Batch Effect Control in an Integrated Human Myeloid Transcriptional Atlas

Yidi Deng

Supervised by: Dr Kim-Anh Le Cao, Dr Jarny Choi

Collaborator: Dr Paul Angel, Prof Christine Wells

Abstract

iMac human myeloid cell atlas is an accessible online resource that can be used to visually explore myeloid cell's molecular phenotype and be benchmarked against broad aspects of human myeloid biology. Construction of such a comprehensive cell atlas requires the incorporation of myeloid cell samples representing all varieties of anatomical, physiological and pathological status. It is necessary for iMac to collect and combine data profiled under different experiment protocols and with different sequencing platforms.

Biological signals in a large integrated dataset are often masked by the batch effects induced by laboratories' technical variants. Batch effect control, therefore, should to be performed before the compilation of a reference cell atlas to avoid the formation of underrepresented cell clusters. Unfortunately, most of existing batch effect correction methods are constrained by their parametric assumption on either the data format or the batch effect, hence will not fit well in a complex data context in which batch sources are extremely diverse. Here, we introduce a simple data normalization technique for controlling batch effects in the heterogeneous datasets that are constructed by extensive integration of RNA-seq profiling data collected from many different studies. We explore the complex data structure of the transformed data, unlock the normal based statistical tools for differential expression analyses and for transcriptional variation partitioning of genes via a second transformation method. Our approach has been successfully applied on iMac to find the gene signatures of cell populations, and to refine its transcriptional landscape by filtering out the genes whose expression profiles are dominated by the technical noises that are missed out by the normalization.

Introduction

iMac is an interactive online reference myeloid atlas that aims to describe and benchmark human myeloid cell population landscapes in terms of their transcriptional profile (Rajab et al. 2019). Initially developed for resolving myeloid cells' phenotypic variation among different culture environments, iMac's function is currently limited by its data composition explicitly designed for the study. Nonetheless, iMac has already shown its great analytical power and the broad prospect of development in myeloid biology. The exact goal of iMac is still under discussion. However, we believe that the accessible cellular map has the potential to contribute to many aspects of myeloid research, ranging from hypothesis generation in fundamental biology to decision making in clinical practice. The possibility of compiling such a powerful resource arises from the rapid advent of affordable, high-throughput sequencing techniques, which has encouraged the explosive proliferation of fine-scale transcriptomic data that are able to resolve stem cells' functional complexity and cellular heterogeneity (Hwang et al. 2018; Rajab et al. 2019). Thus, vast amounts of public and proprietary transcriptomic data that focus on different aspects of myeloid biology are now available for the construction of a comprehensive atlas.

Currently, iMac is composed of transcriptomic data collected form 44 studies and 901 samples representing myeloid cell populations derived from different culture environments and progenitor cells. Datasets integrated by iMac are obtained from Stemformatic.org, an online data portal of stem cell gene expression data collected from over 400 high-quality public studies (Choi et al. 2019). Datasets hold by Stemformatic have been assessed and filtered by a strict quality control pipeline. Thus, data with bad library quality, poor experimental design, and obvious annotation errors were not considered by iMac. Approximately 13000 genes that are measurable in all the sequencing platform used by the above studies were included for the analysis.

Facilitating large scale data integration can face many systematical constrains. Particularity, the pitfall of data heterogeneity that lies under the difference of experimental protocols, laboratory conditions, and technician proficiency between different studies is inevitable in iMac (Goh et al. 2017). In consequence, the presence of batch effects may confound the expression level of myeloid signature genes, resulting in underrepresented clustering of myeloid transcriptional niches on iMac. It is necessary, therefore, to reduce, or even eliminate batch effects from true biological signals before progressing further to downstream

investigations. Existing batch effect correction methods all have some notable limitations imposed by their model assumptions. For example, removeBatchEffect and Combat are two correction algorithms based on the ordinary least square regressions to regresses out the batch effects across each gene in the dataset (Johnson et al. 2007; Ritchie et al. 2015). Combat only differs from removeBatchEffect in a way that it has an additional step to shrink the gene-wise batch parameter estimations to a global trend using Empirical Bayes method, therefore robustly adjusting the batch effects in the small samples. Their application to transcriptomic data relies heavily on the identical composition of sample classes across batches. This assumption is badly violated in iMac as its Batch-Class design is severely imbalanced and even incomplete, resulting in the inflation of class differences and, thus, false removal of batch effects. SVA (Surrogate Variable Analysis) and RUVseq (Remove Unwanted Variance) are two other popular correction methods primarily designed for detecting unknown sources of variation and batch-factor determination (Leek et al. 2012; Risso et al. 2014). However, these two methods cannot adequately distinct biological signals that are conflated with technical variants, potentially leading to the removal of subpopulation variations, which in some cases are directly related to our research interests (Leek and Storey 2007).

Here, we describe a non-parametric data-normalization technique that was performed on iMac's integrated dataset for controlling batch effects from multiple studies. For each individual sample, absolute count sizes of genes are converted to their within-sample percentile ranking. Batch effects that are unable to cause genes' relative expression level changes within a sample will be mitigated upon the transformation. Since batch effects are independently corrected within each sample, the effectiveness of ranking transformation will not be influenced by the magnitude of data integration, the quality of Batch-Class design, and the variety of sample sources. We show that this transformation method can robustly correct batch effects on both the simulated RNA-seq data and the real data from iMac. Prior to this study, the probability distribution of ranking data is unknown, and the statistical test that can be done on iMac were restricted to univariate non-parametric methods (i.e. Kruskal–Wallis test, Mann–Whitney U test by rank). Thus, we subsequently explore ranking data's complex data structure and its implication in the potential downstream statistical analyses. Based on our understanding of ranking data, we propose two approaches to open up access for iMac to parametric statistical tools, which has helped us successfully recover the differentially expressed genes (DEGs) that guide the unsupervised

clustering of myeloid cell population on iMac. In the end, we quantify the technical impurity of each gene according to the amount of batch-induced transcriptional variations that are remained uncorrected by ranking transformation. Genes with high impurities are filtered out from iMac to further reduce the amount of technical noise in the reference transcriptional atlas.

Result

Ranking transformation: a simple data standardization method for batch effect correction.

Suppose we have RNA-seq counting data of n samples (libraries) and m genes. Transcriptional profile of the i^{th} sample can be represented as a vector of length *m*,

$$\vec{y}_{i.} = (y_{i1}, y_{i2}, y_{i3}, \ldots, y_{im})$$

where y_{ij} is the total number of reads mapped to the j^{th} gene in the i^{th} sample. Ranking transformation is simply implemented by ranking the genes according to their expression level (read counts in this case) within each individual sample. The k^{th} order statistic $y_{ij}(k)$ in the i^{th} sample will be mapped to a value of *k*, so the gene with the lowest expression level will be assigned to the value *1*, while the highest be assigned to the value *m*. The ranking transformed vector $rank(\boldsymbol{y_{i.}})$ is then scaled to range *(0, 1)* by dividing each of its components by *m+1*. The scaling factor is modified from the value *m* used in the original iMac to avoid arithmetic errors for the further downstream transformation.

If we consider that transcriptional profile of i^{th} sample $\boldsymbol{y_{i.}}$ is generated from a vector of Negative binomial (or over-dispersed Poisson) random variables,

$$\vec{X}_{i.} = (X_{i1}, X_{i2}, X_{i3}, \ldots, X_{im})$$

then ranking transformation will work perfectly in correcting batch effects that act as consistent mappings of a monotonic function on vector $\boldsymbol{y_{i.}}$ in a component-wise manner. For example, imagine that sequencing platform A's influence on the i^{th} sample is a function of form $g_A(.)$**.** The underlying distribution of i^{th} transcriptional profile can then be described as:

$$\overrightarrow{X^A}_{i.} = (g_A(X_{i1}),\ g_A(X_{i2}),\ g_A(X_{i3}),\ \ldots,\ g_A(X_{im}))$$

Further mathematical derivation shows that, regardless of the inter-gene correlations, the ranking of $g_A(X_{ij})$ follows exactly the same distribution as the ranking of X_{ij}. Ranking transformation, therefore, is able to remove all the technical variations that cannot change genes' relative expression level within a sample.

Simulation results demonstrate that ranking transformation can outperform two popular batch effect correction methods, limma and Combat (Johnson et al. 2007; Ritchie et al. 2015), when batches' influences on transcriptional profiles are non-linear and interactive with biological classes (**figure 1.1**).

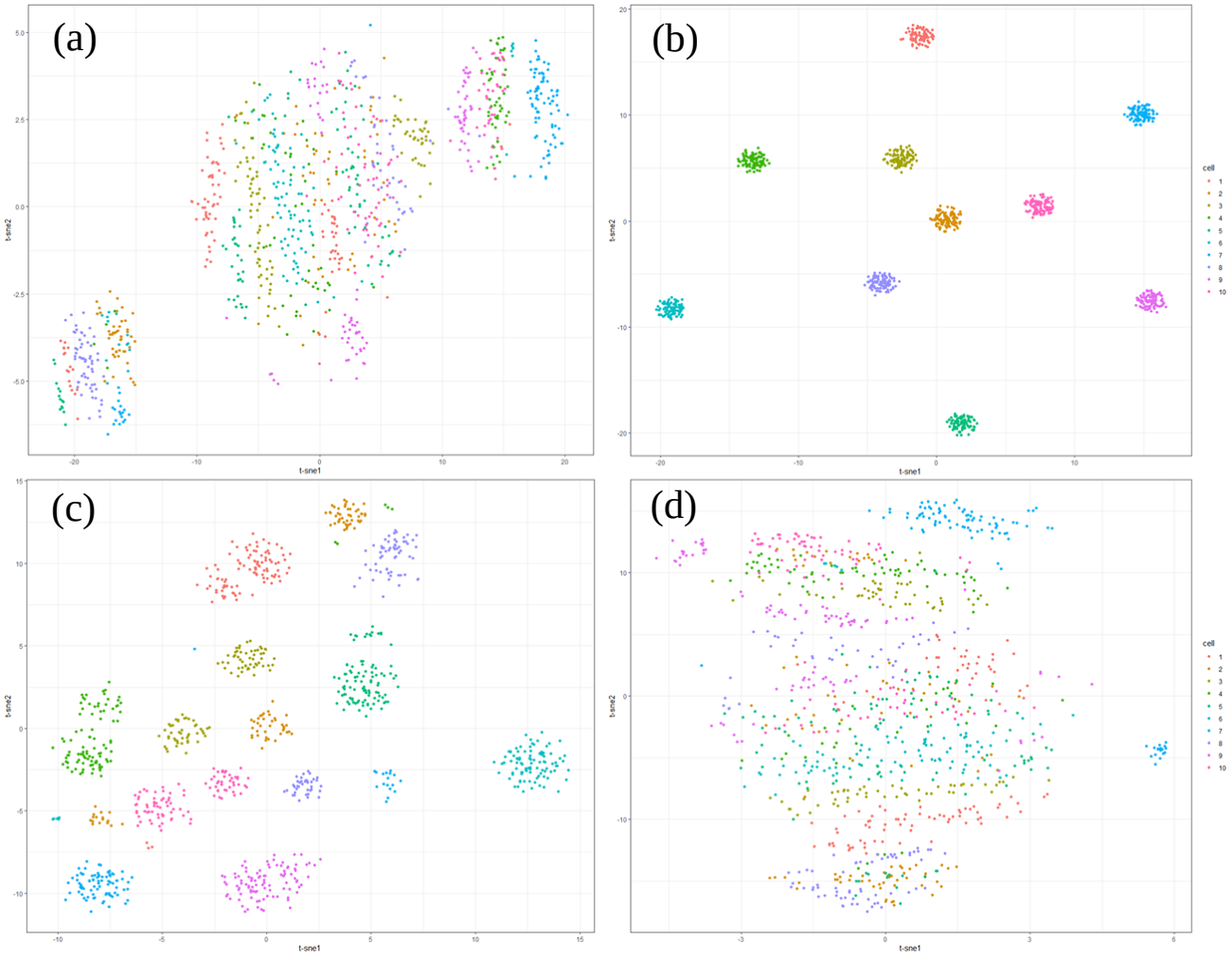

Figure 1.1 ***Batch effect correction with four different methods on simulated count data.*** *Results of corrections are shown with 2D t-SNE plots. 1,000 samples, 10,000 genes, 10 cell types and 4 batches were simulated.* ***Cell type effect****: randomly choose 100~200 genes as DEGs.* ***Batches effect****: g(x) = cx + x^d, where we randomly sampled four pairs of c and d to represent effects of four different technological platforms. Samples are coloured by their biological cell types. The plot* ***(a)*** *is generated before correction, while the rests are generated after correction using* ***(b)*** *ranking transformation,* ***(c)*** *limma plus voom,* ***(d)*** *Combat.*

Moreover, ranking transformation as a non-parametric technique neither requires the specification of batch sources nor the balanced study design. Unknown confounding factors can be automatically corrected as long as the assumption on batch effects is not strongly violated. Ranking transformation can also standardize the samples collected from different studies such that between-study comparisons can be made within the polled dataset. Ranking transformation performs equally well as, or even better than limma and combat on the heterogeneous dataset used for iMac compilation (**figure 1.2**).

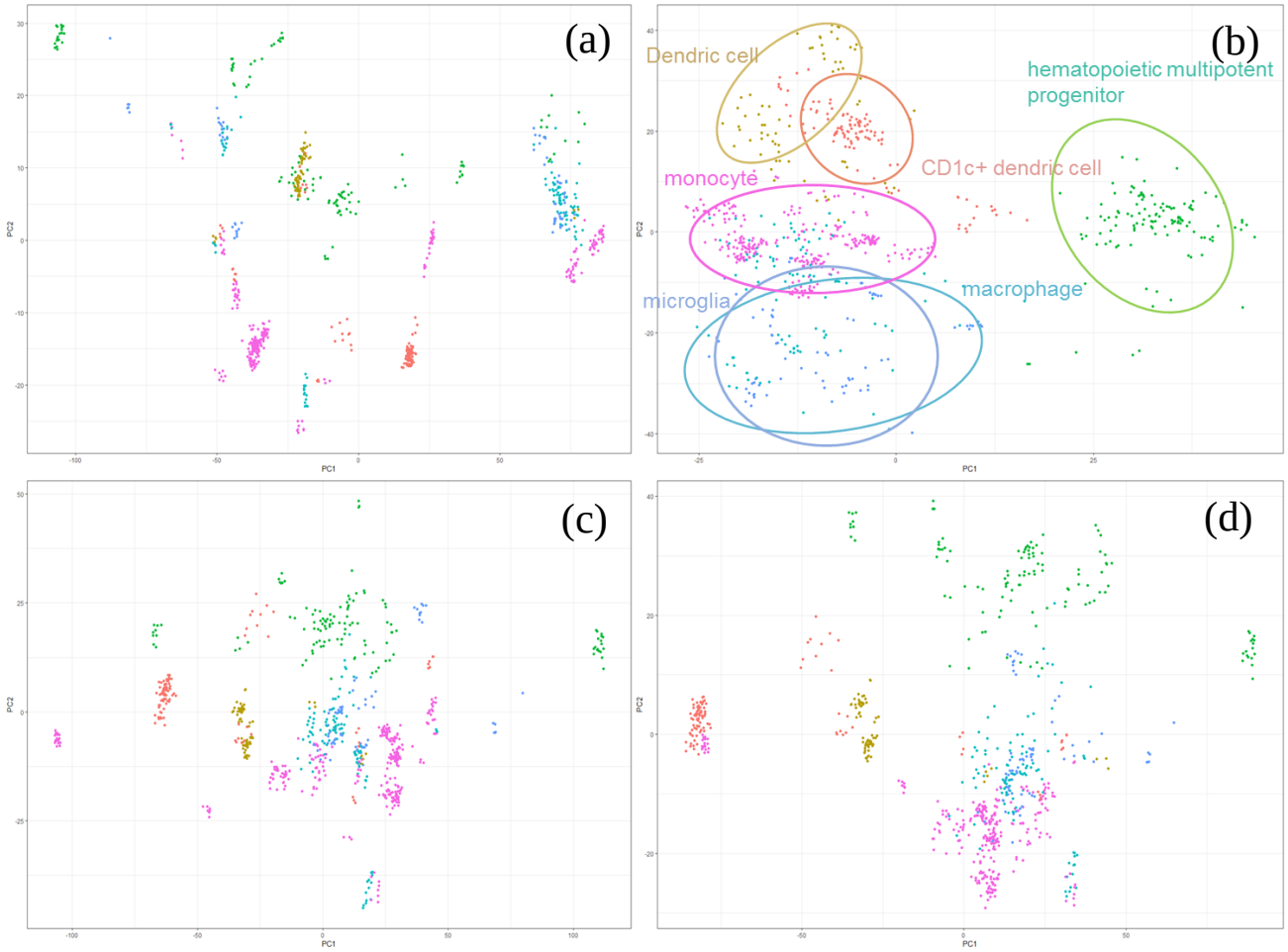

Figure 1.2 ***Batch effect correction with four different methods on the log-RPKM data from iMac.*** *Results of corrections are shown with 2D PCA plots. Samples are coloured by their biological cell types. The plot* ***(a)*** *is generated before correction, and the rests are generated after correction using* ***(b)*** *ranking transformation,* ***(c)*** *limma plus voom,* ***(d)*** *Combat.*

Data structure of ranking data: skewness, unequal variance, and information loss.

To choose a suitable statistical model and effectively quantify the representativeness of genes for cell populations in iMac, we need to fully understand the structure and the characteristic of data after ranking transformation. Firstly, ranking data is asymmetric. The shape of ranking data varies dramatically over the range

of its distribution (**figure 2.1**). Ranking distributions of genes with relatively high expression levels are right-skewed, and that of genes with relatively low expression levels are left-skewed, while genes ranked in the middle are approximately evenly distributed. The closer the gene is ranked to the boundary of the distribution (either highly or lowly ranked), the severer the skewness is. Clearly, ranking data does not follow a normal distribution. Thus, well-developed, Gaussian-based statistical models cannot be fitted directly to the ranking data, and this will greatly limit the types of statistical analyses that can be done.

Secondly, ranking data is naturally heteroscedastic. A distinct quadratic mean-variance relationship can be observed from the ranking transformed RNA-seq count data (**figure 2.1**).

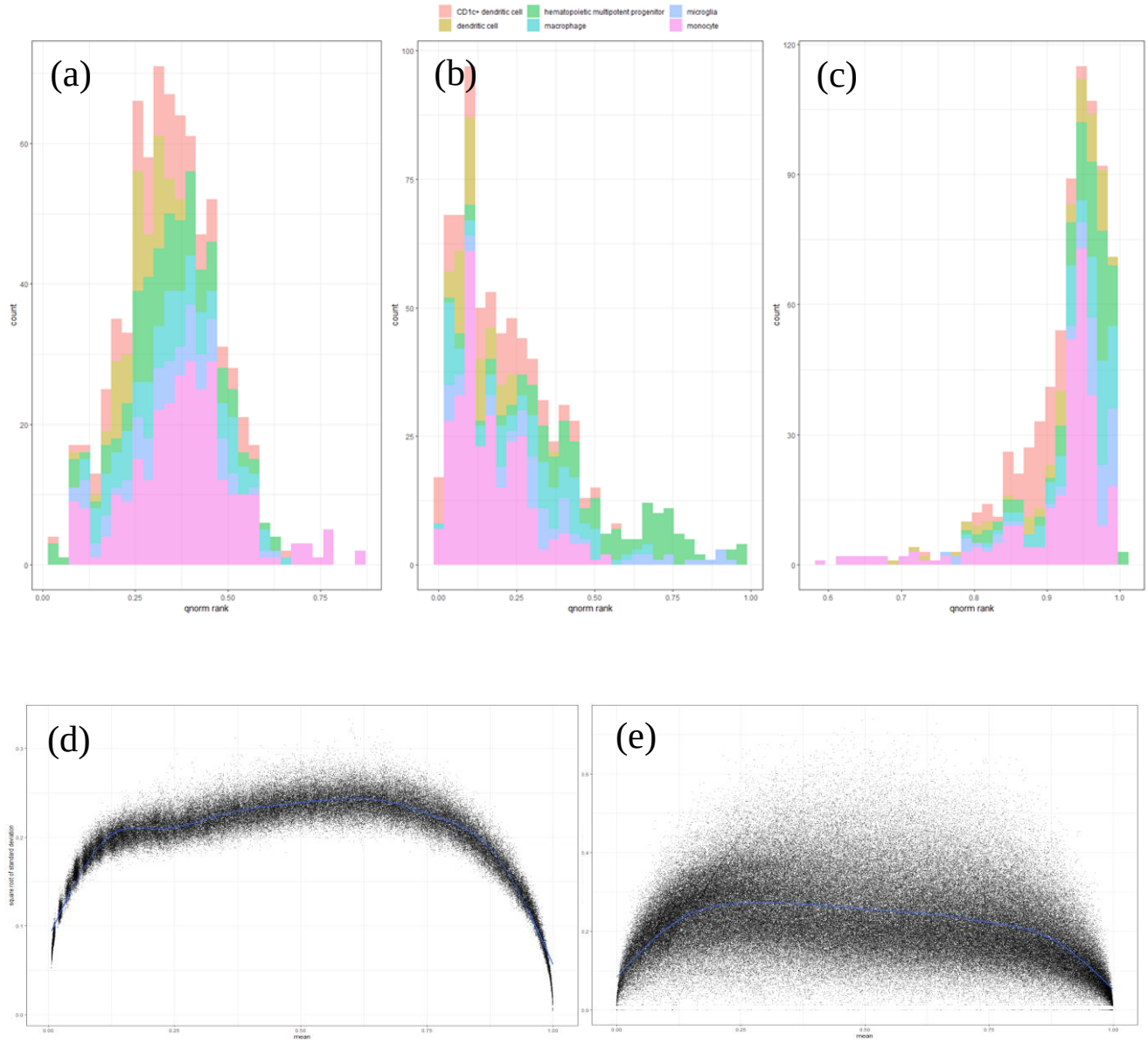

Figure 2.1 ***Properties of ranking data****. Distribution of ranking data are shown with three histograms representing* ***(a)*** *middle ranked gene,* ***(b)*** *lowly ranked* ***gene (c)*** *highly ranked gene in iMac. Quadratic group-wise mean-variance trend fitted by generalized additive model can be observed from both of (d) the simulated ranking data and (e) the ranking data from iMac.*

Genes ranked in the middle display much larger variances than the genes ranked at the upper or the lower boundary of the unit interval. Therefore, highly/lowly ranked genes will have larger statistical confidence in the differential expression analyses compared to the other genes. However, this variance feature is problematic for the ranking data because it may significantly increase the false discovery rates of genes with small variances. In other data formats (i.e., Count, RPKM, CPM and etc.), tiny variances of non-differentially expressed genes usually come together with small effect sizes, whereas it is not always the case for ranking data. Even though not being differentially expressed, a gene's ranking can still be shifted relative to the ranking of the other genes (**figure 2.2**).

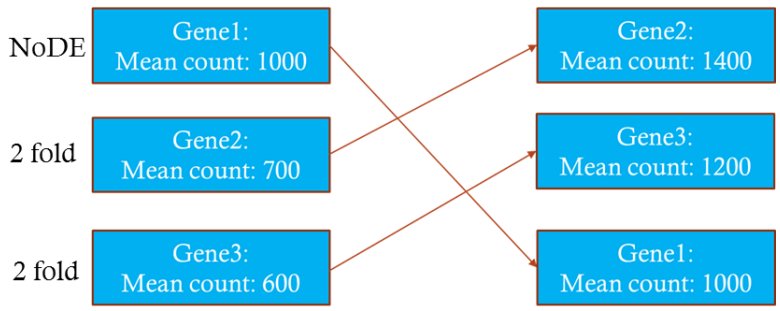

Figure 2.2: ***A schematic diagram shows why ranking transformation may increase false discovery rate.*** *Gene1's ranking is shifted relative to Gene2 and Gene3 even though it is not differentially expressed.*

Consequently, effect sizes of highly/lowly ranked genes will be falsely increased, and their test statistics can be easily inflated to the null hypothesis rejection level due to their variances sizes. This problem is reflected in both the simulated ranking data and the real data from iMac where the highly expressed genes with small fold changes are substantially recovered as DEGs after ranking transformation (**figure 2.2**). Ranking data may also suffer from severe information losses, as absolute count sizes and potential subtle biological variations of genes are dropped during the ranking transformation.

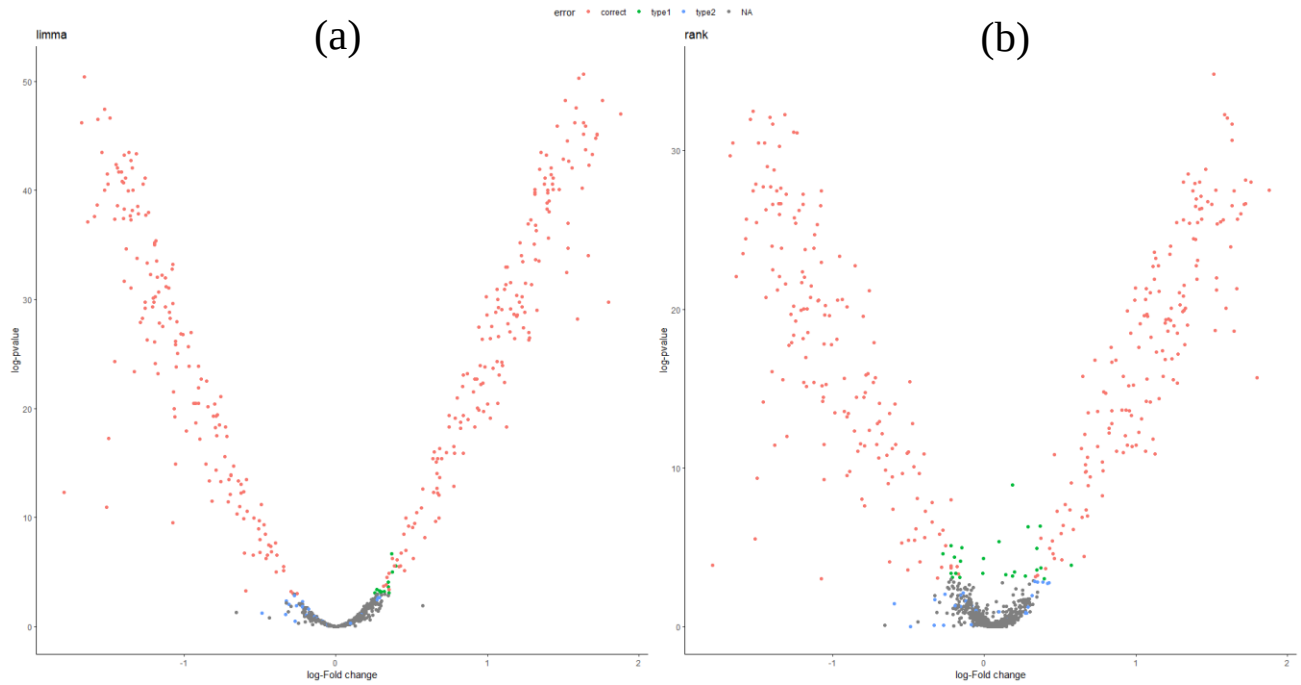

Figure 2.2 ***Volcano plots of simulated RNA-seq count data (a)*** *before and* ***(b)*** *after ranking transformation. Genes are coloured by types of errors to which they have committed. After ranking transformation, genes with small fold changes (in terms of counting) can also be recovered as differentially expressed gene.*

Beta regression: a flexible regression analysis for data ranging between 0 and 1.

We found that the beta regression model that lies under the flexible beta law can be fitted perfectly to the ranking data (Ferrari and Cribari-Neto 2004). The underlying beta distribution modelled by beta regression is formulated as

$$f(y;\ \mu,\phi)\ =\ \frac{\Gamma(\phi)}{\Gamma(\mu\phi)\Gamma((1-\mu)\phi)}y^{\mu\phi\ -1}(1-y)^{(1-\mu)\phi},\ y\in(0,1)\ ,\ \mu\ \in(0,1),\ \phi\ \in\mathbb{R}$$

where the mean and the precision of the distribution are governed by parameter $\boldsymbol{\mu}$ and $\boldsymbol{\Phi}$, respectively. Unlike classical linear regression analysis, which only estimates the means of response variables and assumes equal variances of observations across samples, beta regression can also model the shape and the dispersion of the data, allowing the model to smoothly accommodate to ranking data that are highly skewed and variably dispersed (**figure 3.1**).

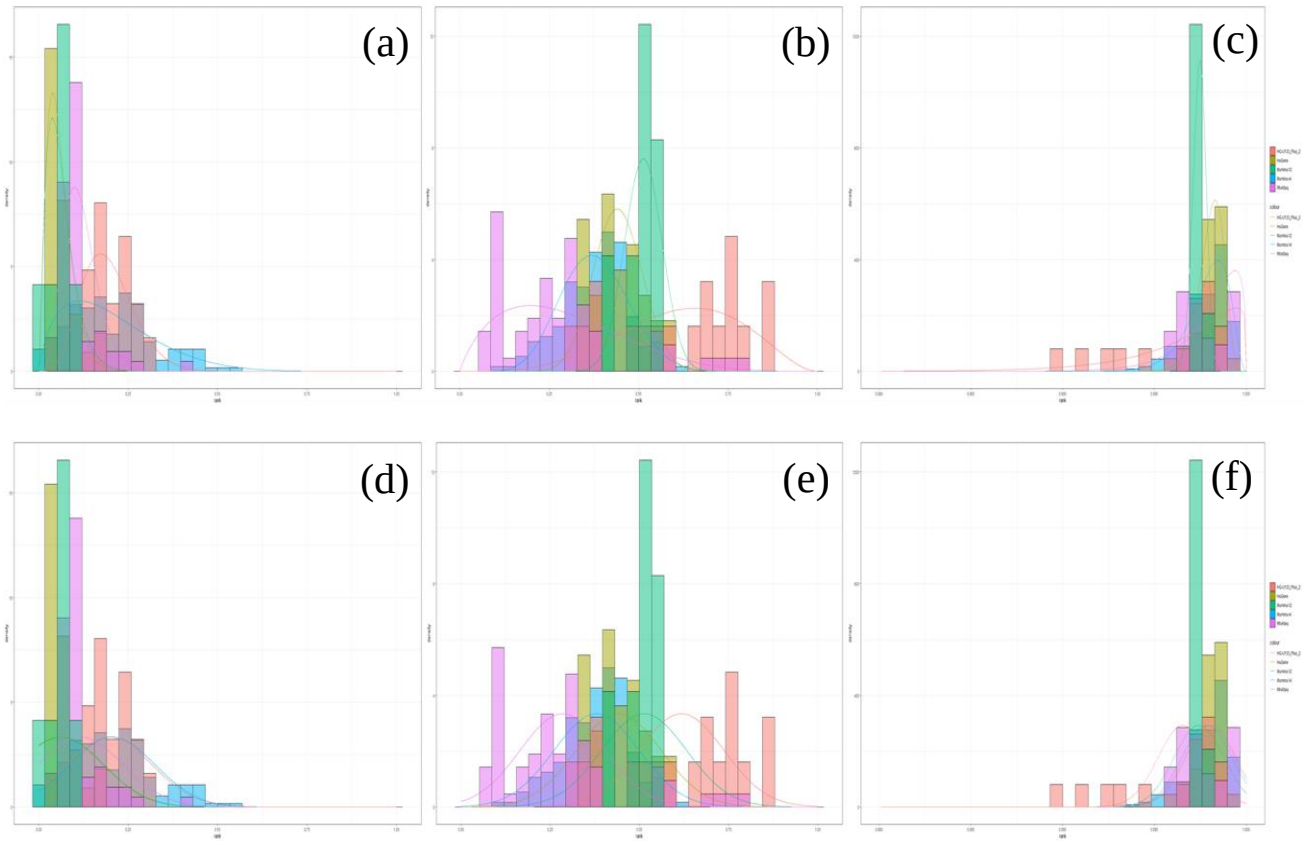

Figure 3.1: ***Histograms of three genes' rankings in monocyte subpopulation in iMac*** *with density lines modelled by* ***(a-c)*** *beta regression and* ***(d-e)*** *ordinary least square regression. Bars and fitted density lines are coloured by different sequencing platforms. Beta regression has perfectly adapted to the shapes of empirical density of monocytes' gene ranking on multiple sequencing platforms.*

Beta regression, nonetheless, does not work well on the data with small sample sizes. To find out at least how many samples are required for the beta regression to capture the genuine underlining distribution of genes, we fitted the model to the simulated ranking data of different sample sizes. Datasets with the rankings of 10,000 genes were simulated, but only the gene that is ranked 5,000th according to its baseline expected count size was used for the model evaluation. This gene has the largest variance and uncertainty among all the simulated genes with respect to their ranking distributions. Hence our simulation result should be generally applicable to all the other genes of different rankings.

We find that the performance of the beta regression varies considerably when the sample size is small (**figure 3.2**).

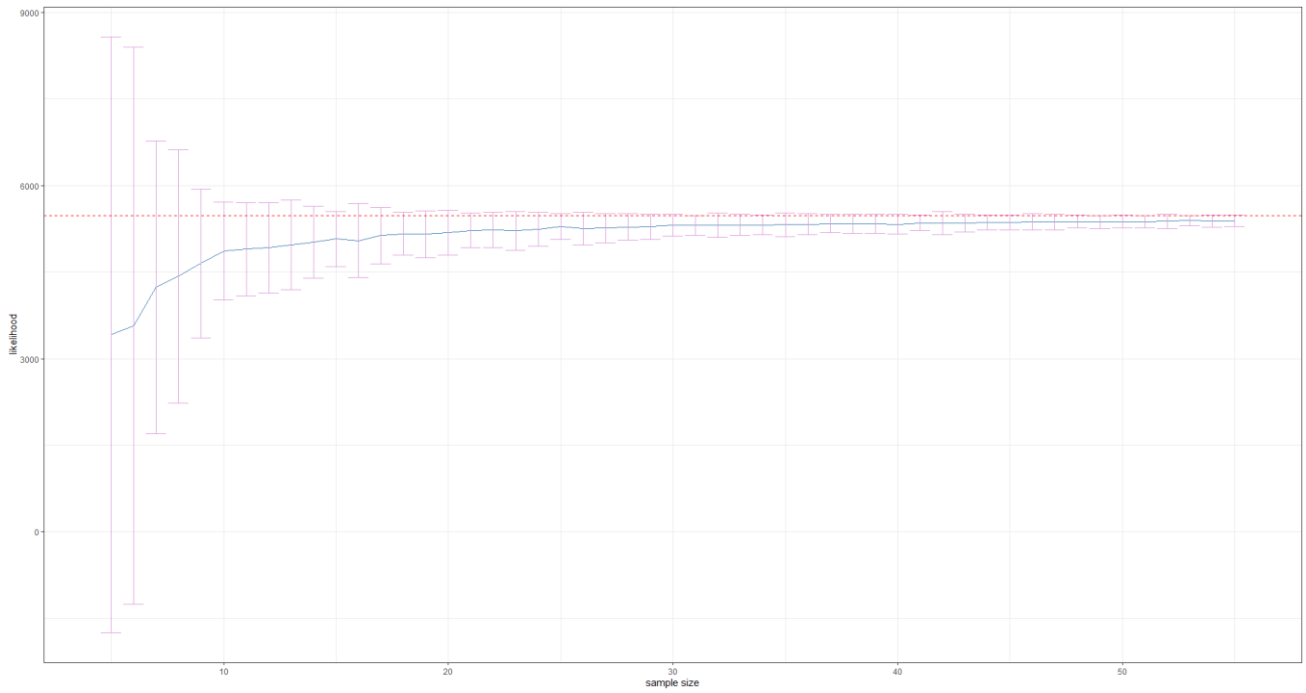

Figure 3.2 ***Performances of beta regression in fitting the simulated ranking data of different sample sizes****. For each sample size, models are fitted iteratively to the ranking data randomly sampled from the training set. Performance scores are quantified by the observed likelihood of fitting the trained models to a large test set, which represents the genuine underline ranking distribution of the gene we choose (see main text for the description of the gene used). Blue line shows the average scores that beta regression gets at each sample size with error bars representing* $\pm$ *1.96 standard errors. Red dashed line shows the maximum likelihood (maximum score) that the model can get for fitting the test set.*

Over 500 independent simulations, beta regression tends to commit more type1 errors (or has higher false-positive rates) than t-tests when comparison is made between two non-differentially expressed sample groups with less than five replicates. The possible reason is that the flexibility of beta regression may make it too sensitive in fitting the data of small sample sizes, potentially leading to over-fittings and false discoveries. However, as the sample size increases, beta regression starts to outperform t-tests in terms of successfully controlling both the type1 and the type2 (or false negative) error rate (**figure 3.3**).

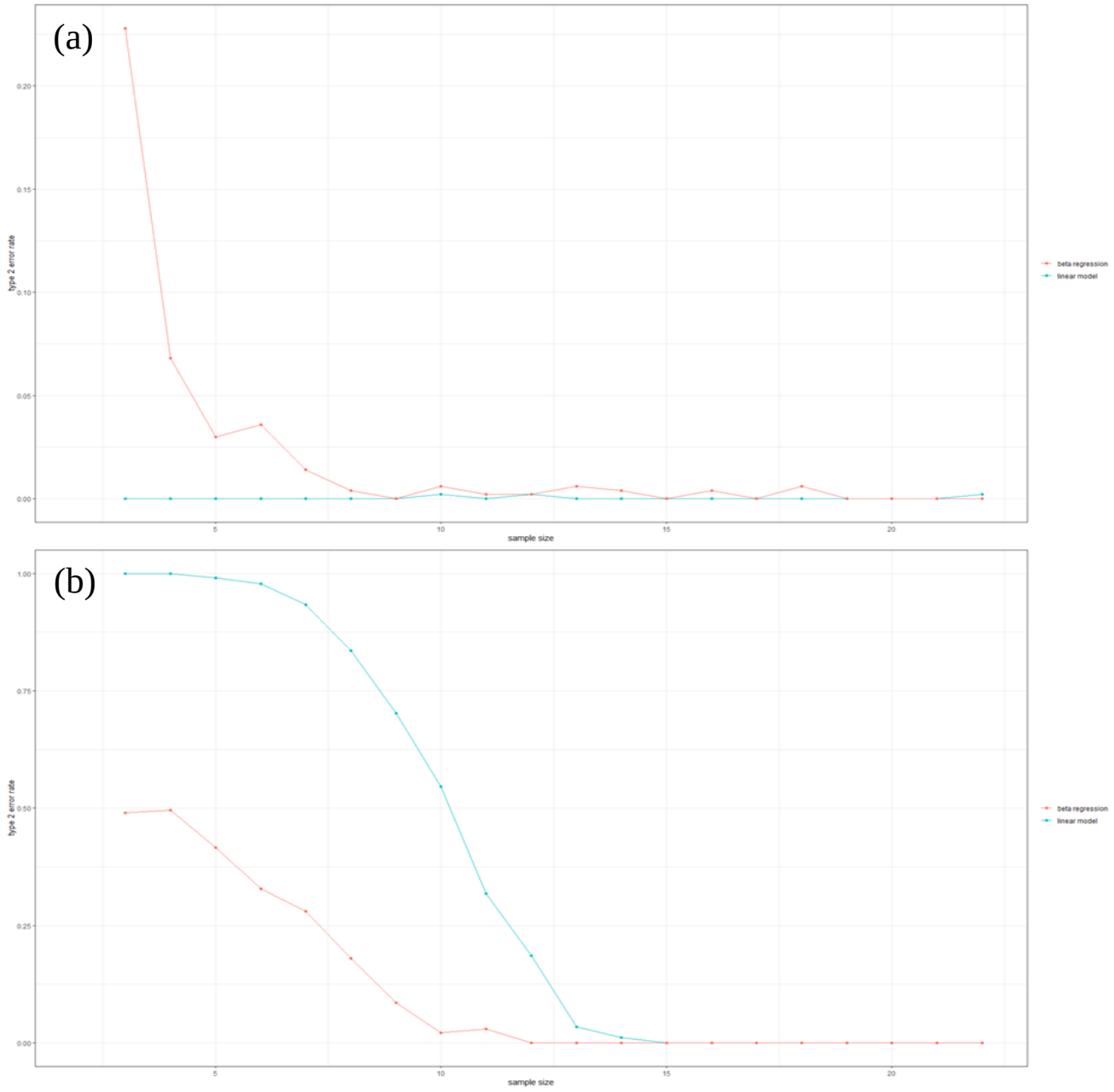

Figure 3.3 ***Beta regression's frequency of committing (a) type 1 and (b) type 2 errors on the simulated ranking data against sample sizes used****. Plot* ***(a)*** *is generated by comparing two groups of ranking data simulated from a non-differentially expressed gene (see main text for the description of the gene used). Comparisons are made 500 times at each sample size. Plot* ***(b)*** *is generated by comparing two groups of ranking data simulated from a same gene, but one group is 1.5-fold differentially expressed compared to the other group. Comparisons are made 500 times at each sample size. The red line shows the results of beta regression. The blue line shows the result of t-tests.*

Probit transformation: restore the normality of the ranking data.

To unlock normal based statistical tools and expand types of statistical inferences that can be done on the ranking data, we need to apply a further transformation to map ranking data back to the whole real line in which standard normal regression analyses can be performed. We expect this second step of transformation to restore the normal assumptions that are strongly violated by the ranking data. For example, to achieve the symmetricity of the data, we need to find a function to stretch the left tails of the distributions of highly ranked genes and the right tails of the distributions of lowly ranked genes. Two functions are found to serve the purpose.

$$\textit{logistic}: g(y_{ij}) = log\left(\frac{y_{ij}}{1+y_{ij}}\right), \quad \textit{probit}: g(y_{ij}) = F_X^{-1}(y_{ij}),\ \textit{where } F \textit{ is the inverse of cumulative distribution function of Normal distribution}$$

In the real practice, the probit function turns out to be a better choice for the ranking data (**figure 4.1, 4.2**).

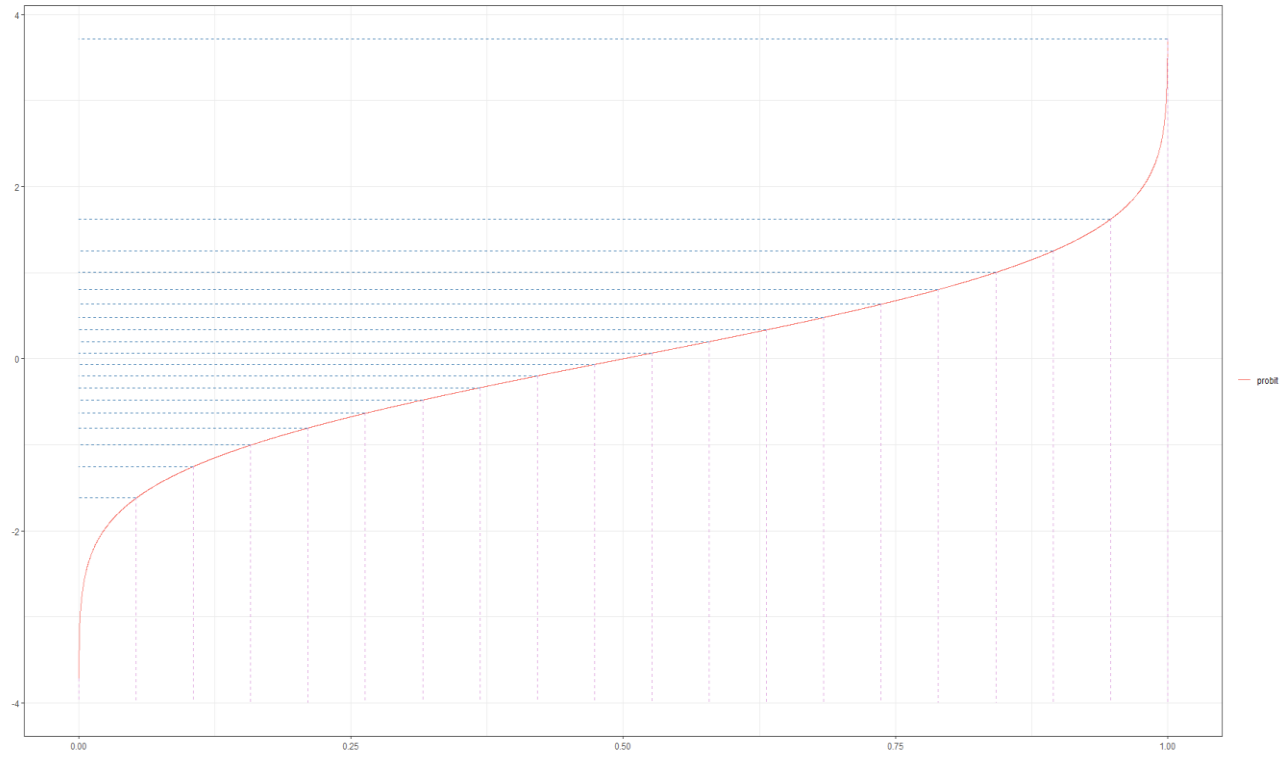

Figure 4.1: ***Mapping of probit function****. Probit mapping is even in the middle, so the symmetricities of middle ranked genes will not be disturbed. Ranking data skewed at the boundary of the unit interval are unevenly mapped by the probit function to achieve the symmetricity of the distribution.*

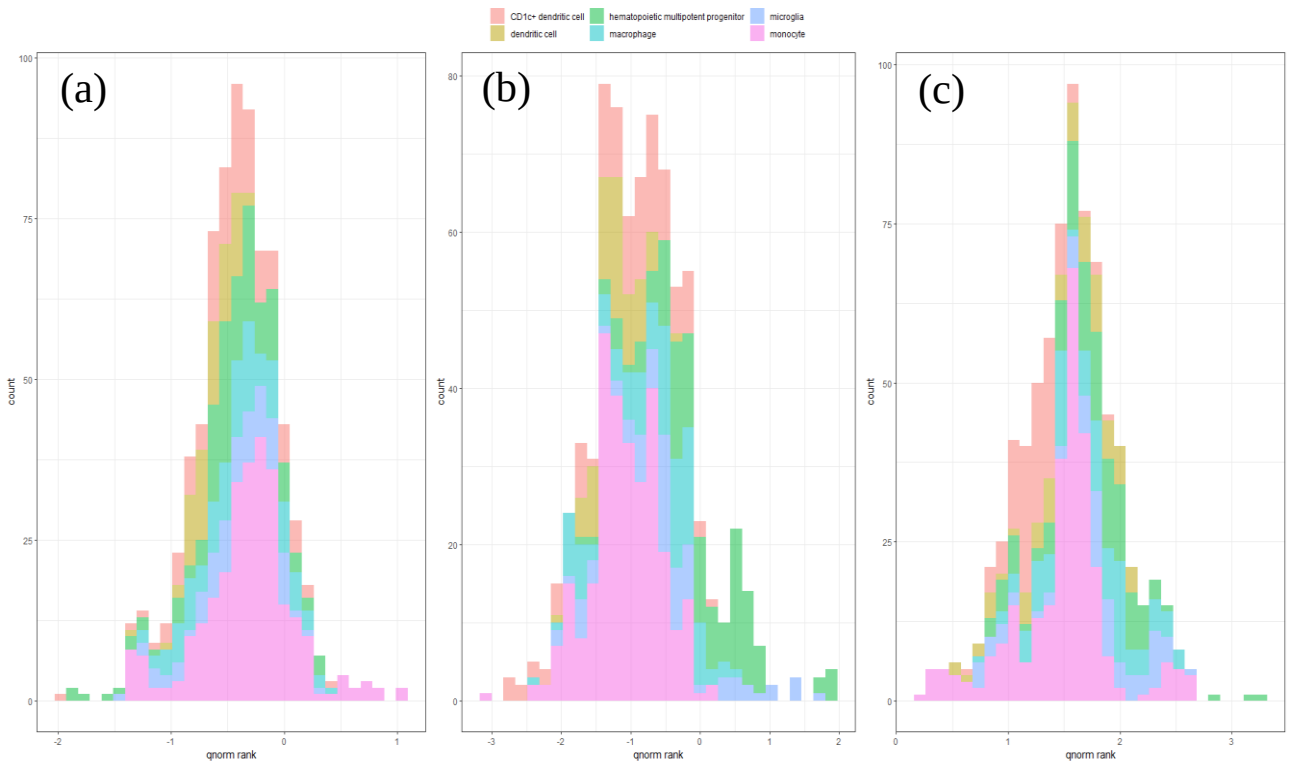

Figure 4.2: ***Histograms of ranking data of three genes after probit transformation****. These genes are the same gene used in figure 21.*

Probit transformation is simply done by applying the probit function to each of the observations in the scaled ranking dataset. After being probit transformed, the distribution of the data in iMac displays an almost perfect bell curve centred at 0, and the skewness of most of the genes have been corrected.

However, the probit transformation does not solve the problem of unequal variances in the heterogeneous dataset. The presence of heteroscedasticity is one of the major concerns of linear regression analyses in which error variances are uniformly estimated across samples. If not being properly addressed, heteroscedasticity will bias the estimation of model parameters' variances, and consequently lead to incorrect statistical conclusions. Here, we model the mean-variance trend of the probit data with generalized additive models, and then interpolate individual observations' variances based on the fitted trend. Subsequently, the inverse of the predicted variances

$$Weight\ on\ y_{ij}:\ W_{ij} = \frac{1}{\widehat{\sigma}_{ij}^2}$$

are assigned as weights to each individual observation in the following linear regression analyses. Our procedure is inspired by the voom's gene-wise mean-variance trend modelling, but the resolution is improved to the groupwise level to accommodate extremely heterogeneous datasets such as the one used to compile iMac (Law et al. 2014). As a consequence, our procedure is more computationally intensive than the voom pipeline.

To evaluate our probit pipeline, we simulated RNA-seq count data of 2000 genes and 10 samples of different libraries sizes. The first 5 samples were treated as group 1, and the rest were treated as group 2. In each of the groups, 100 genes were randomly chosen as differentially expressed genes with fold change rates uniformly sampled from (1, 3). Four different modelling procedures were assessed over 100 simulations: (1) voom on the log-CPM data, (2) Weighted least square regression on the probit data, (3) Ordinary least square regression on the probit data, and (4) Ordinary least square regression on the ranking data. Empirical Bayes method processed by *ebayes* function form the limma package was applied after the model fitting to shrink and stabilize the variance estimations. Beta regression was not assessed due to its inability of modelling the data of small sample sizes. These procedures' performances were evaluated with respect to their frequency of committing type 1 and type 2 errors, fold changes of differentially expressed genes at which type 2 error is made, and the false discovery rates controlled by Benjamini-Hochberg multiple correction method (Benjamini and Hochberg 1995). Results from **figure 8** show that without probit transformation, ranking data's statistical power with small sample sizes is extremely small and is apparently not suitable for the linear regression analyses. Probit data, on the other hand, has greatly improved linear regression's ability in detecting differentially expressed genes, and it is further strengthened by the precise variance weightings. Surprisingly, our probit pipeline is 'almost' as good as the voom regardless of the information losses during the ranking transformation.

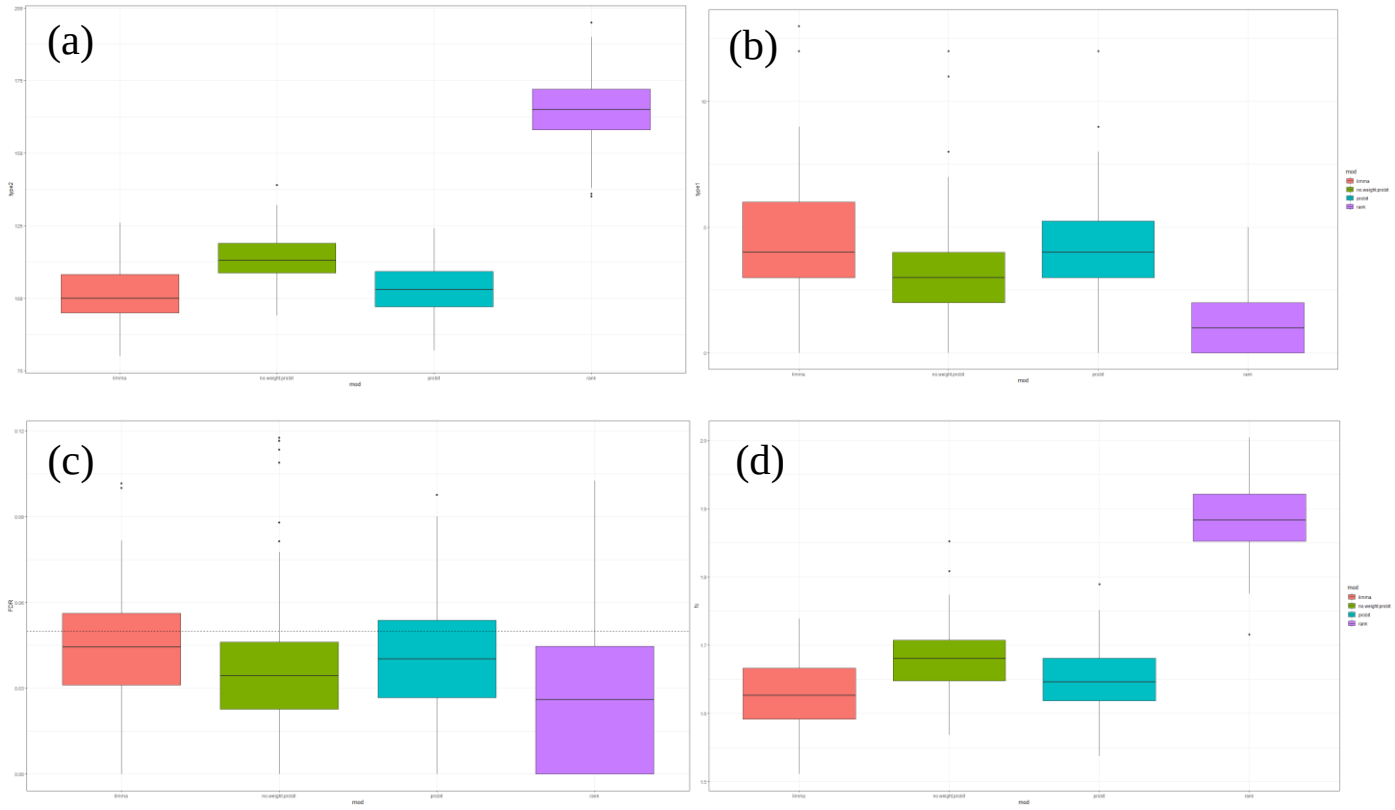

Figure 4.3 ***Comparing four different modelling procedures on simulated RNA-seq count data*** *with respect to their frequency of committing* ***(a)*** *type 2 and* ***(b)*** *type 1 error,* ***(c)*** *False Discovery Rate at 0.05 Benjamini-Hochberg p-value cut-off and* ***(d)*** *Fold changes of differentially expressed gene at which type 2 error is made. The boxplot shows the distributions of model assessment results over 100 simulations. Red box: result of voom pipeline. Green box: result of probit pipeline without precises weighting. Blue box: result of probit pipeline. Purple box: result of ranking transformation with ordinary least square.*

Statistical test: Finding differentially expressed genes in iMac after probit transformation.

iMac is now ready for the differential expression analyses after being processed by the probit pipeline. Two different linear mixed models:

$$y_{i.} = \mu + Z^{(1|Class)}\alpha_{Class} + Z^{(1|Platform)}\alpha_{Platform} + \varepsilon_{i.} \quad \textbf{(Model 1)}$$

$$y_{i.} = \mu + X_{Class}\beta_{Class} + Z^{(1|Platform)}\alpha_{Platform} + \varepsilon_{i.} \quad \textbf{(Model 2)}$$

with random components

$$\alpha_{Class} \sim N\left(0, I\sigma^2_{Class}\right), \quad \alpha_{Platform} \sim N\left(0, I\sigma^2_{Platform}\right)$$

were fitted to each gene in iMac. Platform categories were treated as random factors in both of the models to control for their confounding effects on the biological signals. We proposed two statistical tests that might be biologically relevant to iMac on the fitted models. Firstly, we tested the significance of the biological class effects and the platform effects in each gene using likelihood ratio tests with **model 1**. It is essentially testing whether the ranking variants of a gene are actually caused by the either of these two effects or are just the uncategorized random noises. Secondly, we used Wald tests with **model 2** to examine which class groups should be primarily accounted for each gene's differential expression in iMac. Intuitively, this test is designed to find the class groups that deviates the most from the others in terms of their ranking distribution in each gene, thus can be used for searching the gene signatures of different cell populations.

Both of these two tests have worked pretty well on iMac. The table below shows an example output of the signature genes that are found for the hematopoietic progenitor cell (HPC) at the end (**Table 1**). Many genes in this table are known to associate with HPCs. For example, CD34 is the gene that encodes for CD34 transmembrane phosphoglycoproteins, which are commonly used as the cell markers for HPCs in clinical trials (Sidney et al. 2014). The cyclin-dependent kinase 6 (CDK6) is a key cell-cycle regulator for HPCs, and its function is essential for HPCs' activations (Scheicher et al. 2015). We coloured the samples in the iMac according to the expression level of these two genes **(figure 5**), and it can be clearly seen that both of the genes are specifically highly expressed within the cluster of HPCs.

Symbol id	Ratio	BP test	DE cell type	DE cell type p-val	Cell effect	Batch effect
HCK	131.1558	6.40E-39	hematopoietic multipotent progenitor	0	5.55E-224	1
IL13RA1	21.2313	1.52E-13	hematopoietic multipotent progenitor	0	1.77E-177	7.28E-08
RASSF4	16.99383	1.18E-22	hematopoietic multipotent progenitor	0	6.89E-185	3.33E-07
CD34	9.965965	1.77E-18	hematopoietic multipotent progenitor	0	2.23E-230	5.58E-32
TYROBP	7.255435	2.73E-26	hematopoietic multipotent progenitor	0	6.46E-183	3.50E-26
PRKCQ	6.8828	2.30E-14	hematopoietic multipotent progenitor	0	1.38E-182	1.86E-33
CPXM1	6.07194	2.21E-14	hematopoietic multipotent progenitor	0	1.01E-183	3.34E-29
ANGPT1	3.944755	4.91E-10	hematopoietic multipotent progenitor	0	3.93E-216	1.90E-67
GCSAML	3.747232	2.49E-20	hematopoietic multipotent progenitor	0	6.10E-184	1.26E-20
TBC1D9	12.24588	6.64E-09	hematopoietic multipotent progenitor	4.63E-313	4.27E-174	5.21E-17
MYCT1	11.06673	2.53E-17	hematopoietic multipotent progenitor	3.85E-304	3.12E-167	1.43E-21
PLCB4	1.844484	3.48E-07	hematopoietic multipotent progenitor	3.39E-290	2.45E-165	2.16E-103
TIGAR	21.20098	5.08E-34	hematopoietic multipotent progenitor	3.20E-289	3.30E-159	1.77E-05
MMRN1	9.972017	1.07E-19	hematopoietic multipotent progenitor	6.01E-277	4.43E-162	2.91E-16
DNMT3B	3.785263	7.48E-15	hematopoietic multipotent progenitor	1.30E-272	1.83E-157	2.80E-46
MAFB	56.58852	1.12E-06	hematopoietic multipotent progenitor	8.13E-264	9.91E-197	0.204171
RAB27B	10.32052	3.93E-10	hematopoietic multipotent progenitor	5.89E-263	1.27E-161	1.21E-07
NCF2	15.14618	3.32E-26	hematopoietic multipotent progenitor	1.76E-255	2.13E-156	4.61E-11
CDK6	43.68975	7.57E-06	hematopoietic multipotent progenitor	4.29E-253	1.45E-151	1
LAPTM4B	17.67935	1.55E-33	hematopoietic multipotent progenitor	2.08E-251	1.72E-172	1.00E-09

Table 1: ***An example output of differential expression analyses with iMac atlas using probit pipeline.*** *Only the signature genes found for hematopoietic multipotent progenitor cells are shown.* ***Symbol id****: genes' symbol name.* ***Ratio****: Cell type effect to batch effect variance ratio (see next section for detailed description).* ***BP test****: Breusch–Pagan test for heteroskedasticity.* ***DE cell type****: which cell types' rankings deviate the most form the rankings of other cell types.* ***DE cell type p-val*** *: significance of this deviation.* ***Cell effect*** *: significance of cell type effect reported as p-value.* ***Batch effec****t : significance of batch effect reported as p-value*

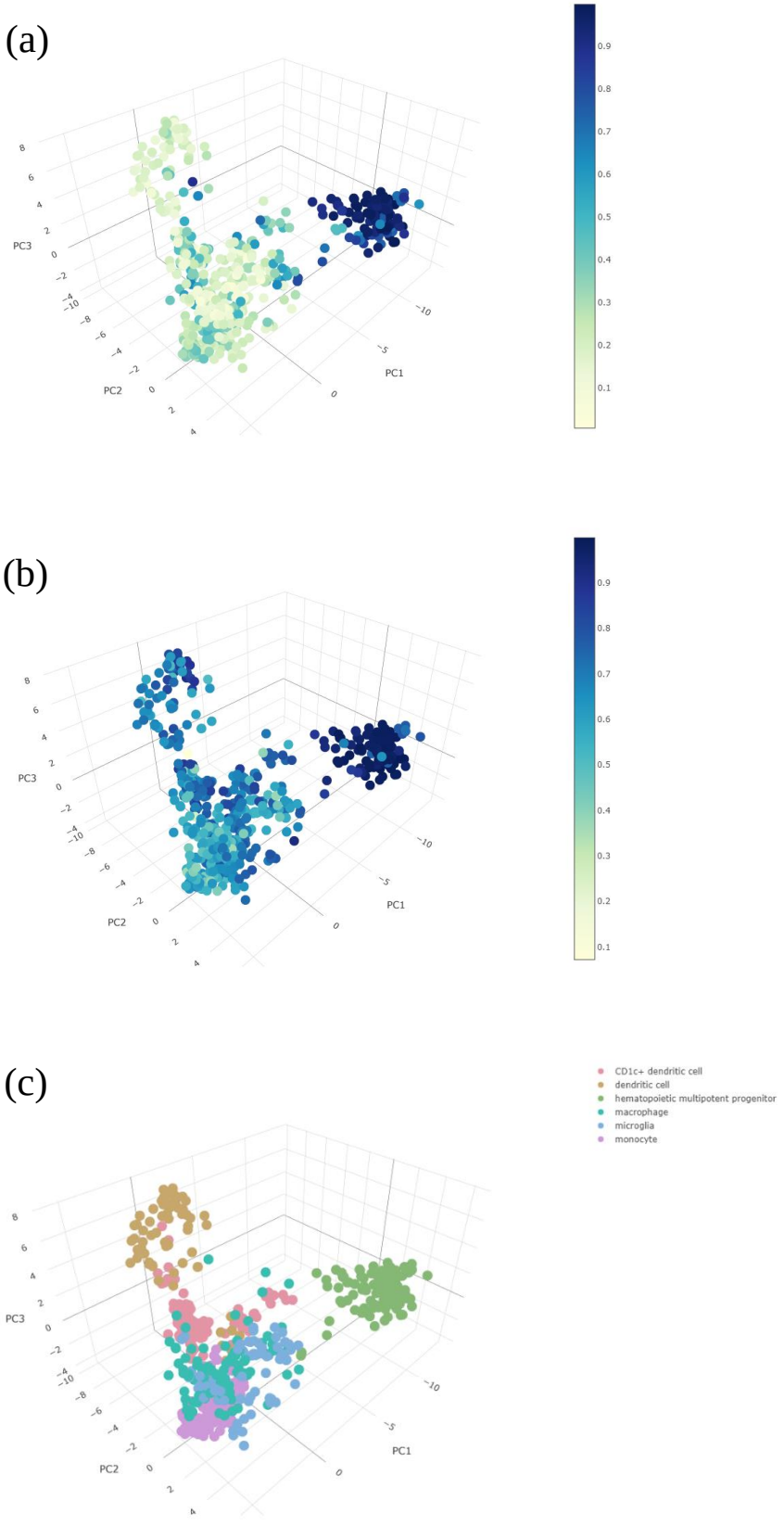

Figure 5: ***3D PCAs of iMac's first 6 largest cell populations****. Libraries in plot (a) are coloured by the within sample rankings of CD34 gene (4th in table 1). Libraries in plot (b) are coloured by the within sample rankings of CDK6 gene (19th in table 1). Libraries in plot (c) are coloured by their biological cell type.*

Variance partition: Quantify cell type effects and batch effects for each gene in iMac.

To make iMac's transcriptional landscape on PCA predominantly biological such that it can be benchmarked against myeloid biology, we need to refine the current iMac by constructing it again using the genes with large cell type effects but low platform dependency. Thus, the second aim of our project is to estimate the contribution of the cell types and the batches to the transcriptional variation of each gene in iMac. It was achieved by implementing the variance partition algorithm on the fitted random effect models (**model 1**) to partition each gene's transcriptional variation into subgroups of variation sources specified as the random factors in the linear mixed model (Hoffman and Schadt 2016). Representativeness of each gene was then quantified by the estimated cell type effect variance to the platform effect variance ratio.

$$r = \frac{\hat{\sigma}^2_{Class}}{\hat{\sigma}^2_{Platform}}$$

Partition outcome is shown in **figure 6.1**.

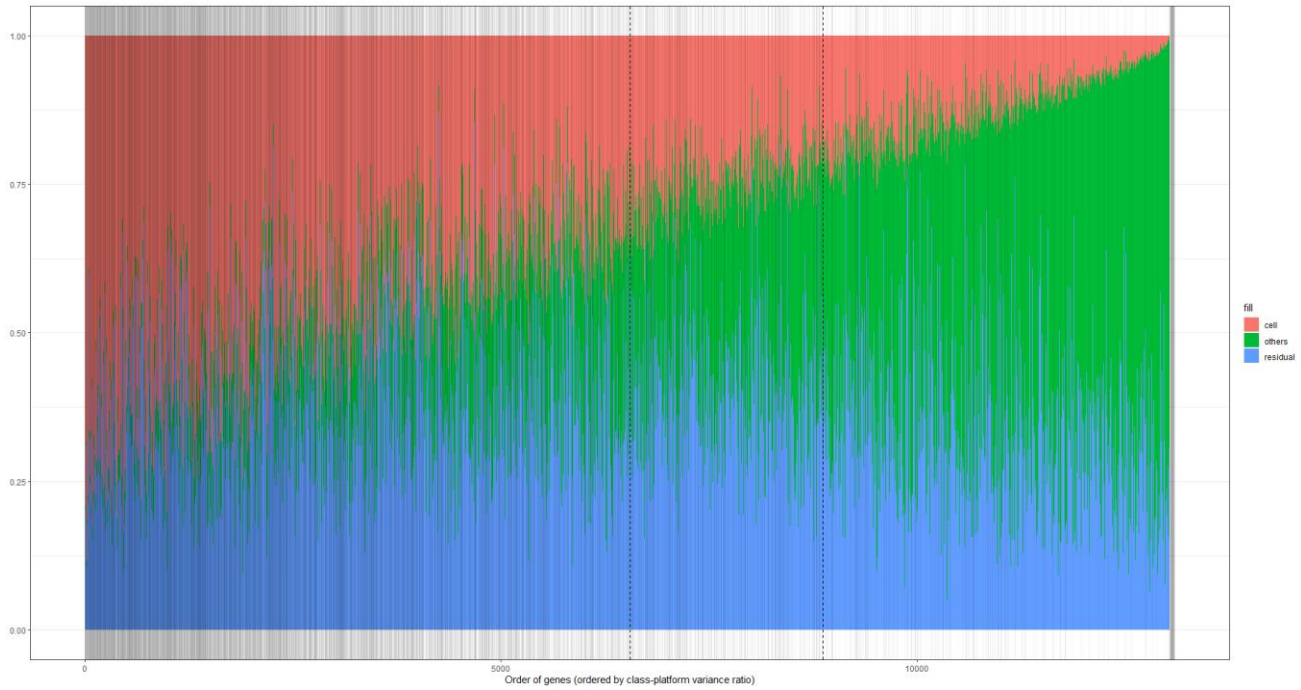

Figure 5.1: ***Variance partition outcome of unfiltered iMac (13,093 genes).*** *Genes (vertical lines) are ordered according to their cell type to batch effect variance ratio, and are coloured by the proportion of transcription variation explained by cell types (red), batches (green), and residuals (blue).* ***Left dashed line****: gene with variance ratio equal to 1.* ***Right dashed line:*** *gene with variance ratio equal to 2.* ***Dark shades:*** *genes that are retained in the filtered iMac (3,607 genes).*

Each vertical line in this plot can be thought of as a gene in iMac that is coloured by its variance composition calculated by the variance partition algorithm, and they are ordered from left to right according to the size of their variance ratio ***r***. Clear we don't want the genes that are placed at the right end of the plot to be included in the iMac since their transcriptional variation is mostly technological. Results of variance partition are in accordance with our differential expression analyses (**figure 6.2**).

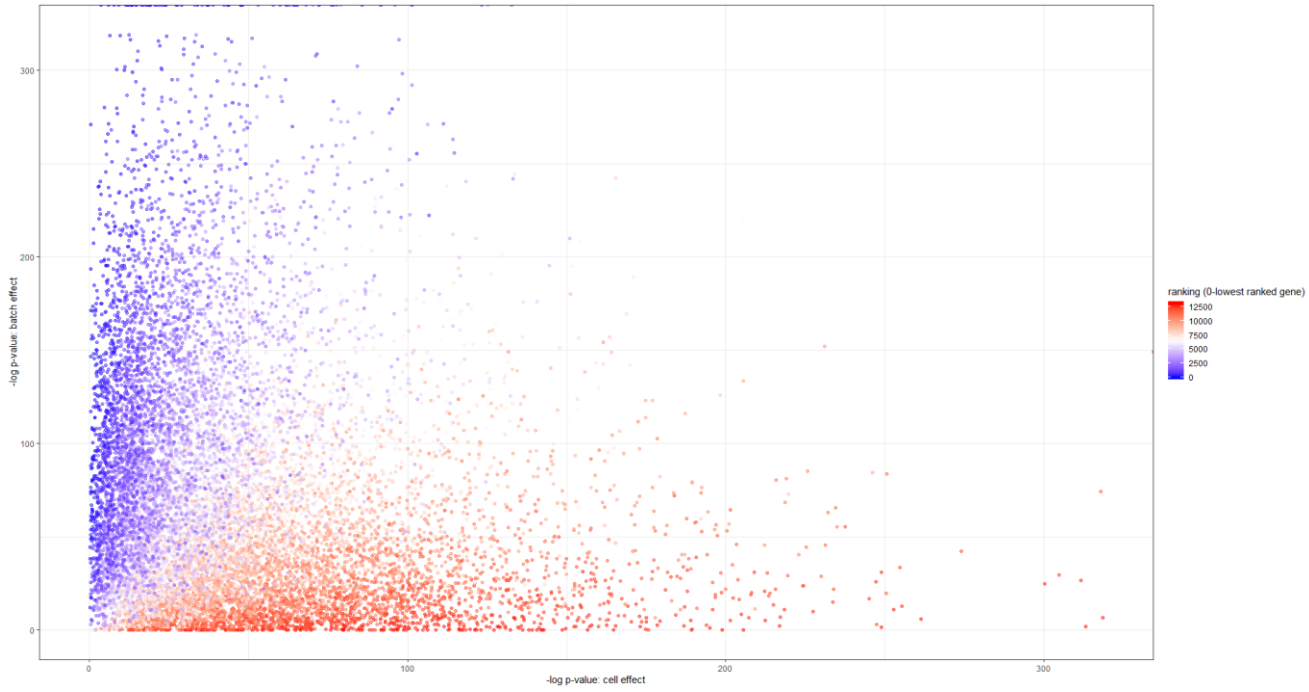

Figure 6.2: ***Checking the consistency between Variance partition algorithm and the result from differential expression analyses.*** *Genes in the unfiltered iMac are depicted as scatter points in this plot, and they are coloured by their ranking with respect to their cell type to platform effect variance ratio.* ***X-axis****: significance of cell type effect reported as negative log2 P-value.* ***Y-axis****: significance of Batch effect reported as negative log2 P-value. Because Variance ratio and significance of effects are calculated by two completely different algorithms, we need to check their consistency to avoid the scenario where large variance ratio comes together with small class effect significance and large batch effect significance.*

Clustering of cell population has been noticeably improved on the reconstructed iMac whose expression space is only characterized by the filtered top 100 genes with the largest variance ratios (**figure 7.3**).

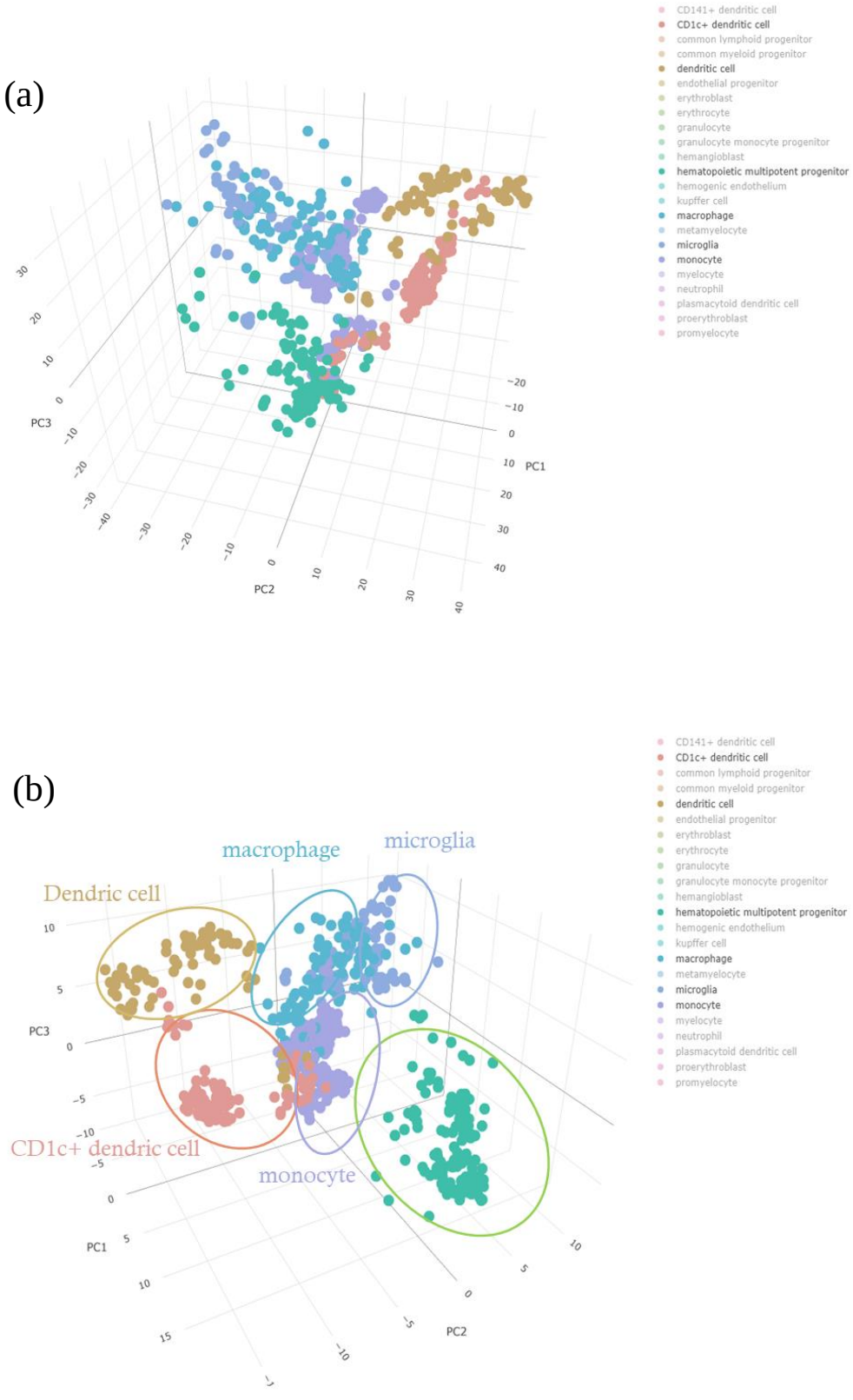

Figure 7.3: ***3D PCA of (a) original iMac (3607 genes) and (b) refined iMac (first 100 genes with largest variance ratio).***

Discussion

Ranking transformation is a preferable practice for data normalization and batch effect correction on the high-throughput heterogeneous datasets whose unwanted technical variation is too complicated to be artificially estimated and separated from our biological interests. We formalized the assumption that is required by ranking transformation to correct batch effects across genes within a sample and investigated its strength and limitations on the simulated count data. Compared to other existing batch effect correction methods, ranking transformation not only makes fewer assumptions on the nature of batch effects but also does not discriminate the datasets that are highly imbalanced or poorly designed (in the case of iMac, there is no study design at all!). Thus, we believe that ranking transformation can outperform most of the existing batch effect correction methods when its assumption is met and can be served as the standard data pre-processing pipeline for large-scale data integration that is essential for the construction of comprehensive cell atlases. Moreover, ranking transformation is a safe technique to implement since it does not bias the information that the original dataset is present to us even if its assumption on the batch effect is strongly violated.

We proposed two separate strategies, beta regression, and probit transformation, for analysing ranking transformed data. We assessed their performance with extensive simulations to see if they can be implemented on the reference human myeloid atlas iMac. Beta regression, regardless of its perfection in fitting the ranking data, is not suitable for statistical inferences on iMac who is currently suffering from the shortage of minor cell populations. The possible reason is that the flexibility of beta regression may make it too sensitive in fitting the data of small sample sizes, potentially leading to over-fittings and false discoveries. In spite of that, the application of beta regression should be seriously considered when the data composition of iMac is improved and becomes complete in the future study. Alternatively, we probit transformed the ranking data, weighted each individual observation according to their variance estimation to unlock the usage of normal based statistical tools on iMac. Despite losing genes' absolute count sizes, our probit pipeline can perform almost equally well as voom's log-CPM-based differential expression analysts in terms of statistical power and FDR control. With probit pipeline, we have successfully identified the signature genes that are responsible for the clustering of cells on iMac while accounting for the batch effects that are failed to be corrected by the ranking transformation. We are also able to visualize each gene based on how much of their transcriptional variation is explained by

biological and technological group effects. Our probit pipeline, therefore, can be a competitive choice for compiling and working with a comprehensive cell transcriptional atlas whose data has been ranking transformed for batch effect control.

There are a few things we need to be aware of while working with ranking transformations. Firstly, batch effects will eventually change the relative expression level of genes within a sample when the number of genes in the dataset is large. To increase the robustness of ranking transformation's assumption on batch effect correction, we highly suggest that a rough gene filtering need to be performed before the transformation to filter out the genes that are severely confounded by the batch effect. Secondly, a gene that is differentially expressed with respect to its ranking does not always indicate that it is also differentially expressed with respect to its absolute count sizes. Thus, before making any conclusions on the DEGs found in the ranking data, we should think about the narrative of differential expression in terms of relative expression level changes and whether do we believe a 'ranking DEG' is differentially expressed even if its absolute expression level remains the same.

With all the statistical tools, the next step of our study would be predicting the cell types of unannotated samples that are projected on the established reference atlas, and quantifying the confidence of the prediction with scores, which indicate the probabilities of the sample belonging to either of the existing cell populations on the atlas or falling into the category of unknown cell types that haven't been included in the atlas yet.

Method

Dataset

All data used for model evaluation were simulated by the same procedure described by Law et al. (Law et al. 2014). The simulated count data were generated from negative binomial distributions with gene specific baseline means and dispersions that ensemble the characteristics of real data produced by Walter and Eliza Hall Institute of Medical Research. Differentially expressed genes were characterized by multiplying their baseline expected count sizes by the desired true fold-changes.

iMac's data were directly obtained from our previous study and are currently hosted by Stemformatics data portal (Rajab et al. 2019).

Groupwise Variance modelling

We calculated the means $\bar{\mu}_{ijk}$ and standard error $\bar{\delta}_{ijk}$ of ranking data for every Class-Batch groups in each gene, where i, j and k denote i^{th} gene, j^{th} class and k^{th} sequencing platform respectively. A generalized additive model (GAM) is fitted to estimate a groupwise mean-variance trend using $\bar{\delta}_{ijk}{}^{0.5}$ as responses and $\bar{\mu}_{ijk}$ as predictors while weights are assigned according to the number of observations in each group n_{ijk}. This gave us a fitted GAM model

$$y = \hat{\beta}_0 + \hat{s}(x_1) + \varepsilon,\ \varepsilon \sim N\left(0, \hat{\sigma}^2\right)$$

where $\hat{s}$ is an estimated smooth function for modelling the non-linear response-predictor relationships. Ranking value of each observation is then plugged into this model in replace of x_1 to get individual variance predction

Software package

All the analyses done in this article were performed with R version 3.6.1 using package: *ggplot2* v3.2.1, *plotly* v4.9.0 for data visualization (Wickham 2009), *betareg* v3.1-2 for beta regression (Ferrari and Cribari-Neto 2004), *limma* v3.40.2 for batch effect correction and differential expression analysis (Ritchie et al. 2015), *sva* for batch effect correction (Leek et al. 2012), *mgcv* v1.8-30 for generalized additive model (Wood 2004), *nlme* v3.1-40 for linear mixed effect model (Bates et al. 2015), *variancePartition* v1.14.0 for variance partition (Hoffman and Schadt 2016), *stat/base* v3.6.1 for ordinary least square regression, t-tests and ranking transformation, *lmtest* v 0.9-37 for likelihood ratio test and Wald test on linear mixed model (Zeileis and Hothorn 2002).

Reference

Bates, Douglas, Martin Mächler, Ben Bolker, and Steve Walker. 2015. “Fitting Linear Mixed-Effects Models Using **Lme4**.” *Journal of Statistical Software* 67 (1).

Benjamini, Yoav, and Yosef Hochberg. 1995. “Controlling The False Discovery Rate - A Practical And Powerful Approach To Multiple Testing.” *J. Royal Statist. Soc., Series B* 57 (November): 289–300.

Choi, Jarny, Chris M Pacheco, Rowland Mosbergen, Othmar Korn, Tyrone Chen, Isha Nagpal, Steve Englart, Paul W Angel, and Christine A Wells. 2019. “Stemformatics: Visualize and Download Curated Stem Cell Data.” *Nucleic Acids Research* 47 (Database issue): D841–46.

Ferrari, Silvia, and Francisco Cribari-Neto. 2004. “Beta Regression for Modelling Rates and Proportions.” *Journal of Applied Statistics* 31 (7): 799–815.

Goh, Wilson Wen Bin, Wei Wang, and Limsoon Wong. 2017. “Why Batch Effects Matter in Omics Data, and How to Avoid Them.” *Trends in Biotechnology* 35 (6): 498–507.

Hoffman, Gabriel E., and Eric E. Schadt. 2016. “VariancePartition: Interpreting Drivers of Variation in Complex Gene Expression Studies.” *BMC Bioinformatics* 17 (1): 483.

Hwang, Byungjin, Ji Hyun Lee, and Duhee Bang. 2018. “Single-Cell RNA Sequencing Technologies and Bioinformatics Pipelines.” *Experimental & Molecular Medicine* 50 (8): 1–14.

Johnson, W. Evan, Cheng Li, and Ariel Rabinovic. 2007. “Adjusting Batch Effects in Microarray Expression Data Using Empirical Bayes Methods.” *Biostatistics* 8 (1): 118–27.

Law, Charity W., Yunshun Chen, Wei Shi, and Gordon K. Smyth. 2014. “Voom: Precision Weights Unlock Linear Model Analysis Tools for RNA-Seq Read Counts.” *Genome Biology* 15 (2): R29.

Leek, Jeffrey T., W. Evan Johnson, Hilary S. Parker, Andrew E. Jaffe, and John D. Storey. 2012. “The Sva Package for Removing Batch Effects and Other Unwanted Variation in High-Throughput Experiments.” *Bioinformatics* 28 (6): 882–83.

Leek, Jeffrey T., and John D. Storey. 2007. “Capturing Heterogeneity in Gene Expression Studies by Surrogate Variable Analysis.” *PLoS Genetics* 3 (9): 1724–35.

Rajab, Nadia, Paul W. Angel, Mariola Kurowska-Stolarska, Simon Milling, Chris M. Pacheco, Matt Rutar, Jarny Choi, and Christine A. Wells. 2019. “IMAC: An Interactive Atlas to Explore Phenotypic Differences between in Vivo, Ex Vivo and in Vitro-Derived Myeloid Cells in the Stemformatics Platform.” *BioRxiv*, July, 719237.

Risso, Davide, John Ngai, Terence P. Speed, and Sandrine Dudoit. 2014. “Normalization of RNA-Seq Data Using Factor Analysis of Control Genes or Samples.” *Nature Biotechnology* 32 (9): 896–902.

Ritchie, Matthew E., Belinda Phipson, Di Wu, Yifang Hu, Charity W. Law, Wei Shi, and Gordon K. Smyth. 2015. “Limma Powers Differential Expression Analyses for RNA-Sequencing and Microarray Studies.” *Nucleic Acids Research* 43 (7): e47.

Scheicher, Ruth, Andrea Hoelbl-Kovacic, Florian Bellutti, Anca-Sarmiza Tigan, Michaela Prchal-Murphy, Gerwin Heller, Christine Schneckenleithner, et al. 2015. "CDK6 as a Key Regulator of Hematopoietic and Leukemic Stem Cell Activation." *Blood* 125 (1): 90–101.

Sidney, Laura E, Matthew J Branch, Siobhán E Dunphy, Harminder S Dua, and Andrew Hopkinson. 2014. "Concise Review: Evidence for CD34 as a Common Marker for Diverse Progenitors." *Stem Cells (Dayton, Ohio)* 32 (6): 1380–89.

Wickham, Hadley. 2009. *Ggplot2: Elegant Graphics for Data Analysis*. 2nd ed. Springer Publishing Company, Incorporated.

Wood, Simon N. 2004. "Stable and Efficient Multiple Smoothing Parameter Estimation for Generalized Additive Models." *Journal of the American Statistical Association* 99 (467): 673–86.

Zeileis, Achim, and Torsten Hothorn. 2002. "Diagnostic Checking in Regression Relationships" *R News* 2(3), 7-10.

Word Count of Main text: Around 3800 Words (in text citation included).