Assessing Reproducibility of High-throughput Experiments in the Case of Missing Data

Roopali Singh[1], Feipeng Zhang[2], Qunhua Li[1]

[1]Department of Statistics, Pennsylvania State University, University Park, Pennsylvania, USA
[2]School of Economics and Finance, Xi'an Jiaotong University, Xi'an, China

Abstract

High-throughput experiments are an essential part of modern biological and biomedical research. The outcomes of high-throughput biological experiments often have a lot of missing observations due to signals below detection levels. For example, most single-cell RNA-seq (scRNA-seq) protocols experience high levels of dropout due to the small amount of starting material, leading to a majority of reported expression levels being zero. Though missing data contain information about reproducibility, they are often excluded in the reproducibility assessment, potentially generating misleading assessments.

In this paper, we develop a regression model to assess how the reproducibility of high-throughput experiments is affected by the choices of operational factors (e.g., platform or sequencing depth) when a large number of measurements are missing. Using a latent variable approach, we extend correspondence curve regression (CCR), a recently proposed method for assessing the effects of operational factors to reproducibility, to incorporate missing values. Using simulations, we show that our method is more accurate in detecting differences in reproducibility than existing measures of reproducibility. We illustrate the usefulness of our method using a single-cell RNA-seq dataset collected on HCT116 cells. We compare the reproducibility of different library preparation platforms and study the effect of sequencing depth on reproducibility, thereby determining the cost-effective sequencing depth that is required to achieve sufficient reproducibility.

Keywords: Correspondence curve, High throughput experiments, Missing Data, Reproducibility, Sequencing depth, Single cell RNA-seq.

1 Introduction

High-throughput technologies are an essential part of modern biological research. Yet, outputs from high-throughput experiments are quite noisy due to many sources of variation in the experimental and analytic pipelines (referred to as workflows thereafter) that generate the data. The reproducibility of outcomes from a high-throughput workflow across replicated experiments provides important information for establishing confidence in measurements and evaluating the performance of the workflow.

The performance of a high-throughput workflow can be affected by many operational factors, such as experimental platforms or protocols, parameter settings in experimental procedures and labs conducting the experiments. Understanding how these factors affect the reproducibility of the outcome is crucial for identifying potential sources of irreproducibility and designing workflows that produce reliable results.

The primary output from a high-throughput workflow typically consists of a list of candidates (e.g., genes) that are evaluated by the workflow and their scores or measurements (e.g., gene expression) assigned by the workflow. One way to assess the reproducibility of a workflow is to compute the Pearson or Spearman correlation between the scores obtained on two replicate samples from the workflow for all candidates. Another way is to profile how consistently candidates are ranked and selected in replicate experiments across a sequence of selection thresholds. Methods based on this sequential approach include the correspondence at the top (CAT) plot [Irizarry et al., 2005], the correspondence curve [Li et al., 2011], the Irreproducible Discovery Rate (IDR) [Li et al., 2011], the Maximum Rank Reproducibility (MaRR) [Philtron et al., 2018] and the correspondence curve regression (CCR) method [Li and Zhang, 2018]. Despite differences, both types of approaches assume that the same set of candidates are observed across all replicate samples. Though this assumption holds for some high-throughput experiments, it is very common that some candidates are not observed in all replicates due to underdetection. For example, most single-cell RNA-seq protocols experience high levels of dropouts, where a gene is observed at a low or moderate expression level in one cell but is not detected in another cell of the same cell type [Kharchenko et al., 2014]. These dropouts occur due to the low amounts of mRNA in individual cells and inefficient mRNA capture, as well as the stochasticity of mRNA expression, making a vast majority of genes report zero expression levels [Qiu, 2020, Jiang et al., 2020].

Because the aforementioned methods do not take missing values into account in the reproducibility assessment, a common practice for dealing with missing values in reproducibility assessments is to consider only the candidates with non-missing measurements, even though missing values do contribute useful information about reproducibility [Wu et al., 2013]. However, when there are a large number of zeros, this approach can lead to problematic results. For example, if only a small proportion of measurements are non-zero on all replicates and agree well across replicates, but the rest are all observed only on a single replicate, then ignoring zeros can lead to a seemingly high reproducibility despite the large amount of discordance in many candidates. Consider a study of HCT116 cells [Wu et al., 2013] where single-cell RNA-seq libraries are made using TransPlex Kit and SMARTer Ultra Low RNA Kit and whether or not including zeros in the data in the calculation of correlation will lead to substantially different conclusions about the reproducibility of these platforms. For example, when transcripts with zero counts are included (24933 transcripts), the Spearman correlation of the gene expression levels between a pair of cells is lower for TransPlex (0.648) than for SMARTer (0.734), whereas it is the opposite when only transcripts that are expressed in both cells are included (TransPlex: 0.501 for 8859 non-zero transcripts vs SMARTer: 0.460 for 6292 non-zero transcripts). This seems to imply that TransPlex is less reproducible than SMARTer if zeros are included, but the opposite if zeros are excluded. However, if the Pearson correlation is used as the reproducibility criterion, it suggests that TransPlex is more reproducible than SMARTer regardless of the inclusion of zeros (Supplementary Table 6). This inconsistency makes it difficult to make informative comparisons of different workflows about their reproducibility. The same issue also arises when profiling the consistency of candidates' ranking, as different ways to handle missing values can lead to different results. A principled approach

that accounts for missing values in reproducibility assessment is desirable.

In this work, we assess how the reproducibility of high-throughput experiments is affected by the choice of operational factors when a large number of measurements are missing due to under detection. Our approach is built on the correspondence curve regression (CCR) model, which assesses how the probability that a candidate consistently passes selection thresholds in different replicates is affected by operational factors in high-throughput experiments. By evaluating this probability at a series of rank-based selection thresholds through a cumulative link model, CCR summarizes the effects of operational factors on the reproducibility of the workflow across candidates at different significance levels as regression coefficients, allowing the effects on reproducibility to be assessed in a concise and interpretable manner. Our method extends this model by incorporating candidates with unobserved measurements through a latent variable approach. In doing so, this extension allows missing data to be properly accounted for in the assessment of the impact of operational factors on reproducibility.

This article is organized as follows. In Section 2, we first discuss the setup of the problem and provide a brief overview of the correspondence curve regression. Section 3 presents our missing data framework and the estimation procedure. In Section 4, we use simulation studies to evaluate the performance of our method. In Section 5, we apply our method to the dataset from a single-cell RNA-seq study. Section 6 concludes the paper.

2 Method

We consider the output from high-throughput experiments generated from S workflows ($S \geq 2$). All the workflows measure the same underlying biological process, but they differ in certain operational factors such as experimental protocols, measurement platforms, or experimental parameters. We denote the vector of operational factors for the workflow s as $\mathbf{x^s}$. Table 1 shows how a dataset in such a setting would look.

Table 1: Output from S different workflows with 2 replicates each.

	Workflow 1		Workflow 2		...	Workflow S	
	Replicate 1	Replicate 2	Replicate 1	Replicate 2		Replicate 1	Replicate 2
Gene 1	y^1_{11}	y^1_{12}	y^2_{11}	y^2_{12}	$\cdots$	y^S_{11}	y^S_{12}
Gene 2	y^1_{21}	y^1_{22}	y^2_{21}	y^2_{22}	$\cdots$	y^S_{21}	y^S_{22}
Gene 3	y^1_{31}	y^1_{32}	y^2_{31}	y^2_{32}	$\cdots$	y^S_{31}	y^S_{32}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
Gene i	y^1_{i1}	NA	NA	y^2_{i2}	$\cdots$	y^S_{i1}	y^S_{i2}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
Gene j	NA	y^1_{j2}	NA	y^2_{j2}	$\cdots$	y^S_{j1}	NA
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

For each workflow, few replicates are available, such that the reproducibility of the findings identified in the workflow can be assessed across replicates. For each replicate, the output consists of a

list of candidates, such as genes, and their significance scores, which are assigned by the workflow to indicate the strength of evidence for a candidate to be a true signal. The scores can be original measurements or statistics derived from them (e.g., p-value), and we use these scores as our data. Because missing data in high-throughput studies are often created due to underdetection, we assume that the unobserved candidates receive a score lower than all the observed candidates. If a candidate is observed in at least one replicate, then it is termed as a partially observed candidate; if it is under the detection limit in all replicates, then it is termed as a missing candidate. Because missing candidates are completely unobserved, the total number of missing candidates is unknown. Without loss of generality, we assume that a score of large value indicates strong evidence and receives a low value in its rank, i.e. the most significant candidate receives rank one. The case of the reversed order, where the least significant candidate receives rank one, is also discussed in Section 3.3.

We will first consider the case that each workflow has two replicates, as this is the case that most existing reproducibility assessment methods are designed for, then extend it to the case of more replicates in Section 3.4. Let $(Y_1^s, Y_2^s) = \{(y_{11}^s, y_{12}^s), (y_{21}^s, y_{22}^s), \ldots, (y_{n1}^s, y_{n2}^s)\}$ be the significance scores of a sample of n candidates on two replicates, assigned by the workflow s. We suppose that the scores y_{ij}^s $(i = 1, 2, \ldots, n; j = 1, 2)$, are a sample of the random variable Y_j^s from an unknown distribution F_j^s. For notational simplicity, we omit the superscript s while describing a single workflow.

2.1 Correspondence curve regression

Correspondence curve regression (CCR) is a cumulative link regression model that evaluates how covariates affect the reproducibility of the high-throughput experiments [Li and Zhang, 2018]. In such experiments, top candidates are often the targets for downstream analyses; hence, the consistency in the top-ranked candidates is important for reproducibility assessment. One challenge in this assessment is that the assessment results would be confounded with the threshold that is used to determine the cutoff for top ranks. CCR addresses this challenge by measuring the reproducibility at a series of thresholds. Specifically, it models the reproducibility at a given percentage threshold t as the probability that a candidate passes the specific threshold on both replicates,

$$\Psi(t) = P(Y_1 \leq F_1^{-1}(t), Y_2 \leq F_2^{-1}(t)). \tag{1}$$

It then uses (1) as the response variable and models its relationship with operational factors at a series of thresholds using a cumulative regression framework. In doing so, the covariate effects on the response are reflected in the regression coefficients without being linked to any threshold, like cumulative regression models in other contexts. Given a set of pre-specified thresholds, $\mathcal{T} = \{t | 0 < t_1 < \ldots < t_M \leq 1\}$, the baseline model is defined as,

$$g(\Psi(t)) = \sum_{k=1}^{K} \alpha_k h_k(t), \quad t \in \mathcal{T}, \tag{2}$$

where $(h_1, \ldots, h_K)$ are pre-specified functions that describe the structure of the dependence between the replicates and $(\alpha_1, \ldots, \alpha_K)$ are unknown parameters that reflect the level of the dependence, i.e. the reproducibility of the workflow. The value of K and the functional forms of h_k and g can be determined based on the empirical data by an algebraic connection between (2) and Archimedean copula models [Li and Zhang, 2018] (introduced below). Now, let $\mathbf{x}$ be a vector of d covariates corresponding to operational factors for a workflow. For categorical variables, $\mathbf{x}$ will be the associated vector of dummy variables. The model with covariates is defined as,

$$g(\Psi(t|\mathbf{x})) = \sum_{k=1}^{K} \alpha_k h_k(t) + \mathbf{W}(t, \boldsymbol{\beta})'\mathbf{x}, \quad t \in \mathcal{T}, \tag{3}$$

where $\mathbf{W}(t, \boldsymbol{\beta})'\mathbf{x}$ is a linear predictor characterizing the effect of the covariates $\mathbf{x}$ on reproducibility and $\boldsymbol{\beta} = (\beta_{11}, \ldots \beta_{1K}, \ldots, \beta_{d1}, \ldots \beta_{dK})^T$ are unknown coefficients to be estimated. Here, $\mathbf{W}(t, \boldsymbol{\beta}) = (W_1(t, \boldsymbol{\beta}_1), \ldots, W_d(t, \boldsymbol{\beta}_d))^T$ with $W_p(t, \boldsymbol{\beta}_p) = \sum_{k=1}^{K} \beta_{pk} h_k(t)$, and β_{pk} measures the covariate effect on $h_k(t)$ due to x_p for $p = 1, \ldots, d$.

To determine the functional form of (2), *Li and Zhang* [Li and Zhang, 2018] noticed that the response variable in (2) is in fact the diagonal section of a bivariate copula, i.e. $\Psi(t) = C(t, t)$. Here, C is the bivariate cumulative distribution function of uniform random variables T_1, T_2 and $C(t_1, t_2) = P(T_1 \leq t_1, T_2 \leq t_2)$. They showed that g, α_k and h_k in (2) take a canonical functional form if the dependence structure between Y_1 and Y_2 follows the dependence described by a certain Archimedean copula. Archimedean copulas are a parametric class of copulas that are used to model the dependence of multiple random variables. They take the form,

$$\psi(C(t_1, \ldots, t_J); \theta) = \psi(t_1; \theta) + \ldots + \psi(t_J; \theta), \tag{4}$$

where $C(t_1, \ldots, t_J) = P(T_1 \leq t_1, \ldots, T_J \leq t_J)$, θ is an association parameter describing the strength of the dependence between uniform random variables $T_1, \ldots, T_J$ and ψ is a generator function specific to each Archimedean copula. Different Archimedean copulas can describe different dependence structures. Thus, the functional form of (2) can be determined based on the copula that fits the empirical dependence structure of (Y_1, Y_2). To select the proper copula, one can plot the rank scatterplot of (Y_1, Y_2), which describes the empirical dependence structure of (Y_1, Y_2) and identify an Archimedean copula whose shape and tail behavior are similar to what is shown in the rank scatterplot. Figure 1 shows the rank scatterplots of data from two 1-parameter Archimedean copulas, the Gumbel-Hougaard copula and the Nelsen 4.2.12 copula. These two copulas reasonably capture the dependence structure of scores from replicated high-throughput experiments and can be represented in the form (2). See Table 2 for their respective functional forms. These two copulas differ in their shape and tail behavior. The Gumbel-Hougaard copula has higher concordance at the upper tail, whereas the Nelsen 4.2.12 copula has higher concordance at the lower tail. Hence, the Gumbel-Hougaard copula is suitable for the case when top candidates have large signals, and the Nelsen 4.2.12 is suitable for the case when top candidates have small signals, since top candidates usually have higher concordance across

replicates than lowly ranked candidates. Consequently, the missing values due to underdetection tend to appear at the lower tail of the Gumbel-Hougaard copula and the upper tail of the Nelsen 4.2.12 copula, respectively, because missing values usually occur among the weaker signals.

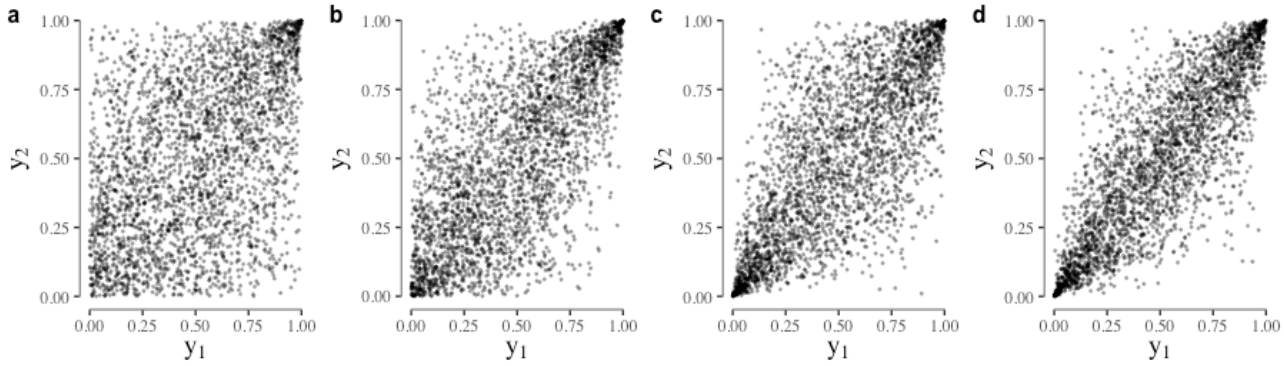

Figure 1: Rank scatterplots of data from the Gumbel-Hougaard copula and the Nelsen 4.2.12 copula with different association parameter θ. (a-b) Gumbel-Hougaard copula with association parameter $\theta = 1.5$ and 2, respectively. (c-d) Nelsen 4.2.12 copula with $\theta = 1.5$ and 2, respectively.

Table 2: Copula function $C(\cdot, \cdot; \theta)$ and the corresponding canonical choices of K, g and $h_k(t)$ for the Gumbel-Hougaard copula and the Nelsen 4.2.12 copula, where θ is the association parameter for the copula models and α's are the coefficients in (3).

Copula name	$C(u, v; \theta)$	K	$g(\Psi(t))$	$h_k(t)$	α_k
Gumbel-Hougaard	$\exp(-[(-\ln u)^\theta + (-\ln v)^\theta]^{1/\theta})$	1	$\log \Psi(t)$	$h_1(t) = \log(t)$	$\alpha_1 = 2^{1/\theta}$
Nelsen 4.2.12	$(1 + [(u^{-1} - 1)^\theta + (v^{-1} - 1)^\theta]^{1/\theta})^{-1}$	2	$\log \frac{\Psi(t)}{1-\Psi(t)}$	$h_1(t) = 1,\ h_2(t) = \log \frac{t}{1-t}$	$\alpha_1 = -\frac{\log 2}{\theta},\ \alpha_2 = 1$

The CCR method has been shown to have good accuracy and power to estimate the difference in reproducibility between different workflows. Hence, we extend it to incorporate missing data.

3 Correspondence Curve Regression with Missing Data

3.1 Missing data framework for upper tail truncation

As mentioned before, some candidates may be unobserved when signals are below the detection level. Consider the case when a small value of y_{ij} indicates strong evidence of a true signal (e.g., p-value) and receives a low rank. Then, a candidate i is observed in replicate j, if its score y_{ij} satisfies $y_{ij} \leq F_j^{-1}(c_j)$. Here, c_j is unknown and needs to be estimated (see Section 3.2). Denote the corresponding observed score as y_{ij}^*, then we can write

$$y_{ij}^* = \begin{cases} y_{ij}, & \text{if } y_{ij} \leq F_j^{-1}(c_j), \\ a_j, & \text{otherwise,} \end{cases} \tag{5}$$

where a_j is a constant that satisfies $a_j > sup\{y_{lj} : y_{lj} \leq F_j^{-1}(c_j); l = 1, 2, \ldots n\}$ for replicate j. For example, if y_{ij} is a p-value and $c_j < 1$, then a_j can be 1. According to the missing status, candidates can be classified into three groups: the ones observed on both replicates, the ones observed on only

one replicate but not the other, and the ones that are not observed on either replicate. They will be referred to as completely observed, partially observed and completely missing, respectively. Without loss of generality, we assume $c_1 \leq c_2$. The sample space of (Y_1, Y_2) can be partitioned by c_1 and c_2 into four categories, $C11, C12, C21$ and $C22$, as shown in Figure 2.

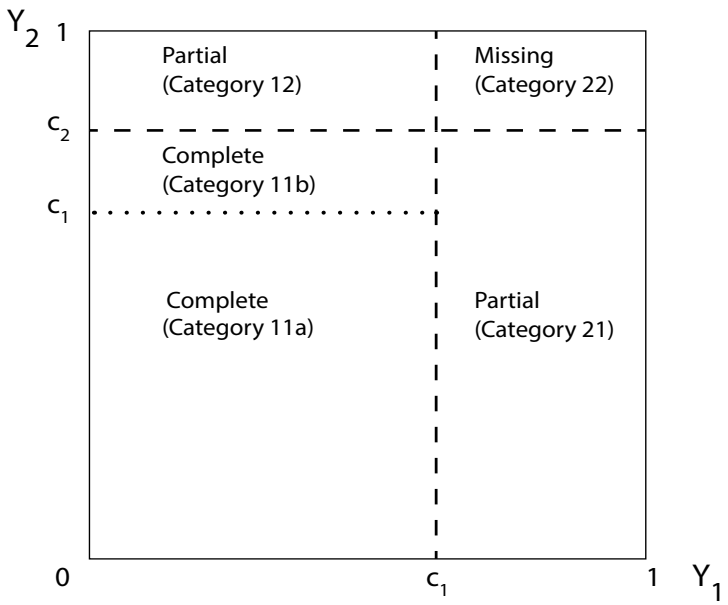

Figure 2: Partition of sample space according to missing status in the case of upper tail truncation.

Category 11: $C11 = \{(Y_1, Y_2) : Y_1 \leq F_1^{-1}(c_1), Y_2 \leq F_2^{-1}(c_2)\}$, i.e. completely observed.

Category 12: $C12 = \{(Y_1, Y_2) : Y_1 \leq F_1^{-1}(c_1), Y_2 > F_2^{-1}(c_2)\}$, i.e. partially observed (observed on Y_1 but missing on Y_2).

Category 21: $C21 = \{(Y_1, Y_2) : Y_1 > F_1^{-1}(c_1), Y_2 \leq F_2^{-1}(c_2)\}$, i.e. partially observed (observed on Y_2 but missing on Y_1).

Category 22: $C22 = \{(Y_1, Y_2) : Y_1 > F_1^{-1}(c_1), Y_2 > F_2^{-1}(c_2)\}$, i.e. completely missing.

Denote the probabilities of falling into these categories as P_{11}, P_{12}, P_{21} and P_{22}, respectively. Then the corresponding probabilities are as follows:

$$
\begin{aligned}
P_{11} &= P(Y_1 \leq F_1^{-1}(c_1), Y_2 \leq F_2^{-1}(c_2)) \qquad (6)\\
P_{12} &= P(Y_1 \leq F_1^{-1}(c_1), Y_2 > F_2^{-1}(c_2)) = P(Y_1 \leq F_1^{-1}(c_1)) - P(Y_1 \leq F_1^{-1}(c_1), Y_2 \leq F_2^{-1}(c_2)) = c_1 - P_{11},\\
P_{21} &= P(Y_1 > F_1^{-1}(c_1), Y_2 \leq F_2^{-1}(c_2)) = P(Y_2 \leq F_1^{-1}(c_2)) - P(Y_1 \leq F_1^{-1}(c_1), Y_2 \leq F_2^{-1}(c_2)) = c_2 - P_{11},\\
P_{22} &= P(Y_1 \leq F_1^{-1}(c_1), Y_2 \leq F_2^{-1}(c_2)) = 1 - P_{11} - P_{12} - P_{21} = 1 - c_1 - c_2 + P_{11}.
\end{aligned}
$$

Denote the numbers of candidates in these categories as n_{11}, n_{12}, n_{21}, and n_{22}, respectively. Note that the occurrences of completely missing candidates are unobserved. Thus, if a category has completely missing candidates, the number of observations in this category is unknown. Hence, the values of n_{11}, n_{12}, n_{21} are known for a dataset, but n_{22} is unknown.

3.2 Likelihood function with missing data due to upper tail truncation

We develop a maximum likelihood approach to estimate the parameters in this model. This approach is built upon the estimation strategy for the CCR model [Li and Zhang, 2018], which partitions observations into non-overlapping ordered categories and fits a multinomial likelihood. We extend this approach to include missing data information using the EM algorithm for truncated data [Dempster et al., 1977]. As we have shown in Section 3.1, when there are missing data, the entire sample space can be partitioned into the category with completely observed candidates (Category 11) and the categories with partial and missing candidates (Category 12, 21, and 22). Therefore, we can conveniently extend the multinomial likelihood in the CCR model by incorporating partial and missing data as additional categories and treating the completely observed category in the similar way as in the CCR model as follows.

For notational simplicity, we first consider the likelihood for a single workflow. Denote the parameters to be estimated as $\phi = \{\boldsymbol{\omega}, c_1, c_2\}$, where $\boldsymbol{\omega} = (\boldsymbol{\alpha}, \boldsymbol{\beta}) = (\alpha_1, \ldots, \alpha_K, \beta_{11}, \ldots, \beta_{dK})$. To compute the multinomial likelihood, we first compute P_{11}, which will be used to compute P_{12}, P_{21} and P_{22} subsequently using (6). To proceed, we first partition $C11$ into two subcategories, $C11^a$ and $C11^b$, where $C11^a = \{(Y_1, Y_2) : Y_1 \leq F_1^{-1}(c_1), Y_2 \leq F_2^{-1}(c_1)\}$ and $C11^b = \{(Y_1, Y_2) : Y_1 \leq F_1^{-1}(c_1), F_1^{-1} < Y_2 \leq F_2^{-1}(c_2)\}$. Denote the probability of these two categories as P_{11a} and P_{11b}, respectively, and the corresponding counts as n_{11a} and n_{11b}, respectively. Then

$$P_{11} = C(c_1, c_2; \theta), P_{11a} = C(c_1, c_1; \theta) = \Psi(c_1), \text{ and } P_{11b} = C(c_1, c_2; \theta) - \Psi(c_1),$$

where $\Psi(\cdot) = g^{-1}(\sum \alpha_k h_k(\cdot) + \mathbf{W}(\cdot, \boldsymbol{\beta})'\mathbf{x})$ and $C(\cdot, \cdot)$ is the copula function corresponding to the regression model, with the copula association parameter θ obtained from the regression coefficient, (α, β), using the correspondence described in the CCR model (See Section 2.1 and Table 2 last column, more details in the supplementary information A.1). For $C11^a$, we partition it further into non-overlapping ordered categories using a series of cutoffs, as in *Li and Zhang*[Li and Zhang, 2018]. Specifically, considering the cutoffs, $0 < t_1 < \ldots < t_M = c_1$, the candidates from each workflow that fall in $C11^a$ can be partitioned into M blocks, $C111, \ldots, C11M$, such that each block consists of the candidates that are deemed reproducible at t_m but not at t_{m-1} (see Figure 2). The observations in block m can be represented as $\mathcal{Y}_m = \{i : \sum_{j=1}^2 I(Y_{ij} \leq F_j{}^{-1}(t_m)|\mathbf{x}) = 2\} \setminus \{i : \sum_{j=1}^2 I(Y_{ij} \leq F_j^{-1}(t_{m-1})|\mathbf{x}) = 2\}$, where Y_{ij} is the score of candidate i on replicate j and F_j is the empirical cumulative distribution function of Y_j, $i = 1, 2, \ldots, n$; $j = 1, 2$. Let $U_{im} = I(i \in \mathcal{Y}_m|\mathbf{x}, c_1, c_2)$ be the binary indicator for candidate i to fall in the mth block. That is, $U_{im} = 1$, if y_{ij} satisfies $t_{m-1} < F_j(y_{ij}) \leq t_m \ \forall j$. Denote the number of observation in block m as n_{11m}; $m = 1, \ldots, M$, and the probability to fall into block m as P_{11m}, then

$$n_{11m} = \sum_{i=1}^{n} U_{im} \text{ and } P_{11m}(\phi) = P(i \in \mathcal{Y}_m|\mathbf{x}) = \Psi(t_m) - \Psi(t_{m-1}).$$

Unlike the case that all data points are observed in *Li and Zhang* [Li and Zhang, 2018], $F_j(\cdot)$ is

not observed directly due to truncation. Yet, we can compute $F_j(y)$ for any observed value y (i.e. $y \leq F_j^{-1}(c_j)$) as $F_j(y) = P(Y_j \leq y) = P(Y_j \leq F_j^{-1}(c_j))P(Y_j \leq y \mid Y_j \leq F_j^{-1}(c_j)) = c_j F_j^*(y)$, where $F_j^*(y) = P(Y_j \leq y \mid Y_j \leq c_j)$ is the cumulative distribution of the truncated distribution. Because the observed values of Y_j are upper truncated at c_j for the jth replicate, $F_j^*(\cdot)$ can be computed from the observed scores.

Following the EM algorithm for truncated data [Dempster et al., 1977], we augment the data to include the number of candidates in $C22$, n_{22}, which is not observed due to truncation. Denote the total number of the candidates in categories $C11$, $C12$ and $C21$ as $n_{obs} = n_{11} + n_{12} + n_{21}$, and the corresponding probability $P(\phi) = P_{11}(\phi) + P_{12}(\phi) + P_{21}(\phi)$. Then $n_{22} \sim$ Negative Binomial$(n_{obs}, P(\phi))$. The complete data likelihood is,

$$L(\phi) = L(\boldsymbol{\omega}, c_1, c_2) = \frac{n_{obs}}{n_{111}!n_{112}!\ldots n_{11M}!n_{11b}!n_{12}!n_{21}!} \frac{(n_{obs} + n_{22} - 1)!}{n_{22}!} \left[\prod_{m=1}^{M} P_{11m}(\phi)^{n_{11m}}\right] P_{11b}(\phi)^{n_{11b}} P_{12}(\phi)^{n_{12}} \tag{7}$$

$$P_{21}(\phi)^{n_{21}}(1 - P(\phi))^{n_{22}}.$$

The detailed EM steps can be found in the supplementary information A.1.

When there are S workflows, the number of observed data from S workflows is denoted as $n_{obs} = n_{obs}^1 + n_{obs}^2 + \ldots + n_{obs}^S = \sum_{s=1}^{S}(n_{11}^s + n_{12}^s + n_{21}^s)$ and the number of candidates in category $C22$ of the sth workflow, $n_{22}^s \sim \text{NB}(n_{obs}^s, P^s(\phi))$. Then, the complete data likelihood for all workflows is,

$$L(\phi) = \prod_{s=1}^{S} L(\boldsymbol{\omega}^s, c_1^s, c_2^s | n_{11}^s, n_{12}^s, n_{21}^s), \tag{8}$$

where $\phi = \{\boldsymbol{\omega}^1, \boldsymbol{\omega}^2, \ldots, \boldsymbol{\omega}^S, c_1^1, c_1^2, \ldots, c_1^S, c_2^1, c_2^2, \ldots, c_2^S\}$, $\boldsymbol{\Omega} = \{\boldsymbol{\omega}^1, \boldsymbol{\omega}^2, \ldots, \boldsymbol{\omega}^S\}$ is the vector of reproducibility parameters for S different workflows and $\mathbf{c_1} = \{c_1^1, c_1^2, \ldots, c_1^S\}$ and $\mathbf{c_2} = \{c_2^1, c_2^2, \ldots, c_2^S\}$ are vectors of truncation parameters for S different workflows. Supplementary information A.1 provides the details of the EM algorithm. We call our proposed method as the Correspondence Curve Regression with Missing Data (mdCCR).

3.3 Inverted missing data framework for lower tail truncation

Sometimes, a large value of y_{ij} indicates strong evidence of a true signal and receives a high rank. In this case, a candidate i is observed in replicate j, if its score y_{ij} satisfies $y_{ij} > F_j^{-1}(c_j)$, and the missing data would appear in the lower tail. Denote the corresponding observed score as y_{ij}^*, then we can write

$$y_{ij}^* = \begin{cases} y_{ij}, & \text{if } y_{ij} > F_j^{-1}(c_j), \\ a_j, & \text{otherwise,} \end{cases} \tag{9}$$

where a_j is a constant that satisfies $a_j < inf\{y_{lj} : y_{lj} > F_j^{-1}(c_j); l = 1, 2, \ldots n\}$ for replicate j. For example, if y_{ij} is a gene expression value and $c_j > 0$, then a_j can be 0. Without loss of generality, we

assume $c_1 \geq c_2$. The whole data set can be partitioned into four categories by c_1 and c_2 as shown in Figure 3. The categories and corresponding probabilities are as follows:

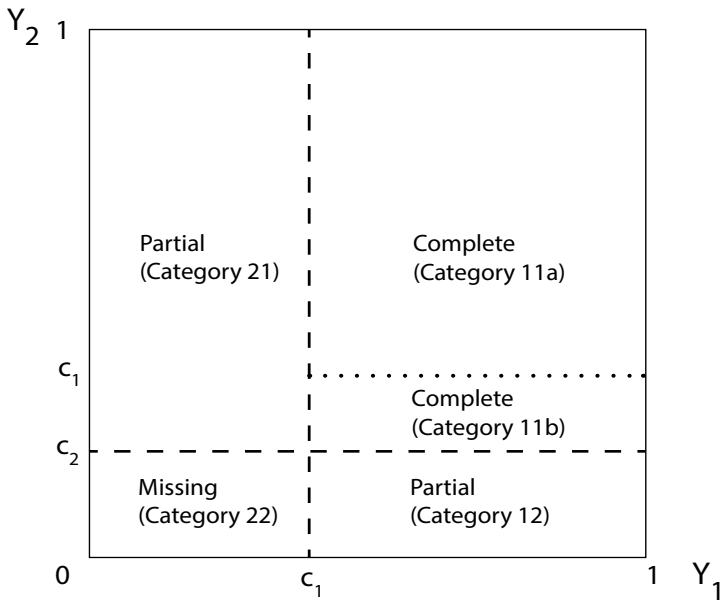

Figure 3: Partition of sample space in the inverted missing data framework (lower tail truncation) according to missing status.

Category 11: $C11 = \{(Y_1, Y_2) : Y_1 > F_1^{-1}(c_1), Y_2 > F_2^{-1}(c_2)\}$ and $P_{11} = 1 - c_1 - c_2 + C(c_1, c_2)$

Category 12: $C12 = \{(Y_1, Y_2) : Y_1 > F_1^{-1}(c_1), Y_2 \leq F_2^{-1}(c_2)\}$ and $P_{12} = 1 - c_1 - P_{11}$

Category 21: $C21 = \{(Y_1, Y_2) : Y_1 \leq F_1^{-1}(c_1), Y_2 > F_2^{-1}(c_2)\}$ and $P_{21} = 1 - c_2 - P_{11}$

Category 22: $C22 = \{(Y_1, Y_2) : Y_1 \leq F_1^{-1}(c_1), Y_2 \leq F_2^{-1}(c_2)\}$ and $P_{22} = c_1 + c_2 + P_{11} - 1$,

where $C(\cdot, \cdot)$ is the copula function corresponding to the regression model, with the copula association parameter obtained from the regression coefficient using the correspondence described in the CCR model.

Denote the numbers of candidates in these categories as n_{11}, n_{12}, n_{21}, and n_{22}, respectively. The estimation for this case can be done in a way similar to the procedure in section 3.2 with a few changes as follows. That is, $C11$ is partitioned into $C11^a$ and $C11^b$, where $C11^a = \{(Y_1, Y_2) : Y_1 > F_1^{-1}(c_1), Y_2 > F_2^{-1}(c_1)\}$ and $C11^b = \{(Y_1, Y_2) : Y_1 > F_1^{-1}(c_1), F_2^{-1}(c_1) \geq Y_2 > F_2^{-1}(c_2)\}$. The corresponding probabilities are

$$P_{11a} = 1 - 2c_1 - \Psi(c_1) \equiv \Psi^*(c_1), \text{ and } P_{11b} = P_{11} - \Psi^*(c_1).$$

Then $C11^a$ is further partitioned with the cutoffs of $c_1 = t_1 < \ldots < t_M = 1$ to form blocks of $\mathcal{Y}_m = \{i : \sum_{j=1}^2 I(Y_{ij} > F_j^{-1}(t_m)|\mathbf{x}) = 2\} \setminus \{i : \sum_{j=1}^2 I(Y_{ij} > F_j^{-1}(t_{m-1})|\mathbf{x}) = 2\}$. The probability of falling in category $C11m$ is $P_{11m}(\phi) = \Psi^*(t_{m-1}) - \Psi^*(t_m)$; $m = 1, 2, \ldots, M$, where $\Psi^*(t_m) = 1 - 2t_m + \Psi(t_m)$.

Because the observed values of Y_j are lower truncated at c_j for the jth replicate, for any observed y (i.e. $y > F_j^{-1}(c_j)$), $F_j(y) = P(Y_j \leq y) = P(Y_j \leq y \mid Y_j > F^{-1}(c_j))(1 - P(Y_j \leq F^{-1}(c_j))) + P(Y_j \leq y \mid Y_j \leq F^{-1}(c_j))P(Y_j \leq F^{-1}(c_j)) = F_j^*(y)(1 - c_j) + 1 \cdot c_j$, where $F_j^*(\cdot)$ is the cumulative distribution of the truncated distribution and can be computed from the observed scores. The complete likelihood for S workflows can be computed using (7) and (8), similar to Section 3.2.

3.4 Framework for more than 2 replicates

Now we extend our model to the case that more than 2 replicate experiments are conducted by a workflow, a situation that is commonly encountered in some high-throughput settings. To assess the reproducibility across multiple replicate experiments, we leverage the information about concordance between pairs of replicates through a pairwise composite likelihood approach. Specifically, suppose that R_0^s replicate experiments are conducted by the sth workflow, then $R^s \equiv \binom{R_0^s}{2}$ pairs of replicates can be formed. For each pair, the likelihood of the concordance between the replicates within the pair takes the form as described in the previous sections. Because each replicate is used in $R^s - 1$ pairs, the likelihood functions for pairwise concordance are not independent across some pairs. Hence, we combine information across pairs using a composite likelihood approach, which is a computationally efficient approach commonly used for approximating joint data models. Composite likelihood assumes independence across individual pairs. It has been shown that this approach produces consistent estimators, though the estimators exhibit some loss of efficiency [Varin et al., 2011]. If R_0^s is moderate or large, one may use one replicate only once in one pair, then $R^s = R_0^s/2$. In this case, the independence assumption in the composite likelihood actually holds.

Specifically, for a given workflow s, we assume that all pairs of replicates share the same set of parameters $(\boldsymbol{\omega}^s, c_1^s, c_2^s)$, because all replicates are generated from the same workflow. Hence, the number of unobserved candidates for the sth workflow from all replicates share the same distribution i.e., $n_{22}^s \sim \text{NB}(n_{obs}^s, P^s(\phi))$, where $n_{obs}^s = \sum_{r=1}^{R^s}(n_{11}^{r,s} + n_{12}^{r,s} + n_{21}^{r,s})$. The complete composite likelihood is,

$$L(\phi) = \prod_{s=1}^{S}\prod_{r=1}^{R^s} L(\boldsymbol{\omega}^s, c_1^s, c_2^s | n_{11}^{r,s}, n_{12}^{r,s}, n_{21}^{r,s}), \tag{10}$$

$\phi = \{\boldsymbol{\omega}^1, \boldsymbol{\omega}^2, \ldots, \boldsymbol{\omega}^S, c_1^1, c_1^2, \ldots, c_1^S, c_2^1, c_2^2, \ldots, c_2^S\}$, $\boldsymbol{\Omega} = \{\boldsymbol{\omega}^1, \boldsymbol{\omega}^2, \ldots, \boldsymbol{\omega}^S\}$ is the vector of reproducibility parameters and $\mathbf{c_1} = \{c_1^1, c_1^2, \ldots, c_1^S\}$, $\mathbf{c_2} = \{c_2^1, c_2^2, \ldots, c_2^S\}$ are vectors of truncation parameters for S different workflows. We estimate the parameters based on the observed data from all the replicates for S workflows using the maximum likelihood estimation via EM algorithm (see details in the supplementary information A.1).

4 Simulation studies

We use simulation studies to examine the performance of our method. In particular, we evaluate how missing data affect the accuracy of estimation, the type I error and the power for detecting differences in reproducibility in the settings resembling high-throughput experiments. To determine if the parameters are accurately estimated from the regression model, we generate (Y_1, Y_2) from the copulas discussed in Section 2 and then estimate the regression coefficient using the corresponding canonical functional choice. In this way, the accuracy of estimation can be determined by comparing the estimated parameters with the association parameter in the copula model.

We simulate the data from two copula models, the Nelsen 4.2.12 and Gumbel-Hougaard copula models. These models fit many high-throughput datasets and have been shown to be suitable for ChIP-seq and microarray datasets [Li and Zhang, 2018]. As mentioned earlier, the two copulas differ in their shape and tail behavior (Figure 1), as well as the location of the missing data in the structure. The Nelsen 4.2.12 copula has missing values at the upper tail and the Gumbel-Hougaard copula has missing values at the lower tail. Hence, the framework in Section 3.1 fits the Nelsen 4.2.12 copula and the inverted framework in Section 3.3 fits the Gumbel-Hougaard copula. After we generate (Y_1, Y_2) from the corresponding copula models, we create missing observations by removing observations under the detection threshold. That is, for each workflow, the missing observations are created by removing $Y_1 > F^{-1}(c_1)$ and $Y_2 > F^{-1}(c_2)$ and vice versa for the inverted case. In all simulations, c_2 is kept equal to c_1. We then estimate the regression coefficient using the corresponding regression model in Section 4. As a comparison, we also perform the analysis using the CCR model, which excludes all the partially observed and missing data and only uses completely observed data.

4.1 Accuracy of estimation and model fitting

To evaluate the accuracy of our estimation procedure, we simulate data from the baseline model (model with a single workflow i.e. (2)) with a single pair of replicates. We simulate from the Nelsen 4.2.12 (case 1) and the Gumbel-Hougaard copulas (case 2). For each copula, we choose the association parameter of the copula, θ, to be 2 and 3, as they reasonably reflect the level of dependence usually found in real data. We vary c_1 such that 10%-40% of the data are missing in each replicate. For each case, we simulate 400 datasets, each of which consists of scores for $n = 10000$ candidates on a pair of replicates. We perform the estimation using our method and the CCR method with $M = 30$ equally spaced cutoff points in $(0, 1)$ for both methods. The standard errors for the estimators in mdCCR are calculated using 400 bootstrap samples.

Case 1: *Nelsen 4.2.12 copula.* The baseline model is $\text{logit}(\Psi(t)) = \alpha + \text{logit}(t)$, where $\alpha = -\log(2)/\theta$, $\theta \in [1, \infty)$. Here, c_1 is chosen to be 0.6, 0.7, 0.8, 0.9, and the proportion of missing data increases as c_1 decreases.

Case 2: *Gumbel-Hougaard copula.* The baseline model is $\log(\Psi(t)) = \alpha \log(t)$, where $\alpha = 2^{1/\theta}$, $\theta \in [1, \infty)$. Here, c_1 is chosen to be 0.1, 0.2, 0.3, 0.4, and the proportion of missing data increases as

c_1 increases.

We use mean integrated squared errors (MISE) as a metric to assess model fitting. MISE for $\Psi(t)$ is the deviation between the empirical correspondence curve ($\hat{\Psi}_n(t)$) and the fitted curve ($\hat{\Psi}(t)$), which is defined as,

$$\text{MISE}(\hat{\Psi}(t)) = \int [\hat{\Psi}(t) - \hat{\Psi}_n(t)]^2 dt. \tag{11}$$

Here, we integrate over all the observed data to keep the comparison of the two methods on the same basis. For the inverted framework, we use MISE for $\Psi^*(t)$, i.e. $\int [\hat{\Psi}^*(t) - \hat{\Psi}^*_n(t)]^2 dt$.

Table 3: Comparison of estimates using mdCCR and CCR for baseline model of the Nelsen 4.2.12 and the Gumbel-Hougaard copula. $\hat{\alpha}$ is the estimate using mdCCR, and $\tilde{\alpha}$ is the estimate using the CCR method.

Nelsen 4.2.12								
Simulation Parameters				mdCCR			CCR method	
θ	α	c_1	n_*	$\hat{\alpha}$	$\hat{c}_1$	MISE	$\tilde{\alpha}$	MISE
2	-0.346	0.9	641.61	-0.343	0.898	0.00007	-0.411	0.00165
		0.8	1388.10	-0.341	0.796	0.00009	-0.446	0.00281
		0.7	2226.22	-0.341	0.694	0.00008	-0.469	0.00346
		0.6	3146.87	-0.342	0.593	0.00008	-0.486	0.00388
3	-0.231	0.9	771.99	-0.229	0.899	0.00007	-0.281	0.00142
		0.8	1604.96	-0.228	0.797	0.00008	-0.315	0.00310
		0.7	2493.41	-0.228	0.695	0.00008	-0.342	0.00441
		0.6	3435.26	-0.228	0.596	0.00007	-0.362	0.00542

Gumbel-Hougaard								
Simulation Parameters				mdCCR			CCR method	
θ	α	c_1	n_*	$\hat{\alpha}$	$\hat{c}_1$	MISE	$\tilde{\alpha}$	MISE
2	1.414	0.1	385.41	1.413	0.100	0.00007	1.485	0.00179
		0.2	1026.94	1.410	0.201	0.00007	1.510	0.00275
		0.3	1821.12	1.409	0.304	0.00008	1.525	0.00332
		0.4	2735.52	1.409	0.406	0.00007	1.535	0.00376
3	1.254	0.1	549.82	1.256	0.101	0.00007	1.331	0.00269
		0.2	1316.45	1.254	0.205	0.00009	1.358	0.00419
		0.3	2192.99	1.254	0.309	0.00010	1.376	0.00518
		0.4	3152.53	1.254	0.411	0.00008	1.389	0.00583

As shown in Table 3, the estimate of α from CCR has a large deviation from the true value, whereas the estimate from our method is close to the true value at all levels of missing data. Our method also estimates the amount of missing data reasonably well. Furthermore, our method shows a much smaller MISE than CCR in all cases, indicating a better fit. The gain of our method over CCR increases with the increase of missing data.

4.2 Power and type I error for detecting differences in reproducibility

Next, we assess the power and type I error for detecting the difference in reproducibility. Here, two workflows are considered, so x takes value 0 or 1. Each workflow has two replicates. We define α_0 and α_1 as the coefficients related to $x = 0$ and $x = 1$, respectively. Similarly, θ_0 and θ_1 are the association parameters related to $x = 0$ and $x = 1$, respectively. We again simulate data from the Nelsen 4.2.12 and the Gumbel-Hougaard copulas with varying proportions of missing data. The corresponding regression model (3) and parameters for each case are summarized below.

Case 1: *Nelsen 4.2.12 copula.* The covariate term is $\mathbf{W}(t, \beta) = \alpha_1 + \text{logit}(t) - (\alpha_0 + \text{logit}(t)) \equiv \beta$ and so the model is $\text{logit}(\Psi(t)) = \alpha_0 + \beta x + \text{logit}(t)$. $\alpha_x = -\log(2)/\theta_x$, where $\theta_x \in [1, \infty)$ for $x = 0, 1$ and $\beta = \alpha_1 - \alpha_0 = -\log(2)/\theta_1 - (-\log(2)/\theta_0)$. This model indicates that the probability that a candidate is reproducibly ranked among top $100t\%$ in workflow $x = 1$ is $\exp(\beta)$ times of that in $x = 0$. Hence, a positive value of β indicates that workflow $x = 1$ is more reproducible than workflow $x = 0$. Here, c_1 is chosen to be 0.6, 0.7, 0.8, 0.9, 1, where a smaller c_1 indicates a larger proportion of missing data and $c_1 = 1$ represents the scenario that all the data are completely observed.

Case 2: *Gumbel-Hougaard copula.* The covariate term is $\mathbf{W}(t, \beta) = \alpha_1 \log(t) - \alpha_0 \log(t) \equiv \beta \log(t)$ and so the model is $\log(\Psi(t)) = (\alpha_0 + \beta x) \log(t)$. $\alpha_x = 2^{1/\theta_x}$, where $\theta_x \in [1, \infty)$ for $x = 0, 1$ and $\beta = \alpha_1 - \alpha_0 = 2^{1/\theta_1} - 2^{1/\theta_0}$. This model indicates that the probability that a candidate is reproducibly ranked among top $100t\%$ in workflow $x = 1$ is t^β times of that in $x = 0$. Hence, a negative value of β indicates that workflow $x = 1$ is more reproducible than workflow $x = 0$. Here, c_1 is chosen to be 0, 0.1, 0.2, 0.3, 0.4, where a larger c_1 indicates a larger proportion of missing data and $c_1 = 0$ represents the scenario that all the data are completely observed.

For assessing the type I error, we set $\theta_0 = \theta_1 = 2$. For assessing the power, we consider two different settings ($\theta_0 = 2$ and $\theta_1 = 1.9$; $\theta_0 = 2$ and $\theta_1 = 1.8$) to evaluate the performance of the method for different effect sizes. For each case, we simulate 400 datasets, each of which consists of $n = 10,000$ pairs of observations. We then fit our model with $M = 30$ equally spaced cutoffs in $(0, 1)$ and test $H_0 : \beta = 0$ using the Wald test at the 5% significance level. Standard errors are calculated using 400 bootstrap samples in case of mdCCR.

As shown in Figure 4, the power of both methods decreases with the increase of the proportion of missing values and the decrease of effect size, when the reproducibility of the two workflows are different. Our method consistently has higher power than CCR for all examined c_1's. The gain in power is more evident with the increase in the proportion of missing data and the decrease in effect size. The type I errors of mdCCR and CCR are both approximately 0.05 for complete data ($c_1 = 1.0$ for Nelsen 4.2.12, or $c_1 = 0.0$ for Gumbel-Hougaard) and then decreases to 0 with the increase in missing data. In addition, the estimates of α_0 and β from our method are consistently more accurate with smaller standard errors than those estimated from the CCR approach (Supplementary Tables 7, 8).

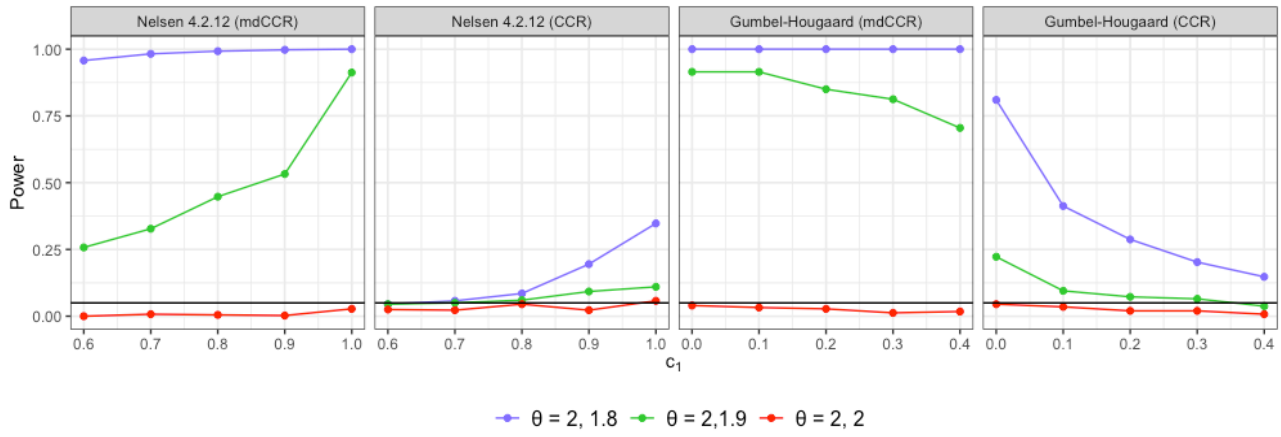

Figure 4: Comparison of power for detecting the difference in reproducibility between two workflows by mdCCR and CCR in simulations. (a, b) are generated from the Nelsen 4.2.12 copula, (c, d) from the Gumbel-Hougaard copula. (a) and (c) are estimated using mdCCR, while (b) and (d) are estimated using the CCR method. Blue ($\theta = 2, 1.8$) and green ($\theta = 2, 1.9$) lines show the power when the two workflows have different reproducibility. Red line ($\theta = 2, 2$) shows the power when the two workflows show no difference in reproducibility, i.e. type I error. The solid black line marks power = 0.05, i.e. the nominal level of type I error. For Nelsen 4.2.12, the amount of missing data increases as c_1 decreases, and for Gumbel-Hougaard, the amount of missing data increases as c_1 increases.

4.3 Power and accuracy with more than 2 replicates

Next, we assess the power and accuracy for detecting the difference in reproducibility when there are more than two replicates per workflow. Again, two workflows are considered, so x takes value 0 or 1 and θ_0 and θ_1 are associated with $x = 0$ and $x = 1$, respectively. We again simulate data from the Gumbel-Hougaard copula and the Nelsen 4.2.12 copula. For each workflow, we simulate R = 1, 3, and 5 pairs of replicates. Because the reproducibility between a pair of replicates from the same workflow may vary across pairs, we incorporate this variation in the simulated data by simulating θ_0 and θ_1 from $N(2, 0.05)$ and $N(1.9, 0.05)$, respectively. For each case, we simulate 400 datasets, each of which consists of $n = 10,000$ pairs of observations with varying c_1. We then fit our model (3) with $M = 30$ equally spaced cutoffs in $(0, 1)$ and compute the likelihood to estimate the parameters using (10). We test $H_0 : \beta = 0$ using the Wald test at the 5% significance level. Standard errors are calculated using 400 bootstrap samples.

As shown in Figure 5, the estimates of α_0 and β are more accurate with smaller standard errors when the number of replicates increases (Supplementary Tables 9, 10). Consequently, the power of mdCCR increases with the increase in the number of replicates. The gain in power is more evident with the increase in the proportion of missing data and the number of replicates.

As previously stated in Section 3.4, composite likelihood estimators may exhibit some loss of efficiency when the assumed independence across individual components is violated. We explore this by repeating the simulation above, except that the replicates are reused in different pairs. Specifically, for each workflow, we simulate $R_0 = 3$ replicates and create $R = 3$ pairs of replicates (i.e. (1,2), (2,3) and (1,3)). These pairs are not entirely independent as each replicate is used in 2 out of 3 pairs. We

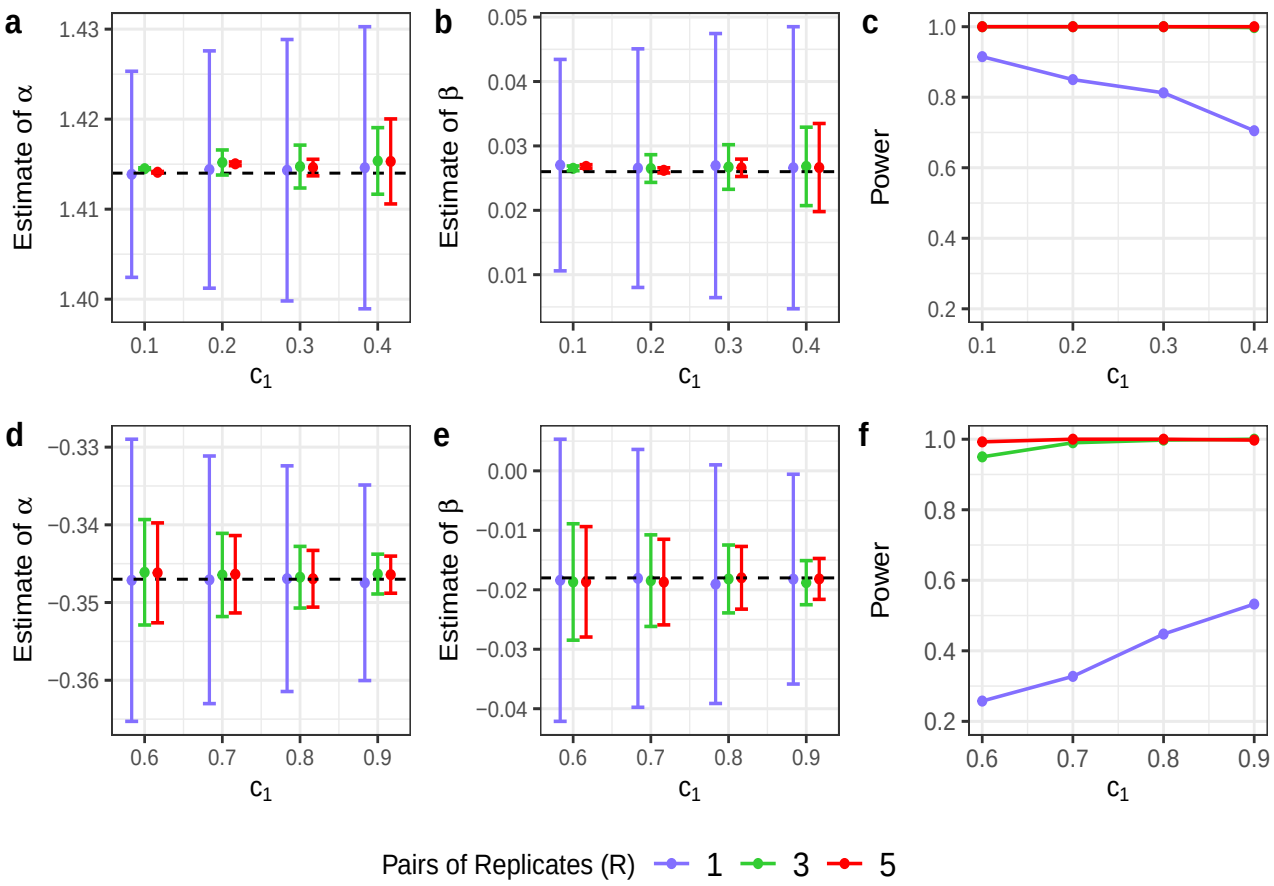

Figure 5: Accuracy in parameter estimation and power of mdCCR for different numbers of replicates. Data are simulated from (a-c) the Gumbel-Hougaard copula and (d-f) the Nelsen 4.2.12 copula, respectively. (a,d) 95% CI of $\hat{\beta}$; (b,e) 95% CI of $\hat{\alpha}$; (c,f) power for detecting the difference in reproducibility between workflows. The dashed line in plots (a-b) and (d-e) signifies the true value of the coefficient. For Gumbel-Hougaard, the amount of missing data increases as c_1 increases and for Nelsen 4.2.12, the amount of missing data increases as c_1 decreases. For (c), the power is the same for replicates = 3 and 5, so the green line and the red line overlap in (c).

estimate the parameters and test the significance of β in the same way as for the case of independent pairs. Supplementary table 11 shows that the estimates of α_0 and β are unbiased for all cases for both copula models, regardless of whether the pairs are dependent or independent. However, it is observed that the variances of $\hat{\alpha}_0$ and $\hat{\beta}$ are larger for dependent pairs than independent pairs, leading to some loss in power. This is likely due to the loss of efficiency for composite likelihood estimators when the independence assumption is violated [Varin et al., 2011].

5 Application: Single-cell RNA-seq study of HCT 116 cells

Single-cell RNA-seq (scRNA-seq) is a powerful high-throughput technology recently developed for measuring gene expression profiles of heterogeneous cell populations for each individual cell. It can reveal regulatory relationships between genes and track the trajectories of distinct cell lineages in development at the single-cell resolution [Hwang et al., 2018]. Several experimental factors affect the

quality of scRNA-seq data, such as RNA library capture, cell quality and sequencing depth. One of the most important factors in the design of scRNA-seq experiments is to determine the desired sequencing depth [Rizzetto et al., 2017]. If sequencing depth is insufficient, many genes may not be detected. Increasing sequencing depth may improve detection but higher sequencing depth inevitably incurs higher costs [Sims et al., 2014]. Therefore, it is important to find the minimum sequencing depth that allows users to obtain reproducible results. Here, we illustrate the usefulness of our method using a single-cell study of HCT116 cells [Wu et al., 2013].

In this study, the authors generated single-cell transcriptomes from cultured HCT116 cells using different platforms. They made 102 single-cell RNA-seq libraries using 3 methods, including two tube-based methods – (i) *TransPlex Kit (Sigma-Aldrich)* and (ii) *SMARTer Ultra Low RNA Kit (Clontech) for cDNA synthesis*, and one microfluidic method – (iii) *SMARTer cDNA synthesis using the C1 microfluidic system.* They evaluated the sensitivity, reproducibility, and accuracy of these single-cell RNA-seq platforms in comparison with bulk total RNA and multiplexed qPCR. They assessed reproducibility of a platform by computing the proportion of genes detected in different replicates and the Pearson correlation of transcript abundance between different pairs of replicates.

As described in Section 1, a major challenge in scRNA-seq data analyses, including the reproducibility assessment, is the large proportion of zeros in the reported gene expression measurements. These zeros occur mainly due to the low amount of mRNA in individual cells and inefficient mRNA capture that are insufficient to produce signals above the detection level, as well as the stochasticity of mRNA expression [Qiu, 2020]. A detailed review on the sources of zeros in single-cell RNA-seq data can be found in [Jiang et al., 2020]. Though several statistical methods have been developed for imputing zeros in scRNA-seq data, a recent study [Sarkar and Stephens, 2021] pointed out that imputing zeros based on the missing mechanisms, such as, missing completely at random or missing at random, should be avoided, because these missing mechanisms, which imply gene expression values are randomly recorded as zeros regardless the underlying true expression level, do not happen in the scRNA-seq measurement process. Our model nevertheless provides a way to take account of the zeros in the reproducibility assessment without imputation. We will use our method to investigate the effect of platforms and sequencing depth on reproducibility and compare with CCR.

5.1 Comparing the reproducibility of tube-based and microfluidic-based libraries

For simplicity, we will refer to method (i) as TransPlex, (ii) as SMARTer, and (iii) as C1. We consider these 3 experimental platforms as different workflows and compare their reproducibility between cells using our method. In this study, C1 has 96 single cells, but TransPlex and SMARTer have only 3 single cells for each platform. Therefore, we formed 3 pairs of replicates for each workflow using 3 single cells for TransPlex, 3 for SMARTer and 6 for C1. On average, each replicate for SMARTer and TransPlex has approximately 50 million raw reads while that for C1 has approximately 3 million raw reads. The average number of zeros in each pair of replicates from the 3 workflows (TransPlex, SMARTer and C1) are 11,794, 14,022 and 15,119, respectively. For CCR, we only used a single pair of single cells for

each platform as the CCR setup does not include the case for more than 2 replicates. We examine the rank scatterplots of the scores on the two biological replicates to select the functional form of the regression model and find that the Gumbel-Hougaard copula is a reasonable fit (see Supplementary Figure B1). Therefore, we fit the following regression model,

$$\log(\Psi(t)) = \alpha_0 \log(t) + \beta_1 \log(t) x_1 + \beta_2 \log(t) x_2 \tag{12}$$

using $M = 30$ equally spaced cutoffs in (0,1), where TransPlex is used as the baseline and x_1, x_2 are dummy variables for SMARTer and C1, respectively.

Table 4: Estimated effect of different libraries by mdCCR and CCR, using Transplex as the baseline.

		mdCCR				CCR	
Platform		Estimate	95% CI	$\hat{c}_1$	$\hat{c}_2$	Estimate	95% CI
TransPlex	α_0	1.375	(1.353, 1.397)	0.648	0.534	1.531	(1.500, 1.563)
SMARTer	β_1	-0.068	(-0.086, -0.050)	0.678	0.585	-0.020	(-0.070, 0.029)
C1	β_2	-0.074	(-0.103, -0.045)	0.663	0.663	-0.049	(-0.097, -0.001)

Table 4 summarizes the results from our analysis. According to the 95% confidence intervals of β's for mdCCR, the three workflows are significantly different in reproducibility. Among them, C1 is the most reproducible and SMARTer is the least reproducible. For example, at $t = 0.05$, the estimated probabilities of being reproducibly reported in the top $100t\%$ are $t^{-0.074} = 1.25$ and $t^{-0.068} = 1.23$ times higher for C1 and SMARTer as for TransPlex, respectively. This agrees with the results in Wu et al. (2013) [Wu et al., 2013], whose assessment also indicated a high proportion of reproducibly detected genes and a high correlation between different pairs of replicates generated by the C1 platform. However, as discussed in Section 1, different types of correlation (with or without zeros) can lead to different conclusions depending upon the choice of pairs of replicates. Nevertheless, our approach uses a principled way to take missing data and all replicates into account and provides quantitative information on the reproducibility at any given threshold. While the estimates of CCR show a similar trend as those of mdCCR, they fail to detect a statistically significant difference in reproducibility between SMARTer and Transplex at the 95% confidence level. This lack of power in CCR is likely related to its ignorance of missing data and restriction to the concordance of a single pair of replicates, as observed in the simulation studies (Section 4).

5.2 Finding the required sequencing depth to achieve sufficient reproducibility

Next, we investigate the effect of sequencing depth on reproducibility. In particular, we aim to find the minimum sequencing depth required to obtain reproducible results. Because Section 5.1 shows that the C1 microfluidic system is the most reproducible platform among the three platforms, we used the C1 data for this analysis. We took 10 cells from C1 data to form 5 pairs of replicates. Their sequencing depth ranges from 1.6M to 5M.

Because differences in sequencing depth across replicates can confound with the reproducibility

analysis, we first downsample all the replicates to a common depth of 1.6M to ensure equality across replicates. To study how sequencing depth affects the reproducibility across replicates, we downsample the reads in the data to create a series of sequencing depths, a common way to study the effect of sequencing depth. Here we downsample the data from 1.6M reads to .8M, 1M, 1.2M, 1.4M reads, and compare the reproducibility across each sequencing depth using our method. See the Appendix for details on the data processing procedure. The number of zeros increases with the reduction of sequencing depth, ranging from 14,000 to 17,000. We consider different downsampling levels as different workflows and use the number of zeros as the starting value for n_*. Rank scatterplots show that the Gumbel-Hougaard copula fits each downsampled dataset well. Using $M = 30$ equally spaced cutoffs in (0,1), we fit the following model,

$$\log(\Psi(t)) = \alpha_0 \log(t) + \beta_1 \log(t)x_1 + \beta_2 \log(t)x_2 + \beta_3 \log(t)x_3 + \beta_4 \log(t)x_4 \tag{13}$$

where the original data (1.6M) is used as the baseline and x_1, x_2, x_3, and x_4 are dummy variables for 1.4M, 1.2M, 1M and 0.8M depth, respectively.

Table 5: Estimated effect of different sequencing depth by mdCCR, using 1.6M as the baseline.

Sequencing Depth	Coefficient	Estimate	95% CI	$\hat{c}_1$	$\hat{c}_2$
Original (1.6M)	α_0	1.3032	(1.2891, 1.3173)	0.677	0.663
1.4M	β_1	0.0026	(-0.0137, 0.0189)	0.683	0.668
1.2M	β_2	0.0072	(-0.0089, 0.0233)	0.694	0.679
1.0M	β_3	0.0095	(-0.0066, 0.0256)	0.699	0.685
0.8M	β_4	0.0161	(0.0002, 0.0320)	0.704	0.689

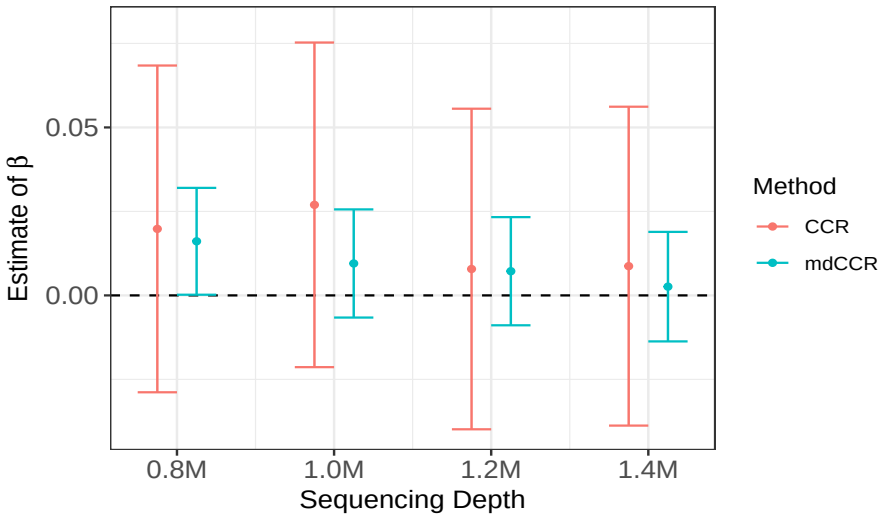

Figure 6: Evaluation of sequencing depth effect on reproducibility for HCT116 single cell RNA-seq data generated using the C1 platform: 95% confidence interval for $\hat{\beta}$ coefficients at different sequencing depths. The results from one pair of replicates (pair 1 in the Supplementary Table 12) are plotted for CCR.

The estimated parameters are summarized in Table 5 and Figure 6. According to the 95% confidence intervals of β's for mdCCR, we observe that when the sequencing depth is below 1M, the work-

flows are significantly less reproducible than the original depth (1.6M). However, after the sequencing depth reaches 1M, the reduction in reproducibility from the original depth is no longer significant. This indicates that a minimal sequencing depth of 1M is required to reach the reproducibility level that is not significantly lower than the original sample of 1.6M reads.

In contrast, unlike mdCCR, which analyzes all replicates at once and provide a unified result, CCR gives different results for different pairs of replicates. Results from CCR using 2 different pairs of replicates are provided in the Supplementary Table 12. It is evident that CCR does not detect the lack of sufficiency when sequencing depth is below 1M.

6 Conclusion

In this paper, we present a regression framework for assessing the influence of experimental factors on the reproducibility of high-throughput experiments when there are missing data. By using a principled approach to take account of missing values, the lack of reproducibility reflected by the missing values is properly accounted for in the assessment of the influence of experimental factors. Using our method, we compared the reproducibility of different single-cell platforms and identified the minimum sequencing depth that can generate sufficiently reproducible results for a scRNA-seq sample. This shows that our method can be used as a principled general approach to understand how influencing factors affect the reproducibility of high-throughput experiments.

Our approach can also be applicable to other experiments where missing values are prevalent, such as Functional Magnetic Resonance Imaging (fMRI) data. The fMRI measures brain activity by detecting changes associated with blood flow using the blood-oxygen-level-dependent (BOLD) contrast. The BOLD signal is frequently corrupted by noise from multiple artifacts such as imaging hardware and temperature, leading to weak or absent signals [Yan et al., 2020, Sammartino et al., 2016]. This hampers studies that investigate whole-brain activity and may result in misinterpretation of findings. The importance of operational factors on the reproducibility of a BOLD signal can be evaluated using our proposed method to help construct reliable data generation protocols.

Acknowledgments

This article is supported by NIH R01GM109453.

Data Availability

The raw data can be obtained from Gene Expression Omnibus using accession code GSE51254.

References

[Afgan et al., 2018] Afgan, E., Baker, D., Batut, B., Van Den Beek, M., Bouvier, D., Čech, M., Chilton, J., Clements, D., Coraor, N., Grüning, B. A., et al. (2018). The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2018 update. Nucleic acids research, 46(W1):W537–W544.

[Dempster et al., 1977] Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. Journal of the Royal Statistical Society: Series B (Methodological), 39(1):1–22.

[ENCODE Project Consortium, 2004] ENCODE Project Consortium (2004). The ENCODE (ENCyclopedia of DNA elements) project . Science, 306(5696):636–640.

[ENCODE Project Consortium, 2012] ENCODE Project Consortium (2012). An integrated encyclopedia of dna elements in the human genome. Nature, 489(7414):57–74.

[Hougaard, 1986] Hougaard, P. (1986). A class of multivanate failure time distributions. Biometrika, 73(3):671–678.

[Hwang et al., 2018] Hwang, B., Lee, J. H., and Bang, D. (2018). Single-cell RNA sequencing technologies and bioinformatics pipelines. Experimental & molecular medicine, 50(8):96.

[Irizarry et al., 2005] Irizarry, R. A., Warren, D., Spencer, F., Kim, I. F., Biswal, S., Frank, B. C., Gabrielson, E., Garcia, J. G., Geoghegan, J., Germino, G., et al. (2005). Multiple-laboratory comparison of microarray platforms. Nature methods, 2(5):345.

[Jiang et al., 2020] Jiang, R., Sun, T., Song, D., and Li, J. J. (2020). Statistics or biology: the zero-inflation controversy about scrna-seq data. bioRxiv, pages 2020–12.

[Kharchenko et al., 2014] Kharchenko, P. V., Silberstein, L., and Scadden, D. T. (2014). Bayesian approach to single-cell differential expression analysis. Nature methods, 11(7):740–742.

[Krueger, 2015] Krueger, F. (2015). Trim Galore: A wrapper tool around Cutadapt and FastQC to consistently apply quality and adapter trimming to FastQ files. https://github.com/FelixKrueger/TrimGalore.

[Langmead and Salzberg, 2012] Langmead, B. and Salzberg, S. L. (2012). Fast gapped-read alignment with Bowtie 2. Nature methods, 9(4):357.

[Li, 2012] Li, H. (2012). Seqtk: Toolkit for processing sequences in FASTA/Q formats. https://github.com/lh3/seqtk.

[Li et al., 2011] Li, Q., Brown, J. B., Huang, H., Bickel, P. J., et al. (2011). Measuring reproducibility of high-throughput experiments. The annals of applied statistics, 5(3):1752–1779.

[Li and Zhang, 2018] Li, Q. and Zhang, F. (2018). A regression framework for assessing covariate effects on the reproducibility of high-throughput experiments. Biometrics, 74(3):803–813.

[Li and Li, 2018] Li, W. V. and Li, J. J. (2018). An accurate and robust imputation method scImpute for single-cell RNA-seq data. Nature communications, 9(1):997.

[Liao et al., 2013] Liao, Y., Smyth, G. K., and Shi, W. (2013). featureCounts: an efficient general purpose program for assigning sequence reads to genomic features. Bioinformatics, 30(7):923–930.

[Martin, 2011] Martin, M. (2011). Cutadapt removes adapter sequences from high-throughput sequencing reads. EMBnet. journal, 17(1):10–12.

[Nelsen, 2006] Nelsen, R. B. (2006). An introduction to copulas. Springer Science & Business Media.

[Philtron et al., 2018] Philtron, D., Lyu, Y., Li, Q., and Ghosh, D. (2018). Maximum rank reproducibility: A nonparametric approach to assessing reproducibility in replicate experiments. Journal of the American Statistical Association, 113(523):1028–1039.

[Qiu, 2020] Qiu, P. (2020). Embracing the dropouts in single-cell rna-seq analysis. Nature communications, 11(1):1–9.

[Rizzetto et al., 2017] Rizzetto, S., Eltahla, A. A., Lin, P., Bull, R., Lloyd, A. R., Ho, J. W., Venturi, V., and Luciani, F. (2017). Impact of sequencing depth and read length on single cell RNA sequencing data of T cells. Scientific Reports, 7(1):12781.

[Sammartino et al., 2016] Sammartino, F., Krishna, V., Sankar, T., Fisico, J., Kalia, S. K., Hodaie, M., Kucharczyk, W., Mikulis, D. J., Crawley, A., and Lozano, A. M. (2016). 3-tesla mri in patients with fully implanted deep brain stimulation devices: a preliminary study in 10 patients. Journal of neurosurgery, 127(4):892–898.

[Sarkar and Stephens, 2021] Sarkar, A. and Stephens, M. (2021). Separating measurement and expression models clarifies confusion in single-cell rna sequencing analysis. Nature Genetics, pages 1–8.

[Schmieder and Edwards, 2011] Schmieder, R. and Edwards, R. (2011). Quality control and preprocessing of metagenomic datasets. Bioinformatics, 27(6):863–864.

[Sims et al., 2014] Sims, D., Sudbery, I., Ilott, N. E., Heger, A., and Ponting, C. P. (2014). Sequencing depth and coverage: key considerations in genomic analyses. Nature Reviews Genetics, 15(2):121.

[Trapnell et al., 2009] Trapnell, C., Pachter, L., and Salzberg, S. L. (2009). TopHat: discovering splice junctions with RNA-Seq. Bioinformatics, 25(9):1105–1111.

[Varin et al., 2011] Varin, C., Reid, N., and Firth, D. (2011). An overview of composite likelihood methods. Statistica Sinica, pages 5–42.

[Wu et al., 2013] Wu, A. R., Neff, N. F., Kalisky, T., Dalerba, P., Treutlein, B., Rothenberg, M. E., Mburu, F. M., Mantalas, G. L., Sim, S., Clarke, M. F., et al. (2013). Quantitative assessment of single-cell RNA-sequencing methods. Nature methods, 11(1):41.

[Yan et al., 2020] Yan, Y., Dahmani, L., Ren, J., Shen, L., Peng, X., Wang, R., He, C., Jiang, C., Gong, C., Tian, Y., et al. (2020). Reconstructing lost bold signal in individual participants using deep machine learning. Nature communications, 11(1):1–13.

[Zhang et al., 2018] Zhang, F., Shen, F., Yang, T., and Li, Q. (2018). Segmented correspondence curve regression model for quantifying reproducibility of high-throughput experiments. arXiv preprint arXiv:1807.00943.

A Supplementary Information

A.1 Estimation Procedure: Maximum likelihood estimation via the EM algorithm

EM algorithm for one pair of replicates from a single workflow First, we consider the case of missing data framework for a single workflow and a single pair of replicates. We first consider the framework in Section 3.1.

As discussed in Sections 3.1 and 3.2, the sample space can be partitioned into categories $C11, C12, C21$ and $C22$, where the observations in $C11$ are completely observed, those in $C12$ and $C21$ are partially observed, and those in $C22$ are completely missing. The numbers of occurrence in these categories, n_{11}, n_{12} and n_{21}, are observed, but n_{22} is missing. The category $C11$ can be partitioned into $C11^a$ and $C11^b$, where $C11^a$ can be further partitioned into M blocks ($C11m, \quad m = 1, 2, \ldots, M$) by the cutoffs, $0 < t_1 < \ldots < t_M = c_1$. The corresponding probabilities for the categories $C11^a, C11^b, C12$, and $C21$ are

$$P_{11}^a = C(c_1, c_1; \theta) = \Psi(c_1), P_{11}^b = C(c_1, c_2; \theta) - \Psi(c_1), P_{12} = c_1 - C(c_1, c_2, ; \theta), P_{21} = c_2 - C(c_1, c_2; \theta),$$

respectively, and that for $C11m$ is $P_{11m}(\phi) = \Psi(t_m) - \Psi(t_{m-1})$, where $\Psi(\cdot) = g^{-1}(\sum \alpha_k h_k(\cdot) + \mathbf{W}(\cdot, \boldsymbol{\beta})'\mathbf{x})$ and $C(\cdot, \cdot; \theta)$ is the copula function corresponding to the regression model, with the copula association parameter θ obtained from its relation with α and β (See Section 2.1). For example, the Nelsen 4.2.12 copula fits this framework and its copula function is defined as $C(u, v; \theta) = \left[1 + \left\{\left(\frac{1}{u} - 1\right)^\theta + \left(\frac{1}{v} - 1\right)^\theta\right\}^{(1/\theta)}\right]^{-1}$, where the association parameter for a single workflow is $\theta = -\frac{\log 2}{\alpha}$ (see Table 2). In the special case of $c_1 = c_2$, the above probabilities can be computed directly from $\Psi(\cdot)$ from the regression model without using the copula function $C(\cdot, \cdot; \theta)$.

Denote the parameters to be estimated as $\phi = \{\boldsymbol{\omega}, c_1, c_2\}$, where $\boldsymbol{\omega} = (\boldsymbol{\alpha}, \boldsymbol{\beta})$. We estimate them using maximum likelihood approach for truncated data via the EM algorithm [Dempster et al., 1977]. Denote the number of observed data from a workflow as $n_{obs} = n_{11} + n_{12} + n_{21}$ and $P(\phi) = P_{11}(\phi) +$

$P_{12}(\phi) + P_{21}(\phi) = c_1 + c_2 - C(c_1, c_2; \theta)$, then the observed data from a workflow can be represented as $\mathbf{z}_{obs} = (\mathbf{y}_1^*, \mathbf{y}_2^*, n_{11}, n_{12}, n_{21})$, where $\mathbf{y}_1^*$ and $\mathbf{y}_2^*$ are observed scores on replicates 1 and 2, respectively.

Following the EM algorithm for truncated data [Dempster et al., 1977], the number of unobserved candidates in category $C22$, $n_{22} \sim$ Negative Binomial$(n_{obs}, P(\phi))$, i.e.,

$$f(n_{22}|\mathbf{z}_{obs}, P(\phi)) = \frac{(n_{obs} + n_{22} - 1)!}{n_{22}!(n_{obs} - 1)!}(1 - P(\phi))^{n_{22}} P(\phi)^{n_{obs}}.$$

The observed data likelihood is,

$$g(\mathbf{z}_{obs}|\phi) = \frac{n_{obs}!}{n_{111}!n_{112}!\ldots n_{11M}!n_{11b}!n_{12}!n_{21}!} \prod_{m=1}^{M} \left(\frac{P_{11m}(\phi)}{P(\phi)}\right)^{n_{11m}} \left(\frac{P_{11b}(\phi)}{P(\phi)}\right)^{n_{11b}} \left(\frac{P_{12}(\phi)}{P(\phi)}\right)^{n_{12}} \left(\frac{P_{21}(\phi)}{P(\phi)}\right)^{n_{21}}.$$

Here $n_{11m} = \sum_{i=1}^{n} U_{im}$, where $U_{im} = I(i \in \mathcal{Y}_m|\mathbf{x}, c_1, c_2)$ and $\mathcal{Y}_m = \{i : \sum_{j=1}^{2} I(Y_{ij} \leq F_j^{-1}(t_m)|\mathbf{x}) = 2\} \setminus \{i : \sum_{j=1}^{2} I(Y_{ij} \leq F_j^{-1}(t_{m-1})|\mathbf{x}) = 2\}$. Because $F_j(\cdot)$ is not observable due to truncation, it is computed as $F_j(y) = P(Y_j \leq y) = P(Y_j \leq F^{-1}(c_j))P(Y_j \leq y \mid Y_j \leq F^{-1}(c_j)) = c_j F_j^*(y)$ for $y \leq F^{-1}(c_j)$, where $F_j^*(y) = P(Y_j \leq y \mid Y_j \leq c_j)$ is cumulative distribution of the truncated distribution and can be computed from the observed scores. (Here we use the empirical distribution of $F_j^*(\cdot)$ in our implementation.) Hence, n_{11m} is a function of t_m, c_1 and c_2, and needs to be updated when the estimates of c_1 and c_2 are updated, unlike the case when all data points are completely observed, where n_{11m} is known and fixed given t_m. Similarly, n_{11b} is also dependent on c_1 and c_2, and needs to be updated when the estimates of c_1 and c_2 are updated.

We augment the data to include the missing counts, n_{22}, then the complete data is $\mathbf{z}_c = \{\mathbf{z}_{obs}, n_{22}\}$. The complete data likelihood is,

$$\begin{aligned} f(\mathbf{z}_c|\phi) &= g(\mathbf{z}_{obs}|\phi) f(n_{22}|\mathbf{z}_{obs}, P(\phi)) \\ &= \frac{n_{obs}!}{n_{111}!n_{112}!\ldots n_{11M}!n_{11b}!n_{12}!n_{21}!} \frac{(n_{obs} + n_{22} - 1)!}{n_{22}!} \left[\prod_{m=1}^{M} P_{11m}(\phi)^{n_{11m}}\right] P_{11b}(\phi)^{n_{11b}} P_{12}(\phi)^{n_{12}} \\ &\quad P_{21}(\phi)^{n_{21}} (1 - P(\phi))^{n_{22}}. \end{aligned}$$

Following the EM methodology for truncation [Dempster et al., 1977], the M-step optimizes the function, $Q(\phi|\phi^{(p)}) = \mathrm{E}[\log f(\mathbf{z}_c|\phi)|\mathbf{z}_{obs}, \phi^{(p)}]$, where

$$\begin{aligned} Q(\phi|\phi^{(p)}) = &\sum_{m=1}^{M} n_{11m} \log(\Psi(t_m) - \Psi(t_{m-1})) + n_{11b} \log(C(c_1, c_2; \theta) - \Psi(c_1)) + n_{12} \log(c_1 - C(c_1, c_2; \theta)) + \\ &n_{21} \log(c_2 - C(c_1, c_2; \theta)) + E(n_{22}) \log(1 - c_1 - c_2 + C(c_1, c_2; \theta)), \end{aligned}$$

and the E-step is defined as,

$$E(n_{22}) = \frac{1 - P(\phi)}{P(\phi)} n_{obs} = \frac{1 - c_1 - c_2 + C(c_1, c_2; \theta)}{c_1 + c_2 - C(c_1, c_2; \theta)} n_{obs}.$$

For the inverted missing data framework in Section 3.3, the probabilities of falling in category $C11m$ are $P_{11m}(\phi) = \Psi^*(t_{m-1}) - \Psi^*(t_m)$; $m = 1, 2, \ldots, M$, where $\Psi^*(t_m) = 1 - 2t_m + \Psi(t_m)$. The probability of falling in category $C11^b$, $C12$ and $C21$ is $P_{11b} = c_1 - c_2 + C(c_1, c_2; \theta) - \Psi(c_1)$, $P_{12}(\phi) = c_2 - C(c_1, c_2; \theta)$, and $P_{21}(\phi) = c_1 - C(c_1, c_2; \theta)$, respectively, where $C(\cdot, \cdot; \theta)$ is the copula function corresponding to the regression model, with the association parameter θ obtained from its relation with α and β. For example, the Gumbel-Hougaard copula fits this framework and its copula function is defined as $C(u, v; \theta) = \exp\left[-\left\{(-\log u)^\theta + (-\log v)^\theta\right\}^{(1/\theta)}\right]$, where the association parameter for a single workflow is $\theta = \frac{\log 2}{\log \alpha}$ (See Table 2). Now, the function for M-step is defined as,

$$Q(\phi|\phi^{(p)}) = \sum_{m=1}^{M} n_{11m} \log(\Psi^*(t_{m-1}) - \Psi^*(t_m)) + n_{11b} \log(c_1 - c_2 + C(c_1, c_2; \theta) - \Psi(c_1)) + n_{12} \log(c_2 - C(c_1, c_2; \theta))$$
$$+ n_{21} \log(c_1 - C(c_1, c_2; \theta)) + E(n_{22}) \log C(c_1, c_2; \theta),$$

and the E-step is defined as,

$$E(n_{22}) = \frac{1 - P(\phi)}{P(\phi)} n_{obs} = \frac{C(c_1, c_2; \theta)}{1 - C(c_1, c_2; \theta)} n_{obs}.$$

EM algorithm for S workflows each with one pair of replicates Now, let us consider the case of S workflows. The number of observed data from S workflows is denoted as $n_{obs} = n^1_{obs} + n^2_{obs} + \ldots + n^S_{obs} = \sum_{s=1}^{S}(n^s_{11} + n^s_{12} + n^s_{21})$ and the number of candidates in category $C22$ of the sth workflow, $n^s_{22} \sim \text{NB}(n^s_{obs}, P^s(\phi))$. Here, the parameters to be estimated are $\phi = \{\boldsymbol{\omega}^1, \boldsymbol{\omega}^2, \ldots, \boldsymbol{\omega}^S, c^1_1, c^2_1, \ldots, c^S_1, c^1_2, c^2_2, \ldots, c^S_2\}$. So, the M-step optimizes the following function,

$$Q(\phi|\phi^{(p)}) = \sum_{s=1}^{S} \left[\sum_{m=1}^{M} n^s_{11m} \log P^s_{11m}(\phi) + n^s_{11b} \log P^s_{11b}(\phi) + n^s_{12} \log P^s_{12}(\phi) + n^s_{21} \log P^s_{21}(\phi) + E(n^s_{22}) \log(1 - P^s(\phi)) \right]$$

and the E-step is defined as,

$$E(n^s_{22}) = \frac{1 - P^s(\phi)}{P^s(\phi)} n^s_{obs}$$

where $P^s(\phi) = P^s_{11}(\phi) + P^s_{12}(\phi) + P^s_{21}(\phi)$ and n^s_{obs} is the total number of candidates observed from workflow s.

EM algorithm for S workflows each with R pairs of replicates When each workflow has R pairs of replicates, the number of observed data from S workflows is denoted as $n_{obs} = \sum_{s=1}^{S} \sum_{r=1}^{R} (n^{r,s}_{11} + n^{r,s}_{12} + n^{r,s}_{21})$. For a given workflow s, we assume that all pairs of replicates share the same set of reproducibility parameters, $\boldsymbol{\omega}^s$, and truncation parameter, c^s_1, because all replicates are generated from the same workflow. Hence, the number of unobserved candidates for s^{th} workflow from all replicates share the same distribution, i.e., $n^s_{22} \sim \text{NB}(n^s_{obs}, P^s(\phi))$. Consequently, the number of parameters to be estimated remain the same i.e., $\phi = \{\boldsymbol{\omega}^1, \boldsymbol{\omega}^2, \ldots, \boldsymbol{\omega}^S, c^1_1, c^2_1, \ldots, c^S_1, c^1_2, c^2_2, \ldots, c^S_2\}$, but these parameters are estimated based on the observations from all the R pairs of replicates. Specifically, the function

to be optimized in the M-step is

$$Q(\phi|\phi^{(p)}) = \sum_{s=1}^{S}\sum_{r=1}^{R}\left[\sum_{m=1}^{M} n_{11m}^{r,s}\log P_{11m}^{s}(\phi) + n_{11b}^{r,s}\log P_{11b}^{s}(\phi) + n_{12}^{r,s}\log P_{12}^{s}(\phi) + n_{21}^{r,s}\log P_{21}^{s}(\phi) + \right.$$
$$E(n_{22}^{s})\log(1-P^{s}(\phi))$$

and the E-step is defined as

$$E(n_{22}^{s}) = \frac{1-P^{s}(\phi)}{P^{s}(\phi)} n_{obs}^{s}$$

where n_{obs}^{s} is the average number of candidates observed from workflow s across R pairs of replicates and $P^{s}(\phi) = P_{11}^{s}(\phi) + P_{12}^{s}(\phi) + P_{21}^{s}(\phi)$. Algorithm 1 describes the general algorithm to estimate ϕ for S workflows with R pairs of replicates. We use bootstrapping method for calculating the standard errors.

Algorithm 1 EM algorithm for estimating ϕ

Initialize $\phi^{(0)} = \{\mathbf{\Omega}^{(0)}, \mathbf{c_1}^{(0)}, \mathbf{c_2}^{(0)}\} = \{\boldsymbol{\omega}^{1,(0)}, \boldsymbol{\omega}^{2,(0)}, \ldots, \boldsymbol{\omega}^{S,(0)},\ c_1^{1,(0)}, c_1^{2,(0)}, \ldots, c_1^{S,(0)},\ c_2^{1,(0)}, c_2^{2,(0)}, \ldots, c_2^{S,(0)}\}$.

(1) E-step: Compute $E(n_{22}^{s}) = \dfrac{1-P^{s}(\phi)}{P^{s}(\phi)} n_{obs}^{s}$ for $s = 1, 2, \ldots, S$.
Update $n_{11m}^{r,s}, n_{11b}^{r,s};\ m = 1, 2, \ldots, M;\ r = 1, 2, \ldots, R;\ s = 1, 2, \ldots, S$.

(2) M-step: Maximize $Q(\phi|\phi^{(p)}) = \mathrm{E}[\log f(\mathbf{z}_c|\phi)|\mathbf{z}_{obs}, \phi^{(p)}]$ w.r.t ϕ.

Repeat E-step and M-step until convergence.

A.2 Downsampling and pre-processing of raw reads

The raw data were obtained from Gene Expression Omnibus using accession code GSE51254. We downsampled the data (randomly selected raw reads from the data) using Seqtk [Li, 2012] and then removed the adapter sequences using Trim Galore [Krueger, 2015]. Alignment and mapping were done using Bowtie 2 [Langmead and Salzberg, 2012] against the reference genome, hg19. Gene expression was measured using featureCounts [Liao et al., 2013]. All the pre-processing steps were done using usegalaxy.org [Afgan et al., 2018].

B Supplementary Figures

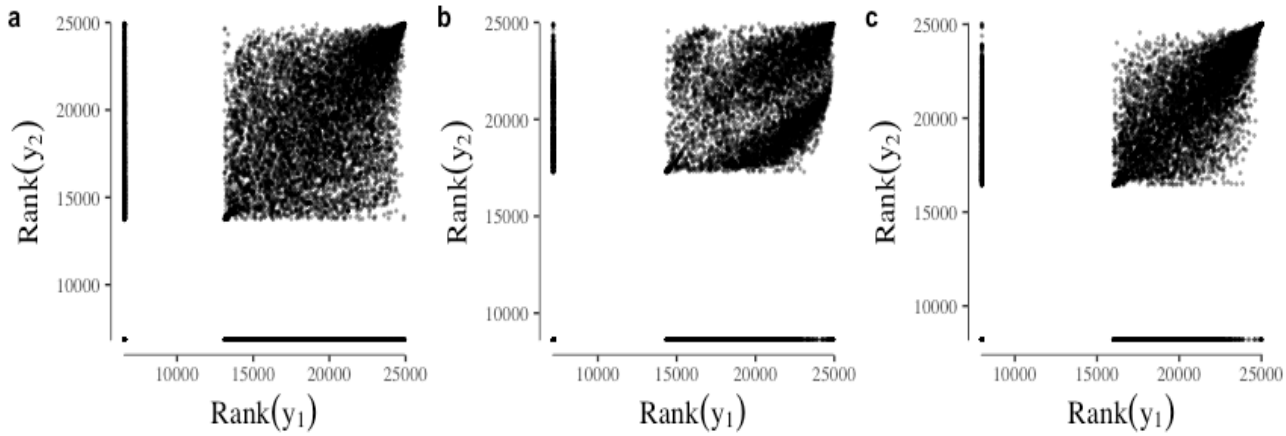

Figure 7: Rank scatterplots of gene expression on two replicates from (a) TransPlex, (b) SMARTer and (c) C1. The dots on the left and bottom sides of the plot (appearing like lines) are the partial data points as the genes are only observed on a single replicate. This resembles the inverted framework described in Section 3.3.

C Supplementary Tables

Table 6: Proportion of zeros and correlation coefficients calculated for 2 replicates from 3 platforms. Correlation coefficients are measured with and without the zeros.

		Spearman's rank correlation		Pearson's correlation	
Platform	Proportion of zeros	With 0	Without 0	With 0	Without 0
TransPlex	0.430	0.648	0.501	0.962	0.984
SMARTer	0.612	0.734	0.460	0.477	0.789
C1	0.568	0.766	0.751	0.685	0.984

Table 7: Estimation of α and β in the simulation study of two workflows for different values of c_1 in the Nelsen 4.2.12 copula, using our proposed method (mdCCR) and CCR method.

True parameters					mdCCR							CCR						
θ_0	θ_1	α_0	β	c_1	$\hat{\alpha}_0$	SE($\hat{\alpha}_0$)	SD($\hat{\alpha}_0$)	$\hat{\beta}$	SE($\hat{\beta}$)	SD($\hat{\beta}$)	Power (β)	$\hat{\alpha}_0$	SE($\hat{\alpha}_0$)	SD($\hat{\alpha}_0$)	$\hat{\beta}$	SE($\hat{\beta}$)	SD($\hat{\beta}$)	Power (β)
.0	2.0	-0.35	0	1.0	-0.35	0.004	0.004	0.00	0.006	0.005	0.03	-0.34	0.017	0.017	-0.00	0.024	0.024	0.05
				0.9	-0.34	0.006	0.005	0.00	0.009	0.007	0.00	0.02	0.018	0.015	0.00	0.025	0.022	0.02
				0.8	-0.34	0.007	0.006	0.00	0.010	0.007	0.01	0.34	0.019	0.015	0.00	0.027	0.021	0.02
				0.7	-0.34	0.008	0.005	0.00	0.011	0.007	0.01	0.62	0.020	0.014	0.00	0.028	0.020	0.01
				0.6	-0.34	0.009	0.006	0.00	0.012	0.008	0.00	0.91	0.022	0.015	0.00	0.031	0.021	0.00
.0	1.9	-0.35	-0.02	1.0	-0.35	0.004	0.004	-0.02	0.006	0.006	0.91	-0.34	0.017	0.016	-0.02	0.025	0.023	0.11
				0.9	-0.35	0.006	0.005	-0.02	0.009	0.007	0.53	0.02	0.018	0.018	-0.01	0.025	0.025	0.09
				0.8	-0.35	0.007	0.005	-0.02	0.010	0.007	0.45	0.33	0.019	0.018	-0.01	0.027	0.027	0.06
				0.7	-0.35	0.008	0.005	-0.02	0.011	0.007	0.33	0.62	0.020	0.019	-0.01	0.029	0.027	0.05
				0.6	-0.35	0.009	0.006	-0.02	0.012	0.008	0.26	0.91	0.022	0.020	-0.01	0.031	0.029	0.05
.0	1.8	-0.35	-0.04	1.0	-0.35	0.004	0.004	-0.04	0.006	0.006	1.00	-0.34	0.017	0.016	-0.04	0.025	0.024	0.35
				0.9	-0.35	0.006	0.005	-0.04	0.009	0.007	1.00	0.02	0.018	0.017	-0.03	0.025	0.025	0.20
				0.8	-0.35	0.007	0.005	-0.04	0.010	0.007	0.99	0.33	0.018	0.019	-0.02	0.027	0.024	0.09
				0.7	-0.35	0.008	0.005	-0.04	0.011	0.007	0.98	0.62	0.020	0.018	-0.02	0.029	0.026	0.06
				0.6	-0.34	0.009	0.006	-0.04	0.012	0.008	0.96	0.91	0.022	0.019	-0.01	0.031	0.026	0.05

Table 8: Estimation of α and β in the simulation study of two workflows for different values of c_1 in the Gumbel-Hougaard copula, using our proposed method (mdCCR) and CCR method.

True parameters					mdCCR							CCR						
θ_0	θ_1	α_0	β	c_1	$\hat{\alpha}_0$	SE($\hat{\alpha}_0$)	SD($\hat{\alpha}_0$)	$\hat{\beta}$	SE($\hat{\beta}$)	SD($\hat{\beta}$)	Power (β)	$\hat{\alpha}_0$	SE($\hat{\alpha}_0$)	SD($\hat{\alpha}_0$)	$\hat{\beta}$	SE($\hat{\beta}$)	SD($\hat{\beta}$)	Power (β)
2.0	2.0	1.41	0	0.0	1.41	0.006	0.006	0.00	0.008	0.008	0.04	1.41	0.014	0.014	0.00	0.020	0.020	0.04
				0.1	1.41	0.006	0.006	0.00	0.008	0.008	0.03	1.96	0.021	0.018	0.00	0.030	0.025	0.02
				0.2	1.41	0.007	0.006	0.00	0.009	0.008	0.03	2.49	0.030	0.024	0.00	0.042	0.034	0.01
				0.3	1.41	0.007	0.007	0.00	0.010	0.008	0.01	3.12	0.041	0.033	0.00	0.058	0.046	0.01
				0.4	1.41	0.008	0.007	0.00	0.011	0.009	0.02	3.94	0.057	0.045	0.00	0.081	0.062	0.01
2.0	1.9	1.41	0.03	0.0	1.41	0.006	0.006	0.03	0.008	0.008	0.92	1.41	0.014	0.014	0.02	0.020	0.021	0.22
				0.1	1.41	0.006	0.005	0.03	0.008	0.007	0.92	1.96	0.021	0.019	0.02	0.031	0.026	0.10
				0.2	1.41	0.007	0.006	0.03	0.009	0.007	0.85	2.49	0.030	0.023	0.03	0.042	0.035	0.07
				0.3	1.41	0.007	0.005	0.03	0.010	0.007	0.81	3.12	0.041	0.030	0.04	0.058	0.049	0.07
				0.4	1.42	0.008	0.006	0.03	0.011	0.008	0.71	3.94	0.057	0.044	0.04	0.082	0.063	0.04
2.0	1.8	1.41	0.06	0.0	1.41	0.006	0.005	0.06	0.008	0.008	1.00	1.41	0.014	0.013	0.06	0.020	0.020	0.81
				0.1	1.41	0.006	0.005	0.06	0.009	0.008	1.00	1.96	0.021	0.018	0.06	0.031	0.026	0.41
				0.2	1.42	0.007	0.005	0.06	0.010	0.007	1.00	2.49	0.030	0.024	0.06	0.043	0.035	0.29
				0.3	1.42	0.007	0.005	0.06	0.011	0.007	1.00	3.12	0.041	0.033	0.08	0.059	0.047	0.20
				0.4	1.42	0.008	0.006	0.06	0.011	0.009	1.00	3.94	0.057	0.045	0.09	0.083	0.065	0.15

Table 9: Estimation of α and β in the simulation study of two workflows with $R = 1, 3$ and 5 pairs of replicates per workflow for different values of c_1 in the Nelsen 4.2.12 copula, using the mdCCR method. Here, $\theta_0 = 2$ and $\theta_1 = 1.9$ i.e. $\alpha_0 = -0.347$ and $\beta = -0.018$.

R	c_1	$\hat{\alpha}_0$	SE($\hat{\alpha}_0$)	SD($\hat{\alpha}_0$)	$\hat{\beta}$	SE($\hat{\beta}$)	SD($\hat{\beta}$)	Power (β)
1	0.9	-0.347	0.006	0.005	-0.018	0.009	0.007	0.53
	0.8	-0.346	0.007	0.005	-0.018	0.010	0.007	0.45
	0.7	-0.346	0.008	0.005	-0.019	0.011	0.007	0.33
	0.6	-0.346	0.009	0.006	-0.019	0.012	0.008	0.26
3	0.9	-0.346	0.001	0.003	-0.019	0.002	0.005	1.00
	0.8	-0.347	0.002	0.003	-0.018	0.003	0.004	0.99
	0.7	-0.346	0.003	0.003	-0.018	0.004	0.005	0.99
	0.6	-0.346	0.003	0.003	-0.019	0.005	0.005	0.95
5	0.9	-0.346	0.001	0.003	-0.018	0.002	0.004	1.00
	0.8	-0.347	0.002	0.003	-0.018	0.003	0.004	1.00
	0.7	-0.346	0.003	0.003	-0.019	0.004	0.004	1.00
	0.6	-0.346	0.003	0.003	-0.019	0.005	0.004	0.99

Table 10: Estimation of α and β in the simulation study of two workflows with $R = 1, 3$ and 5 pairs of replicates per workflow for different values of c_1 in the Gumbel-Hougaard copula, using the mdCCR method. Here, $\theta_0 = 2$ and $\theta_1 = 1.9$ i.e. $\alpha_0 = 1.414$ and $\beta = 0.026$.

R	c_1	$\hat{\alpha}_0$	SE($\hat{\alpha}_0$)	SD($\hat{\alpha}_0$)	$\hat{\beta}$	SE($\hat{\beta}$)	SD($\hat{\beta}$)	Power (β)
1	0.1	1.414	0.006	0.005	0.026	0.008	0.007	0.92
	0.2	1.414	0.007	0.006	0.027	0.09	0.007	0.85
	0.3	1.414	0.007	0.005	0.027	0.010	0.007	0.81
	0.4	1.415	0.008	0.006	0.028	0.011	0.008	0.71
3	0.1	1.414	0.000	0.003	0.027	0.000	0.004	1.00
	0.2	1.415	0.001	0.003	0.026	0.001	0.004	1.00
	0.3	1.415	0.001	0.003	0.027	0.002	0.005	1.00
	0.4	1.415	0.002	0.004	0.027	0.003	0.005	0.99
5	0.1	1.414	0.000	0.003	0.027	0.000	0.004	1.00
	0.2	1.415	0.000	0.003	0.026	0.000	0.004	1.00
	0.3	1.415	0.000	0.003	0.027	0.001	0.003	1.00
	0.4	1.415	0.002	0.003	0.027	0.003	0.004	1.00

Table 11: Estimation of α and β in the simulation study of two workflows with $R = 3$ independent (created using 6 replicates) and dependent (created using 3 replicates only) pairs of replicates per workflow for different values of c_1 in the Nelsen 4.2.12 and Gumbel-Hougaard copula, using the mdCCR method. Here, $\theta_0 = 2$ and $\theta_1 = 1.9$. Relative efficiency $\left(\frac{\text{Variance of parameter estimate from dependent pairs}}{\text{Variance of parameter estimate from independent pairs}}\right)$ is reported.

Nelsen 4.2.12 copula

True parameters			Dependent Pairs					Independent pairs					Relative Efficiency	
α_0	β	c_1	$\hat{\alpha}_0$	SD($\hat{\alpha}_0$)	$\hat{\beta}$	SD($\hat{\beta}$)	Power($\hat{\beta}$)	$\hat{\alpha}_0$	SD($\hat{\alpha}_0$)	$\hat{\beta}$	SD($\hat{\beta}$)	Power($\hat{\beta}$)	RE($\hat{\alpha}_0$)	RE($\hat{\beta}$)
-0.347	-0.018	0.9	-0.347	0.009	-0.019	0.013	0.92	-0.346	0.003	-0.019	0.005	1.00	3.00	2.60
		0.8	-0.346	0.008	-0.019	0.013	0.88	-0.347	0.003	-0.018	0.004	1.00	2.67	3.25
		0.7	-0.347	0.009	-0.018	0.013	0.82	-0.346	0.003	-0.018	0.005	0.99	3.00	2.60
		0.6	-0.347	0.010	-0.019	0.014	0.74	-0.346	0.003	-0.019	0.005	0.95	3.33	2.80

Gumbel-Hougaard copula

True parameters			Dependent Pairs					Independent pairs					Relative Efficiency	
α_0	β	c_1	$\hat{\alpha}_0$	SD($\hat{\alpha}_0$)	$\hat{\beta}$	SD($\hat{\beta}$)	Power($\hat{\beta}$)	$\hat{\alpha}_0$	SD($\hat{\alpha}_0$)	$\hat{\beta}$	SD($\hat{\beta}$)	Power($\hat{\beta}$)	RE($\hat{\alpha}_0$)	RE($\hat{\beta}$)
1.414	0.026	0.1	1.414	0.012	0.028	0.017	1.00	1.414	0.003	0.027	0.004	1.00	4.00	4.25
		0.2	1.414	0.012	0.027	0.018	0.99	1.415	0.003	0.026	0.004	1.00	4.00	4.50
		0.3	1.416	0.012	0.027	0.016	0.99	1.415	0.003	0.027	0.005	1.00	4.00	3.20
		0.4	1.415	0.011	0.027	0.017	0.99	1.415	0.004	0.027	0.005	1.00	2.75	3.40

Table 12: Estimated effect of different sequencing depth by CCR, using 1.6M as the baseline. Table shows two different results based on two different pairs of replicates.

	Sequencing Depth	Coefficient	Estimate	95% CI
Pair 1	Original (1.6M)	α_0	1.410	(1.380,1.447)
	1.4M	β_1	0.009	(-0.039,0.056)
	1.2M	β_2	0.008	(-0.040,0.056)
	1.0M	β_3	0.027	(-0.021, 0.075)
	0.8M	β_4	0.020	(-0.029, 0.068)
Pair 2	Original (1.6M)	α_0	1.500	(1.465,1.537)
	1.4M	β_1	-0.003	(-0.054,0.049)
	1.2M	β_2	0.003	(-0.049,0.055)
	1.0M	β_3	-0.005	(-0.057, 0.047)
	0.8M	β_4	-0.017	(-0.070, 0.035)