Submitted to the Annals of Applied Statistics

A COMPOSITIONAL MODEL TO ASSESS EXPRESSION CHANGES FROM SINGLE-CELL RNA-SEQ DATA

By Xiuyu Ma[*], Keegan Korthauer[‡], Christina Kendziorski[†], and Michael Newton[*,†]

Department of Statistics[*], Department of Biostatistics and Medical Informatics[†], University of Wisconsin - Madison; Dana Farber[‡]

On the problem of scoring genes for evidence of changes in the distribution of single-cell expression, we introduce a empirical Bayesian mixture approach and evaluate its operating characteristics in a range of numerical experiments. The proposed approach leverages cell-subtype structure revealed in cluster analysis in order to boost gene-level information on expression changes. Cell clustering informs gene-level analysis through a specially-constructed prior distribution over pairs of multinomial probability vectors; this prior meshes with available model-based tools that score patterns of differential expression over multiple subtypes. We derive an explicit formula for the posterior probability that a gene has the same distribution in two cellular conditions, allowing for a gene-specific mixture over subtypes in each condition. Advantage is gained by the compositional structure of the model, in which a host of gene-specific mixture components are allowed, but also in which the mixing proportions are constrained at the whole cell level. This structure leads to a novel form of information sharing through which the cell-clustering results support gene-level scoring of differential distribution. The result, according to our numerical experiments, is improved sensitivity compared to several standard approaches for detecting distributional expression changes. **plus connection to bursting and other**

1. Introduction. The ability to measure genome-wide gene expression at single-cell resolution has accelerated the pace of biological discovery. Overcoming data analysis challenges caused by the scale and unique variation properties of single-cell data will surely fuel further advances in immunology (Papalexi and Satija, 2017), developmental biology (Marioni and Arendt, 2017), cancer (Navin, 2015), and other areas (Nawy, 2013). Computational tools and statistical methodologies created for data of lower-resolution (e.g., bulk RNA-seq) or lower dimension (e.g., flow cytometry) guide our response to the data science demands of new measurement platforms, but they remain inadequate for efficient knowledge discovery in this rapidly advancing domain (Bacher and Kendziorski, 2016).

An important feature of single-cell studies that could be leveraged better statistically is the fact that cells populate distinct, identifiable subtypes determined by lineage history, epigenetic state, the activity of various transcriptional programs, or other distinguishing factors. Extensive research on clustering cells has produced tools for identifying subtypes, including: `SC3` (Kiselev et al., 2017), `CIDR` (Lin, Troup and Ho, 2017) and `ZIFA` (Pierson and Yau, 2015). We hypothesize that such subtype information may be usefully injected into other inference procedures in order to improve their operating characteristics.

Assessing the magnitude and statistical significance of changes in gene expression associated with changes in cellular condition has been a central statistical problem in genomics for which new tools specific to the single-cell RNAseq data structure have been deployed: `MAST` (Finak et al., 2015), `SCDD` (Korthauer et al., 2016), `D3E` (Delmans and Hemberg, 2016), etc. These tools respond to scRNAseq characteristics, such as high prevelance of zero counts and gene-level multimodality, but they do not fully exploit cellular-subtype information. We address this limitation, aiming to increase power to detect differential distribution. The proposed method measures changes in a gene's marginal mixture distribution, and acquires sensitivity to a wide variety of distributional effects by how it integrates genome-wide with gene-level data. It is implemented in software in the R package `scDDboost` [1]. Modularity in the necessary elements provides some methodological advantages. For example, improvements in clustering may be used in place of the default clustering without altering the form of downstream analysis. Also, by avoiding Markov chain Monte Carlo, `scDDboost` computations are relatively inexpensive for a Bayesian procedure.

Through the compositional model underlying `scDDboost`, subtypes inferred by clustering inform the analysis of gene-level expression. The proposed methodology merges two lines of computation after cell clustering: one concerns patterns of differential expression among the cellular subtypes, and here we take advantage of the powerful `EBseq` method for detecting patterns in negative-binomially-distributed expression data (Leng et al., 2013). The second concerns the counts of cells in various subtypes; for this we propose a Double-Dirichlet-Mixture distribution to model the pair of multinomial probability vectors for subtype counts in two experimental conditions. Further elements are developed, on the selection of the number of subtypes and on accounting for uncertainty in the clus-

[1] http://github.com/wiscstatman/scDDboost/

ter output, in order to provide an end-to-end solution to the differential distribution problem.

To set the context by way of example, Figure 1 shows expression data from 91 human stem cells known to be in the G1 phase of the cell cycle, as well as from 76 such cells known to be in the G2/M phase (Leng et al., 2013). The three panels on the left compare expression of G1 and G2/M cells at three genes (BIRC5, HMMR, and CKAP2), which we happen to know from prior studies have differential activity between G1 and G2/M (Li and Altieri, 1999; Sohr and Engeland, 2008; Dominguez et al., 2016). Standard statistical tools applied to the data behind Figure 1 do not find the observed differences in any of these genes to be statistically significant when controlling the false discovery rate (FDR) at 5% (Supplementary Material, Section 1). But `scDDboost` does include these genes on its 5% FDR list. Considering prior studies, these subtle distributional changes are probably not false discoveries. The right panel in Figure 1 shows these three among many other genes also known to be involved in cell-cycle regulation but not identified by standard tools as altered between G1 and G2/M at the 5% FDR level. The color panel above the heatmap hints at why `scDDboost` has identified these genes. Cells (columns) are clustered by their genome-wide expression profiles into distinct cellular subtypes, as indicated by the color panel. Evidently, these subtypes have changed in their proportions between G1 and G2/M; for instance, there is a lower proportion of *red* cells and a greater proportion of *orange* cells in G2/M. These proportion shifts, inferred from genome-wide data, infuse information into gene-specific tests which measure changes between conditions in the mixture distribution of expression. We note that `scDDboost` agrees with other statistical tools on very strong differential-distribution signals (not shown), but it has the potential to increase power for subtle signals owing to its unique approach to leveraging cell subtype information.

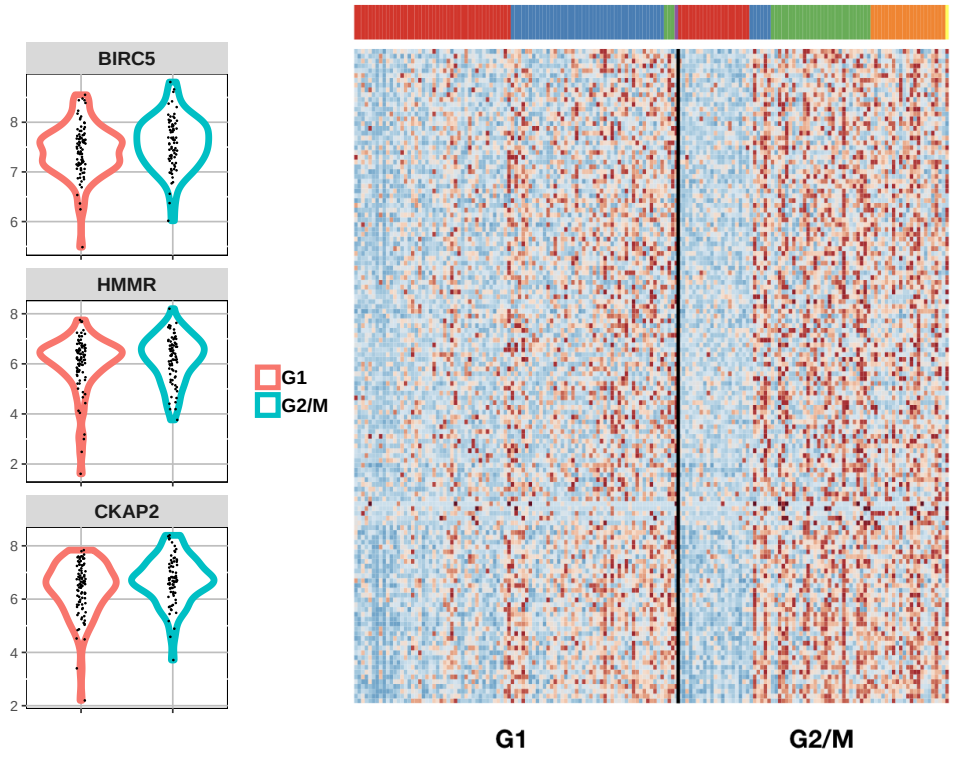

Fig 1: Genes involved in cell-cycle that are identified by scDDboost, but not standard approaches, as differentially distributed between cell-cycle phases G1 and G2/M in human embryonic stem cells. Density estimates on the left show expression data (log2 scale) of three genes identified by scDDboost at 5% FDR, but not similarly identified by other approaches. Prior studies have shown that the expression of BIRC5, HMMR, and CKAP2 is dependent on the phase of cell-cycle, suggesting that these subtle shifts are not false positives. Heatmap (right) shows these three genes among 145 other cell-cycle genes (GO:0007049).

Numerical experiments on both synthetic and published scRNA-seq data bear out the incidental finding in Figure 1, that `scDDboost` has sensitivity for detecting subtle distribution changes. In these experiments we take advantage of `splatter` for generating synthetic data (Zappia, Phipson and Oshlack, 2017) as well as the compendium of scRNA-seq data available through `conquer` (Soneson and Robinson, 2017). Additional numerical experiments show a relationship between `scDDboost` findings and more mechanistic attempts to parameterize transcriptional activation (Delmans and Hemberg, 2016). Finally, we establish first-order asymptotic results for the methodology.

On manuscript organization, we present the modeling and methodology elements in Section 2, numerical experiments in Section 3, asymptotic analysis in Section 4, and a discussion in Section 5. We relegate some details to an appendix and many others to a Supplementary Material document.

2. Modeling.

2.1. *Data structure, sampling model, and parameters.* In modeling scRNASeq data, we imagine that each cell c falls into one of $K > 1$ classes, which we think of as subtypes or subpopulations of cells. For notation, $z_c = k$ means that cell c happens to be of subtype k, with the vector $z = (z_c)$ recording the states of all sampled cells. Knowledge of this class structure prior to measurement is not required, as it will be inferred as necessary from available genomic data. We expect that cells arise from multiple experimental conditions, such as by treatment-control status or some other factors measured at the cell level, but we present our development for the special case of two conditions. Notationally, $y = (y_c)$ records the experimental condition, say $y_c = 1$ or $y_c = 2$. Let's say condition j measures $n_j = \sum_c 1[y_c = j]$ cells, and in total we have $n = n_1 + n_2$ cells in the analysis. The examples in Section 3 involve hundreds to thousands of cells. Further let

$$t_k^j = t_k^j(y, z) = \sum_c 1[y_c = j, z_c = k] \tag{1}$$

denote the number of cells of subtype k in condition j; we infer something about these counts using genome-wide data. As for molecular data, the normalized expression of gene g in cell c, say $X_{g,c}$, is one entry in a typically large GENES by CELLS data matrix X. Thus, the data structure entails an expression matrix X, a treatment label vector y, and a vector z of latent subtype labels.

We treat subtype counts in the two conditions, $t^1 = (t_1^1, t_2^1, \cdots, t_K^1)$ and $t^2 = (t_1^2, t_2^2, \cdots, t_K^2)$, as independent multinomial vectors, reflecting the experimental design. Explicitly,

$$t^1|y \sim \text{Multinomial}_K(n_1, \phi) \quad \text{and} \quad t^2|y \sim \text{Multinomial}_K(n_2, \psi) \tag{2}$$

for probability vectors $\phi = (\phi_1, \phi_2, \cdots, \phi_K)$ and $\psi = (\psi_1, \psi_2, \cdots, \psi_K)$ that characterize the populations of cells from which the n observed cells are sampled. This follows from the more basic sampling model: $P(z_c = k|y_c = 1) = \phi_k$ and $P(z_c = k|y_c = 2) = \psi_k$.

Our working hypothesis, referred to as the *compositional model*, is that any differences in the distribution of expression $X_{g,c}$ between $y_c = 1$ and $y_c = 2$ (i.e., any condition effects) are attributable to differences between the conditions in the underlying composition of cell types; i.e., owing to $\phi \neq \psi$. We reckon that cells of any given subtype k will present data according to a distribution reflecting technical and biological variation specific to that class of cells, regardless of the condition the cell finds itself in. Some care is needed in this, as an overly broad cell subtype (e.g., *epithelial cells*) could have further subtypes that show differential response to some treatment, for example, and so cellular condition (treatment) would then affect the distribution of expression data within the subtype, which is contrary to our working hypothesis. Were that the case, we could have refined the subtype definition to allow a greater number of population classes K in order to mitigate the problem of within-subtype heterogeneity. A risk in this approach is that K could approach n, as if every cell were its own subtype. We find, however, that data sets often encountered do not display this theoretical phenomenon when considering a broad class of within-subtype expression distributions. We revisit the issue in Section 4, but for now we proceed assuming that cellular condition affects the composition of subtypes but not the distribution of expression within a subtype.

Within the compositional model, let $f_{g,k}$ denote the sampling distribution of expression measurement $X_{g,c}$ assuming that cell c is from subtype k. Then for the two cellular conditions, and at some expression level x, the marginal distributions over subtypes are finite mixtures:

$$f_g^1(x) = \sum_{k=1}^{K} \phi_k f_{g,k}(x) \quad \text{and} \quad f_g^2(x) = \sum_{k=1}^{K} \psi_k f_{g,k}(x).$$

In other words, $X_{g,c}|[y_c = j] \sim f_g^j$ and $X_{g,c}|[z_c = k, y_c = j] \sim f_{g,k}$.

We say that gene g is *differentially distributed*, denote DD_g and indicated $f_g^1 \neq f_g^2$, if $f_g^1(x) \neq f_g^2(x)$ for some x, and otherwise it is equivalently distributed (ED_g). Motivated by findings from bulk RNAseq data analysis, we further set each $f_{g,k}$ to have a a negative-binomial form, say with mean $\mu_{g,k}$ and shape parameter σ_g; e.g. Leng et al. (2013), Anders and Huber (2010), and Love, Huber and Anders (2014). This choice is effective in our numerical experiments though it is not critical to the modeling formulation. The use of mixtures per gene has proven useful in related model-based approaches (e.g., Finak *et al.* 2015; McDavid *et al.* 2016; Huang *et al.* 2018). Our perspective is that genome-wide data may usefully inform the mixing proportions.

We seek methodology to prioritize genes for evidence of DD_g. Interestingly, even if we have evidence for condition effects on the subtype frequencies, it does not follow that a given gene will have $f_g^1 \neq f_g^2$; that depends on whether or not the subtypes show the right pattern of *differential expression* at g, to use the standard terminology from bulk RNAseq. For example, if two subtypes have different frequencies between the two conditions ($\phi_1 \neq \psi_1$ and $\phi_2 \neq \psi_2$) but the same aggregate frequency ($\phi_1 + \phi_2 = \psi_1 + \psi_2$), and also if $\mu_{g,1} = \mu_{g,2}$ then, other things being equal, $f_g^1 = f_g^2$ even though $\phi \neq \psi$. The fact is so central that we emphasize:

Key issue: A gene that does not distinguish two subtypes will also not distinguish the cellular conditions if those subtypes appear in the same aggregate frequency in the two conditions, regardless of changes in the individual subtype frequencies.

We formalize this issue in order that our methodology has the necessary functionality. To do so, first consider the parameter space $\Theta = \{\theta = (\phi, \psi, \mu, \sigma)\}$, where $\phi = (\phi_1, \phi_2, \cdots, \phi_K)$ and $\psi = (\psi_1, \psi_2, \cdots, \psi_K)$ are as before, where $\mu = \{\mu_{g,k}\}$ holds all the subtype-and-gene-specific expected values, and where $\sigma = \{\sigma_g\}$ holds all the gene-specific negative-binomial shape parameters. Critical to our construction are special subsets of Θ corresponding to partitions of the K cell subtypes. A single partition, say π, is a set of mutually exclusive and exhaustive blocks, b, say, each a subset of $\{1, 2, \cdots, K\}$, and we write $\pi = \{b\}$. Of course, the set Π containing all partitions π of $\{1, 2, \cdots, K\}$ has cardinality that grows rapidly with K. We carry along an example involving $K = 7$ cell types, and one three-block partition taken from the set of 877 possible partitions of $\{1, 2, \cdots, 7\}$ (Figure 1).

For any partition $\pi = \{b\}$, consider aggregate subtype frequencies

$$\Phi_b = \sum_{k \in b} \phi_k \quad \text{and} \quad \Psi_b = \sum_{k \in b} \psi_k,$$

and extend the notation, allowing vectors $\Phi_\pi = \{\Phi_b : b \in \pi\}$ and similarly for Ψ_π. Recall the partial ordering of partitions based on refinement, and note that as long as π is not the most refined partition (every cell type its own block), then the mapping from (ϕ, ψ) to (Φ_π, Ψ_π) is many-to-one. Further, define sets

$$A_\pi = \{\theta \in \Theta : \ \Phi_b = \Psi_b \, \forall b \in \pi\}. \tag{3}$$

and

$$M_{g,\pi} = \{\theta \in \Theta : \ \mu_{g,k} = \mu_{g,k'} \iff k, k' \in b, b \in \pi\}. \tag{4}$$

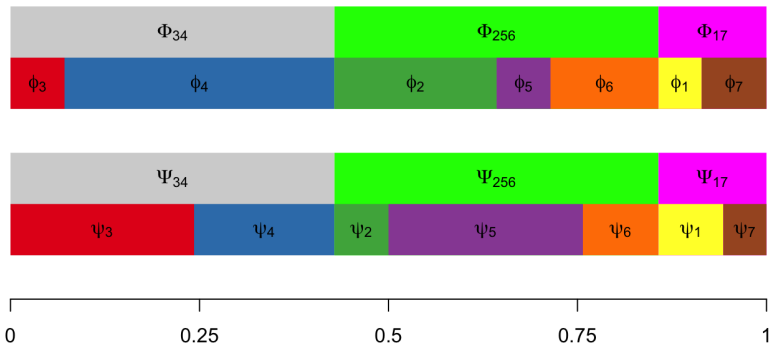

Fig 2: Proportions of $K = 7$ cellular subtypes in different conditions. Aggregated proportions of subtypes 3 and 4, subtypes 2, 5, and 6, and subtypes 1, and 7 remain same across conditions, while individual subtype frequencies change. Depending on the changes in average expression among subtypes, these frequency changes may or may not induce changes between two conditions in the marginal distribution of some gene's expression.

Under A_π there are constraints on cell subtype frequencies; under $M_{g,\pi}$ there is equivalence in the gene-level distribution of expression between certain subtypes. These sets are precisely the structures needed to address differential distribution DD_g (and it complement, equivalent distribution, ED_g) at a given gene g, since:

THEOREM 1. *Let $C_{g,\pi} = A_\pi \cap M_{g,\pi}$. For partitions $\pi_1 \neq \pi_2$, $C_{g,\pi_1} \cap C_{g,\pi_2} = \emptyset$. Further, at any gene g, equivalent distribution is*

$$\mathrm{ED}_g = \bigcup_{\pi \in \Pi} C_{g,\pi}.$$

With additional probability structure on the parameter space, we immediately obtain from Theorem 1 a formula for local false discovery rates:

$$1 - P(\mathrm{DD}_g | X, y) = P(\mathrm{ED}_g | X, y) = \sum_{\pi \in \Pi} P\left(A_\pi \cap M_{g,\pi} | X, y\right). \tag{5}$$

Such local false discovery rates are important empirical Bayesian statistics in large-scale testing (e.g., Efron, 2007; Muralidharan, 2010; Newton *et al.* 2004). For example, the conditional false discovery rate of a list of genes is the arithmetic mean of the associated local false discovery rates. The

partition representation guides construction of a prior distribution (Section 2.3) and a model-based method (Section 2.2) for scoring differential distribution. Setting the stage, Figure 2 shows the dependency structure of the proposed compositional model and the partition-reliant prior specification.

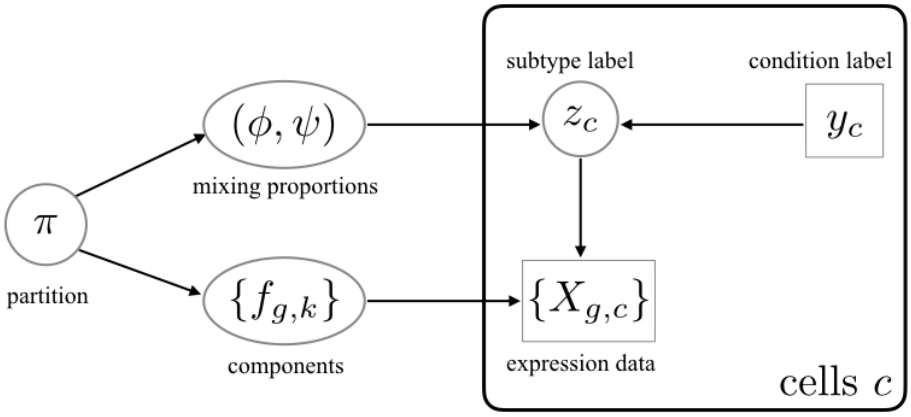

Fig 3: Directed acyclic graph structure of compositional model and partition-reliant prior. The plate on the right side indicates i.i.d. copies over cells c, conditionally on mixing proportions and mixing components. Observed data are indicated in rectangles/squares, and unobserved variables are in circles/ovals.

Key to computing the gene-specific local false discovery rate $P(\text{ED}_g|X, y)$ is evaluating probabilities $P\left(A_\pi \cap M_{g,\pi}|X, y\right)$ for any subtype partition π and gene g. The dependence structure (Figure 2) implies a useful reduction of this quantity, at least conditionally upon subtype labels $z = (z_c)$.

THEOREM 2. $P\left(A_\pi \cap M_{g,\pi}|X, y, z\right) = P\left(A_\pi|y, z\right)\ P\left(M_{g,\pi}|X, z\right).$

In what follows, we develop the modeling and computational elements necessary to efficiently evaluate inference summaries (5) taking advantage of Theorems 1 and 2. Roughly, the methodological idea is that subtype labels z have relatively low uncertainty, and may be estimated from genome-wide clustering of cells in the absence of condition information y (up to an arbitrary label permutation). The modest bit of uncertainty in z we handle through a computationally efficient randomized clustering scheme. Theorem 2 indicates that our computational task then separates into two parts

given z. On one hand, cell subtype frequencies combine with condition labels to give $P(A_\pi|y,z)$. Then gene-level data locally drive the posterior probabilities $P\left(M_{g,\pi}|X,z\right)$ that measure differential expression between subtypes. Essentially, the model provides a specific form of information sharing between genes that leverages the compositional structure of single-cell data in order to sharpen our assessments of between-condition expression changes.

2.2. *Method structure and clustering.* We leverage the extensive research on how to cluster cells into subtypes using scRNA-seq data: for example, SC3 (Kiselev et al., 2017), CIDR (Lin, Troup and Ho, 2017), and ZIFA (Pierson and Yau, 2015). We propose clustering on the full set of profiles in a way that is blind to the condition label vector y, in order to have as many cells as possible to inform the subtype structure. We investigated several clustering schemes in numerical experiments and allow flexibility in this choice within the scDDboost software. Associating clusters with subtype labels $\hat{z}_c$ estimates the actual subtypes z_c, and prepares us to use Theorems 1 and 2 in order to compute separate posterior probabilities $P(A_\pi|y,\hat{z})$ and $P(M_{g,\pi}|X,\hat{z})$ that are necessary for scoring differential distribution. The first probability concerns patterns of cell counts over subtypes in the two conditions, and has a convenient closed form within the double-Dirichlet model (Section 2.3). The second probability concerns patterns of changes in expected expression levels among subtypes, and this is also conveniently computed for negative-binomial counts using EBSeq (Leng et al., 2013). Algorithm 1 summarizes how these elements combine to get the posterior probability of differential distribution per gene, conditional on an estimate of the subtype labels.

We invoke $K-$medoids (Kaufman and Rousseeuw, 1987) as the default clustering method in scDDboost, and customize the cell-cell distance by integrating two measures. The first assembles gene-level information by cluster-based-similarity partitioning (Strehl and Ghosh, 2003). Separately at each gene, modal clustering (Dahl (2009) and Appendix B) partitions the cells, and then we define dissimilarity between cells as the mahattan distance of those gene specific partition labels. A second measure defines dissimilarity by one minus the Pearson correlation between cells, which is computationally inexpensive, less sensitive to outliers than Euclidean distance, and effective at detecting cellular clusters in scRNA-seq (Kim et al., 2018). We combine the two measures by a weighted average, with $w_C = \frac{\sigma_P}{\sigma_C+\sigma_P}$ and $w_P = 1 - w_C$. where $w_C, \sigma_C, w_P, \sigma_P$ are the weights and standard deviations of cluster-based distance and Pearson-correlation dis-

Algorithm 1 scDDBoost-core

Input:

genes by cells expression data matrix $X = (X_{g,c})$

cell condition labels $y = (y_c)$

cell subtype labels (estimated) $\hat{z} = (\hat{z}_c)$

Output: posterior probabilities of differential distribution from estimated subtypes

procedure scDDBoost-core($X, y, \hat{z}$)
number of cell subtypes $K = \text{length}(\text{unique}(\hat{z}))$
  subtype differential expression: $\forall g, \pi$ compute $P(M_{g,\pi}|X, \hat{z})$ using `EBSeq`
  cell frequency changes: $\forall \pi$ compute $P(A_\pi|y, \hat{z})$ using Double Dirichlet model
  posterior probability: $\forall g$, $P(\text{ED}_g|X, y, \hat{z}) \leftarrow \sum_\pi P(M_{g,\pi}|X, \hat{z})\, P(A_\pi|y, \hat{z})$
  return $\forall g$, $P(\text{DD}_g|X, y, \hat{z}) = 1 - P(\text{ED}_g|X, y, \hat{z})$
end procedure

tance, respectively. The final distance matrix is denoted $D = \left(d_{i,j}\right)$.

Any clustering method entails classification errors, and so $\hat{z}_c \neq z_c$ for some cells. To mitigate the effects of this uncertainty, scDDboost averages output probabilities from scDDboost-core over randomized clusterings $\hat{z}^*$. These are not uniformly random, but rather are generated by applying $K-$medoids to a randomized distance matrix $D^* = \left(d_{i,j}/w_{i,j}\right)$, where $w_{i,j}$ are non-negative weights $w_{i,j} = (e_i + e_j)$, and where (e_i) are independent and identically Gamma distributed deviates with shape $\hat{a}/2$ and rate $\hat{a}$, and where $\hat{a}$ is estimated from D. (Thus $w_{i,j}$ is Gamma$(\hat{a}, \hat{a})$ and has unit mean.) The distribution of clusterings induced by this simple computational scheme approximates a Bayesian posterior analysis, as we argue in the Appendix, where we also present pseudo-code for the resulting scDDboost Algorithm 2. Averaging over results from randomized clusterings gives additional stability to the posterior probability statistics.

Computations become more intensive the larger is the number K of cell subtypes. Version 1.0 of `scDDboost` is restricted to $K \leq 9$. Further, taking K to be too large may inflate the false positive rate (Supplementary Figure S8). To mitigate these issues, the approach taken in `scDDboost` is to set K using the validity score (Ray and Turi, 2000), which measures changes in within-cluster sum of squares as we increase K. Our specific implementation is in Supplementary Material Section 2.2.4.

2.3. $P(A_\pi|y, z)$. We introduce the Double Dirichlet Mixture (DDM), which is the partition-reliant prior $p(\phi, \psi)$ indicated in Figure 2, in order to derive an explicit formula for $P(A_\pi|y, z)$. We lose no generality here

by defining $A_\pi = \{(\phi, \psi) : \Phi_b = \Psi_b \; \forall b \in \pi\}$, rather than as a subset of the full parameter space as in (3). Each A_π is closed and convex subset of the product space holding all possible pairs of length-K probability vectors.

We propose a spike-slab-style mixture prior with the following form:

$$p(\phi, \psi) = \sum_{\pi \in \Pi} \omega_\pi \, p_\pi(\phi, \psi). \tag{6}$$

Each mixture component $p_\pi(\phi, \psi)$ has support A_π; the mixing proportions ω_π are positive constants summing to one. To specify component p_π, notice that on A_π there is a 1-1 correspondence between pairs (ϕ, ψ) and parameter states:

$$\{(\tilde{\phi}_b, \tilde{\psi}_b, \Phi_b), \; \forall b \in \pi\}, \tag{7}$$

where

$$\tilde{\phi}_b = \frac{\phi_b}{\Phi_b}, \quad \tilde{\psi}_b = \frac{\psi_b}{\Psi_b}, \quad \text{and} \quad \Phi_b = \sum_{k \in b} \phi_k = \sum_{k \in b} \psi_k = \Psi_b.$$

For example, $\tilde{\phi}_b$ is a vector of conditional probabilities for each subtype given that a cell from the first condition is one of the subtypes in b.

We introduce hyperparameters $\alpha_k^1, \alpha_k^2 > 0$ for each subtype k, and set $\beta_b = \sum_{k \in b} \left(\alpha_k^1 + \alpha_k^2\right)$ for any possible block b. Extending notation, let α_b^j be the vector of α_k^j for $k \in b$, β_π be the vector of β_b for $b \in \pi$, ϕ_b and ψ_b be vectors of ϕ_k and ψ_k, respectively, for $k \in b$, and Φ_π and Ψ_π be the vectors of Φ_b and Ψ_b for $b \in \pi$. The proposed double-Dirichlet component p_π is determined in the transformed scale by assuming $\Psi_\pi = \Phi_\pi$ and further:

$$\begin{aligned} \Phi_\pi &\sim \text{Dirichet}_{N(\pi)}[\beta_\pi] \\ \tilde{\phi}_b &\sim \text{Dirichlet}_{N(b)}[\alpha_b^1] \qquad \forall b \in \pi \\ \tilde{\psi}_b &\sim \text{Dirichlet}_{N(b)}[\alpha_b^2] \qquad \forall b \in \pi \end{aligned} \tag{8}$$

where $N(\pi)$ is the number of blocks in π and $N(b)$ is the number of subtypes in b, and where all random vectors in (8) are mutually independent. Mixing over π as in (6), we write $(\phi, \psi) \sim \text{DDM}\left[\omega = (\omega_\pi), \alpha^1 = (\alpha_k^1), \alpha^2 = (\alpha_k^2)\right]$.

We record some properties of the component distributions p_π:

Property 1: In $p_\pi(\phi, \psi)$, ψ and ϕ are dependent, unless π is the null partition in which all subtypes constitute a single block.

Property 2: With $k \in b$, marginal means are:

$$E_\pi(\phi_k) = \frac{\alpha_k^1}{\sum_{k' \in b} \alpha_{k'}^1} \frac{\beta_b}{\sum_{b' \in \pi} \beta_{b'}} \quad \text{and} \quad E_\pi(\psi_k) = \frac{\alpha_k^2}{\sum_{k' \in b} \alpha_{k'}^2} \frac{\beta_b}{\sum_{b' \in \pi} \beta_{b'}}.$$

Recall from (1) the vectors t^1 and t^2 holding counts of cells in each subtype in each condition, computed from y and z. Relative to a block $b \in \pi$, let $t_b^j = \sum_{k \in b} t_k^j$, for cell conditions $j = 1, 2$, and, let t_π^j be the vector of these counts over $b \in \pi$. The following properties refer to marginal distributions in which (ϕ, ψ) have been integrated out of the joint distribution involving (2) and the component p_π.

Property 3: t^1 and t^2 are conditionally independent given y, t_π^1 and t_π^2.

Property 4: For $j = 1, 2$,

$$p_\pi(t^j | t_\pi^j, y) = \prod_{b \in \pi} \left\{ \left[\frac{\Gamma(t_b^j + 1)}{\prod_{k \in b} \Gamma(t_k^j + 1)} \right] \left[\frac{\Gamma(\sum_{k \in b} \alpha_k^j)}{\prod_{k \in b} \Gamma(\alpha_k^j)} \right] \left[\frac{\prod_{k \in b} \Gamma(\alpha_k^j + t_k^j)}{\Gamma(t_b^j + \sum_{k \in b} \alpha_k^j))} \right] \right\}$$

Property 5:

$$p_\pi(t_\pi^1, t_\pi^2 | y) = \left[\frac{\Gamma(n_1 + 1)\Gamma(n_2 + 1)}{\prod_{b \in \pi} \Gamma(t_b^1 + 1)\Gamma(t_b^2 + 1)} \right] \left[\frac{\Gamma(\sum_{b \in \pi} \beta_b)}{\prod_{b \in \pi} \Gamma(\beta_b)} \right] \left[\frac{\prod_{b \in \pi} \Gamma(\beta_b + t_b^1 + t_b^2)}{\Gamma(n_1 + n_2 + \sum_{b \in \pi} \beta_b)} \right].$$

Let's look at some special cases to dissect this result.

Case 1. If π has a single block equal to the entire set of cell types $\{1, 2, \cdots, K\}$, then $t_b^j = n_j$ for both $j = 1, 2$, and Property 5 reduces, correctly, to $p_\pi(t_\pi^1, t_\pi^2 | y) = 1$. Further,

$$p_\pi(t^j | t_\pi^j, y) = \left[\frac{\Gamma(n_j + 1)}{\Gamma(n_j + \sum_{k=1}^K \alpha_k^j)} \right] \left[\frac{\Gamma(\sum_{k=1}^K \alpha_k^j)}{\prod_{k=1}^K \Gamma(\alpha_k^j)} \right] \left[\prod_{k=1}^K \frac{\Gamma(\alpha_k^j + t_k^j)}{\Gamma(t_k^j + 1)} \right]$$

which is the well-known Dirichlet-multinomial predictive distribution for counts t^j (Wagner and Taudes, 1986). E.g, taking $\alpha_k^j = 1$ for all types k we get the uniform distribution

$$p_\pi(t^j | t_\pi^j, y) = \frac{\Gamma(n_j + 1)\Gamma(K)}{\Gamma(n_j + K)}.$$

Case 2. At the opposite extreme, π has one block b for each class k, so $\phi = \psi$. Then $p_\pi(t^j | t_\pi^j, y) = 1$, and further, writing $b = k$,

$$p_\pi(t_\pi^1, t_\pi^2 | y) = \left[\frac{\Gamma(n_1 + 1)\Gamma(n_2 + 1)}{\prod_{k=1}^K \Gamma(t_k^1 + 1)\Gamma(t_k^2 + 1)} \right] \left[\frac{\Gamma(\sum_{k=1}^K \beta_k)}{\prod_{k=1}^K \Gamma(\beta_k)} \right] \left[\frac{\prod_{k=1}^K \Gamma(\beta_k + t_k^1 + t_k^2)}{\Gamma(n_1 + n_2 + \beta_k)} \right].$$

which corresponds to Dirichlet-multinomial predictive distribution for counts $t^1 + t^2$ since t^1 and t^2 are identical distributed given (ϕ, ψ) in this case.

The properties above are useful in establishing:

THEOREM 3. *The DDM model is conjugate to multinomial sampling of t^1 and t^2:*

$$(\phi, \psi)|y, z \sim \text{DDM}\left[\omega^{\text{post}} = (\omega_\pi^{\text{post}}), \alpha^1 + t^1, \alpha^2 + t^2\right]$$

where

$$\omega_\pi^{\text{post}} \propto p_\pi(t^1|t_\pi^1, y)\, p_\pi(t^2|t_\pi^2, y)\, p_\pi(t_\pi^1, t_\pi^2|y)\, \omega_\pi. \tag{9}$$

The target probability $P(A_\pi|y, z)$ is an integral of the posterior distribution in Theorem 3. To evaluate it, we need to contend with the fact that sets $\{A_\pi : \pi \in \Pi\}$ are not disjoint. Relevant overlaps have to do with partition refinement. Recall that a partition π^r is a refinement of a partition π^c if for any $b \in \pi^c$ there exists $s \subset \pi^r$ such that $\underset{b' \in s}{\cup} b' = b$. We say π^c coarsens π^r when π^r refines π^c. Any partition both refines and coarsens itself, as a trivial case. Generally, refinements increase the number of blocks. If subtype frequency vectors (ϕ, ψ) satisfy the constraints in A_{π^r} then they also satisfy the constraints of any π^c that coarsens π^r: i.e., $A_{\pi^r} \subset A_{\pi^c}$. Refinements reduce the dimension of allowable parameter states. For the double-Dirichlet component distributions P_π, we find:

Property 6: For two partitions $\tilde{\pi}$ and π,

$$P_{\tilde{\pi}}(A_\pi|y, z) = \begin{cases} 1 & \text{if } \tilde{\pi} \text{ refines } \pi \\ 0 & \text{otherwise} \end{cases}$$

This supports the main finding of this section:

$$P(A_\pi|y, z) = \sum_{\tilde{\pi} \in \Pi} \omega_{\tilde{\pi}}^{\text{post}}\, 1[\tilde{\pi} \text{ refines } \pi]. \tag{10}$$

2.4. $P(M_{g,\pi}|X, z)$. Here we leverage well-established modeling tools for transcript analysis. On $M_{g,\pi}$, cells within the same subtype, say all c such that $z_c = k$, have $X_{g,c}|z_c = k \sim$ Negative Binomial$(\sigma_g, \mu_{g,k})$, we use the Beta prior from EBSeq (Leng et al., 2013) and the posterior probability is calculated from

$$P(M_{g,\pi}|X, z) = \frac{P(M_{g,\pi}, X, z)}{\sum_{\pi \in \Pi} P(M_{g,\pi}, X, z)}$$

where

$$\begin{aligned} P(M_{g,\pi}, X, z) &= P(M_{g,\pi})P(X, z|M_{g,\pi}) \\ &= P(M_{g,\pi})\prod_{b\in\pi} P(X_{g,c}, z_c \in b|\mu_{g,i} = \mu_{g,j}, \forall i, j \in b) \end{aligned}$$

Marginal density of $M_{g,\pi}$ are assumed to be shared by all genes and we estimate hyper-parameters through first-order EM algorithm. (Details at supplementary material 2.2.1)

3. Numerical experiments.

3.1. *Synthetic data.* `splatter` is an effective simulation system for generating sythetic scRNA-seq data (Zappia, Phipson and Oshlack, 2017). We utilized this software (version 1.2.0) to generate data for which the DD status of genes is known, thereby allowing us to measure operating characteristics of `scDDboost`. Specifically, we entertained 12 different parameter settings for a two-group comparison on 17421 genes with 10% DD genes. Settings varied according to the number of subtypes K, the subtype frequency profiles (ϕ, ψ), as well as the `splatter`-specific parameters θ and γ controlling location and scale characteristics of expression levels. These settings were chosen to cover a range of scenarios we might expect to see in practice. Two replicate data sets were simulated under each parameter setting. Further details are provided in Supplementary Material Section 3.1.

Figures 4 and 5 summarize the true positive rate and false discovery rate of `scDDboost` compared to three other methodologies: `MAST (1.4.0)`, `scDD (1.2.0)`, and `DESeq2 (1.18.1)`. The proposed `scDDboost` exhibits very good operating characteristics in this study, as it controls the false discovery rate in all cases while also returning a higher rate of true positive genes in most case. ***maybe something else not yet shown; other FDR levels, other scenarios????***

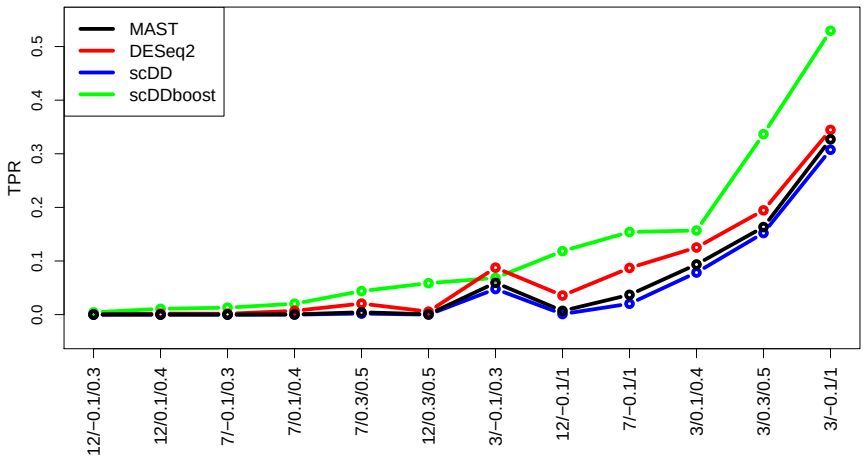

Fig 4: True positive rate, averaged over replicates, of four DD detection methods in 12 synthetic-data scenarios. Scenarios are labeled for $K/\theta/\gamma$ and ranked by scDDboost values. Each method is targeting a 5% false discovery rate (FDR).

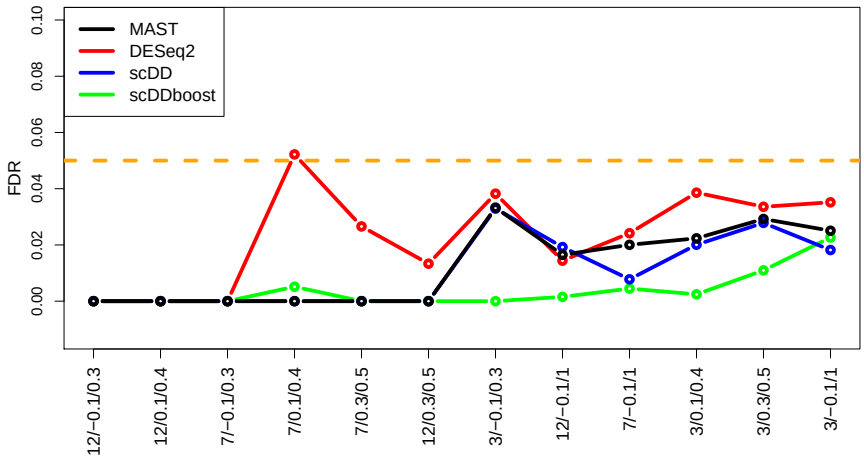

Fig 5: Empirical false discovery rate of methods in scenarios from Figure 4

3.2. *Empirical study.* We applied `scDDboost` to a collection of previously published data sets that are recorded at `conquer` (Soneson and Robinson, 2017). Though not knowing the truly DD genes, we can examine how `scDDboost` output compares to output from other standard methods. We selected 12 data sets from `conquer` representing different species and experimental settings and involving hundreds to thousands of cells. Supplementary Table 1 provides details. Figure 6 compares methods in terms of the size of the reported list of DD genes at the 5% FDR target level. We see a consistent improvement of gene yeild compared to the other tested methods. For reference, one of these data sets (GSE64016) happens to be the data behind Figure 1, where we know from other information that some of the uniquely identified genes are likely not to be false positives.

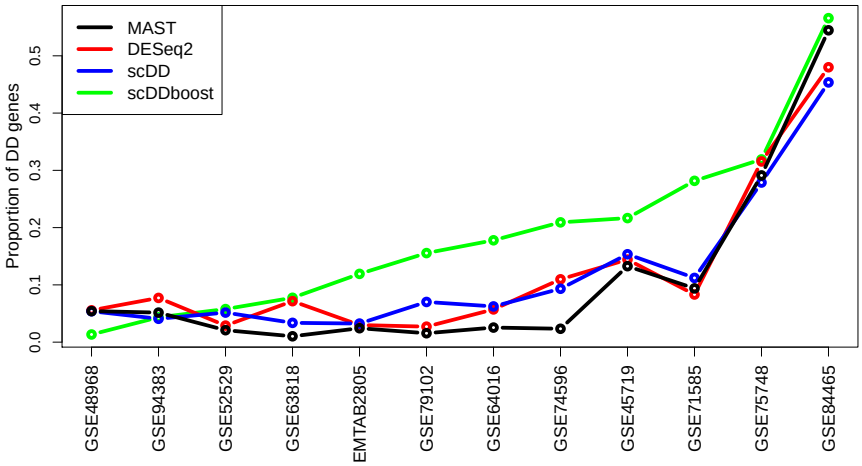

Fig 6: Proportion of DD genes at 5% threshold with respect to total number of genes identified by each method. Ranked by scDDboost list size

To check that the high yeild of `scDDboost` is not associated with an increased rate of false positives, we applied it to a series of random splits of single-condition data sets (Appendix Table 2). Figure 7 confirms a very low call rate in cases where no changes in distribution are expected. ***other methods use BH adjusted pvalue***

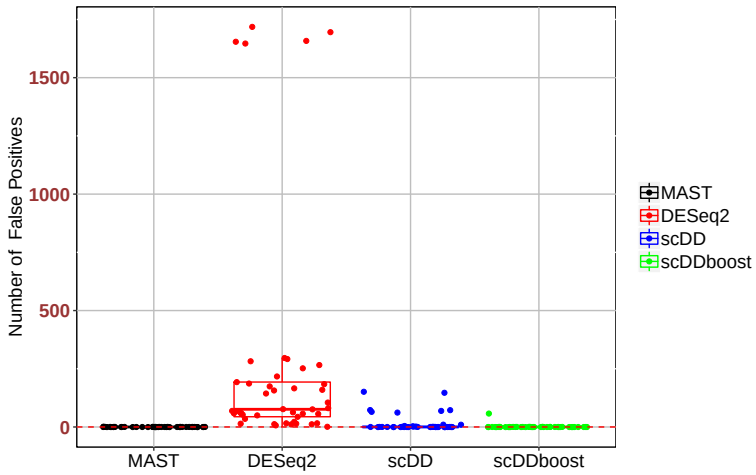

Fig 7: False positive counts by several methods on null 5 random splits of 9 single condition data sets from Appendix Table 2

We conjecture that `scDDboost` gains power through its novel approach to *borrowing strength*; i.e., that the genomic data are providing information about cell subtypes and mixing proportions, leaving gene-level data to handle gene-specific mixture components. One way to drill into this idea is presented in Figure 8. In three data sets, we report on genes that are identified by `scDDboost`; those in red are uniquely identified by `scDDboost`; the blue ones are also identifed by MAST, scDD, or DESeq2. The vertical axis reports a model-based cluster size associated with each gene. By virtue of the `EBseq` analysis inside `scDDboost`, any gene may be assigned to a cluster of genes that all have the same highest-probability pattern of equality/inequality of means across the subtypes. Say $\hat{\pi}_g = \text{argmax}_\pi P(M_{g,\pi}|\hat{z}, X)$.

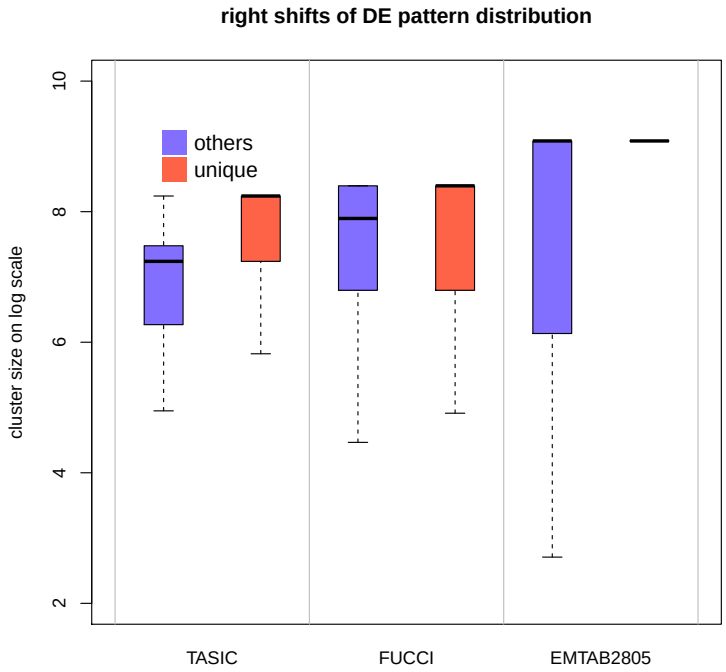

Fig 8: boxplot of log transformed cluster size of DD genes of two categories. The red one are corresponding to the genes that have also been identified by other methods. The blue one are corresponding to the genes uniquely identified by scDDboost. We grouped genes to the same cluster if they share the same map DE pattern between subtypes inferred from EBSeq We use the number of genes within a cluster as the size for that cluster. We observe a right shift for between blue and right, which indicates our uniquely identified DD genes are more likely to associate with DE patterns of larger sizes Datasets used for demonstration from left to right are GSE71585, GSE64016, EMTAB2805. Here we used 0.01 threshold as it is more friendly for graphical demostration, 0.05 cases are in the supplementary fig *** (still have right shift, but hard to see from box plot as most of genes are associated with the largest cluster, the right shift did not change the quantile too much)

Number of subtypes K is a crucial factor controlling the accuracy of our modeling. Too small K may end up in an underfit such that cells within same subtype can still be very different, such heterogeneity within the subtype introduced by insufficient number of clusters tends to make our posterior inference of the DE patterns between subtypes more favorable for all equivalent expressed and reducing the power of scDDboost. Too big K may end up in an overfit such that two subtypes can be very similar,

given we have fixed number of samples (cells), allowing more clusters will introduce more patterns (both for mean expression change and proportion change) to infer and thus reduce our estimation efficiency on those patterns. Overestimating K in scDDboost may losing FDR control (supplementary Figure 8).

***Benefits from share information from other genes in EBSeq. ***

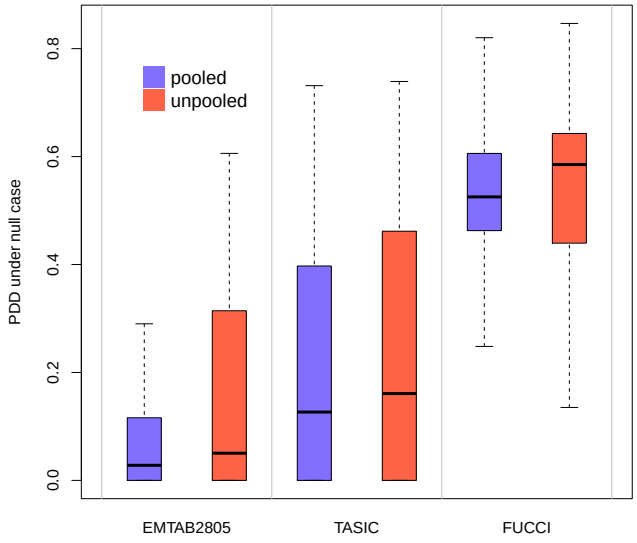

Fig 9: PDD of the same datasets used in Figure 7, we have types of PDD, the purple box is the PDD calculated by using the proportion of DE patterns inferred from the whole genome. The magenta box is the PDD using the trivial proportion of DE pattern (i.e. all DE patterns between subtypes have same probabilities), basically we do not borrow information from other genes. In the null case, we observed lower PDD when using information from the whole genome and this is consistent with the underlying ED structure. Recall in Figure 7, genes uniquely identified by scDDboost tend to associate with DE pattern with large size. From the null case such behavior could make our PDD more correct

3.3. *Bursting.* D3E(Delmans and Hemberg, 2016) is a distributional method that can identify bursting parameters of transcripts. Rate of promoter activation, rate of promoter inactivation and the rate of transcription when the promoter is in the active state are estimated by D3E. We investigate DD genes identified by scDDboost and their change of those three parameters

on dataset GSE71585

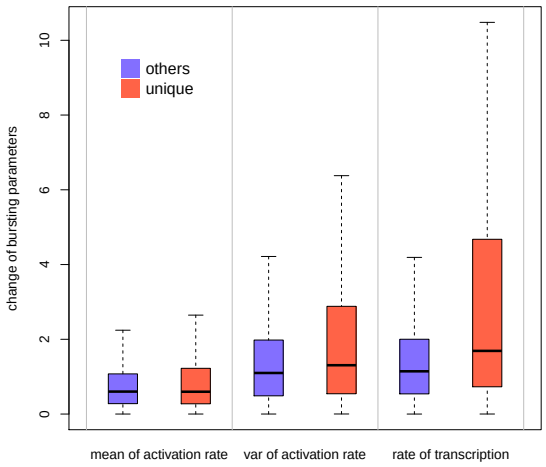

Fig 10: Absolute value of log fold change of bursting parameters estimated by D3E methods across conditions (DEC vs EC). Purple box refers to those DD genes identified by other methods, magenta box refers to DD genes uniquely identified by scDDboost. In the D3E model, it assumes the counts are sampled from a poisson-beta model. Specifically, $x \sim \text{Poi}(x|\gamma * p), p \sim \text{Beta}(\alpha, \beta)$, In D3E paper, it call α the rate of promoter activation, β the rate of promoter inactivation and γ the rate of transcripts when the promoter is in the active state. Such poisson-beta model should be the stationary distribution of the transcriptional bursting model. For each condition each gene D3E would estimate a triplet (α, β, γ). Instead of directly looking at the change of α, β. I looked at the change of mean and variance of Beta prior for each condition. i.e. the change for mean and var for p from the poisson-beta model. I think it would be more natural to look at p, as transcripts are more directly related to p than α and β, $p = 0$ would mean the promoter is off and $p = 1$ means the promoter is on, so p is more like a measure for how likely a promoter is on, I call the mean and var of p as the mean and var of activation rate. Then I plot the absolute value of log fold change of mean, var activation rate and rate of transcripts γ. Those uniquely identified genes by scDDboost are more consistent with the estimations from D3E since they are corresponding to more significant change of those bursting parameters

We observed that genes uniquely identified by scDDboost are associated with more significant changes between estimated bursting parameters compare to genes commonly identified by all four RNAseq methods. From previous study (Fucci data Figure 1) our mixture model could be sensitive to the subtle changes of genes expressions. Using the bursting

model, such shifts of distributions we identified are informative for the changes of bursting parameters and probably are not falsely discovered.

4. Asymptotics of the Double Dirichlet Mixture. Summary statistics $P(A_\pi|y,z)$, from Section 2.3, are amenable to a first-order asymptotic analysis that provides further insight into DDM model behavior. The fact that support sets A_π for component distributions $p_\pi(\phi,\psi)$ are not disjoint becomes an important issue. Consider distinct partitions π_1 and π_2 of subtypes $\{1,2,\cdots,K\}$, and recall that $N(\pi)$ counts the number of blocks in partition π. In case π_2 refines π_1, then $N(\pi_1) < N(\pi_2)$, and we also know that $A_{\pi_2} \subset A_{\pi_1}$, since refinement imposes additional constraints on the pair (ϕ,ψ) of probability vectors. If the data-generating state $(\phi,\psi) \in A_{\pi_2}$, one might ask how posterior probability mass tends to be allocated among the other mixture components whose support sets also contain this state. The question is addressed by the following:

THEOREM 4. *Let π_1 and π_2 denote two partitions for which $N(\pi_1) < N(\pi_2)$ and $A_{\pi_1} \cap A_{\pi_2}$ is non-empty. Let $(\phi,\psi) \in A_{\pi_1} \cap A_{\pi_2}$ denote the data generating state for subtype labels $z_1, z_2, \cdots, z_n$ given i.i.d. Bernoulli condition labels $y_1, y_2, \cdots, y_n$, and recall the posterior mixing proportions ω_π^{post} from equation (9) with hyper-parameters $\alpha_i^j \geq 1$ for $i = 1,\cdots,K$, $j = 1,2$. Then*

$$\frac{\omega_{\pi_1}^{\text{post}}}{\omega_{\pi_2}^{\text{post}}} \longrightarrow_{a.s.} 0 \qquad \text{as } n \longrightarrow \infty.$$

Essentially, mixing mass is transferred to components associated with the most refined partition consistent with a given parameter state. To be precise, let $H(\phi,\psi) = \{\pi : (\phi,\psi) \in A_\pi\}$ record all the partitions associated with one state. Typically, there is a most refined partition, say $\pi^* = \pi^*(\phi,\psi)$, such that

$$A_{\pi^*} = \bigcap_{\pi \in H(\phi,\psi)} A_\pi. \tag{11}$$

This always happens when $K \leq 3$. In Supplementary Material Section 4 we characterize the exceptional set of states where (11) does not hold. Notably, if (11) does hold for state (ϕ,ψ), then for any $\pi \in H(\phi,\psi)$, using Theorem 4 and (10), we have

$$P(A_\pi|y_1,\cdots,y_n;z_1,\cdots z_n) \longrightarrow_{a.s.} 1 \quad \text{as } n \longrightarrow \infty.$$

5. Discussion. We have presented scDDboost, a compositional model for detecting differential distributed genes from scRNA-seq data. To account for the over-dispersion and multi-modality of single-cell data, scDDboost modeled transcripts as mixture distributed. Unlike previous invented methods (e.g. Deseq2, MAST and scDD), which conducts genewise DD test in an isolated manner. scDDboost make whole genome information shared at gene level by further assuming the mixture distribution of transcripts is a mixture over the subtypes of cells. Another advantage of scDDboost is its' flexibility to allow user specified clustering methods of cells, with more and more studies of the scRNA-seq data, there will be more accurate distance matrix between cells, which will yield better estimation of subtypes and inference of DD genes. We combine estimations of changes of subtypes' proportions across conditions and changes of mean expressions across subtypes to infer distributional changes of transcripts. To estimate changes of subtypes' proportions across conditions, we use empirical Bayes and developed a double Dirichlet prior distribution. We invented a random weighting scheme that stabilize our DD inference as well as approximating the results as if we have done a fully bayesian clustering analysis based on Dirichlet prior. We demonstrated that scDDboost outperforms existing approaches in simulation and tends to be more powerful than existing methods on a wide range of public available empirical datasets.

One limitation of scDDboost is that current EBseq inference of the DE patterns is computationally not feasible for big number of subtypes. Given the noise level among the single cell data and especially if we want to identify DD genes among conditions containing thousands of cells, allowing a big number of subtypes would make cells under same subtype more homogeneous and result in a more accurate estimations for the distribution of transcripts. Further research is needed for acceleration of EBseq, one direction is to reduce the calculation on those patterns that would have small posterior probabilities.

References.

Anders, S. and Huber, W. (2010). Differential expression analysis for sequence count data. *Genome Biology* **11** R106–R106.

Bacher, R. and Kendziorski, C. (2016). Design and computational analysis of single-cell RNA-sequencing experiments. *Genome Biology* **17** 63. .

Dahl, D. B. (2009). Modal clustering in a class of product partition models. *Bayesian Anal.* **4** 243–264.

Delmans, M. and Hemberg, M. (2016). Discrete distributional differential expression (D3E) - a tool for gene expression analysis of single-cell RNA-seq data. *BMC Bioin-*

formatics **17** 110. .

Dominguez, D., Tsai, Y.-H., Gomez, N., Jha, D. K., Davis, I. and Wang, Z. (2016). A high-resolution transcriptome map of cell cycle reveals novel connections between periodic genes and cancer. *Cell Research* **26** 946 EP -.

Finak, G., McDavid, A., Yajima, M., Deng, J., Gersuk, V., Shalek, A. K., Slichter, C. K., Miller, H. W., McElrath, M. J., Prlic, M., Linsley, P. S. and Gottardo, R. (2015). MAST: a flexible statistical framework for assessing transcriptional changes and characterizing heterogeneity in single-cell RNA sequencing data. *Genome Biology* **16** 278. .

Kaufman, L. and Rousseeuw, P. (1987). *Clustering by means of medoids*. North-Holland.

Kim, T., Chen, I. R., Lin, Y., Wang, A. Y.-Y., Yang, J. Y. H. and Yang, P. (2018). Impact of similarity metrics on single-cell RNA-seq data clustering. *Briefings in Bioinformatics* bby076.

Kiselev, V. Y., Kirschner, K., Schaub, M. T., Andrews, T., Yiu, A., Chandra, T., Natarajan, K. N., Reik, W., Barahona, M., Green, A. R. and Hemberg, M. (2017). SC3: consensus clustering of single-cell RNA-seq data. *Nature Methods* **14** 483 EP -.

Korthauer, K. D., Chu, L.-F., Newton, M. A., Li, Y., Thomson, J., Stewart, R. and Kendziorski, C. (2016). A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. *Genome Biology* **17** 222. .

Leng, N., Dawson, J. A., Thomson, J. A., Ruotti, V., Rissman, A. I., Smits, B. M. G., Haag, J. D., Gould, M. N., Stewart, R. M. and Kendziorski, C. (2013). EBSeq: an empirical Bayes hierarchical model for inference in RNA-seq experiments. *Bioinformatics* **29** 1035-1043.

Li, F. and Altieri, D. C. (1999). The Cancer Antiapoptosis Mouse Survivin Gene. *Cancer Research* **59** 3143.

Lin, P., Troup, M. and Ho, J. W. K. (2017). CIDR: Ultrafast and accurate clustering through imputation for single-cell RNA-seq data. *Genome Biology* **18** 59. .

Love, M. I., Huber, W. and Anders, S. (2014). Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biology* **15** 550. .

Marioni, J. C. and Arendt, D. (2017). How Single-Cell Genomics Is Changing Evolutionary and Developmental Biology. *Annual Review of Cell and Developmental Biology* **33** 537-553. PMID: 28813177.

Navin, N. E. (2015). The first five years of single-cell cancer genomics and beyond. *Genome Research* **25** 1499–1507.

Nawy, T. (2013). Single-cell sequencing. *Nature Methods* **11** 18 EP -.

Papalexi, E. and Satija, R. (2017). Single-cell RNA sequencing to explore immune cell heterogeneity. *Nature Reviews Immunology* **18** 35 EP -.

Pierson, E. and Yau, C. (2015). ZIFA: Dimensionality reduction for zero-inflated single-cell gene expression analysis. *Genome Biology* **16** 241. .

Ray, S. and Turi, R. H. (2000). Determination of Number of Clusters in K-Means Clustering and Application in Colour Image Segmentation.

Sohr, S. and Engeland, K. (2008). RHAMM is differentially expressed in the cell cycle and downregulated by the tumor suppressor p53. *Cell Cycle* **7** 3448–3460.

Soneson, C. and Robinson, M. D. (2017). Bias, Robustness And Scalability In Differential Expression Analysis Of Single-Cell RNA-Seq Data. *bioRxiv*.

Strehl, A. and Ghosh, J. (2003). Cluster Ensembles — a Knowledge Reuse Framework for Combining Multiple Partitions. *J. Mach. Learn. Res.* **3** 583–617.

Wagner, U. and Taudes, A. (1986). A Multivariate Polya Model of Brand Choice and Purchase Incidence. *Marketing Science* **5** 219–244.

Yakowitz, S. J. and Spragins, J. D. (1968). On the Identifiability of Finite Mixtures. **39** 209–214.

Zappia, L., Phipson, B. and Oshlack, A. (2017). Splatter: simulation of single-cell RNA sequencing data. *Genome Biology* **18** 174. .

Appendix.

Proof of Theorem 1. If $\theta \in \bigcup_{\pi\in\Pi}\left[A_\pi \cap M_{g,\pi}\right]$, then there exists a partition π for which $\theta \in A_\pi$ and $\theta \in M_{g,\pi}$. By construction

$$f_g^1(x) = \sum_{k=1}^{K} \phi_k f_{g,k}(x) = \sum_{b\in\pi}\sum_{k\in b} \phi_k f_{g,k}(x) = \sum_{b\in\pi} \Phi_b f_{g,k^*(b)}(x),$$

where $k^*(b)$ indexes any component in b, since all components in that block have the same component distribution owing to constraint $M_{g,\pi}$. Continuing, using the constraint $\theta \in A_\pi$,

$$f_g^1(x) = \sum_{b\in\pi} \Psi_b f_{g,k^*(b)}(x) = f_g^2(x) \qquad \forall x.$$

That is, $\theta \in \mathrm{ED}_g$.

If $\theta \in \mathrm{ED}_g$, then $f_g^1(x) = f_g^2(x)$ for all x. Noting that both are mixtures over the same set of components $\{f_{g,k}\}$, let $\{h_{g,l} : l = 1, 2, \cdots, L\}$ be the set of distinct components over this set, and so

$$f_g^1(x) = \sum_{k=1}^{k} \phi_k f_{g,k}(x) = \sum_{l=1}^{L} c_{g,l}(\phi) h_{g,l}(x) = \sum_{l=1}^{L} c_{g,l}(\psi) h_{g,l}(x) = f_g^2(x)$$

where

$$c_{g,l}(\phi) = \sum_{k=1}^{K} \phi_k 1[f_{g,k} = h_{g,l}] \qquad c_{g,l}(\psi) = \sum_{k=1}^{K} \psi_k 1[f_{g,k} = h_{g,l}]. \tag{12}$$

Finite mixtures of distinct negative binomial components are identifiable (Proposition 5 from Yakowitz and Spragins (1968)), and so the equality of f_g^1 and f_g^2 implies $c_{g,l}(\phi) = c_{g,l}(\psi)$ for all $l = 1, 2, \cdots, L$. Identifying the partition blocks $b_l = \{k : f_{g,k} = h_{g,l}\}$, and the partition $\tilde{\pi} = \{b_l\}$, we find $\theta \in A_{\tilde{\pi}} \cap M_{g,\tilde{\pi}}$. The accumulated probabilities in (12) correspond to $\Phi_{\tilde{\pi}}$ and $\Psi_{\tilde{\pi}}$, which are equal on $A_{\tilde{\pi}}$.

Randomizing distances for approximate posterior inference. One way to frame the subtype problem is to suppose that subtype labels $z = (z_i)$ satisfy $z = f(\Delta)$, where $\Delta = \left(\delta_{i,j}\right)$ is a $n \times n$ matrix holding *true*, unobservable distances, such as $\delta_{i,j}$ between cells i and j, and that f is some assignment function, like the one induced by the $K-$medoids algorithm. Then posterior uncertainty in z would follow directly from posterior uncertainty

in Δ. On one hand, we could proceed via formal Bayesian analysis, say under a simple conjugate prior in which $1/\delta_{i,j} \sim \text{Gamma}(a_0, d_0)$, for hyperparameters a_0 and d_0, and in which the observed distance $d_{i,j}|\delta_{i,j} \sim \text{Gamma}(a_1, a_1/\delta_{i,j})$. This would assure that $\delta_{i,j}$ is the expectation of $d_{i,j}$, with shape parameter a_1 affecting variation of measured distances about their expected values. Not accounting for any constraints imposed by both D and Δ being distance matrices, we would have the posterior distribution $1/\delta_{i,j}|D \sim \text{Gamma}(a_0 + a_1, d_0 + a_1 d_{i,j})$. For any threshold $c > 0$, we would find

$$P\left(\delta_{i,j} \leq c|D\right) = P\left(U \geq \frac{d_0 + a_1 d_{i,j}}{c(a_0 + a_1)}\right) \tag{13}$$

where $U \sim \text{Gamma}(a_0 + a_1, a_0 + a_1)$

Alternatively, we could form randomized distances $d^*_{i,j} = d_{i,j}/w_{i,j}$ where $w_{i,j}$ is the analyst-supplied random weight distributed as $\text{Gamma}(\hat{a}, \hat{a})$ as in Section 2.2. Notice that

$$P(d^*_{i,j} \leq c|D) = P(w_{i,j} > d_{i,j}/c|D)$$

which is also an upper tail probability for a unit-mean Gamma deviate with shape and rate equal to $\hat{a}$. Comparing to (13), by setting $\hat{a}$ to equal $a_0 + a_1$, and if a_0 and d_0 are relatively small, we find

$$P(d^*_{i,j} \leq c|D) \approx P(\delta_{i,j} \leq c|D).$$

In other words, the randomized distance procedure is providing approximate posterior draws of the underlying distance matrix. In spite of limitations of this procedure for full Bayesian inference, it provides an elementary scheme to account for uncertainty in subtype allocations. Numerical experiments in Supplementary Material make comparisons to a full, Dirichlet-process-based, posterior analysis.

Pseudo-code.

Address of the First and Second authors
Usually a few lines long
E-mail: ma79@wisc.edu
kendzior@biostat.wisc.edu

Address of the Third author
Usually a few lines long
Usually a few lines long
E-mail: newton@stat.wisc.edu
URL: http://www.foo.com

Algorithm 2 scDDboost

Input:

genes by cells expression data matrix $X = (X_{g,c})$

cell condition labels $y = (y_c)$

number of cell subtypes K

number of randomized clusterings n_r

Output: posterior probabilities of differential distribution

procedure scDDboost(X, y, K, n_r)

distance matrix: $D = \text{dist}(X) \leftarrow$ pairwise distances between cells (columns of X)

hyper-parameters $(a_0, a_1, d_0) \leftarrow \text{hyper}(D)$. Set $\hat{a} = a_0 + a_1$.

repeat

Gamma noise vector: e, with components $\sim \text{Gamma}(\hat{a}/2, \hat{a})$

randomized distance matrix: $D^* \leftarrow D/(e\mathbf{1}^T + \mathbf{1}e^T)$

$\hat{z}^* \leftarrow K-\text{medoids}(D^*)$

$P^* \leftarrow$ scDDboost-core$(X, y, \hat{z}^*)$

until n_r randomized distance matrices

return $\forall \text{genes}\, g$, $P(\text{DD}_g | X, y) = \frac{1}{n_r} \sum_{D^*} P^*_g$

end procedure