VENICE: A NEW ALGORITHM FOR FINDING MARKER GENES IN SINGLE-CELL TRANSCRIPTOMIC DATA

Hy Vuong BioTuring Inc
Thao Truong BioTuring Inc
Tan Phan BioTuring Inc
Son Pham BioTuring Inc

November 16, 2020

ABSTRACT

Most widely used tools for finding marker genes in single cell data (SeuratT/NegBinom/Poisson, CellRanger, EdgeR, limmatrend) use a conventional definition of *differentially expressed genes*: genes with different mean expression values. However, in single-cell data, a cell population can be a mixture of many cell types/cell states, hence the mean expression of genes cannot represent the whole population. In addition, these tools assume that gene expression of a population belongs to a specific family of distribution. This assumption is often violated in single-cell data. In this work, we define marker genes of a cell population as genes that can be used to distinguish cells in the population from cells in other populations. Besides log-fold change, we devise a new metric to classify genes into up-regulated, down-regulated, and transitional states. In a benchmark for finding up-regulated and down-regulated genes, our tool outperforms all compared methods, including Seurat, ROTS, scDD, edgeR, MAST, limma, normal t-test, Wilcoxon and Kolmogorov–Smirnov test. Our method is much faster than all compared methods, therefore, enables interactive analysis for large single-cell data sets in BioTuring Browser. Venice algorithm is available within Signac package: https://github.com/bioturing/signac [1]).

1 Introduction

Single-cell RNA-sequencing provides unprecedented insights for cell heterogeneity. At the same time, it yields many new formidable computational challenges. Finding marker genes is among such problems. Most widely used methods for finding marker genes (Seurat-wilcox/t/negbinom/poisson, CellRanger, EdgeR, limmatrend) rely on a conventional definition of *differentially expressed genes* from bulk RNA-seq: *genes with different means of expression values*.

However, single-cell data is very different to bulk RNA-seq data. Generally, a cell population can be a mixture of many cell types, or cell states, hence a single parameter (mean) cannot represent the whole population. For instance, given a population with three sub-populations A, B, C, gene g expresses at intermediate, low, and high levels, respectively. This expression patterns often appears in cells at a transitional state (population B) [1]. When this transitional population is compared against the remaining population (cells in A and C), which contain both up-regulated and down-regulated sub-populations, the mean expression value can not be distinguished. Hence, these *transitional genes* can not be revealed by normal *differentially-expressed gene* procedures. In addition, the assumption that gene expression of a population belongs to a specific family of distribution is strongly violated, therefore, the result can be inaccurate.

Some authors had recognized this problem, and incorporated non-parametric tests into their tools, e.g. SigEMD [2], and EMDomics [3], D3E [4] or using a mixture model, e.g. scDD [5]. However, those tools do not have a intuitive statistic like log-fold change to control the quality of the marker genes, and they are also very computational expensive.

In this work, we define marker genes of a cell population as the genes that can be use to distinguish the cells in the population from cells in other populations, and based on that idea, we introduce two simple statistics to quantify the

[1]Contact email: sonpham@bioturing.com

distinctiveness and to classify up/down-regulated or transitional marker genes. This method is also much faster than all compared methods, therefore, enables interactive analysis for large single-cell data sets (up to millions cells).

2 Method

We define marker genes of a cell population as genes that can be used to distinguish cells in the population from cells outside of the population. From this idea, we construct the ideal classifier that tries to identify the population of each cell given the expression level of each specific gene. We approximate the classifier's accuracy and use it as a metric to quantify the "marker quality" of the gene.

2.1 Classifier accuracy

Assume that we need to find the marker genes of a specific cell population, denoted as C_1. And we denote the remaining cell population as C_2.

For a gene g, let $A(g)$ be the accuracy of the classifier on identifying the population of the cells, given expression level of gene g for each cell. We can split our accuracy into two parts,

$$A(g) = \frac{1}{2}(A_1(g) + A_2(g)) \tag{1}$$

Where $A_1(g)$ is the accuracy of identifying the population of cells from C_1, and $A_2(g)$ is the accuracy of identifying the population of cells from C_2. In particular,

$$A_i(g) = E_{c \in C_i}[P(c \in C_i | g_c)] \tag{2}$$

Where g_c is the expression level of gene g in a random cell c, and $P(c \in C_i|g_c)$ is the our classifier predicted probability of cell c with gene expression g_c belonging to group C_i.

Using the Bayes' theorem, we have

$$P(c \in C_i \mid g_c) = \frac{P(g_c \mid c \in C_i)P(c \in C_i)}{P(g_c)} = \frac{P(g_c \mid c \in C_i)P(c \in C_i)}{P(g_c \mid c \in C_1)P(c \in C_1) + P(g_c \mid c \in C_2)P(c \in C_2)} \tag{3}$$

where $i \in \{1, 2\}$.

To avoid the biases toward either population, we simply assign $P(c \in C_1) = P(c \in C_2) = \frac{1}{2}$. And from (1), (2), and (3), we can deduce the accuracy of our ideal classifier,

$$A(g) = \frac{1}{2} \sum_{g_c} \frac{P(g_c \mid c \in C_1)^2 + P(g_c \mid c \in C_2)^2}{P(g_c \mid c \in C_1) + P(g_c \mid c \in C_2)} \tag{4}$$

However, $P(g_c \mid c \in C_i)$ is a latent parameter and the ideal classifier model needs to find its value exactly. As we cannot compute $A(g)$ directly, we may estimate $A(g)$ using the estimation of $P(g_c \mid c \in C_i)$ from our observed data,

$$P(g_c \mid c \in C_i) \approx \frac{\text{Number of cells with expression } g_c \text{ in group } C_i}{\text{Number of cells in group } C_i}$$

However, using above substitution to estimate $A(g)$, in many cases, can cause our estimation to have a very high variance. We try to alleviate the problem by grouping the expression level into k intervals: $G_1, G_2, ..., G_k$[1] and denote the new statistic as

$$A'(g) = \frac{1}{2} \sum_{i=1}^{k} \frac{P(g_c \in G_i \mid c \in C_1)^2 + P(g_c \in G_i \mid c \in C_2)^2}{P(g_c \in G_i \mid c \in C_1) + P(g_c \in G_i \mid c \in C_2)} \approx A(g) \tag{5}$$

Now, we can substitute the following estimation,

[1]The detail of our grouping strategy is discussed in the Supplementary

$$P(g_c \in G_i \mid c \in C_j) \approx \frac{\text{Number of cells of group } C_j \text{ that have expression level in interval } G_i}{\text{Number of cell in group } C_j} \triangleq D_{j,i}$$

And we have our final estimation of $A(g)$,

$$A''(g) = \frac{1}{2}\sum_{i=1}^{k} \frac{D_{1,i}^2 + D_{2,i}^2}{D_{1,i} + D_{2,i}} \approx A'(g) \approx A(g)$$

One of the unmentioned problems is that $A''(g)$ is a biased estimation of $A'(g)$. However, as the number of data points in single-cell studies is high, this bias is insignificant. Nonetheless, we provide a correction for the bias, which leads to better score consistency to handle populations with small numbers of cells. We denote the corrected value $A'''(g)$, which is defined in the Supplementary.

In our implementation, we rescaled value of $A'''(g)$ to $2A'''(g) - 1$, and called it dissimilarity score. The dissimilarity score can be negative due to bias correction.

2.2 Up-down score

Conventionally, we want to classify our differential expressed genes into up-regulated and down-regulated genes. The log-fold change provides a simple metric for the distinction. However, such distinction is only suitable for simple distribution families of gene expression. For genes with more complex distributions, e.g. multi-modal distributions, such distinction cannot always be made.

Instead we propose a different definition of up-regulated and down-regulated genes. We say that a gene is up-regulated in group 1 *iif* for every $p \in (0, 1)$, the p-quantile of the expression in the group 1 is higher than the p-quantile of the expression in the group 2 and vise versa for down regulated genes. For genes that do not fit into either cases, the p-quantile of the group 1 expression is greater compared to group 2 for $p \in P_+$, and smaller for $p \in P_-$. For those genes, we assign them the values ranged from -1 to 1 that represent the difference between the measure of P_+ and P_- and call them *transitional genes*.

Let Q_1, and Q_2 be the quantile function for the expression level of population C_1, and C_2, respectively.

Then, we have the measure of P_+ and P_-,

$$U = \mu(P_+) = \int_0^1 [Q_1(p) > Q_2(p)]\mathrm{d}p$$

$$D = \mu(P_-) = \int_0^1 [Q_1(p) < Q_2(p)]\mathrm{d}p$$

Where $[P] = \begin{cases} 1 & P \\ 0 & \neg P \end{cases}$.

Finally, we define our up-down score,

$$S = \frac{U - D}{U + D}$$

To complete our definition, we need define our quantile function. Given a sorted data set $A = (a_1, a_2, ..., a_n)$, with $a_i \leq a_{i+1} \forall i \in \{1, 2, ..., n-1\}$, we define our quantile function as

$$Q_A(p) = (1 - \gamma)x_i + \gamma x_{i+1}$$
$$\text{where } i = \lfloor np \rfloor, \gamma = np - i, a_0 = a_1.$$

2.3 P-value approximation

Our null hypothesis is that the probability that expression level $g_c \in G_i$ does not depend on which population that cell c comes from.

$$H_0 : \forall i \in \{1, 2, ..., k\}, P(g_c \in G_i \mid c \in C_1) = P(g_c \in G_i \mid c \in C_2)$$

To test the null hypothesis, for each gene g, we calculate our statistic $A''(g)$ and approximate its p-value using the following theorem (See more details in the Supplement).

Theorem (P-value theorem)**.** *Under null hypothesis, and fixed intervals G_i, we have*

$$2(2A''(g) - 1)\left(\frac{2n_1n_2}{n_1 + n_2} - 1\right) \xrightarrow{d} \chi^2_{k-1}$$

as the sample sizes n_1, n_2 of both populations tends to infinity.

Where $A \xrightarrow{d} B$ means the distribution of A converges to the distribution of B

We also implement a permutation test, which simulates the sampling process from the null sample, in order to estimate the p-value. The permutation test is more accurate in the data with very small cell population. Besides these extreme cases, both methods produce similar results (See the Supplement).

3 Results

Simulated Data We benchmark Venice precision and performance against 15 other methods using two simulated data sets.

- **Data set 1**: We use scDD, a Bayesian modeling framework, to characterize different expression patterns [6]. In particular, scDD simulates count data from mixtures of negative binomial distribution for two groups of 500 vs 500 cells. The simulated data set contains 10,000 genes, divided into 6 groups: 500 differential expression (DE), 500 differential modality (DM), 500 differential proportion (DP), 500 both differential modality and 500 different component means (DB), 4000 equivalent expression (EE), and 4000 equivalent proportions of cells belonging to each component (EP) (See Figure 1).

- **Data set 2**: To simulate data sets with small number of cells, we sub-sample the above data set to create a new set of 30 samples. Each sample contains two populations, one with 50 cells and the other with 100 cells. Each sample has 1000 differential distribution genes (250 genes for each type of differential distribution) and 3000 equivalent expression genes (1500 EE genes and 1500 EP genes).

scDD classifies differential-distributed genes into four different types (DE, DP, DM, and DB), which are visualized in Figure 1, and non-differential genes into 2 types depending on the modality of the gene expression distribution. EE represents genes with uni-modal distribution and EP represents genes with multi-modal distribution.

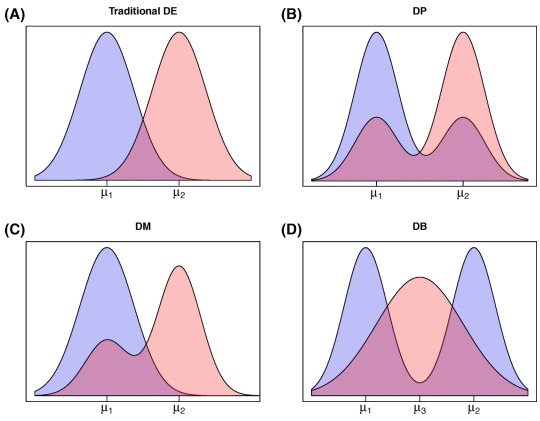

Figure 1: Four types of differential distribution in gene expression. (A) traditional differential expression (DE), (B) differential proportion within each mode (DP), (C) differential modality (DM), and (D) both differential modality and differential expression (DB).

3.1 Data set 1

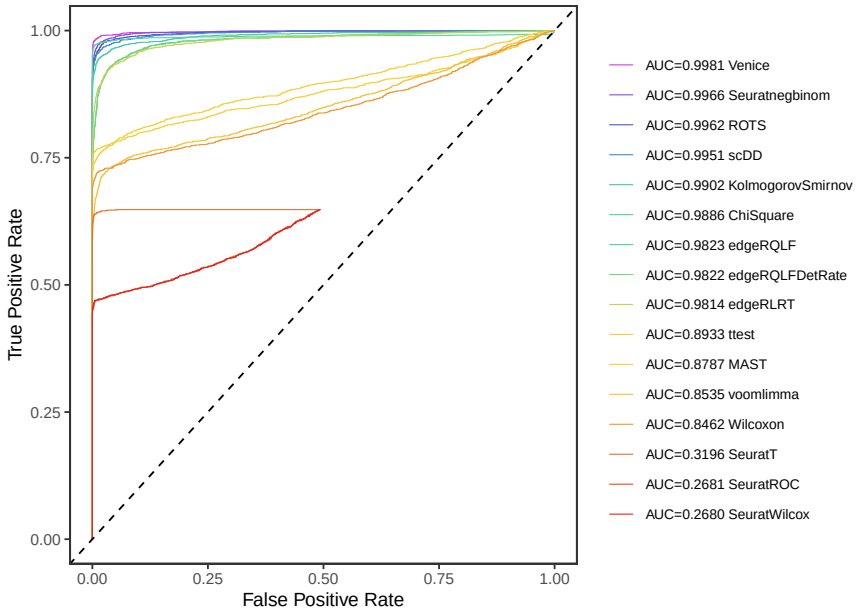

Figure 2: *ROC curves of 16 different methods on data set 1. Venice has the best AUC*

In the first data set, Venice outperforms all other tools in the benchmark in the AUC score. There is a significant discrepancy between 2 groups of tools. Venice, SeuratNegBinom, ROTS, scDD, KS test, edgeR have AUC scores greater than 0.98, while other tools have AUC scores smaller than 0.9. scDD provided two settings to run. We use the default one, which uses the KS test. The other setting, which uses permutation test for its the Bayes factor, is very computational expensive and doesn't produce better result in our test. However the KS test of scDD outperforms the standard KS test. SeuratNegBinom also performs really well despite using a simple distribution model. This may stem from the fact that the uni-modal genes are simulated from a negative binomial distribution.

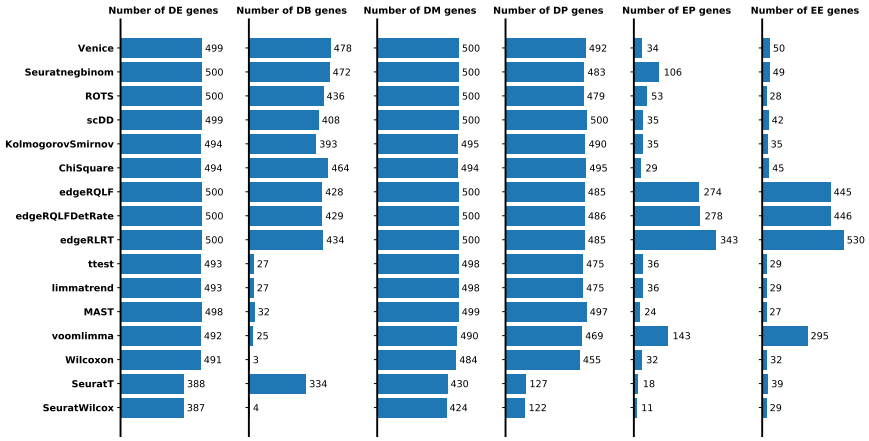

Figure 3: The number of genes in each group that each tool called using a false discovery rate cut-off of 0.05. As explained, DE, DB, DM, DP are true positive; EE, EP are false positive.

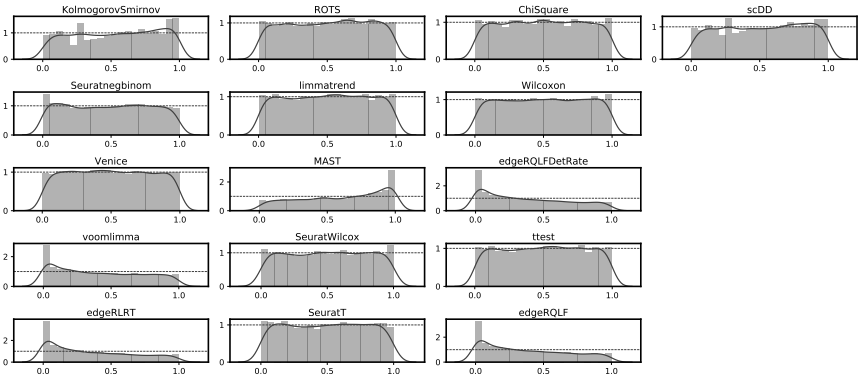

Figure 4: The p-value histogram of non-marker genes of each tool

When using the false discovery rate cut-off of 0.05, edgeR methods produce a high number of false positives. Venice and ROTS controls the false discovery rate best as shown in figure 3 and 4. Venice is the tool that detects the most DB genes, which can be the transitional genes.

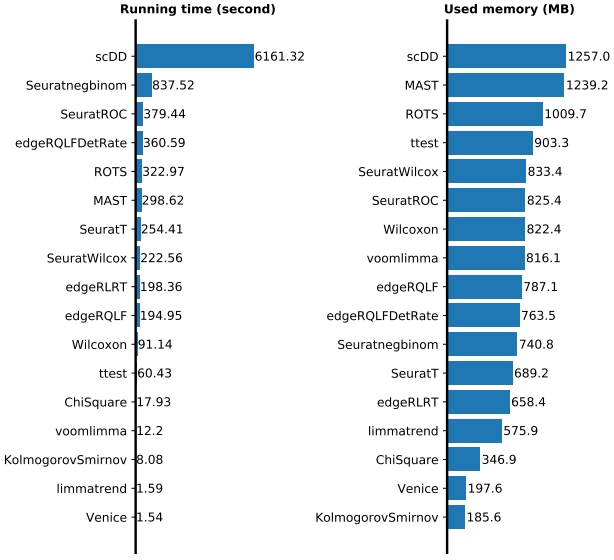

Figure 5: The running time and memory consumption of each tool

Amongst the tools with high AUC scores, only Venice and KS test run under 3 minutes on this data set, especially Venice only takes less than 2 seconds. Venice and KS test also use significantly less memory compared to other tools.

3.2 Data set 2

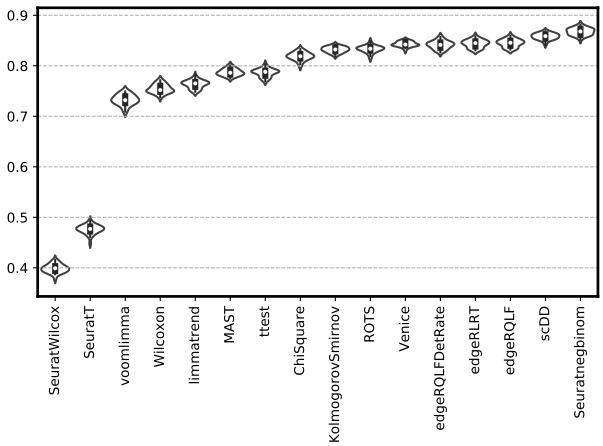

Figure 6: The average AUC of each tool on data set 2

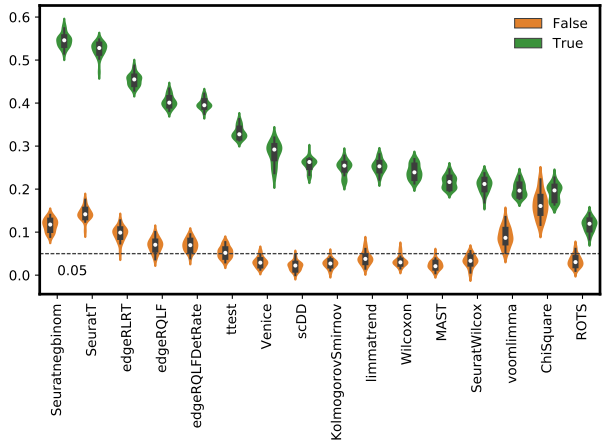

Figure 7: The average false/true positive rate of each tool on data set 2 with the false discovery rate cut-off of 0.05

In smaller data sets (which are uncommon for single-cell data), our method doesn't produce the best result. However, the tools with better AUC scores (edgeR, SeuratNegBinom) are assuming the negative binomial distribution. This assumption can improve the power of these tests, however, it causes biases in the p values. Consequentially, it invalidates the false discovery rates. (Figures 7 and 8).

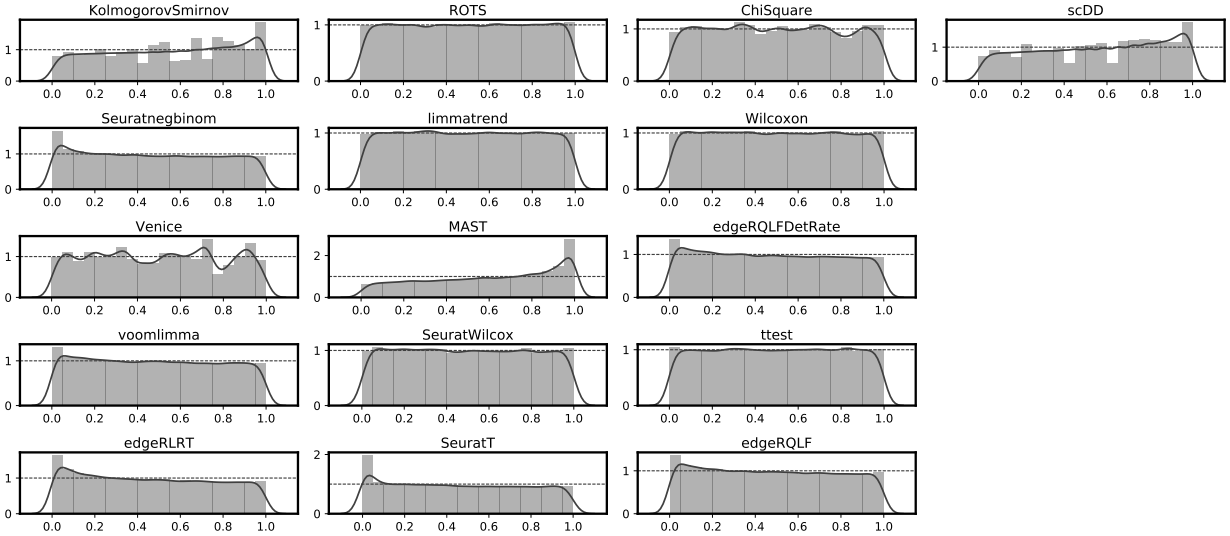

Figure 8

4 Supplementary

4.1 Grouping

4.1.1 Grouping strategy

First, we want to have about $k = \left(\frac{2n_1n_2}{n_1+n_2}\right)^{0.6}$ groups. Next, we want each group have a similar amount of cells and we don't want each group to have too few cells, specifically less than 10 cells. Therefore, we define $t = \max\left(\left\lfloor\frac{n_1+n_2}{k}\right\rfloor, 10\right)$ as our expected number of cells in a group .

In our grouping process, first we sort the cells increasingly by gene expression of a gene, we pick out the first t cells and put them into a group. However there the next cells may have the same expression level as the t^{th} cell. In that case, we also put those cell into that group. We continue our process until don't have enough cells to form a group. If the number of remaining cells is less than our minimal threshold (10 cells), we merge them into the previous group. Otherwise, we form a new group with our remaining cells.

4.1.2 Effects on the approximation

$$\sum_{g_c \in G_i} \frac{P(g_c \mid c \in C_1)^2 + P(g_c \mid c \in C_2)^2}{P(g_c \mid c \in C_1) + P(g_c \mid c \in C_2)} \geq \frac{P(g_c \in G_i \mid c \in C_1)^2 + P(g_c \in G_i \mid c \in C_2)^2}{P(g_c \in G_i \mid c \in C_1) + P(g_c \in G_i \mid c \in C_2)}$$

This inequality can be deduced from the Cauchy-Schwarz inequality in Engel's form $\sum \frac{a_i^2}{b_i} \geq \frac{(\sum a_i)^2}{\sum b_i}$

Hence, $A(g) \geq A'(g)$

Grouping reduces our predicting power, hence causes the accuracy to decrease. However, this is necessary, since there are not always enough data to estimate the expected gene expression for each possible expression level.

4.2 p-value

Our main theorem is

Theorem 4.1 (P-value theorem)**.** *Under null hypothesis, and fixed intervals G_i, we have*

$$2(2A''(g) - 1)\left(\frac{2n_1n_2}{n_1+n_2} - 1\right) \xrightarrow{d} \chi^2_{k-1}$$

as the sample sizes n_1, n_2 of both populations tends to infinity.

Coincidentally, our statistic $A''(g)$ is very similar to the Chi-square test statistic after a bit of transformation. Conveniently, to prove this theorem, we can use the same technique that shows Chi-square test statistic follows a chi-square distribution.

First, we needs a few definition.

First we denote $p_i = P(g_c \in G_i)$

Under the null hypothesis, it also means that $p_i = P(g_c \in G_i \mid c \in C_1) = P(g_c \in G_i \mid c \in C_2)$

Next, we denote $D_i = (D_{i,1}, D_{i,2}, ..., D_{i,k})$ as the observed count vector for population i. More specifically, $D_{i,j}$ is the number of cells of population i that have the expression level in interval G_j. We can model $D_1 \sim \text{Multi}(n_1, p)$, and $D_2 \sim \text{Multi}(n_2, p)$ under the null hypothesis.

And then, we denote the proportion vector $d_{x,i} = \frac{D_{x,i}}{n_x}$ and the average proportion vector $\hat{p_i} = \frac{1}{2}(d_{1,i} + d_{2,i})$

We derived an unbiased estimator S of $d_1 - d_2$ co-variance matrix.

$$S = \frac{1}{c}\begin{pmatrix} \hat{p}_1(1-\hat{p}_1) & -\hat{p}_1\hat{p}_2 & \dots & -\hat{p}_1\hat{p}_k \\ -\hat{p}_2\hat{p}_1 & \hat{p}_2(1-\hat{p}_2) & \dots & -\hat{p}_2\hat{p}_k \\ \vdots & \vdots & \ddots & \vdots \\ -\hat{p}_k\hat{p}_1 & -\hat{p}_k\hat{p}_2 & \dots & \hat{p}_k(1-\hat{p}_k) \end{pmatrix}$$

where $c = \dfrac{n_1 n_2}{n_1+n_2} - \dfrac{1}{2}$

We create S^* by removing the last row and column of S.

$$S^* = \frac{1}{c}\begin{pmatrix} \hat{p}_1(1-\hat{p}_1) & -\hat{p}_1\hat{p}_2 & \dots & -\hat{p}_1\hat{p}_{k-1} \\ -\hat{p}_2\hat{p}_1 & \hat{p}_2(1-\hat{p}_2) & \dots & -\hat{p}_2\hat{p}_{k-1} \\ \vdots & \vdots & \ddots & \vdots \\ -\hat{p}_{k-1}\hat{p}_1 & -\hat{p}_{k-1}\hat{p}_2 & \dots & \hat{p}_{k-1}(1-\hat{p}_{k-1}) \end{pmatrix}$$

Though S is not invertible, as the sum of each column/row is zero, S^* is invertible and its inverse can be found using the Sherman–Morrison formula.

Inverse of S^*

$$(S^*)^{-1} = c\begin{pmatrix} \frac{1}{\hat{p}_1}+\frac{1}{\hat{p}_k} & \frac{1}{\hat{p}_k} & \dots & \frac{1}{\hat{p}_k} \\ \frac{1}{\hat{p}_k} & \frac{1}{\hat{p}_2}+\frac{1}{\hat{p}_k} & \dots & \frac{1}{\hat{p}_k} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{\hat{p}_k} & \frac{1}{\hat{p}_k} & \dots & \frac{1}{\hat{p}_{k-1}}+\frac{1}{\hat{p}_k} \end{pmatrix}$$

We also denote d_i^* as the vector d_i with the last component removed.

In the following proves, we will ignore the cases when $d_{1,i} = d_{2,i} = 0$ for some i, because as $n \to \infty$, the probability of those cases happening become zero, hence the our theorem below is still hold without explicitly handle those cases.

Before proving the theorem, we need some lemmas. Note that we always assume the null hypothesis and fixed grouping intervals G_i.

Lemma 4.2. *We have* $(d_1^* - d_2^*)^T (S^*)^{-1}(d_1^* - d_2^*) = 4c(2A'' - 1)$

Proof. $(d_1^* - d_2^*)^T (S^*)^{-1}(d_1^* - d_2^*) = c\left(\frac{1}{\hat{p}_k}\left(\sum_{i=1}^{k-1} d_{1,i} - d_{2,i}\right)^2 + \sum_{i=1}^{k-1}\frac{(d_{1,i}-d_{2,i})^2}{\hat{p}_i}\right)$

Furthermore, we have that $\sum_{i=1}^{k-1} d_{1,i} - d_{2,i} = (1 - d_{1,k}) - (1 - d_{2,k}) = d_{2,k} - d_{1,k}$, since $\sum_{i=1}^{k} d_{x,i} = 1$.

Therefore, we have

$$(d_1^* - d_2^*)^T (S^*)^{-1}(d_1^* - d_2^*) = c\sum_{i=1}^{k}\frac{(d_{1,i}-d_{2,i})^2}{\hat{p}_i} = 2c\sum_{i=1}^{k}\frac{(d_{1,i}-d_{2,i})^2}{d_{1,i}+d_{2,i}}$$

$$= 2c\left(2\sum_{i=1}^{k}\frac{d_{1,i}^2+d_{2,i}^2}{d_{1,i}+d_{2,i}} - \sum_{i=1}^{k}\frac{(d_{1,i}+d_{2,i})^2}{d_{1,i}+d_{2,i}}\right) = 2c\left(4A'' - \sum_{i=1}^{k}(d_{1,i}+d_{2,i})\right)$$

$$= 2c(4A'' - 2) = 4c(2A'' - 1)$$

□

Lemma 4.3. *e have* $(d_1^* - d_2^*)^T (S^*)^{-1}(d_1^* - d_2^*) \xrightarrow{d} \chi^2_{k-1}$ *as* $n_1, n_2 \to \infty$

Proof. First we assume that $\lim_{n_1,n_2\to\infty} \frac{n_1}{n_2} = t$ (t can be ∞),

Using the central limit and Slutsky's theorem, we have

$$(S^*)^{-1/2}(d_1^* - d_2^*) \xrightarrow{p} \sqrt{c}(\Sigma^*)^{-1/2}(d_1^* - d_2^*) = \sqrt{c}(\Sigma^*)^{-1/2}(d_1^* - p^*) - \sqrt{c}(\Sigma^*)^{-1/2}(d_2^* - p^*)$$

$$\xrightarrow{p} \frac{\sqrt{n_1}}{\sqrt{1+t}}(\Sigma^*)^{-1/2}(d_1^* - p^*) - \frac{\sqrt{n_2}}{\sqrt{1+\frac{1}{t}}}(\Sigma^*)^{-1/2}(d_2^* - p^*)$$

$$\xrightarrow{d} N\left(0, \left(\frac{1}{1+t} + \frac{1}{1+\frac{1}{t}}\right) I_{k-1}\right) \xrightarrow{d} N(0, I_{k-1})$$

Therefore,

$$(d_1^* - d_2^*)^T (S^*)^{-1} (d_1^* - d_2^*) \xrightarrow{d} \chi^2_{k-1}$$

Hence,

$$2(2A'' - 1)\left(\frac{2n_1n_2}{n_1+n_2} - 1\right) \xrightarrow{d} \chi^2_{k-1}$$

We also need to handle the cases where $t = 0$ and $t = \infty$ to complete this proof.

Here we still have to assume hat $\frac{n_1}{n_2}$ has a limit (maybe infinite). We outline a proof that this assumption is not necessary in the following paragraph

From the Bolzano–Weierstrass theorem, we can show that every sequence of $\frac{n_1}{n_2}$ will contain a sub-sequence with a limit (infinite when the sequence is unbounded). Hence, if we only assume $(n_1, n_2) \to \infty$, then for every infinite sub-sequence of (n_1, n_2), we can always find an sub-sub-sequence such that $(S^*)^{-1/2}(d_1^* - d_2^*)$ converges to $N(0, I_{k-1})$. We can apply the lemma 4.4 to show that that $(S^*)^{-1/2}(d_1^* - d_2^*)$ converges to $N(0, I_{k-1})$ without the assumption that $\frac{n_1}{n_2}$ has a limit.

□

Lemma 4.4 (Convergence)**.** *Given a sequence (a_n) such that every infinite sub-sequence will have a sub-sequence that converges to a common limit L, then the sequence converges to the same limit L*

Proof. We will proof this lemma by contradiction.

Assume that the said sequence doesn't converge to L, then by negating the definition of limit, there will exist an $\epsilon > 0$, such that there is infinite of a_n, such that $|a_n - L| \geq \epsilon$. Then, we construct an infinite sub-sequence from these a_n, denoted (b_n). By the assumption, b_n has a sub-sub-sequence that converges to L, hence there exists b_n such that $|b_n - L| < \epsilon$, which contradicts the construction of b_n.

□

Here is the proof of theorem 4.1

Proof. From the lemma 4.2, we have $(d_1^* - d_2^*)^T (S^*)^{-1} (d_1^* - d_2^*) = 4c(2A'' - 1) = 2(2A''(g) - 1)\left(\frac{2n_1n_2}{n_1+n_2} - 1\right)$

Furthermore, we also have $(d_1^* - d_2^*)^T (S^*)^{-1} (d_1^* - d_2^*) \xrightarrow{d} \chi^2_{k-1}$ as $n_1, n_2 \to \infty$ from lemma 4.3.

Hence, $2(2A''(g) - 1)\left(\frac{2n_1n_2}{n_1+n_2} - 1\right) \xrightarrow{d} \chi^2_{k-1}$ □

About the fixed grouping assumption, we think it can be loosen up. If the grouping strategy converges to a fixed grouping where every group have a non-zero probability of having a cell when n_1 and n_2 go to infinity, then we think our theorem is still hold.

Though the theorem only provides the asymptotic distribution of the statistic, this distribution seems to approximate the statistic well in common cases.

4.3 bias correction

Our estimation of the accuracy is not an unbiased estimator due to grouping the expression level into G_i intervals and the A'' estimation of A'. Grouping may cause the accuracy to reduce ($A' \leq A$), while the A'' cause upward bias ($\mathrm{E}[A''] \geq \mathrm{E}[A']$). The later fact can be shown using the Jensen's inequality since A' is a convex function of $P(g_c \in G_i \mid c \in C_j)$.

Assume fixed grouping intervals G_i, D is a sufficient statistic for the distribution of A'. We can assume a estimator A'' of A' is a function of $D_{j,i}$. Consequently, $\mathrm{E}[A'']$ is a polynomial in $P(g_c \in G_i, \mid c \in C_j)$. On the contrary, A' cannot be represented as a polynomial in $P(g_c \in G_i, \mid c \in C_j)$. Hence there is no unbiased estimator for A'.

However, we can do a small correction that is empirically reducing the bias.

First we try to estimate the expected value using the Taylor's approximation of A'' at $D_{j,i} = P_{j,i} = P(g_c \in G_i \mid c \in C_j)$,

$$e = E[A''] - A' \approx \sum_{i=1}^{k} \frac{P_{1,i}P_{2,i}(n_1 P_{1,i}(1 - P_{2,i}) + n_2 P_{2,i}(1 - P_{1,i}))}{n_1 n_2 (P_{1,i} + P_{2,i})^3}$$

Then, we estimate our bias as,

$$e' = \sum_{i=1}^{k} \frac{D_{1,i}D_{2,i}(n_1 D_{1,i}(1 - D_{2,i}) + n_2 D_{2,i}(1 - D_{1,i}))}{n_1 n_2 (D_{1,i} + D_{2,i})^3} \approx e$$

Hence, we have the corrected estimator,

$$A''' = A'' - e'$$

In Venice package, we provide a parameter to turn on and off the correction; it is on by default.

5 Acknowledgements

The authors feel very grateful for all the supports from everyone in the BioTuring team, which can range from ideas, suggestions, testings, integration of Venice algorithm to BBrowser, to encouragements or even some pats on the shoulders during some moments when nothing works.

References

[1] Jihwan Park, Rojesh Shrestha, Chengxiang Qiu, Ayano Kondo, Shizheng Huang, Max Werth, Mingyao Li, Jonathan Barasch, and Katalin Suszták. Single-cell transcriptomics of the mouse kidney reveals potential cellular targets of kidney disease. *Science*, 360(6390):758–763, 2018.

[2] Tianyu Wang and Sheida Nabavi. Sigemd: A powerful method for differential gene expression analysis in single-cell rna sequencing data. *Methods*, 145:25–32, 2018.

[3] Sheida Nabavi, Daniel Schmolze, Mayinuer Maitituoheti, Sadhika Malladi, and Andrew H Beck. Emdomics: a robust and powerful method for the identification of genes differentially expressed between heterogeneous classes. *Bioinformatics*, 32(4):533–541, 2016.

[4] Mihails Delmans and Martin Hemberg. Discrete distributional differential expression (d 3 e)-a tool for gene expression analysis of single-cell rna-seq data. *BMC bioinformatics*, 17(1):110, 2016.

[5] Keegan D Korthauer, Li-Fang Chu, Michael A Newton, Yuan Li, James Thomson, Ron Stewart, and Christina Kendziorski. scdd: A statistical approach for identifying differential distributions in single-cell rna-seq experiments. *bioRxiv*, page 035501, 2015.

[6] Keegan D Korthauer, Li-Fang Chu, Michael A Newton, Yuan Li, James Thomson, Ron Stewart, and Christina Kendziorski. A statistical approach for identifying differential distributions in single-cell rna-seq experiments. *Genome biology*, 17(1):222, 2016.