the algorithms over the randomized replicates is contained in the files `/inst/script/pickrell/diffExpr.R` (the specificity analysis run on the Pickrell et al. [32] dataset) and and `/inst/script/bottomly/diffExpr.R` (for the sensitivity and precision analysis run on the Bottomly et al. [13] dataset). The *Cuffdiff 2* commands are contained in the `/inst/script/pickrell/` and `/inst/script/bottomly/` directories.

Supplemental Tables

$i \in \{1, \ldots, n\}$	gene index
$j \in \{1, \ldots, m\}$	sample index
$r \in \{0, \ldots, p-1\}$	covariate index, with intercept $r = 0$
K_{ij}	counts of reads for gene i, sample j
μ_{ij}	fitted mean
α_i	gene-specific dispersion
s_j	sample-specific size factor
s_{ij}	gene- and sample-specific size factor
q_{ij}	proportional to true concentration of fragments
x_{jr}	elements of the design matrix X
β_{ir}	the logarithmic fold change for gene i and covariate r
$\bar{\mu}_i$	mean of normalized counts of gene i
σ_d^2	prior variance for logarithmic dispersions
σ_{lde}^2	sampling variance of logarithmic dispersion estimator
s_{lr}^2	variance estimate for logarithmic residuals of dispersion
α_i^{gw}	gene-wise dispersion estimate
$\alpha_{\mathrm{tr}}(\bar{\mu}_i)$	trended dispersion fit
α_i^{MAP}	maximum *a posteriori* estimate of dispersion
σ_r^2	prior variance for logarithmic fold change r
Σ_i	covariance matrix for $\vec{\beta}_i$

Supplementary Table S1: Notation

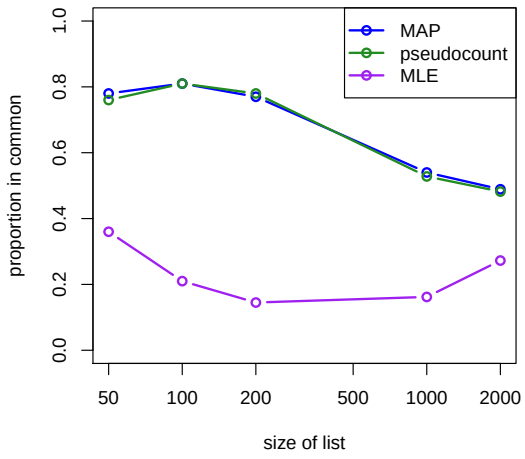
m	p	α	theor. var.	sample var.
6	2	0.05	0.645	0.677
6	2	0.20	0.645	0.644
8	2	0.05	0.395	0.411
8	2	0.20	0.395	0.396
8	3	0.05	0.490	0.532
8	3	0.20	0.490	0.462
16	2	0.05	0.154	0.160
16	2	0.20	0.154	0.138
16	3	0.05	0.166	0.169
16	3	0.20	0.166	0.155

Supplementary Table S2: Theoretical and sample variance of logarithmic dispersion estimates for various combinations of sample size m, number of parameters p and true dispersion α. The estimates are the *DESeq2* gene-wise estimates from 4000 simulated genes with Negative Binomial counts with a mean of 1024. The sample variance of the logarithmic dispersion estimates is generally close to the approximation of theoretical variance.

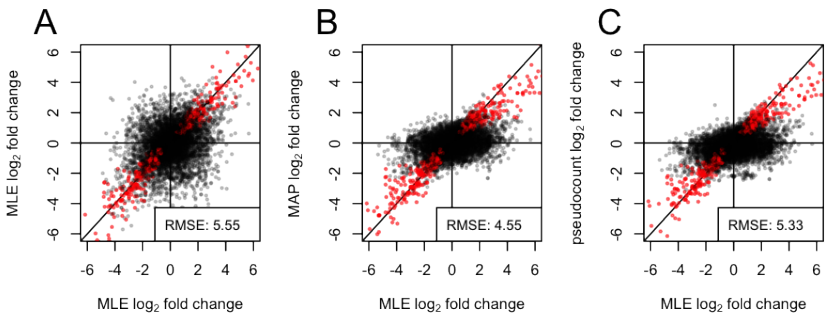
function/package	version	additional information
DESeq (old)	1.14.0	using the GLM test
DESeq2	1.3.40	
edgeR	3.4.2	using trended dispersion estimation
DSS	1.8.0	
voom: *limma*	3.18.12	
SAMseq: *samr*	2.0	
Cuffdiff 2	2.1.1	

Supplementary Table S3: Versions of software used in benchmark

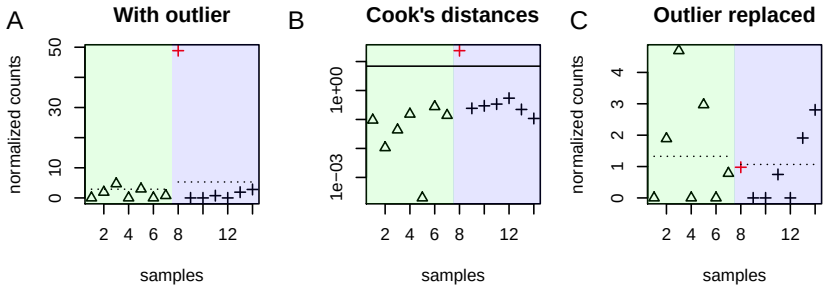
Supplemental Figures



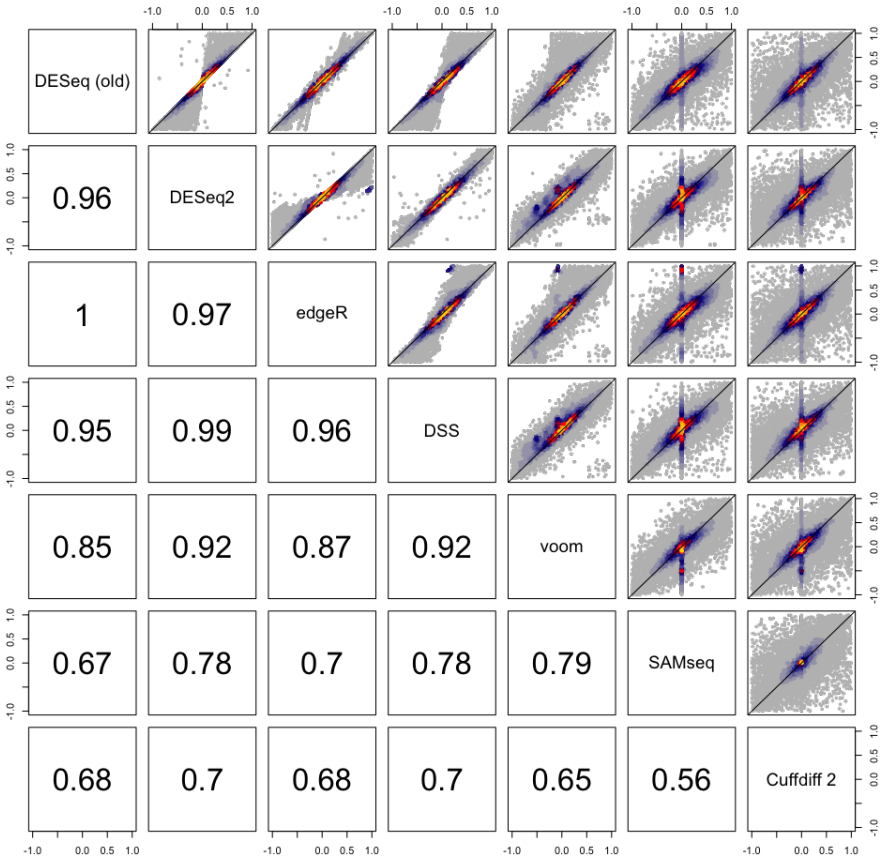
Supplementary Figure S1: **"Concordance at the top" plot.** *DESeq2* is run on equally split halves of the data of Bottomly et al. [13] and the proportion of genes in common after ranking by absolute logarithmic fold changes is compared [55]. On the y-axis is the number of genes in common between the splits divided by the size of the top-ranked list. The MAP estimate of logarithmic fold change and the MLE after adding a pseudocount of 1 to all samples provide nearly the same concordance for various cutoffs, while ranking by the MLE on raw counts has generally low concordance.



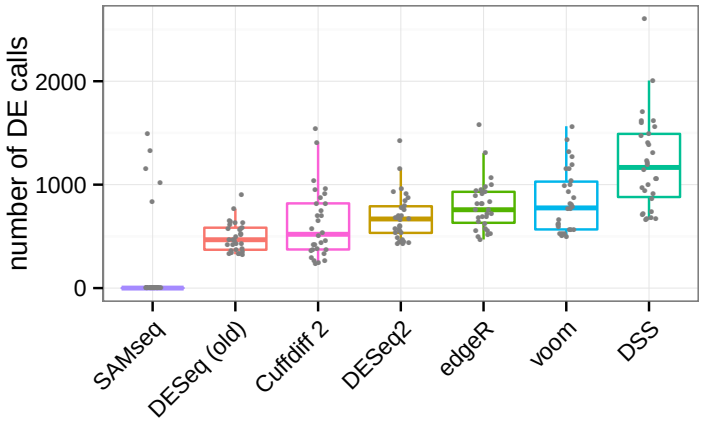
Supplementary Figure S2: **Stability of logarithmic fold changes.** *DESeq2* is run on equally split halves of the data of Bottomly et al. [13], and the logarithmic fold changes from the halves are plotted against each other. In these three panels, the x-axis presents the MLE LFCs from group I, while the y-axis presents LFCs for group II: (A) the MLE LFCs, (B) the MAP LFCs, and (C) the MLE LFCs after adding a pseudocount of 1 to every sample. Fixing the x-axis to the unbiased MLE LFCs of group I allows for a comparison of the stability of the MAP estimators against the stability of the pseudocount-based estimators. Though the MLE LFCs for group I have high variance, this should affect the group II estimators equally. Red points indicate genes with adjusted p-value less than 0.1. The legend displays the root mean squared error of the varying estimates in group II to the MLE LFCs from raw counts in group I, which is minimized for the MAP estimator (B).



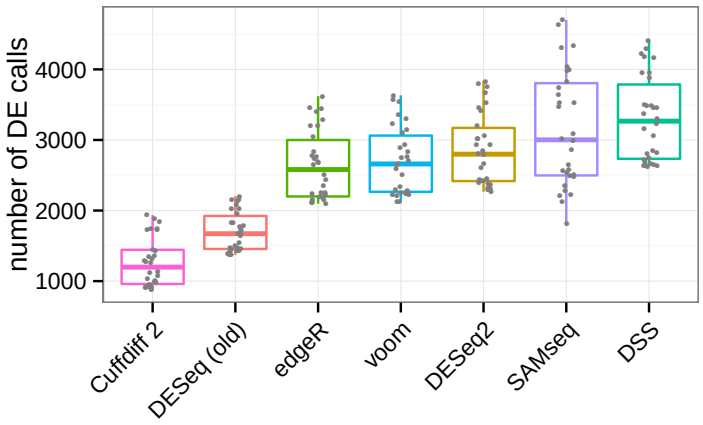
Supplementary Figure S3: **Cook's distance outlier detection**. Shown are normalized counts and Cook's distances for a 7 by 7 comparison of the Bottomly et al. [13] dataset. (A) Normalized counts for a single gene, samples divided into groups by species (light green and light blue). Dotted segments represent fitted means. A potential outlier is highlighted in red. (B) The Cook's distances for each sample for this gene, and the 99% quantile of the $F(p, m-p)$ cutoff used for flagging outliers. (C) The normalized counts after replacing the outlier with the trimmed mean over all samples, scaled by size factor. The fitted means now are less affected by the single outlier sample.



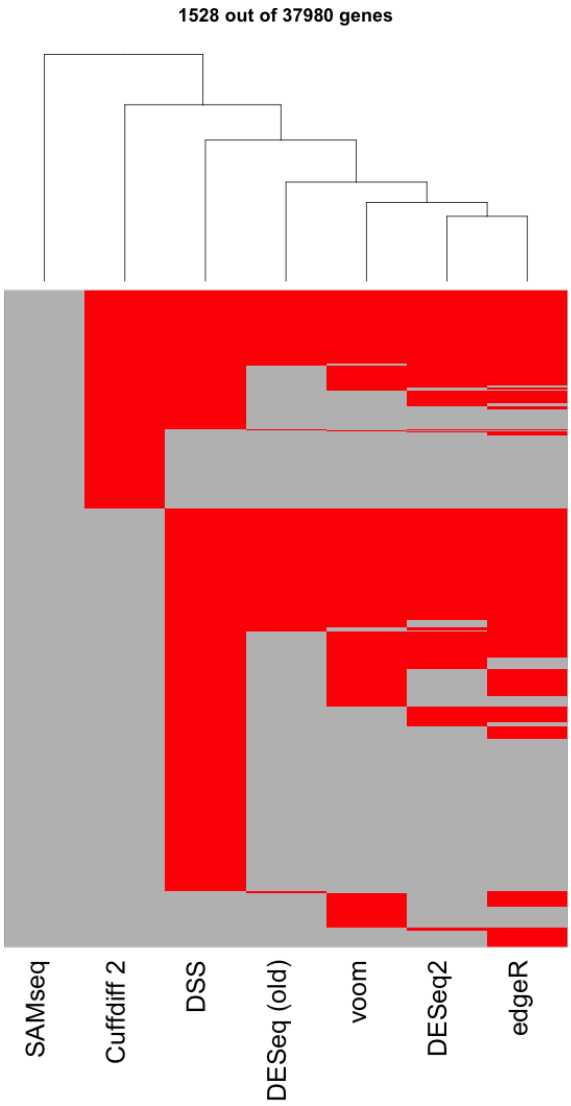
Supplementary Figure S4: **Smooth scatterplots of estimated logarithmic fold changes from all algorithms.** $\log_2$ fold changes are estimated from one of the verification sets of the Bottomly et al. [13] dataset. The blue, red, and yellow colors indicate regions of increasing density of points. Bottom panels display the Pearson correlation coefficients. We note that the direction of the estimate of differential expression for *DESeq2* and *Cuffdiff 2* accorded for the majority of genes called differentially expressed: among genes which were called differentially expressed by either of these two algorithms, both agreed on the sign of the estimated logarithmic fold change for 96% of genes (averaged over all 30 replicates) in the evaluation set and for 96% of genes in the verification set.



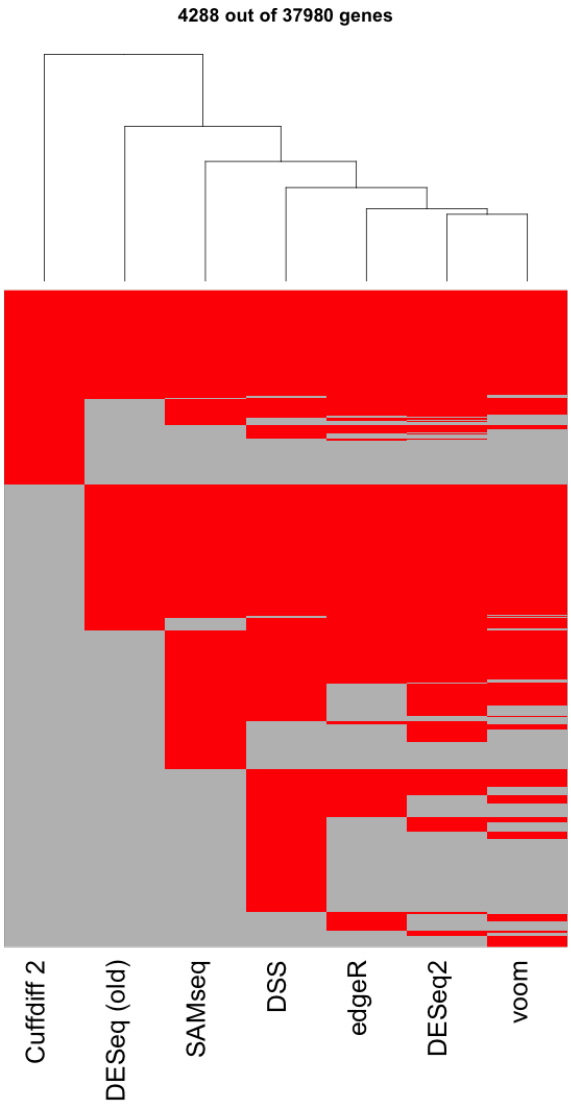
Supplementary Figure S5: **Number of total calls in the evaluation set (3 vs 3 samples)** of the sensitivity/precision analysis using the Bottomly et al. [13] dataset thresholding at adjusted p-value < 0.1, over 30 replications.



Supplementary Figure S6: **Number of total calls in the verification set (7 vs 8 samples)** of the sensitivity/precision analysis using the Bottomly et al. [13] dataset thresholding at adjusted p-value < 0.1, over 30 replications.



Supplementary Figure S7: **Clustering of each algorithm's calls on the evaluation set (3 vs 3 samples) for one replicate of the sensitivity/precision benchmark.** Genes are on the vertical axis and algorithms on the horizontal axis. Red lines indicate a gene had adjusted p-value < 0.1 in the evaluation set. Genes in which no algorithm had a call are not shown. Clustering is based on the Jaccard index. *DESeq2* calls are closest by a Jaccard-index-based distance to *edgeR* and *voom*.



Supplementary Figure S8: **Clustering of algorithm calls on the verification set (7 vs 8 samples) for one replicate of the sensitivity/precision benchmark.** Genes are on the vertical axis and algorithms on the horizontal axis. Red lines indicate a gene had adjusted p-value < 0.1 in the verification set. Genes in which no algorithm had a call are not shown. Clustering is based on the Jaccard index. *DESeq2* calls are closest by a Jaccard-index-based distance to *voom* and *edgeR*.