Gene expression

DiPhiSeq: robust comparison of expression levels on RNA-Seq data with large sample sizes

Jun Li^{1,*} and Alicia T. Lamere²

¹Department of Applied and Computational Mathematics and Statistics, University of Notre Dame, Notre Dame, IN 46556, USA and ²Mathematics Department, Bryant University, Smithfield, RI 02917, USA

*To whom correspondence should be addressed.

Associate Editor: Inanc Birol

Received on June 15, 2018; revised on October 22, 2018; editorial decision on November 14, 2018; accepted on November 16, 2018

Abstract

Motivation: In the analysis of RNA-Seq data, detecting differentially expressed (DE) genes has been a hot research area in recent years and many methods have been proposed. DE genes show different average expression levels in different sample groups, and thus can be important biological markers. While generally very successful, these methods need to be further tailored and improved for cancerous data, which often features quite diverse expression in the samples from the cancer group, and this diversity is much larger than that in the control group.

Results: We propose a statistical method that can detect not only genes that show different average expressions, but also genes that show different diversities of expressions in different groups. These ‘differentially dispersed’ genes can be important clinical markers. Our method uses a redescending penalty on the quasi-likelihood function, and thus has superior robustness against outliers and other noise. Simulations and real data analysis demonstrate that DiPhiSeq outperforms existing methods in the presence of outliers, and identifies unique sets of genes.

Availability and implementation: DiPhiSeq is publicly available as an R package on CRAN: <https://cran.r-project.org/package=DiPhiSeq>.

Contact: jun.li@nd.edu

Supplementary information: [Supplementary data](#) are available at *Bioinformatics* online.

1 Introduction

Describing and quantifying the transcriptome is important in understanding biological processes or disease mechanisms. Because of its ability in detecting new transcriptome elements and better accuracy in measuring transcript abundance, ultra-high-throughput sequencing of the transcriptome (RNA-Seq) is replacing microarrays as the *de facto* standard in describing and measuring the transcriptome. Different from microarrays, which measure gene expression with a real number that is often modeled by a Gaussian distributed variable, RNA-Seq measures gene expression with the number of short sequence readouts (called ‘reads’) mapped to the gene, which is a nonnegative integer that is better modeled by discrete distributions, like the negative binomial. The non-Gaussian nature of the RNA-Seq data brings many interesting and challenging statistical problems in inference on the data, making this area very attractive.

A very important problem in transcriptome analysis is to identify genes and other transcript elements such as isoforms (we call them ‘genes’ from now on for simplicity) that express differently in different groups of samples, say in cancerous samples versus normal samples, as these genes can be related to the phenotype differences. So far, these ‘differently expressed genes’ have been recognized as ‘differentially expressed’ (DE) genes, that is, genes whose *average* expression levels are different in different groups. Often, the count data from RNA-Seq is modeled by a negative binomial distribution that has two parameters, mean μ and dispersion ϕ , and DE genes are genes that have different means in different groups. Testing the mean difference is not a trivial problem when the sample size is small (say less than 10 within each group), as in this case the dispersion parameter is hard to estimate accurately, but the statistical significance of mean difference highly depends on the value of the

dispersion. Many sophisticated methods (e.g. Anders and Huber, 2010; Hardcastle and Kelly, 2010; Love et al., 2014; Robinson et al., 2010; Van De Wiel et al., 2012; Wu et al., 2012) have been proposed to borrow information from other genes to estimate the dispersion parameter of each gene. These methods have proven to increase the power of DE detection in the small sample size cases.

Without doubt, the difference between means is perhaps the most important difference between distributions, and hence DE genes are a natural choice of the ‘differently expressed genes.’ However, it is not the only important difference. In this paper, we propose a statistical method to test the difference between the dispersions of distributions, and we call these genes ‘differentially dispersed’ (DD) genes. A gene can be a DE gene, a DD gene, both, or neither. Our proposed method identifies genes that are DE, DD, or both.

DD genes can be biologically important. For example, while genes that consistently express in a higher/lower level in the cancer group may be important driver genes or signature genes for cancer, that is, the cancer will not form or develop without the over-expression/under-expression of these genes, some other genes may have much higher expression than normal samples in some subsets of cancer samples and much lower expression in other subsets of cancer samples. Even if these genes are not driver genes, they can be genes that are responsive to cancer and may determine whether the tumor can withstand medical treatment or has the ability of metastasis, and hence may be important clinical targeting genes. However, these genes are likely to be missed by DE analysis as their average expression may not be very different from the normal group, or at least their fold change will not be large and thus they will not be ranked highly by DE analysis.

Our proposed method can only be used for RNA-Seq data with relatively large sample sizes, say, no less than a couple dozen replicates in each group. Note that this sample size may still be regarded as ‘small’ in classic statistical literature. In negative binomial distributions, the dispersion parameter is usually much harder to estimate accurately compared to the mean, that is, the uncertainty/variance of the estimate is much larger, and thus no sufficient power can be gained in data with small sample sizes. While most of the past and current RNA-Seq datasets are still of small sample sizes, which is also a reason why state-of-art DE detection methods are very popular, the development of multiplexing and single-cell technology has significantly lowered the cost of sequencing each sample, making data with larger sample sizes more and more common. Also, in some applications such as cancer-related research, large sample sizes are required as a few samples are not enough to characterize the high heterogeneity, with some even behaving like ‘outliers,’ of the cancer samples. In cancer-related research, high statistical power is often not the primary concern of analysis, as the difference of mean expressions between groups are often very large and are thus easy to detect—instead, reliability is much more critical.

Our method has been designed to behave very robustly to possible outliers. Starting from Li and Tibshirani (2013), who brought the existence of outliers and importance of reducing their impact to the table, various nonparametric methods have been proposed for DE analysis, and existing parametric methods are robustified. The most widely used way of robustifying parametric methods relies on the Huber function, which has been shown by many statistical literatures to have limited capacity of robustness (e.g. Cantoni and Ronchetti, 2001; Huber and Ronchetti, 2009; Maronna et al., 2006). Especially, a dramatic outlier can still change the inference result completely. Instead, based on the statistical framework developed by Cantoni and Ronchetti (2001) and Aeberhard et al. (2014),

we use a redescending cost function, which minimizes the influence of dramatic outliers by putting very small or even zero weights on them.

2 Materials and methods

2.1 A negative binomial model and a naive likelihood ratio test

Our setting is two-class (two-group) RNA-Seq data, and we model each count by a negative binomial distribution. The negative binomial distribution is a generalization of the Poisson distribution to take the overdispersion of counts across samples into account. A negative binomial distribution with mean μ and dispersion parameter ϕ , denoted as $NB(\mu, \phi)$, has probability mass function:

$$P(X = x; \mu, \phi) = \frac{\Gamma(x + 1/\phi)}{\Gamma(1/\phi)\Gamma(x + 1)} (\phi\mu + 1)^{-1/\phi} \left(\frac{\phi\mu}{\phi\mu + 1} \right)^x, \quad (1)$$

for $X = 0, 1, 2, \dots$, and has variance $\mu + \phi\mu^2$.

Suppose we have data from p RNA-Seq experiments, each measuring the expression levels of n genes. Let x_{ij} be the number of reads mapped to gene i in experiment j , $i = 1, \dots, n$, and $j = 1, \dots, p$. Let S_k be the index set of group k , $k = 1, 2$. That is, $j \in S_k$ means sample j is in group k . Let d_j be the sequencing depth, a measure of total number of reads generated, of experiment j . Let $\exp(\beta_{i1})$ and $\exp(\beta_{i2})$ be the expression of gene i in group 1 and group 2, respectively. Then we assume

$$X_{ij} \sim NB\left(\sum_{k=1,2} d_j \exp(\beta_{ik}) I_{j \in S_k}, \sum_{k=1,2} \phi_{ik} I_{j \in S_k}\right), \quad (2)$$

where I is the indicator function, which equals 1 if the condition in the subscript is satisfied and 0 otherwise.

Negative binomial models used by previous methods (e.g. Anders and Huber, 2010; Hardcastle and Kelly, 2010; Robinson et al., 2010) assume the dispersion parameter for a gene is the same across classes, that is, $\phi_{i1} = \phi_{i2}$ for $i = 1, \dots, n$. This assumption enables these methods to use all samples to estimate the dispersion parameter, which is an efficient way to lower the variance of estimation when the sample size is small, although it may bring bias to the estimate. In our model as defined in Equation (2), this assumption is released.

In Equation (2), d_j is usually estimated beforehand using all data (see e.g. edgeR, DESeq, PoissonSeq) and assumed known. β_{i1} , β_{i2} , ϕ_{i1} and ϕ_{i2} are unknown and need to be estimated.

Differential expression (DE) analysis tests the hypothesis $\beta_{i1} = \beta_{i2}$. As described in the Section 1, in this paper we also want to test the hypothesis $\phi_{i1} = \phi_{i2}$. We call genes that have $\phi_{i1} \neq \phi_{i2}$ differentially dispersed, or DD for short. A gene can be DE, DD, both, or neither.

A naive way of identifying DD genes is to use the likelihood ratio test. Let l_{i1} and l_{i0} be the log likelihood of the model defined in Equation (2) without constraint and with constraint $\phi_{i1} = \phi_{i2}$. Then $2(l_{i1} - l_{i0})$ follows χ^2_1 distribution under the null hypothesis, and a P -value can be obtained accordingly for gene i .

Unfortunately, this naive solution does not work well when applying to real data because of the outliers. Figure 1 shows an example from a real dataset of KIRC samples (see Section 3.4 for more details about this dataset). Gene SLC4A7 has significantly larger dispersion in the second group, but it seems that this large dispersion is caused by a single outlier. If this outlier is excluded from the analysis, then the two groups have quite comparable dispersions (P -value = 0.658). Outliers, while having been recognized to distort

Fig. 1. Expression of gene SLC4A7 in KIRC dataset. Without removing the outlier, the dispersions for the cancerous and non-cancerous data appear significantly different. However, once the outlier is excluded, the two groups have quite comparable dispersions, with a P -value = 0.658

testing of DE genes, have even larger effects in dispersion estimation, and thus they have to be carefully handled.

2.2 A robust estimator of mean and dispersion parameters

We propose a robust estimate of dispersion parameters in Equation (2). Note that the parameters in the two groups are separable, that is, samples in group 1 do not provide any information for estimating β_{i2} and ϕ_{i2} , and samples in group 2 do not provide any information for estimating β_{i1} and ϕ_{i1} . Thus, we estimate β_{ik} and ϕ_{ik} by using samples in group k only, $k = 1, 2$. This makes our problem a rather simple generalized linear model for negative binomial distribution with the following log link

$$\log(\mu_{ijk}) = \eta_{ijk} = \log d_j + \beta_{ik}. \quad (3)$$

This is a log link with an offset $\log d_j$.

We apply the robust M-estimator proposed by Cantoni and Ronchetti (2001) and Aeberhard et al. (2014) to this model, and the solution of β_{ik} and ϕ_{ik} is given by the following two equations

$$U_\beta(\beta_{ik}, \phi_{ik}) = \sum_{j \in S_k} [\psi_{c_\beta}(r_{ijk}) V_{ijk}^{-1/2} \mu_{ijk} - a_{ijk}] = 0, \quad (4)$$

$$U_\phi(\beta_{ik}, \phi_{ik}) = \sum_{j \in S_k} \left[\frac{\psi_{c_\phi}(r_{ijk})}{r_{ijk}} \Psi_{ijk} - b_{ijk} \right] = 0. \quad (5)$$

In the above equations,

- μ_{ijk} is the expectation of x_{ij} when in group k , as defined by Equation (3).
- $V_{ijk} = \mu_{ijk} + \phi_{ik} \mu_{ijk}^2$ is the variance of x_{ij} when in group k .
- $r_{ijk} = (x_{ijk} - \mu_{ijk}) / \sqrt{V_{ijk}}$ is the Pearson residual.
- $\psi_c(r) = \begin{cases} ((r/c)^2 - 1)^2 r & |r| \leq c \\ 0 & |r| > c \end{cases}$ is Tukey's biweight function. $c > 0$ is a pre-specified constant that determines the degree of robustness of the estimator. In Equation (4), we use $c = c_\beta$. In Equation (5), we use $c = c_\phi$. We will discuss how to choose c_β and c_ϕ later.
- $\Psi_{ijk} = (-1/\phi_{ik}^2)[F(x_{ijk} + 1/\phi_{ik}) - F(1/\phi_{ik}) - \log(\phi_{ik} \mu_{ijk} + 1) - \frac{\phi_{ik}(x_{ijk} - \mu_{ijk})}{\phi_{ik} \mu_{ijk} + 1}]$ is the partial derivative of the log likelihood function to ϕ_{ik} , where $F(x) = \partial \log \Gamma(x) / \partial x$ denotes the digamma function.

- $a_{ijk} = E[\psi_{c_\beta}(r_{ijk})] V_{ijk}^{-1/2} \mu_{ijk}$ and $b_{ijk} = E\left[\frac{\psi_{c_\phi}(r_{ijk})}{r_{ijk}} \Psi_{ijk}\right]$ are two constants that ensure the Fisher consistency of the estimator.

Estimators 4 and 5 seem complicated, but actually have simple and clear meanings. Equations (4) and (5) are the robustified quasi-likelihood equations for β_{ik} and ϕ_{ik} , respectively. Note that if $\psi(r) = r$, then the two equations become the regular quasi-likelihood equation for β_{ik} and ϕ_{ik} . Thus, the estimate is like placing a weight $\psi(r_{ijk})/r_{ijk}$ on sample j . The robustification is achieved by using Tukey's biweight function, which plays a similar role as Huber's function $\psi_{c, \text{Huber}}(r) = \max(-c, \min(c, r))$, but is much more robust than Huber's function to dramatic outliers. This is because for an outlier with very large estimated $|r|$, Tukey's biweight function will have $\psi_c/r = 0$, meaning this presumed dramatic outlier is dropped out from the quasi-likelihood function, while Huber's function will have $\psi_c/|r| > 0$, meaning this outlier is still included in the quasi-likelihood. Numerous theoretical and empirical work have shown that while estimates using Huber's function have quite limited robustness in real data, estimates using Tukey's biweight are reliable (e.g. Cantoni and Ronchetti, 2001; Huber and Ronchetti, 2009; Maronna et al., 2006).

Parameter c determines the level of robustness, as well as the efficiency of the estimator: a smaller c gives a more robust but less efficient estimator, while a larger c means less robustness but more efficiency. When $c \rightarrow +\infty$, $\psi_c(r)/r \rightarrow 1$, and thus the estimator becomes the regular quasi-likelihood estimate. In general, one can choose different c_β and c_ϕ to give different robustness in estimating β and ϕ . Aeberhard et al. (2014) chose $c_\beta = c_\phi = 4$, which gives a very robust estimator with high efficiency on a real dataset they used. We explored this choice through simulations and found that it is still a reasonable choice for our RNA-seq applications, and therefore adopt it for all simulations and real data analysis for this paper. Details about our simulation study of the influence of c values can be found in the Supplementary Material.

The estimate of β_{ik} and ϕ_{ik} is given by Equations (4) and (5). The basic idea of solving them is by alternatively solving ϕ_{ik} from $U_\phi(\beta_{ik}, \phi_{ik}) = 0$ while fixing the value of β_{ik} , and solving β_{ik} from $U_\beta(\beta_{ik}, \phi_{ik}) = 0$ while fixing the value of ϕ_{ik} . Theoretically, since the primitive function of ϕ_c , and hence the primitive functions of U_ϕ and U_β , are not convex, it is possible that multiple sets of values of U_ϕ and U_β can satisfy Equations (4) and (5). However, we never encountered any difficulties in finding a reasonable (and actually well-performing, see simulations and real data analysis) solution for this simple two-variable optimization problem by using Algorithm 1. In the algorithm, we start with a naive high breakpoint estimate of β_{ik} and a naive high breakpoint estimate of ϕ_{ik} , and searching for nearby updates in each iteration. The solution seems to be quite insensitive to the starting values.

Algorithm 1 Solving robust quasi-likelihood Equations (4) and (5)

Step 1 Initialize β_{ik} and ϕ_{ik} :

- **Step 1.1** Let $w_{ij} = x_{ij}/d_j$ for $j \in S_k$. Rank them from the smallest to largest. Let T_{ik} be the set of j such that w_{ij} is in the δ 's and $(1 - \delta)$'s quantiles. Here δ is a constant between 0 and 0.5. We use $\delta = 0.2$ for all our simulations and real data analysis. Let $\beta_{ik}^{(0)} = \log \left[\frac{\sum_{j \in T_{ik}} x_{ij}}{\sum_{j \in T_{ik}} d_j} \right]$.
- **Step 1.2** Let $\mu_{ijk} = \exp(\log d_j + \beta_{ik}^{(0)})$ and $\phi_{ik} = [(x_{ijk} - \mu_{ijk})^2 - \mu_{ijk}]/\mu_{ijk}^2$. Let ϕ_{ik} be the δ -trimmed mean of ϕ_{ijk} . Then we let $\phi_{ik}^{(0)} = \text{median}_{i=1, \dots, n, k=1, 2} \phi_{ik}$, which is common for all genes.

Step 2 For $t = 0, \dots, t_{\text{MAX}}$,

- **Step 2.1** Let $\beta_{ik} = \beta_{ik}^{(t)}$, search for the root of Equation (5) near to $\phi_{ik}^{(t)}$. Let this root be $\phi_{ik}^{(t+1)}$. The search is done by:
 - **Step 2.1.1** For $l = 1, \dots, l_{\text{MAX}}$, let $\phi_1 = \phi_{ik}^{(t)} \times 1.2^{-l}$ and $\phi_2 = \phi_{ik}^{(t)} \times 1.2^l$, let $\phi_{\text{low}} = \phi_1$ and $\phi_{\text{up}} = \phi_{ik}^{(t)}$ and go to Step 2.1.2 if $U_\phi(\beta_{ik}^{(t)}, \phi_{ik}^{(t)})U_\phi(\beta_{ik}^{(t)}, \phi_1) \leq 0$, and let $\phi_{\text{low}} = \phi_{ik}^{(t)}$ and $\phi_{\text{up}} = \phi_2$ and go to Step 2.1.2 if $U_\phi(\beta_{ik}^{(t)}, \phi_{ik}^{(t)})U_\phi(\beta_{ik}^{(t)}, \phi_2) \leq 0$.
 - **Step 2.1.2** Search for the solution of $U_\phi(\beta_{ik}^{(t)}, \phi) = 0$ in the interval of $[\phi_{\text{low}}, \phi_{\text{up}}]$. This can be done by using the uniroot function in R, or by using binary search.
- **Step 2.2** Let $\phi_{ik} = \phi_{ik}^{(t)}$, search for the root of Equation (4) near to $\beta_{ik}^{(t)}$. Let this root be $\beta_{ik}^{(t+1)}$. The search is done by:
 - **Step 2.2.1** For $l = 1, \dots, l_{\text{MAX}}$, let $\beta_1 = \beta_{ik}^{(t)} - 0.1l$ and $\beta_2 = \beta_{ik}^{(t)} + 0.1l$, let $\beta_{\text{low}} = \beta_1$ and $\beta_{\text{up}} = \beta_{ik}^{(t)}$ and go to Step 2.2.2 if $U_\beta(\beta_{ik}^{(t)}, \phi_{ik}^{(t)})U_\beta(\beta_1, \phi_{ik}^{(t)}) \leq 0$, and let $\beta_{\text{low}} = \beta_{ik}^{(t)}$ and $\beta_{\text{up}} = \beta_2$ and go to Step 2.2.2 if $U_\beta(\beta_{ik}^{(t)}, \phi_{ik}^{(t)})U_\beta(\beta_2, \phi_{ik}^{(t)}) \leq 0$.
 - **Step 2.2.2** Search for the solution of $U_\beta(\beta, \phi_{ik}^{(t)}) = 0$ in the interval of $[\beta_{\text{low}}, \beta_{\text{up}}]$. This can be done by using the uniroot function in R, or by using binary search.
- **Step 2.3** If $|\beta_{ik}^{(t+1)} - \beta_{ik}^{(t)}| < \Delta_\beta$ and $|\phi_{ik}^{(t+1)} - \phi_{ik}^{(t)}| < \Delta_\phi$, go to Step 3. Here Δ_β and Δ_ϕ are small constants. We used 0.01 and 0.005, respectively.

Step 3 Let $\hat{\beta}_{ik} = \beta_{ik}^{(t+1)}$ and $\hat{\phi}_{ik} = \phi_{ik}^{(t+1)}$.

2.3 Identifying DE and DD genes

Aeberhard *et al.* (2014) have shown that asymptotically the estimates follow a Gaussian distribution jointly,

$$\sqrt{|S_k|} \begin{pmatrix} \hat{\beta}_{ik} - \beta_{ik} \\ \hat{\phi}_{ik} - \phi_{ik} \end{pmatrix} \rightarrow N(0, M_{ik}^{-1} Q_{ik} M_{ik}^{-T}), \quad (6)$$

where M_{ik} and Q_{ik} are 2×2 matrices with elements given in the [Supplementary Material](#). We let $A_{ik} = \frac{1}{|S_k|} M_{ik}^{-1} Q_{ik} M_{ik}^{-T}$, then $\text{var} \hat{\beta}_{ik} = A_{ik,11}$ and $\text{var} \hat{\phi}_{ik} = A_{ik,22}$. It is then straightforward to test for DE genes and DD genes. To test DE,

$$z_{i, \text{DE}} = \frac{\hat{\beta}_{i2} - \hat{\beta}_{i1}}{\sqrt{A_{i2,11} + A_{i1,11}}} \sim N(0, 1), \quad (7)$$

under the null hypothesis $\beta_{i2} = \beta_{i1}$, and the P -value of DE can be obtained accordingly. To test DD,

$$z_{i, \text{DD}} = \frac{\hat{\phi}_{i2} - \hat{\phi}_{i1}}{\sqrt{A_{i2,22} + A_{i1,22}}} \sim N(0, 1), \quad (8)$$

under the null hypothesis $\phi_{i2} = \phi_{i1}$, and the P -value of DD can be obtained accordingly.

The above tests for DE and DD should be highly robust to outliers. We have implemented these tests and the robust estimations of the parameters as an R package, which we have named DiPhiSeq, as it accounts for differences in the dispersion parameter ϕ .

It is important to note that in most applications, because we are testing the mean and the dispersion for each gene, multiple testing

Fig. 2. Type I error and power plots across different mean count values, on simulated data with no outliers (**A**) and simulated data with outliers (**B**). Each line represents a different dispersion combination for the two experimental conditions. The dashed line represents the $y = x$ line. On both simulated data with and without outliers, type I error appears to be well controlled across all dispersion and mean combinations. Power is negatively impacted as the difference between the two dispersions becomes smaller, but this effect is lessened as the mean increases. Outliers do not seem to hurt the performance of our method

becomes an issue. Consequently, DiPhiSeq uses FDR-corrected P -values as proposed by [Benjamini and Hochberg \(1995\)](#).

3 Results

3.1 Type I error and power control when testing differential dispersion

We first wished to explore the effectiveness of our method at identifying differentially dispersed genes by examining the power and type I error control for simulated gene counts across specific values for the dispersion and mean. For each simulation, we generated 1000 genes across two experimental conditions, each with 30 samples. We considered all possible combinations of values for the dispersions from $\phi_1, \phi_2 = (0.1, 0.3, 0.5, 1)$, and for the mean from $\mu = (20, 50, 100, 1000)$. Note that the same mean was used for both experimental conditions because we are testing the performance of our method's ability to identify differential dispersion only. Sequencing depths d were randomly generated from $2^{\text{uniform}(-3,3)}$. Finally, counts were randomly generated using the distributions $X_{1i} \sim \text{NB}(d_i \mu, \phi_1)$ and $X_{2i} \sim \text{NB}(d_i \mu, \phi_2)$.

The results of these simulations are plotted in [Figure 2A](#). In the left-hand plots, we observe barely any deviations from the diagonal, meaning the type I error rate is accurately controlled across all variable combinations. Looking at the power, we observe that the differences between the dispersions for the two experimental conditions have the greatest effect on the power of our test. As would be expected, the dispersion combinations with the larger difference have a much larger power. The largest difference, 0.1 versus 1, reaches a power of 1 by a threshold of 0.05. As the difference between the two dispersions decreases, the power is significantly impacted. As the mean is increased, however, we observe an improvement in power for all dispersion combinations.

An important feature of our method is its robustness, so we also examined the effect of outliers on its power and type I error control. For these simulations, outliers were added for each gene count by

randomly generating two count values using $X_i \sim \text{NB}(10d_i\mu, \phi)$ as their distribution. The results of these simulations can be found in Figure 2B. We observed little change in the type I error rate and a slight decrease in power. Overall, this minor impact clearly demonstrates the effectiveness of our robust estimator in handling the presence of outliers when estimating the parameters of both distributions.

3.2 Simulations based on real data

We wished to examine DiPhiSeq further by exploring its performance with more realistic simulations and in comparison to several widely used parametric methods for estimating differential expression. For these simulations, we tested the performance of each method under several different scenarios representing different assumptions, all containing data generated based on a real dataset. First, we simulated data under the simplest case, in which only differentially expressed genes are present. This is the ideal case for methods such as edgeR, DESeq and baySeq. Next, to create a more realistic dataset, we added differentially dispersed genes to the simulations. Finally, it has been observed in some RNA-Seq datasets that mean and dispersion values can be negatively associated, so we also performed simulations using these ‘trended’ dispersions.

All of these simulations were based on a real dataset, which we also used for a real data analysis. Count values X_{ij} for each condition i were generated using a negative binomial distribution. Please see the [Supplementary Material](#) for more details about each simulation structure.

We compared our method’s performance with three popular parametric methods for estimating differential expression: edgeR (Robinson *et al.*, 2010, version 3.12.1), DESeq (Anders and Huber, 2010, version 1.22.1) and baySeq (Hardcastle and Kelly, 2010, version 2.8.0). For edgeR, the recommended likelihood ratio test was used to generate P -values for each gene, and both trended and tag-wise dispersion options were estimated to test how the method handled the inclusion of differentially dispersed genes. Similarly, for

DESeq the negative binomial test was used to generate P -values for each gene, and dispersions were estimated with and without the ‘gene-est-only’ option to allow for trended dispersions. Finally, for baySeq, priors were estimated using the quasi-likelihood estimate, with equal dispersions not assumed for simulations with differentially dispersed genes included, and with the associated false discovery rates for each gene’s estimated likelihood used as P -values. For all methods, sequencing depths were assumed unknown and estimated based on the simulated datasets.

To evaluate each method’s performance, we used the Receiver Operating Characteristic (ROC) curve. We calculated these curves for each simulation, and then plotted the average partial ROC curve for each method across the 50 simulations, focusing on the region where the false positive rate is 0.2 or lower—as this is the area of greatest interest to biologists.

3.2.1 Differentially expressed genes

First, we simulated 50 datasets to reflect the naive case where only differential expression occurs, as is assumed by many DE detection methods. For each dataset, we simulated $n=4000$ genes across $p=30$ experiments under condition 1 and condition 2, for which 70% of genes have no differential expression and 30% are differentially expressed. To emulate a real data analysis, we then filtered these genes to retain the 1000 most highly expressed, ensuring to maintain 30% were still differentially expressed. To add outliers to our simulations, we randomly selected 2% of simulated data counts throughout each simulated dataset and multiplied those count values by a factor of 10.

The average partial ROC curves for each method are shown in the left side of Figure 3A. We see the three comparison methods slightly out-performing ours (plotted in red) when no outliers are included. However, when outliers are added, we observe our method outperforming all others. In fact, our method displays the smallest change in performance after the addition of outliers—DESeq, edgeR and baySeq all show large drops in performance, well below DiPhiSeq. Clearly, outliers are impacting the estimation of the dispersions for the comparison methods, demonstrating the robustness of the non-convex weight used in our parameter estimation.

3.2.2 Differentially expressed and dispersed genes

We also simulated 50 datasets to reflect the more realistic scenario where differential expression and dispersion both occur. For each of these datasets, we simulated $n=4000$ genes across $p=30$ experiments under condition 1 and condition 2, for which 49% have no differential expression or dispersion, 21% are differentially expressed only, 21% are differentially dispersed only and 9% are differentially expressed and dispersed. Details of the simulation are given in [Supplementary Material](#).

The average partial ROC curves for each method are shown in the right side of Figure 3A. Surprisingly, when differentially dispersed genes are added, edgeR and DESeq continue to outperform DiPhiSeq in the unrealistic scenario of no outliers. It appears that their assumptions of equal dispersions between the two distributions do not have a significant effect on their performance. This may be explained, as Boneau (1960) has shown in the case of the t -test, by the fact that violations of equal variance assumptions do not have a large impact on performance when sample sizes are still the same—as is the case for our simulations. These results also match what Sonesson and Delorenzi (2013) observed in their comparison study, in which they also observed a comparable worsened performance by

Fig. 3. Partial ROC curve for real data-based simulations with non-trended dispersions (A) and trended dispersions (B). The dashed line represents $y=x$, or the equivalent of a random guess. (A) The inclusion of differentially dispersed genes only appears to have a large impact on baySeq’s performance, while outliers have a large effect on all comparison methods. The robustness of DiPhiSeq allows it to outperform edgeR, DESeq and baySeq in the more realistic scenarios where outliers are included. (B) edgeR and DESeq’s methods for handling trended dispersions only improve their performance when outliers are not included. When outliers are added to the simulation, we once again see DiPhiSeq outperforming all comparison methods

baySeq after the addition of differentially dispersed genes for larger sample sizes.

Outliers were also added to this simulation, using the same method as described above. After their addition to the simulation, DESeq, edgeR and baySeq again show large drops in performance, well below DiPhiSeq. As the strength of the outliers drops, DiPhiSeq loses its advantage over the other methods in detecting DE genes. Under our simulation setting (2% outliers), DiPhiSeq is less powerful in detecting DE genes when the outlier strength is below 6. However, DiPhiSeq is the only method that detects DD genes, and it successfully detects DD genes regardless of the outlier strength. For plots and more discussion see the [Supplementary Material](#).

3.2.3 Simulating with trended dispersions

It has been observed in real datasets that mean and dispersion values are sometimes negatively associated. Methods such as edgeR and DESeq make use of this association by borrowing information from other genes to improve their dispersion estimates. We wanted to explore the effect of the presence of these ‘trended’ dispersions on each method’s performance for larger sample sizes to see if this would provide them any advantage over DiPhiSeq.

Fig. 4. Plots of the true positive counts versus the P -value threshold for all simulations including DD genes. With no outliers, all methods appear to identify a comparable number of DE genes. However, when outliers are added, both DESeq and edgeR identify fewer true positives. The solid red line depicts the total number of DE and DD genes identified by DiPhiSeq, illustrating the fact that our method effectively doubles the number of true discoveries by considering DD genes

To simulate trended dispersions, we adjusted our first two simulations to use a multivariate distribution to generate correlated values for our β 's and ϕ 's, following the method described in [Specht and Li \(2015\)](#). The correlated parameter value pairs were then used in the distributions used to sample counts, following the same structure for each of the differential expression or dispersion cases used in the un-trended simulations.

The results of these simulations are shown in [Figure 3B](#). While edgeR and DESeq both have methods of addressing trended dispersions, these only improve their performance when outliers are not included. The relatively large sample sizes used appear to allow edgeR and DESeq to develop dispersion estimates that are not improved by borrowing information from other genes, particularly when outliers are added. DiPhiSeq outperforms all comparison methods once outliers are incorporated, demonstrating once again the robustness of our estimator.

3.3 Identifying DD genes

Across all four real data simulation structures, our method shows a consistent, reliable performance that is little affected by outliers. The inclusion of trended dispersions does not appear to provide DESeq or edgeR any leverage over DiPhiSeq, particularly when outliers are added.

The analysis of these simulations thus far, though, does not capture the novel information that our method is able to identify through its testing of differential dispersion. To illustrate this feature, we examined the number of true positives captured by each method for the simulations incorporating differentially dispersed genes (see [Fig. 4](#)). Here, a true positive is defined as a gene correctly identified as either differentially dispersed or expressed.

Again, our method is shown in red—with three different lines representing genes correctly identified as differentially expressed, differentially dispersed and differentially expressed and/or dispersed. While all methods appear to identify a comparable number of differentially expressed genes when no outliers are included, outliers appear to have a large effect on DESeq’s performance, cutting the number of true positives approximately in half. baySeq performs comparably to DiPhiSeq in its discovery of truly differentially expressed genes—however, if we reflect back on the partial ROC curves, these discoveries are accompanied by a large proportion of false positives in the case of baySeq.

More striking, though, is the solid red line included in each plot. This represents the total number of correctly identified differentially expressed and/or dispersed genes found by DiPhiSeq. Through its novel identification of differential dispersion, our method nearly doubles the number of true discoveries.

3.4 Real data analysis

To evaluate the usefulness of our method in a real world application, we examined a dataset consisting of Kidney Renal Clear Cell Carcinoma (KIRC) samples. An estimated 62 700 Americans will be diagnosed with kidney cancer in 2017 ([Siegel et al., 2016](#)), with KIRC as the most common form—responsible for approximately 92% of diagnoses ([Chow et al., 2010](#)). Our dataset was obtained from the GDC Data Portal as part of the Cancer Genome Atlas Kidney Renal Clear Cell Carcinoma data collection ([Akin et al., 2016](#)), and consisted of raw RNA-Seq counts from 35 tumor and 35 normal samples.

For our analysis of the KIRC dataset, we focused on differential dispersion and expression between cancerous and non-cancerous conditions. We filtered the dataset to keep only those genes with a

Fig. 5. Results on a real dataset. **(A)** Venn Diagram of the top one hundred genes identified by each method for the KIRC dataset. Genes identified by DiPhiSeq as differentially expressed or differentially dispersed are shown in red and blue, respectively. We observe little overlap between any of the methods, demonstrating that DiPhiSeq has the ability to identify novel differentially expressed genes. The fact that there is little overlap between DiPhiSeq's DD genes and the other methods also illustrates that many of these differentially dispersed genes would be missed by methods that only consider differential expression. **(B)** Boxplots showing the log($x+1$) transformed and sequencing depth normalized expression counts for the genes only identified by our method through differential dispersion. The expression from tumor samples are plotted in blue, while the expression from normal samples are plotted in red. Looking at the large differences in variation between tumor and normal tissue, these genes clearly demonstrate differential dispersion. Notably, the greater dispersion occurs in the normal samples for some of the genes, e.g. ALB and PTPRO

row mean count greater than 300 for both normal and tumor samples. We then filtered these genes to keep those above a relative interquartile range threshold of 0.7, resulting in 4006 genes for analysis.

We searched for differential dispersion and differential expression with DiPhiSeq, and looked for differential expression with edgeR, DESeq and baySeq. We examined the top one hundred genes

identified. The results are shown in the Venn diagram contained in Figure 5A. There is little overlap between our method's discoveries and those of the comparison methods, demonstrating that DiPhiSeq has the ability to identify novel differentially expressed genes.

As evidence of differential dispersion as an occurring biological phenomenon, we looked at the expression profiles for the 94 genes identified only by DiPhiSeq through differential dispersion. Figure 5B contains the boxplots of each gene's log expression counts. It can be clearly seen that all of these genes display dramatic differential dispersion in their expression profiles. This provides strong evidence that not only is differential dispersion an occurring phenomenon, but that it does not coincide necessarily with differential expression and hence genes exhibiting this behavior could be missed by other existing methods.

We also searched the Renal Cancer Gene Database(RCGD) (Ramana, 2012) for our uniquely identified differentially expressed and dispersed genes. Several were found to have direct associations with KIRC. The genes KRT19, PTPRO and TFCP2L1 were only captured as differentially dispersed, and the genes COL4A3 and ENO2 were only identified as differentially expressed by DiPhiSeq, and were present in the RCGD. KRT19 has been shown to be associated with patient outcome and can serve as a prognostic tool for KIRC patients (Mertz *et al.*, 2008). The genes PTPRO and COL4A3 are known tumor suppressors, and have exhibited down regulation in KIRC studies (Dalgin *et al.*, 2007; Xu *et al.*, 2010). TFCP2L1, a developmental transcription factor, has displayed changes in expression—thereby disrupting normal renal development—in KIRC tumor samples (Tun *et al.*, 2010). The protein encoded by ENO2, enolase 2, is a widely known diagnostic marker for tumors, and has exhibited upregulation in KIRC samples (Sun *et al.*, 2010). Again, none of these genes were identified within the top 100 most significant genes by any of the comparison methods.

4 Conclusion and discussion

DiPhiSeq is a powerful and important tool for the study of both differential expression and differential dispersion. We have shown through simulations and a real data application that the use of a robust estimator is fundamental to properly estimating the mean, and more importantly, the dispersion of RNA-Seq count distributions.

In simulations, we observe superior performance by DiPhiSeq when outliers are included, regardless of whether differentially expressed or both differentially expressed and dispersed genes are incorporated, or trended dispersions. In our real data analysis, we also demonstrate that differential dispersion is an occurring biological phenomenon, and that it does not necessarily coincide with differential expression—therefore genes displaying differential dispersion could be missed by other differential expression-focused methods.

DiPhiSeq handles outliers by using a robust M-estimator proposed by Cantoni and Ronchetti (2001) and Aeberhard *et al.* (2014). An obvious alternative is a two-step strategy that first detects outliers and then removes them from the analysis. However, this 'simple' strategy may actually be both problematic and more complicated. First, detecting outliers under negative binomial models, especially considering that different RNA-seq experiments have different sequencing depths, is not trivial. Second, it will be hard to choose the cutoff for declaring 'outliers'. On one hand, some outliers may not be that strong. On the other hand, 'regular' samples from tails of negative-binomial distributions look like outliers. There will be a nontrivial multiple-test-correction problem in choosing this

cutoff. Third, removing ‘outliers’, which are imputed rather than true, from the following DE/DD analysis will change the asymptotic variance of the testing statistics, but this change is very difficult to measure quantitatively.

When neither outliers nor differential dispersion are present, DiPhiSeq does lose some efficiency due to its robust estimation of dispersions for each condition. The current version of the DiPhiSeq R package includes a simple test for the presence of outliers. Also, if a regular run of DiPhiSeq detects no DD genes, one may run DiPhiSeq again to detect DE genes by setting equal dispersions for the two conditions. A new version of DiPhiSeq in the near future will include this setting as an option. At the same time, we believe that our model, with both DD and outliers, is usually a more realistic characterization of RNA-Seq data.

This paper explored the performance of tests for DE and DD genes, but our estimator could also be applied to test other types of measurement of difference in distributions. For example, in GFOLD, the genes are ranked according to their log fold change (Feng *et al.*, 2012). To avoid being too liberal, GFOLD uses the lower or upper bound (depending on the sign of the log fold change) of the 99% confidence interval of the log fold change. Our estimator readily enables a corresponding test. The conserved log fold change can be calculated by

$$g_{i,z} = S_{z_{1-z/2}\sqrt{A_{i2,11}+A_{i1,11}}}(\hat{\beta}_{i2} - \hat{\beta}_{i1}) \\ = \text{sign}(\hat{\beta}_{i2} - \hat{\beta}_{i1})(|\hat{\beta}_{i2} - \hat{\beta}_{i1}| - z_{1-z/2}\sqrt{A_{i2,11} + A_{i1,11}})_+$$

On the above, S is the soft-thresholding function. Once again, due the robustness of our estimator, we would expect to observe superior performance of our corresponding test on realistic datasets containing outliers and differentially dispersed genes.

The improvement in performance observed for DiPhiSeq over comparison methods when identifying differential expression is possible because of the large size of the samples. In the case where only smaller sample sizes are available, methods such as edgeR, DESeq and baySeq will have a distinct advantage. However, for large samples, DiPhiSeq has the ability to identify novel and biologically useful genes.

Acknowledgements

We thank Siyuan Zhang from Biological Sciences at University of Notre Dame and Hui Jiang from Biostatistics at University of Michigan for valuable discussions. We also thank three anonymous reviewers for their constructive suggestions.

Funding

This work was supported by the National Institutes of Health [R03CA212964 to J.L.].

Conflict of Interest: none declared.

References

- Aeberhard, W.H. *et al.* (2014) Robust inference in the negative binomial regression model with an application to falls data. *Biometrics*, **70**, 920–931.
- Akin, O. *et al.* (2016) Data from the cancer genome atlas kidney renal clear cell carcinoma [tcga-kirc] collection. *The Cancer Imaging Archive*.
- Anders, S. and Huber, W. (2010) Radiology data from the cancer genome atlas kidney renal clear cell carcinoma [TCGA-KIRC] collection. *The Cancer Imaging Archive*. <http://doi.org/10.7937/K9/TCIA.2016.V6PBVTDR>.
- Benjamini, Y. and Hochberg, Y. (1995) Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J. R. Stat. Soc. Ser. B (Methodological)*, **57**, 289–300.
- Boneau, C.A. (1960) The effects of violations of assumptions underlying the t test. *Psychol. Bull.*, **57**, 49.
- Cantoni, E. and Ronchetti, E. (2001) Robust inference for generalized linear models. *J. Am. Stat. Assoc.*, **96**, 1022–1030.
- Chow, W.-H. *et al.* (2010) Epidemiology and risk factors for kidney cancer. *Nat. Rev. Urol.*, **7**, 245–257.
- Dalgin, G.S. *et al.* (2007) Identification and characterization of renal cell carcinoma gene markers. *Cancer Inf.*, **3**, 65–92.
- Feng, J. *et al.* (2012) Gfold: a generalized fold change for ranking differentially expressed genes from rna-seq data. *Bioinformatics*, **28**, 2782–2788.
- Hardcastle, T.J. and Kelly, K.A. (2010) bayseq: empirical bayesian methods for identifying differential expression in sequence count data. *BMC Bioinformatics*, **11**, 422.
- Huber, P.J. and Ronchetti, E.M. (2009) *Robust Statistics*, 2nd edn. John Wiley & Sons, Hoboken, NJ.
- Li, J. and Tibshirani, R. (2013) Finding consistent patterns: a nonparametric approach for identifying differential expression in RNA-seq data. *Stat. Methods Med. Res.*, **22**, 519–536.
- Love, M.I. *et al.* (2014) Moderated estimation of fold change and dispersion for RNA-seq data with *DESeq2*. *Genome Biol.*, **15**, 550.
- Maronna, R. *et al.* (2006) *Robust Statistics*, vol. 1. John Wiley & Sons, Chichester, UK.
- Mertz, K.D. *et al.* (2008) Association of cytokeratin 7 and 19 expression with genomic stability and favorable prognosis in clear cell renal cell cancer. *Int. J. Cancer*, **123**, 569–576.
- Ramana, J. (2012) RCDB: renal cancer gene database. *BMC Res. Notes*, **5**, 246.
- Robinson, M.D. *et al.* (2010) edgeR: a bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, **26**, 139–140.
- Siegel, R.L. *et al.* (2016) Cancer statistics, 2016. *CA Cancer J. Clin.*, **66**, 7–30.
- Soneson, C. and Delorenzi, M. (2013) A comparison of methods for differential expression analysis of rna-seq data. *BMC Bioinformatics*, **14**, 91.
- Specht, A.T. and Li, J. (2015) Estimation of gene co-expression from RNA-seq count data. *Stat. Its Interface*, **8**, 507–515.
- Sun, C.Y. *et al.* (2010) Proteomic analysis of clear cell renal cell carcinoma. Identification of potential tumor markers. *Saudi Med. J.*, **31**, 525–532.
- Tun, H.W. *et al.* (2010) Pathway signature and cellular differentiation in clear cell renal cell carcinoma. *PLoS One*, **5**, e10696.
- Van De Wiel, M.A. *et al.* (2012) Bayesian analysis of rna sequencing data by estimating multiple shrinkage priors. *Biostatistics*, **14**, 113–128.
- Wu, H. *et al.* (2012) A new shrinkage estimator for dispersion improves differential expression detection in rna-seq data. *Biostatistics*, **14**, 232–243.
- Xu, C.-x. *et al.* (2010) The expression of tumstatin is down-regulated in renal carcinoma. *Mol. Biol. Rep.*, **37**, 2273–2277.