Compound models and Pearson residuals for single-cell RNA-seq data without UMIs

Jan Lause[1,2], Christoph Ziegenhain[3], Leonard Hartmanis[4], Philipp Berens[1,2], and Dmitry Kobak[1,2]

[1] *Hertie Institute for AI in Brain Health, University of Tübingen, Germany*
[2] *Tübingen AI Center, Tübingen, Germany*
[3] *Department of Medical Biochemistry & Biophysics, Karolinska Institutet, Sweden*
[4] *Department of Cell & Molecular Biology, Karolinska Institutet, Sweden*
[1] name.surname@uni-tuebingen.de
[3] name.surname@ki.se

July 24, 2024

Abstract

Recent work employed Pearson residuals from Poisson or negative binomial models to normalize UMI data. To extend this approach to non-UMI data, we model the additional amplification step with a compound distribution: we assume that sequenced RNA molecules follow a negative binomial distribution, and are then replicated following an amplification distribution. We show how this model leads to compound Pearson residuals, which yield meaningful gene selection and embeddings of Smart-seq2 datasets. Further, we suggest that amplification distributions across several sequencing protocols can be described by a broken power law. The resulting compound model captures previously unexplained overdispersion and zero-inflation patterns in non-UMI data.

1 Introduction

Single-cell RNA sequencing (scRNA-seq) data are affected by count noise and technical variability due to the total number of sequenced molecules varying from cell to cell. Removing this technical variation by normalization and variance stabilization is an important step in common analysis pipelines (Luecken and Theis, 2019; Heumos et al., 2023). The standard approach for this has been to use the $\log(1 + x/s)$ transformation, where s is a size factor of the cell. While the log-transform often performs well in practice (Ahlmann-Eltze and Huber, 2023), it has well-known theoretical limitations and can produce biased results (Lun, 2018).

Recently, a number of count modelling approaches like `sctransform` (Hafemeister and Satija, 2019), GLM-PCA (Townes et al., 2019), Sanity (Breda et al., 2021), and analytic Pearson residuals (Lause et al., 2021) have been suggested for pre-processing scRNA-seq data. These methods are based on explicit statistical models of the count generation process, rather than on heuristics such as the log-transform. One limitation of all of these methods is that they are tailored to data obtained using sequencing protocols based on unique molecular identifiers (UMIs), and are not appropriate for non-UMI technologies such as Smart-seq2 (Picelli et al., 2013). In this paper, we develop a count model and corresponding analytic Pearson residuals (Lause et al., 2021) for non-UMI sequencing data.

Single-cell sequencing protocols usually require an amplification step by polymerase chain reaction (PCR) to obtain enough starting material for sequencing (Figure 1). The process is imperfect, and different molecules will get amplified to a different extent. As a result, the number of sequenced molecules of a given gene (called *read count*) does not reflect the original number of RNA molecules in the cell.

In UMI protocols, a random DNA sequence called UMI is appended to each original reverse-transcribed RNA molecule prior to the amplification, and is then amplified and sequenced together with it (Islam et al., 2014). Because the UMI uniquely identifies each original molecule, one can later remove amplification duplicates by counting each UMI only once ('de-duplication'), giving rise to the *UMI counts* instead of *read counts* (Figure 1), effectively reducing amplification noise (Grün et al., 2014). Note that the UMI count does not necessarily equal the number of original RNA molecules present in the cell, as

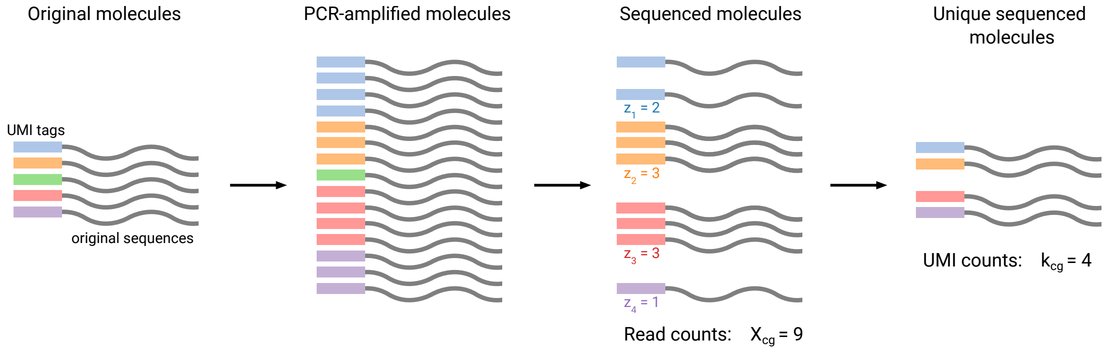

Figure 1: Important quantities in single-cell RNA sequencing. A cell c contains some number (here, 5) of RNA molecules of gene g. In UMI-based protocols, the original molecules are tagged by unique molecular identifiers (UMIs) before PCR amplification and sequencing. Both processes are imperfect, such that not all original molecules get amplified by the same factor, and not all amplified molecules get sequenced. In the end, each of the unique molecules may get sequenced one or several times, which is its *copy number* (z_i). Here z_i values are 2, 3, 3, and 1, and one original molecule (green) is not detected at all. The sum of all copy numbers gives the observed *read count* X_{cg} (here, 9). UMIs allow to compute a *UMI count* k_{cg} by only counting unique sequenced molecules (here, 4). In non-UMI protocols, amplification and sequencing work the same, but the final de-duplication step is not possible, meaning that only X_{cg} is observable. Note that the shown numbers are not to scale: A typical cell might contain on the order of 10^5 RNA molecules across all genes (Ziegenhain et al., 2022); yielding on the order of 10^{10} molecules after amplification; and producing on the order of 10^6 sequenced reads.

some molecules can get lost during sample preparation for the sequencing ('library preparation') or fail to get sequenced due to low capture rate ('depth') in the sequencing step (Figure 1).

While UMI protocols are popular in the scRNA-seq community, non-UMI technologies remain important. Indeed, UMIs only mark one end of the original molecules, so UMI counts are not available for the internal reads (most protocols involve a fragmentation step that cuts each molecule into small fragments prior to sequencing). Full-length sequencing methods, such as Smart-seq2 (Picelli et al., 2013), are typically more sensitive than UMI protocols (Ziegenhain et al., 2017; Ding et al., 2020), and are often used to detect rare cell types (Tasic et al., 2018; Yao et al., 2021) or splicing variants (Feng et al., 2021), or in low-throughput experiments such as Patch-seq (Lipovsek et al., 2021). In the resulting datasets, UMI counts are not available, and computational analysis has to be based on the read counts that still contain amplification-induced variability.

Only few normalization methods have been developed specifically to account for the amplification noise in read counts. The *Census* (Qiu et al., 2017) and *quasi-UMIs* (Townes and Irizarry, 2020) methods are two transformations that are designed to make the shape of the read count distribution approximately match the shape of the UMI count distribution. Afterwards, the transformed data still requires a UMI-like normalization. However, neither Census nor quasi-UMIs derive their transforms from a principled statistical model and rather rely on heuristics.

Here, we develop a new theoretically motivated method for normalization of non-UMI data that explicitly accounts for the amplification noise: *compound Pearson residuals.* We do so by extending the null model behind the analytic Pearson residuals (Lause et al., 2021) from UMI counts to read counts, based on an explicit statistical model for the amplification step. This yields a generative model for read counts that reproduces characteristic patterns of non-UMI data. We demonstrate that our compound Pearson residuals can efficiently normalize complex read count datasets.

2 Results

2.1 Analytic Pearson residuals for normalization of UMI data

In this section we briefly summarize the normalization approach based on Pearson residuals, originally developed by Hafemeister and Satija (2019) for UMI data. Pearson residuals compare the observed data to a null model that captures only technical variability due to count noise and variations in sequencing depth. The null model assumes perfect biological homogeneity, and so any deviation from it suggests biological variability.

Under the null model (Lause et al., 2021), a gene g takes up a certain constant fraction p_g of the total n_c RNA molecules sequenced in cell c, and the observed UMI counts k_{cg} follow a negative binomial (NB) distribution:

$$k_{cg} \sim \mathrm{NB}(\mu_{cg}, \theta), \tag{1}$$

$$\mu_{cg} = n_c p_g, \tag{2}$$

where θ is the inverse overdispersion parameter. Higher values of θ yield smaller variance, and for $\theta = \infty$, the NB distribution reduces to the Poisson distribution. Note that θ in this formulation is shared between all genes; based on negative control UMI data, Lause et al. (2021) argued that θ can be set to $\theta = 100$, which is close to Poisson. Sarkar and Stephens (2021) even suggest pure Poisson as measurement model.

Given an observed UMI count matrix, the maximum likelihood estimate of μ_{cg} is given by:

$$\hat{\mu}_{cg} = \frac{\sum_j k_{cj} \cdot \sum_i k_{ig}}{\sum_{ij} k_{ij}}, \tag{3}$$

which is exact in the Poisson case and holds only approximately in the NB case (Lause et al., 2021). This yields the analytic formula for *UMI Pearson residuals* (difference between observed UMI count values and model prediction, divided by the model standard deviation):

$$R^{\mathrm{UMI}}_{cg} = \frac{k_{cg} - \hat{\mu}_{cg}}{\sqrt{\hat{\mu}_{cg} + \hat{\mu}^2_{cg}/\theta}}, \tag{4}$$

where $\hat{\mu}_{cg} + \hat{\mu}^2_{cg}/\theta$ is the variance of the NB distribution with mean $\hat{\mu}_{cg}$ and overdispersion parameter θ. The variance of Pearson residuals does not depend on p_g, and in a homogeneous dataset is close to 1 for all genes. This ensures variance stabilization across all levels of gene expression.

This algorithm is similar to the one implemented in `sctransform` (Hafemeister and Satija, 2019; Choudhary and Satija, 2021) and is equivalent to a rank-one GLM-PCA (Townes et al., 2019), but it is simpler and faster to compute than either of these methods. When followed by singular value decomposition (SVD), the Poisson ($\theta = \infty$) version of UMI Pearson residuals is also known as correspondence analysis (Hsu and Culhane, 2023).

2.2 Compound Pearson residuals for non-UMI read count data

To apply Pearson residuals to scRNA-seq data without UMIs, we need to change the null model, because read counts do not follow the NB distribution (Svensson, 2020; Cao et al., 2021). As in the UMI case, we assume that the number of unique sequenced RNA molecules k_{cg} follows a Poisson or a NB distribution. However, during the amplification step, each of these k_{cg} unique molecules could have been duplicated multiple times before sequencing (Figure 1). For the i-th unique molecule, we call the number of its sequenced duplicates its *copy number* z_i. We assume that copy numbers follow some distribution Z, which we call *amplification distribution*. Our assumption is that the amplification distribution is the same for all genes and all cells, and only depends on the details of the PCR amplification and the sequencing protocol (see the note below about the variable gene length).

The read count X_{cg} of a given gene g in cell c is thus modeled as the sum of k_{cg} independent and identically distributed (i.i.d.) positive integer copy numbers drawn from Z:

$$X_{cg} = \sum_{i=1}^{k_{cg}} z_i, \tag{5}$$

$$z_i \sim \mathrm{Z} \text{ with } z_i \in \mathbb{N}^+, \tag{6}$$

$$k_{cg} \sim \mathrm{NB}(\mu_{cg}, \theta), \tag{7}$$

$$\mu_{cg} = n_c p_g. \tag{8}$$

For example, $k_{cg} = 4$ means that four unique RNA molecules of gene g were sequenced in a cell c; if their copy numbers were 2, 3, 3, and 1, this would yield the read count value $X_{cg} = 2 + 3 + 3 + 1 = 9$ (c.f. Figure 1). The resulting distribution of X_{cg} can be called *compound NB distribution* (see Methods).

In the above formulation, our model does not explicitly account for gene length. In most sequencing protocols, longer transcripts are cut into more fragments before amplification. This will result in more unique sequenced fragments k_{cg} for longer molecules (Phipson et al., 2017). In our model, this increase amounts to a constant length factor l_g per gene, which can be absorbed by our per-gene expression fraction p_g. The variance of Pearson residuals does not depend on p_g (see above), so for simplicity, we do not explicitly put gene lengths into the model.

To obtain Pearson residuals for this null model, we need to obtain expressions for its mean and variance. The mean of a compound NB distribution is equal to the product of the NB mean μ_{cg} and the mean of the amplification distribution Z:

$$\mathbb{E}[X_{cg}] = \mathbb{E}[Z] \cdot \mathbb{E}[k_{cg}] = \mathbb{E}[Z] \cdot \mu_{cg}. \tag{9}$$

We can use the observed read count matrix X to estimate the means of the maximum likely null compound model as follows:

$$\hat{\mathbb{E}}[X_{cg}] \approx \mathbb{E}[Z] \cdot \hat{\mu}_{cg} = \frac{\mathbb{E}[Z]^2}{\mathbb{E}[Z]} \cdot \frac{\sum_j k_{cj} \cdot \sum_i k_{ig}}{\sum_{ij} k_{ij}} \approx \frac{\sum_j X_{cj} \cdot \sum_i X_{ig}}{\sum_{ij} X_{ij}}. \tag{10}$$

Note that this expression has the same form as Equation 3: the outer product of the row and the column sums of the count matrix, normalized by its total sum.

The mean-variance relationship of the compound NB distribution takes the form (see Methods):

$$\mathrm{Var}[X_{cg}] = \alpha_Z \mathbb{E}[X_{cg}] + \frac{\mathbb{E}[X_{cg}]^2}{\theta}, \tag{11}$$

$$\text{where } \alpha_Z = \mathbb{E}[Z] + \mathbb{FF}[Z] = \mathbb{E}[Z] + \frac{\mathrm{Var}[Z]}{\mathbb{E}[Z]}. \tag{12}$$

This expression is similar to the mean-variance relationship of the NB distribution but contains a scaling parameter α_Z equal to the sum of the mean and the Fano factor (denoted $\mathbb{FF}[Z]$) of Z. Note that in the compound Poisson case ($\theta = \infty$), the compound variance is proportional to the compound mean.

Using these equations, we can compute the Pearson residuals of the compound NB null model, which we call the *compound Pearson residuals*:

$$R_{cg} = \frac{X_{cg} - \hat{\mathbb{E}}[X_{cg}]}{\sqrt{\alpha_Z \hat{\mathbb{E}}[X_{cg}] + \hat{\mathbb{E}}[X_{cg}]^2/\theta}}. \tag{13}$$

Following the UMI case and the arguments in Lause et al. (2021), we set the overdispersion parameter to $\theta = 100$. The scalar α_Z is a function of the mean and variance of the amplification distribution and remains as the only free parameter of the model. Following Hafemeister and Satija (2019) and Lause et al. (2021), we clip the residuals to $[-\sqrt{n}, \sqrt{n}]$, where n is the number of cells in the dataset.

This formalism naturally generalizes the UMI Pearson residuals. Indeed, in the UMI case, each sequenced molecule is counted only once, thanks to the UMIs, meaning that the Z distribution is a delta peak $\delta(1)$ with $\mathbb{E}[Z] = 1$ and $\mathrm{Var}[Z] = 0$, and hence $\alpha_Z = 1$. In this case, Equation 13 reduces to Equation 4.

Conveniently, compound Pearson residuals are equivalent to UMI Pearson residuals of the read count matrix scaled by $1/\alpha_Z$:

$$R_{cg}(X_{cg}; \alpha_Z, \theta) = \frac{(X_{cg} - \hat{\mathbb{E}}[X_{cg}])/\alpha_Z}{\sqrt{\left(\alpha_Z \hat{\mathbb{E}}[X_{cg}] + \hat{\mathbb{E}}[X_{cg}]^2/\theta\right)/\alpha_Z^2}} = R_{cg}^{\mathrm{UMI}}(X_{cg}/\alpha_Z; \theta). \tag{14}$$

Importantly, the necessary scaling factor is not equal to $\mathbb{E}[Z]$, as could be naïvely expected, but rather to $\alpha_Z = \mathbb{E}[Z] + \mathrm{Var}[Z]/\mathbb{E}[Z]$.

Compound Pearson residuals have the same computational complexity as UMI Pearson residuals, and can be computed in seconds even for large datasets with >10 000 cells. The matrix of Pearson residuals is dense and will thus require more memory than the sparse matrix of raw counts. For very large datasets it may be prohibitive to hold the full matrix in memory, but memory demand can be reduced by advanced implementations (Irizarry, 2021) or by subsetting to highly variable genes (Lause et al., 2021), allowing to process datasets with millions of cells.

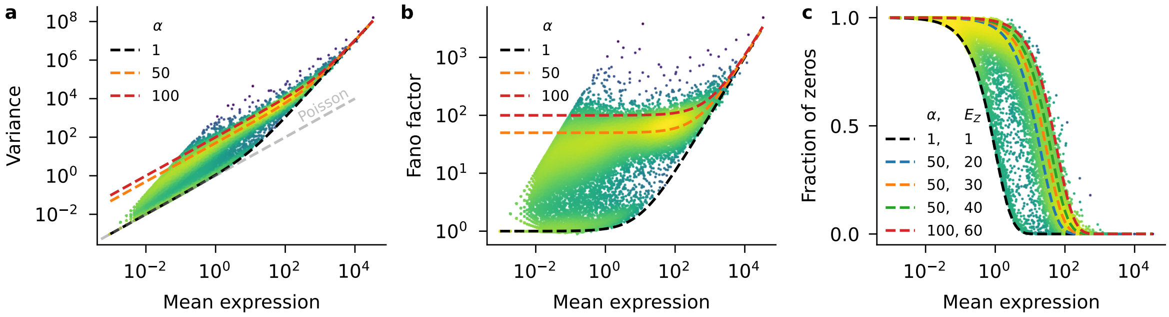

Figure 2: The compound NB model captures statistics of homogeneous read count data. The data are a homogeneous subset of a mouse visual cortex dataset (Tasic et al., 2018) sequenced with Smart-seq2 (*L6 IT VISp Penk Col27a1* cluster; 1 049 cells, 33 914 genes). Each dot represents a gene. Brighter colors indicate higher density of points. Dashed lines show the behavior of the compound negative binomial model ($\theta = 10$). **a:** Mean-variance relationship. Gray line illustrates the Poisson case where mean equals variance. **b:** Relationship between mean expression and Fano factor (variance/mean). **c:** Relationship between mean expression and fraction of zero counts.

2.3 Compound model can fit homogeneous read count data

The compound model introduced above is designed to capture only technical, but not biological variance in non-UMI read count data. Therefore, it should provide a good fit to data that contain little biological variation. To test this, we took scRNA-seq data from adult mouse neocortex sequenced with Smart-seq2 (Tasic et al., 2018), and focused on a subset of cells corresponding to one specific cell type, assuming that there is little biological variability within a cell type. We chose the *L6 IT VISp Penk Col27a1* type (as annotated by the authors of the original study), containing 1 049 excitatory neurons (Figure 2).

The mean-variance relationship across genes (Figure 2a) showed that most genes exhibited overdispersion compared to the Poisson model (gray line, $\alpha_Z = 1$ and $\theta = \infty$). Most genes also showed more variance than expected from a NB model without amplification (black line, $\alpha_Z = 1$ and $\theta = 10$). In contrast, compound models accounting for amplification with $\alpha_Z \in [10, 100]$ (colored lines) were able to approximate the mean-variance relationship for the majority of the genes. Note that we used $\theta = 10$ for illustrations in Figure 2 because this value fit the within-cell-type data better than $\theta = 100$, in agreement with the idea that even biologically homogeneous data can show some additional variability on top of the purely technical variability (Lause et al., 2021). Cell-to-cell variation in sequencing depth also contributed to this increase in overdispersion (Supplementary Figure S1).

The relationship between the mean and the Fano factor across genes (Figure 2b) allowed us to further constrain the amplification parameter α_Z. Indeed, for genes with low average expression, the Fano factor of the read counts is approximately equal to α_Z (Equation 11 and Equation 29 in the Methods). The Fano factors of most genes were bounded by models with $\alpha_Z = 1$ from below (black), and by models with $\alpha_Z = 100$ from above (red). The bulk of the genes followed a model with $\alpha_Z = 50$ (orange).

While knowing α_Z is sufficient to compute Pearson residuals, we can obtain separate estimates of $\mathbb{E}[Z]$ and $\mathrm{Var}[Z]$ by studying the relationship between the average expression and the fraction of zeros. This relationship only depends on $\mathbb{E}[Z]$ (see Methods, Equation 30), allowing to estimate this term directly. We observed that in the Smart-Seq2 data, the fraction of observed zeros decreased with increasing mean expression (Figure 2c). There were more observed zeros than expected from a NB model with $\alpha_Z = 1$ (black), hinting at why read count data have in the past often been modeled using a zero-inflated negative binomial (ZINB) distribution (e.g. Lopez et al., 2018; Chen et al., 2018). Our compound NB model with $\mathbb{E}[Z] \approx 30$ (orange) provided a good qualitative fit to the observed data, without any explicit zero-inflation terms. From here we can compute $\mathbb{FF}[Z] = \alpha_Z - \mathbb{E}[Z] \approx 20$ and hence $\mathrm{Var}[Z] \approx 600$.

The compound NB model with $\alpha_Z = 50$ and $\mathbb{E}[Z] = 30$ described the majority of genes well. However, some genes were instead following the model without any amplification (black line in Figure 2), as if their transcripts were not amplified by the PCR. To understand this pattern, we obtained gene type annotations from `mygene.info`. This revealed that protein-coding genes generally followed our best-fitting compound model (Supplementary Figure S2a–c), while most of the seemingly non-amplified genes were pseudogenes (Supplementary Figure S2d–f). This observation was not limited to the Smart-seq2 data, but also occurred for all sequencing protocols studied in Ziegenhain et al. (2017) (Supplementary Figure S3). Exonic transcript lengths from the `mygene.info` database were shorter for the non-amplified genes (Supplementary Figure S4).

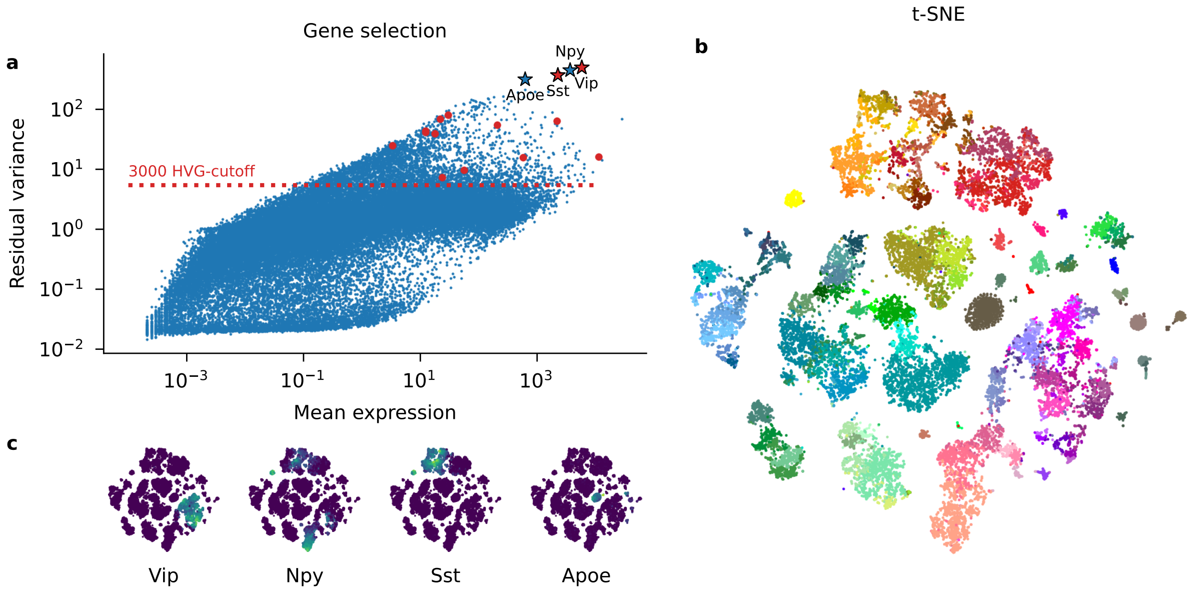

Figure 3: Compound Pearson residuals work well for preprocessing a heterogeneous Smart-seq2 dataset. Here, we used the raw counts of a mouse visual cortex dataset sequenced with Smart-seq2 (Tasic et al., 2018) (23 822 cells, 38 510 genes). We used a compound NB model with amplification parameter $\alpha = 50$ and overdispersion parameter $\theta = 100$. **a:** Highly variable gene (HVG) selection by largest residual variance. Each dot is a gene; genes above the red line were included in the selection of 3 000 HVGs. Stars indicate the top four HVGs shown in panel (c). Red dots and stars correspond to the following well-known marker genes taken from Tasic et al. (2016), from left to right: *Itgam* (microglia), *Bgn* (smooth muscle cells), *Pdgfra* (oligodendrocyte precursors), *Aqp4* (astrocytes), *Flt1* (endothelial cells), *Foxp2* (layer 6 excitatory neurons), *Mog* (oligodendrocytes), *Rorb* (layer 4 excitatory neurons), *Pvalb* (subset of inhibitory neurons), *Slc17a7* (excitatory neurons), *Gad1* (inhibitory neurons), *Sst* (subset of inhibitory neurons), *Vip* (subset of inhibitory neurons), *Snap25* (neurons). **b:** t-SNE embedding on compound Pearson residuals following HVG selection (to 3 000 HVGs) and PCA (down to 1 000 PCs). Each dot is a cell, colored by the original cluster assignments from Tasic et al. (2018). Warm colors: inhibitory neurons. Cold colors: excitatory neurons. Brown and gray colors: non-neural cells. **c:** t-SNE embeddings as in panel (b), colored by expression strength of the four most variable genes according to compound Pearson residual variance. For expression, we show square-root-transformed, depth-normalized counts.

In summary, we showed that our compound NB model fits a biologically homogeneous example dataset. In particular, our model matched the main statistical properties of protein coding genes (mean, variance, and fraction of zeros).

2.4 Compound Pearson residuals for normalization of heterogeneous read count data

Next, we computed compound Pearson residuals with $\alpha_Z = 50$ (Equation 13) to preprocess the entire dataset from Tasic et al. (2018), which is highly heterogeneous and includes both neural and non-neural cells from two areas of the mouse neocortex.

For highly variable gene (HVG) selection, we used the variance of compound Pearson residuals for each gene (Figure 3a). Most genes had residual variance close to 1, indicating that they followed the null model. The interpretation is that those genes did not show biological variability and were not differentially expressed between cell types. In contrast, genes with residual variance $\gg 1$ had more variability than predicted by the null model, implying nontrivial biological variability. For downstream analysis, we selected the 3 000 HVGs with highest residual variances. Among those were well-known marker genes of specific cortical cell types (Figure 3a, red dots). In particular, the four genes with highest residual variance (star symbols in Figure 3a) marked different groups of inhibitory neurons (*Npy*, *Vip*, *Sst*) and astrocytes (*Apoe*). As before, we confirmed that genes with very low residual variances $\ll 1$ were mostly pseudogenes (Supplementary Figure S5).

Next, we used PCA and t-SNE to visualize the single-cell composition of the mouse cortex using compound Pearson residuals of the selected HVGs. The resulting embedding showed rich structure that corresponded well to the cell type annotations originally determined by Tasic et al. (2018) (Figure 3b):

individual cell types formed mostly clearly delineated clusters, while related cell types (having similar colors) mostly stayed close to each other. The expression of most variable genes according to the residual variance was typically localized in one part of the embedding space (Figure 3c).

Calculating compound Pearson residuals requires to set the amplification parameter α_Z. To investigate the influence of this parameter, we computed compound Pearson residuals for a range of α_Z values covering three orders of magnitude (Supplementary Figure S6). We found that for $\alpha \gg 1$, the exact value did not lead to large differences in the HVG selection or t-SNE representation. In contrast, when we used UMI Pearson residuals of the null model without amplification ($\alpha_Z = 1$), the HVG selection failed to include some of the most important marker genes (Supplementary Figure S6a) and the embedding quality visibly degraded (Supplementary Figure S6b). This shows that it is not appropriate to apply the original formulation of UMI Pearson residuals (Lause et al., 2021) to non-UMI data, and that it is important to explicitly account for the PCR-induced variance. Reassuringly, the exact value of the α_Z parameter did not have a large influence on the downstream performance.

We compared our approach to existing methods for read count normalization: qUMI (Townes et al., 2019) and Census (Qiu et al., 2017). Both use heuristics to estimate UMI counts from read count data, and one can then apply standard UMI methods for further processing. We found that both methods, when combined with UMI Pearson residuals, gave results that were similar to our compound Pearson residuals (Supplementary Figure S7). This is unsurprising, as Census amounts to dividing read counts by a cell-specific constant, and compound Pearson residuals are equivalent to UMI Pearson residuals after appropriate scaling of the data matrix (Equation 14). The qUMI transformation is non-linear but gave similar results for our data. Importantly, both Census and qUMI rely on heuristics (see Discussion), while our approach is based on an explicit statistical model.

We also compared this approach to the default preprocessing implemented in the `Scanpy` library (Wolf et al., 2018) based on depth normalization, `log1p()` transform, and `Seurat` HVG selection. We found that many high-expression genes did not get selected by this method, including known marker genes like *Snap25* (Supplementary Figure S8a). The t-SNE embedding based on the default `Scanpy` preprocessing was similar to ours, but arguably showed less local structure (Supplementary Figure S8b). In the absence of ground truth cell labels, it is impossible to assess the representation quality objectively; however, based on the variance of known marker genes, we argue that compound Pearson residuals provide a more meaningful representation of the data.

As noted above, computing compound Pearson residuals was fast. For this dataset with ca. 23 000 cells and 38 000 genes it took ~15 s on a single CPU. The resulting dense matrix of residuals used 3.4 Gb of RAM instead of 1.6 Gb for the sparse matrix of read counts. Census and qUMI had slower runtimes (3 h and 3 min respectively).

2.5 Compound Pearson residuals recover ground truth

To confirm that compound Pearson residuals are indeed able to recover true marker genes and true cell classes, we simulated read count data with known ground truth based on the Tasic et al. (2018) dataset. In our simulations, sequencing depths n_s, gene fractions p_g, and class identities were taken from the real data, and we used a compound model with NB($\theta = 100$) as UMI distribution and a geometric distribution as amplification distribution to simulate counts within each class. The true α_Z in this case is equal to 199 (see Methods).

In *Simulation I*, we allowed only a small set of known marker genes to vary between classes. Compound Pearson residuals showed high residual variance only for those ground truth marker genes (even when α_Z was misspecified, Supplementary Figure S9a–b). In contrast, UMI Pearson residuals ($\alpha_Z = 1$) showed high residual variance for many non-variable genes (Supplementary Figure S9c).

In *Simulation II*, we mirrored the full cluster structure of the data by using cluster-specific p_g values for all genes. Using compound Pearson residuals, we obtained reasonable residual variances and embeddings recovering ground truth clusters (regardless of the exact value of $\alpha \gg 1$, Supplementary Figure S9d–e, g–h). At the same time, UMI Pearson residuals failed to stabilize the variance and incorrectly merged many clusters (Supplementary Figure S9f,i). In summary, both simulation experiments confirmed that compound Pearson residuals can recover true marker genes and cell types.

2.6 The broken zeta distribution as amplification model

So far, we did not specify the amplification distribution Z. Instead, we only characterized its mean and variance through the α_Z parameter. While this was sufficient to compute compound Pearson residuals,

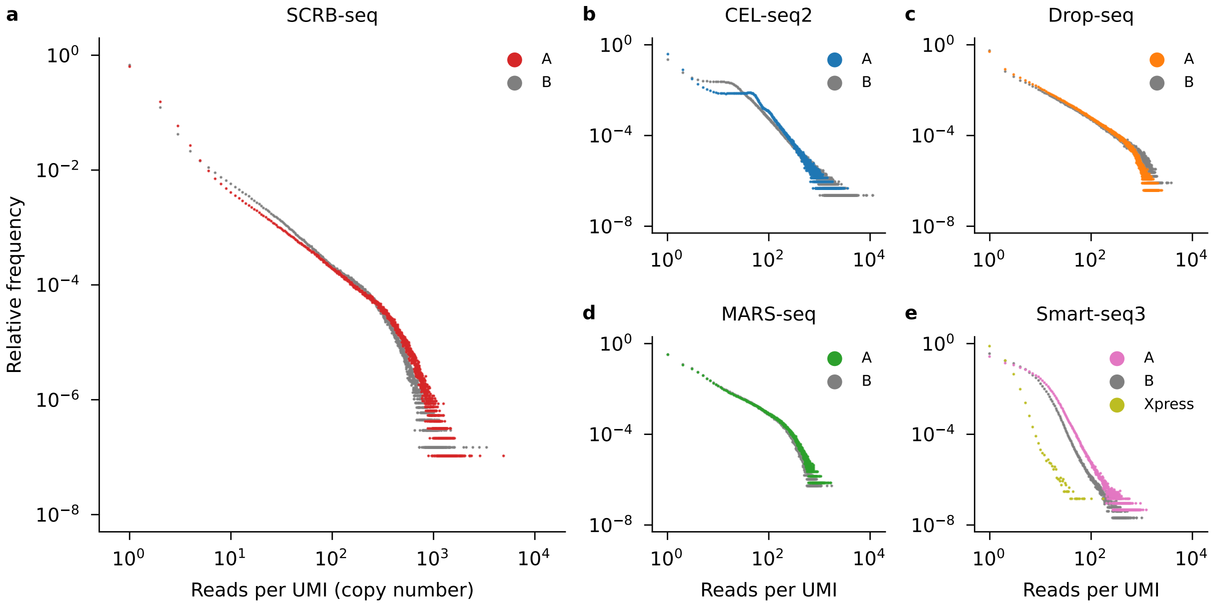

Figure 4: Observed amplification distributions follow a similar shape across protocols. Each panel shows a distribution of UMI copy numbers for a given UMI protocol. **a–d:** SCRB-seq, CEL-seq2, Drop-seq, and MARS-seq data from Ziegenhain et al. (2017). For each protocol two identical runs were performed (A and B). **e:** Smart-seq3 protocols. Data from a single-end experiment (A), a paired-end experiment (B) (Hagemann-Jensen et al., 2020), and a Smart-seq3 Xpress experiment (Hagemann-Jensen et al., 2022).

an explicit amplification distribution is needed for the complete specification of the compound model, enabling likelihood calculation or using it as a generative model. To find an appropriate statistical model, we obtained empirical amplification distributions from experimental data generated with several UMI-based protocols: CEL-seq2, Drop-seq, MARS-seq, and SCRB-seq (Ziegenhain et al., 2017), as well as Smart-Seq3 (Hagemann-Jensen et al., 2020) and Smart-Seq3xpress (Hagemann-Jensen et al., 2022). Together, we analyzed 106.3 million UMIs from 856 cells.

For each UMI barcode, we computed the number of times it occurred in the sequenced reads (its copy number). The normalized histogram of these copy numbers provided an empirical characterization of the amplification distribution Z (Figure 4). Across all sequencing protocols, the copy number histograms in log-log coordinates showed a characteristic elbow shape: Higher copy numbers were less frequent, and the distribution followed two separate decreasing trends in two ranges of copy numbers. This shape can be described by a broken power law, i.e. two separate power laws for low and high copy numbers. The exact shape of the distribution differed between sequencing protocols, leading to different values of mean and variance (Table 1). These values are influenced by how deeply a sample is sequenced and how many cycles of amplification are employed: e.g. the Smart-seq3 Xpress dataset was sequenced shallower than the other two Smart-seq3 datasets (ca. 94 000 vs ca. 572 000–806 000 reads per cell) and with fewer PCR cycles, leading to substantially lower $\mathbb{E}[Z]$ values.

To study how stable the amplification distribution was across cells in the same sample, we computed per-cell estimates of $\mathbb{E}[Z]$ and $\alpha_Z = \mathbb{E}[Z] + \mathbb{FF}[Z]$. The estimates showed some variability across cells (Supplementary Figure S10), but it was small enough that our assumption of the shared amplification parameters seems justified in practice. Moreover, the per-cell estimates of α_Z were correlated with the total number of read counts per cell (Supplementary Figure S11), and this between-cell variability is accounted for in our model in any case.

Note that for all protocols, the empirical distribution of copy numbers was monotonically decreasing, meaning that $z = 1$ was the most likely copy number, despite many cycles of amplification. This may seem counter-intuitive, but previous work (Best et al., 2015) showed that a mechanistic model of the amplification process followed by Poisson sampling at the sequencing stage can give rise to similar copy number histograms with power law behaviour.

As copy numbers are positive integers, we modeled the distribution of copy numbers z with a discrete broken power law. The discrete probability distribution with the mass function following a power law is called zeta distribution. For the broken power law, we adopted the term *broken zeta distribution*, which

Protocol	Run	Cells	UMIs	$\mathbb{E}[Z]$	$\mathbb{FF}[Z]$	α_Z	$\max(z_i)$
CEL-seq2	A	34	2 140 365	27.2	107.5	134.8	3 476
CEL-seq2	B	37	4 303 956	24.4	199.8	224.2	11 092
Drop-seq	A	42	2 506 244	29.6	284.2	313.8	2 463
Drop-seq	B	34	1 272 895	31.1	414.1	445.2	3 718
MARS-seq	A	29	1 342 232	24.1	132.1	156.2	1 624
MARS-seq	B	36	1 903 673	20.9	107.0	128.0	1 719
SCRB-seq	A	39	9 429 371	9.7	218.8	228.5	4 840
SCRB-seq	B	45	6 800 371	9.4	159.3	168.7	3 276
Smart-seq3	A	145	21 549 849	5.4	8.0	13.4	1 216
Smart-seq3	B	319	48 073 893	3.8	4.5	8.3	989
Smart-seq3 Xpress		96	6 940 889	1.3	0.3	1.6	173
Smart-seq2		23 822	—	—	—	—	—

Table 1: Key statistics of the observed amplification distribution across protocols. Top part: Each row in the table corresponds to one of the datasets presented in Figure 4. The number of UMIs shows how many observed copy numbers z_i were used to compute the statistics for that dataset. Bottom row: The Smart-seq2 dataset analyzed in Figure 3, where UMIs were not observed.

we define as having the following probability mass function (PMF):

$$p(z) \propto \begin{cases} z^{-a_1} & z < b \\ b^{a_2-a_1} z^{-a_2} & z \geq b \end{cases}, \tag{15}$$

where $a_1 > 0$ and $a_2 > 0$ are negative slopes of the PMF in log-log coordinates, and $b \in \mathbb{N}$ is the breakpoint between the two slopes. (Figure 5a, inset). We could choose the values for these three parameters such that the broken zeta distribution approximately matched the observed copy number histograms. For example, we obtained a good match for the Drop-seq protocol using $a_1 = 1.4$, $a_2 = 4.5$, and $b = 500$ (Figure 5a).

The fitted model could reproduce several key statistics of the experimental data, such as the mean, the variance, and the Fano factor (Figure 5b–e). However, the broken zeta distribution produced sample maxima that were larger than empirically observed maxima (given the same sample size) (Figure 5d). This is a limitation of the broken zeta model as it tends to allocate non-zero probability mass to very high copy numbers that are not observed in practice. A more flexible model that limits the probability of very large copy numbers could potentially fit the data even better, but we considered the broken zeta distribution sufficient for our purposes.

2.7 Compound NB model with broken zeta amplification captures trends in read count data

The compound NB model (Equations 5–8) together with the broken zeta amplification distribution (Equation 15) provides a generative probabilistic model of the read counts in a biologically homogeneous population. To confirm that the model gives rise to realistic data, we used it to sample read counts and compared them to observed read count histograms in a biologically homogeneous dataset (Figure 6). We used the same dataset as above in Figure 2. For the amplification distribution, we used broken zeta parameter settings ($a_1 = 0.36$, $a_2 = 5.1$, $b = 56$) that led to an amplification model with $\alpha_Z = 50$ and $\mathbb{E}[Z] = 30$, as we showed earlier that these values fit the protein-coding genes in this dataset well (Figure 2).

We found that the empirical count distributions of real genes (Figure 6a–f, grey) could be well matched by the compound NB model (Figure 6a–f, orange). Note that there is only one free parameter per gene in our simulation: p_g, the fraction of RNA molecules taken up by this gene. The entire mass function is then determined by this single parameter. While the compound model did not fit every example gene perfectly (Figure 6b,e), it correctly captured the shapes of the distributions. In particular, for low-expression genes, the compound model predicted strong zero inflation and monotonically decreasing probability of non-zero counts (Figure 6a–b), while predicting a bell-shaped distribution without excess zeros for high expression genes (Figure 6f).

In order to study these patterns more systematically, we fitted a zero-inflated negative binomial (ZINB) distribution to the count histograms of each gene separately. ZINB models have been used

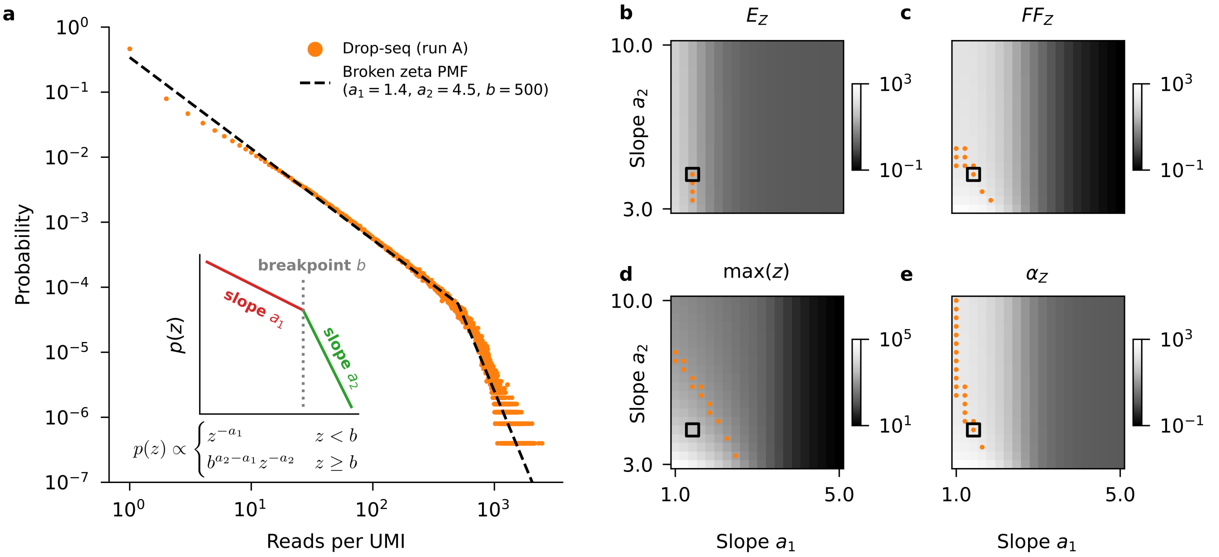

Figure 5: Broken zeta model can fit observed amplification distribution. a: Observed amplification distribution for Drop-seq (same as in Figure 4c) (orange dots) and the PMF of a broken zeta model (black line). The inset illustrates the parameters of the broken zeta model. **b:** The heatmap shows how the two slope parameters affect the mean of the broken zeta distribution. For each combination of a_1 and a_2, we sampled from a broken zeta model with these slope parameters and fixed $b = 500$. We used the same sample size as in the observed data in (a). The orange dots highlight the simulations yielding the sample mean close the observed mean (±10%). The black square shows the simulation corresponding to the fit shown in (a). **c–e:** Same as (b), but showing the sample Fano factor, sample maximum, and sample α_Z.

to model read counts before (Lopez et al., 2018), as read count data commonly exhibit zero-inflation compared to NB (Cao et al., 2021; Chen et al., 2018). A ZINB distribution has three parameters: the mean μ, the inverse overdispersion θ, and the zero-inflation parameter ψ. Its mass function is simply a negative binomial mass function with additional mass ψ on zero:

$$p(z) = \begin{cases} \psi + (1-\psi) \cdot p_{\mathrm{NB}}(0, \mu, \theta) & \text{for } z = 0 \\ (1-\psi) \cdot p_{\mathrm{NB}}(z, \mu, \theta) & \text{for } z > 0 \end{cases} \tag{16}$$

where p_{NB} is the NB probability mass function (see Methods). The ZINB distribution reduces to the NB distribution when $\psi = 0$. As in NB, the overdispersion parameter θ controls the shape of the distribution: $\theta = 1$ corresponds to the geometric distribution with monotonically decreasing $p(x)$, while higher values of θ result in more Poisson-like bell shapes ($\theta = \infty$ corresponds to the Poisson case).

By fitting the ZINB model, we obtained independent estimates $\hat{\theta}_g$ and $\hat{\psi}_g$ for each gene (Figure 6h–i). These estimates exhibited the same two patterns illustrated above for single genes. First, with increasing mean expression, genes tended to have a higher $\hat{\theta}_g$, corresponding to smaller variance, and transitioned from a geometric-like to a Poisson-like shape. Second, with increasing mean expression, genes tended to have a lower $\hat{\psi}_g$, corresponding to less pronounced zero inflation.

The ZINB model cannot explain these trends, as all parameters θ_g and ψ_g can be chosen independently. In contrast, our compound model naturally gives rise to both effects. To demonstrate this, we repeated the ZINB fitting procedure on counts sampled from various compound NB models (see Methods for the broken zeta parameters), and reproduced both observations over a wide range of mean expressions (Figure 6h–i, colored lines). As expected, the model matching this dataset's amplification parameters ($\alpha_Z = 50$ and $\mathbb{E}[Z] = 30$, orange line, cf. Figure 2) provided the best match to the bulk of the distribution.

As a sanity check, sampling read counts from a NB model without amplification ($\alpha_Z = 1$) and fitting ZINB distribution to the resulting samples recovered the original parameters (Figure 6h–i, black lines): constant overdispersion $\theta = 10$ and absent zero inflation $\psi = 0$. This again shows that a NB model without amplification cannot describe the properties of the read count data. However, our results suggest that it is not necessary to include explicit zero-inflation like in a ZINB model, as it is naturally arising through the compound model.

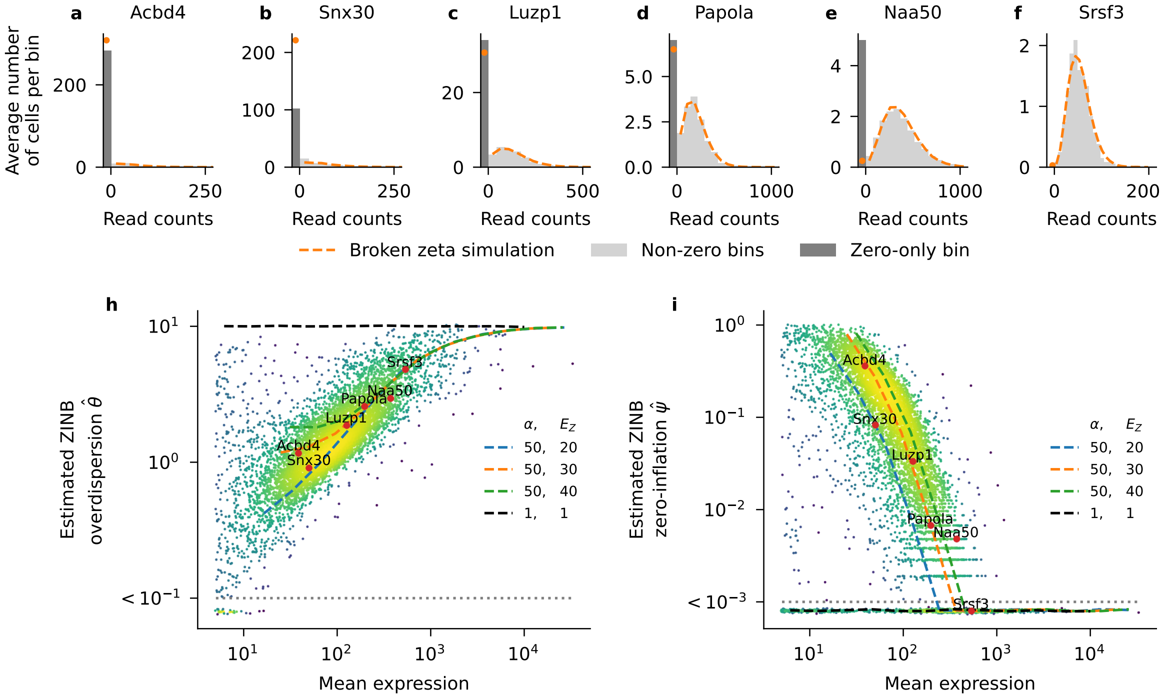

Figure 6: Broken zeta compound model simulations reproduce trends in read counts. Using a homogeneous subset ($n = 1\,049$) of the mouse cortex data (from Figure 2), we fitted a zero-inflated negative binomial (ZINB) distribution to each gene individually. We also used the broken zeta model ($a_1 = 0.36$, $a_2 = 5.1$, $b = 56$ to sample read counts with a given mean expression. **a–f:** Each panel shows the observed read counts for a certain gene (gray histogram) and the histogram of counts sampled from the broken zeta model (orange line). Exact zeros are shown in a separate bin (orange dots, dark-gray bar). To show the zero-bin (width 1) and non-zero bins (width $\gg 1$) on the same scale, the y-axis shows the average count per bin. Genes are ordered from left to right by mean expression. Note that with higher expression, fraction of zeros decreases, and the histogram shape changes from a geometric-looking distribution to a Poisson-looking distribution. **h:** Estimated ZINB overdispersion parameter $\hat{\theta}$ as a function of mean expression. Each dot is a gene, colored by local density of points. Values $\hat{\theta} < 10^{-1}$ were clipped. Only genes with mean expression ≥ 5 are shown. Colored lines show $\hat{\theta}$ for samples from four different broken zeta models. For each model, we sampled counts over a range of expression fractions p_g for 10^5 cells, each with fixed sequencing depth of $n_c = 100\,000$. We used overdispersion $\theta = 10$ for all genes. Broken zeta parameters: see Table 2. The black line corresponds to a negative binomial distribution (UMI model without amplification). Red dots highlight genes from panels (a)–(f). **i:** Estimated ZINB zero-inflation parameter $\hat{\psi}$ as a function of mean expression. Otherwise same as (h); values $\hat{\psi} < 10^{-3}$ were clipped.

3 Discussion

In this paper, we derived a parsimonious and theoretically grounded statistical model describing scRNA-seq read count data without UMIs. Furthermore, we showed that our compound model leads to analytic compound Pearson residuals, a fast, simple, and effective normalization approach for non-UMI data.

Despite the popularity of UMI protocols (Svensson et al., 2020), full-length non-UMI protocols such as Smart-seq2 (Picelli et al., 2013) remain relevant as they have higher sensitivity (Ziegenhain et al., 2017; Ding et al., 2020) and allow quantification of reads over full transcripts. This makes read count data indispensable for detection of splicing variants (Feng et al., 2021) or profiling of complex tissues with rare cell types (Tasic et al., 2018; Yao et al., 2021). The recently developed Smart-seq3/Smart-seq3xpress protocols (Hagemann-Jensen et al., 2020, 2022) contain UMIs on the 5'-end reads but do not have UMIs on internal reads, so our treatment remains relevant for Smart-seq3/Smart-seq3xpress as well.

While UMI counts can be modeled by a Poisson or a negative binomial (NB) distribution (Grün et al., 2014; Chen et al., 2018; Hafemeister and Satija, 2019; Townes et al., 2019; Svensson, 2020; Grün, 2020; Sarkar and Stephens, 2021; Rosales-Alvarez et al., 2023; Neufeld et al., 2023), read counts can not (Chen et al., 2018; Cao et al., 2021). Instead, they are often modeled by a more flexible zero-inflated negative binomial distribution (ZINB) (Pierson and Yau, 2015; Zappia et al., 2017; Chen et al., 2018; Risso et al., 2018; Lopez et al., 2018). However, this leaves unexplained what causes zero inflation and why there are relationships between the gene-specific ZINB parameters, such as less zero inflation for higher mean expression (Figure 6i).

Our compound model answers these questions. We showed that read counts in biologically homogeneous data can be well described by a compound negative binomial distribution, arising from simple statistical assumptions about the amplification and sequencing processes. Furthermore, we showed empirically that the distribution of copy numbers approximately follows a broken zeta distribution. Together, our compound NB model with amplification modeled by broken zeta yields a generative model reproducing zero-inflation and overdispersion patterns similar to what is observed in read count data. Compared to the ZINB model with three per-gene parameters (Equation 16), our model contains only one free per-gene parameter (Equations 5–8), and the varying zero-inflation and overdispersion naturally emerge as a function of a gene's mean expression.

We observed that the distribution of copy numbers in UMI-containing data followed a similar shape across various protocols (Figure 4), implying that this is a general property of scRNA-seq data. We argued that this shape could be described by a broken power law, and hence we modelled it with a broken zeta distribution. This model is phenomenological, but previous work on mechanistic modelling of PCR amplification followed by Poisson sampling showed that these processes can give rise to similar copy number distributions (Best et al., 2015). We note that fitting parameters for power-law-like data is intrinsically difficult: common approaches such as least squares often return unstable estimates due to low-probability events (Clauset et al., 2009), which is why we avoided automatic parameter fitting. Instead, we qualitatively showed that the broken zeta model can give rise to realistic read count distributions.

Our compound NB model did not describe all genes perfectly: we found that a subset of genes, mostly pseudogenes, did not follow the compound model, but rather behaved as if they were not amplified (Figures 2 and 3). Similar bimodal patterns in gene variance have been be observed in previous works (e.g. Brennecke et al., 2013; Ziegenhain et al., 2017). Pseudogenes are copies of functional genes that contain a mutation making the copy dysfunctional. We can only speculate about the reason causing pseudogene read counts to have less variance: they may behave differently during amplification or sequencing, or perhaps their counts are an artifact of the alignment algorithm (all datasets we analyzed used STAR (Dobin et al., 2013)). In practice, such pseudogenes have less variance than expected under the compound NB model, so will be filtered out by the gene selection step in our suggested workflow (Figure 3a).

On the practical side, we used the compound NB model to derive a fast and theory-based normalization procedure for read counts: compound Pearson residuals. They constitute an extension of the UMI Pearson residuals normalization, that has proven to be effective for gene selection and normalization of UMI data (Hafemeister and Satija, 2019; Lause et al., 2021). We showed that compound Pearson residuals work well for processing complex read count datasets, leading to a biologically meaningful gene selection and embeddings. Importantly, we also showed that normalization and gene selection using the non-compound UMI Pearson residuals leads to suboptimal results on read count data, underscoring the importance of an adequate statistical model.

Compound Pearson residuals only require to set the α_Z parameter of the amplification distribution. Whereas α_Z can be observed directly from the copy number distribution for UMI-containing data (i.e. reads per UMI, Table 1), it is unknown in a Smart-seq2 experiment. Reassuringly, we found that the results of compound Pearson residuals do not strongly depend on the exact value of α_Z — as long as it is set within a reasonable range $\gg 1$, such as $\alpha_Z \in [10, 1000]$. When working with Smart-seq2 data, we recommend using $\alpha_Z = 50$ by default. Furthermore, it is possible to empirically adjust α_Z to a given dataset from any sequencing protocol. Indeed, under the common assumption that most genes are not differentially expressed, the majority of genes should have residual variance close to one. Thus, adjusting α_Z until this condition is fulfilled will typically lead to a reasonable setting (Supplementary Figure S6).

Typical approaches to read count data normalization consist of scaling read counts by a size factor to account for sequencing depth (CPM: counts per million) and sometimes gene length (TPM: transcripts per million (Li and Dewey, 2011), or RPKM: reads per kilobase per million (Mortazavi et al., 2008)), followed by a log-transform (Luecken and Theis, 2019; Andrews et al., 2021; Slovin et al., 2021). Various methods have been suggested to estimate the required size factors, going beyond CPM/TPM/RPKM (Vallejos et al., 2017): via spike-ins (Brennecke et al., 2013; Lun et al., 2017), cell pooling (Lun et al., 2016), housekeeping genes (Andrews et al., 2021), separate scaling for groups of genes (Bacher et al., 2017), or a Bayesian approach (Tang et al., 2020). However, all of these methods depend on the log-transform for variance stabilization, which is inherently limited (Lun, 2018) and fails to fully stabilize the variance (Ahlmann-Eltze and Huber, 2023). In contrast, our compound Pearson residuals use the mean-variance relationship that follows from simple statistical assumptions, and the resulting residuals are variance-stabilized by design and do not require any explicit normalization by the gene length.

Two existing methods aim to transform read counts so that their distribution matches the distribution of UMI counts. Census (Qiu et al., 2017) linearly scales the read counts within each cell to set the mode of the count distribution to 1, while qUMI (Townes and Irizarry, 2020) performs quantile normalization within each cell to transform the entire distribution to the typical shape of within-cell UMI counts. In both cases, the transformations are heuristics not based on any generative statistical model, and the transformed data still require UMI-specific normalization. In contrast, our compound Pearson residuals perform necessary normalization directly on the read counts. In practice, we observed that qUMI and Census lead to comparable normalization results as our compound Pearson residuals, but our method follows from an explicit statistical model that offers theoretical insights into the data generation process underlying read counts. For example, as described above, our model captures previously unexplained patterns in the zero-inflation and overdispersion of read count data.

One limitation of our model is that it assumes that the amplification distribution is the same for all genes and cells and uses the single amplification parameter α_Z shared by all cells. This is not the case in Census and qUMI, which both use cell-specific adjustments. Reassuringly, we did not observe strong cell-to-cell variability in α_Z estimates (Supplementary Figure S10), and furthermore found that α_Z correlated with total counts per cell (Supplementary Figure S11) — a factor which our model explicitly accounts for.

In summary, we show that the compound NB distribution is the appropriate statistical model for read count data, naturally giving rise to compound Pearson residuals as an effective, convenient and theoretically motivated way of data pre-processing.

4 Methods

4.1 Datasets and preprocessing

Our example read count dataset throughout this paper is the mouse brain dataset from Tasic et al. (2018), GEO accession GSE115746. It contains cells from the primary visual cortex (VISp) and the anterior lateral motor area (ALM), and was sequenced with Smart-seq2. We downloaded the data for both areas from `https://portal.brain-map.org/atlases-and-data/rnaseq/mouse-v1-and-alm-smart-seq` and used only the exonic counts and applied the same cell filtering as Tasic et al. (2018), leading to 23 822 cells and 42 776 genes with at least one count. We used the Python package `mygene` to query the `mygene.info` database (Wu et al., 2013) for gene type annotations (`type_of_gene` field) with the Entrez gene identifiers. We queried `BioMart` to obtain transcript lengths for all Ensemble mouse genes (database *GRCm39*, column 'Transcript length (including UTRs and CDS)', Cunningham et al. (2022)).

From these data, we assembled a biologically homogeneous subset by selecting only cells from one of the largest neuronal clusters (named *VISp Penk Col27a1* by Tasic et al. (2018), 1 049 cells, 33 914 detected genes). These data were used without further filtering in Figures 2 and 6 and Supplementary Figures S1 and S2.

To assemble a heterogeneous dataset, we took the full dataset but filtered out genes that were detected in less than 5 cells as in previous work on UMI Pearson residuals (Hafemeister and Satija, 2019; Lause et al., 2021), leading to 23 822 cells and 38 510 genes.

To study copy number distributions across protocols, we used the following UMI datasets:

- Mouse embryonic stem cells profiled by CEL-seq2, Drop-seq, MARS-seq, and SCRB-seq (Ziegenhain et al., 2017); GEO accession GSE75790;
- Mouse fibroblasts profiled with Smart-seq3 paired-end (Johnsson et al., 2022); accession E-MTAB-10148, sample `plate2`;
- Mouse fibroblasts profiled with Smart-seq3 single-end (Hagemann-Jensen et al., 2020); accession E-MTAB-8735, sample `Smartseq3.Fibroblasts.smFISH`;
- HEK293 cells profiled with Smart-seq3Xpress (Hagemann-Jensen et al., 2022); accession E-MTAB-11467.

For UMI deduplication, we used Hamming distance correction with a threshold of 1. See Table 1 for numbers of cells and UMIs per dataset. The reads-per-UMI tables are available at `https://zenodo.org/record/8172702`.

We used `scanpy 1.9.0` (Wolf et al., 2018) and `anndata 0.8.0` (Virshup et al., 2021) for all scRNA data handling in `Python 3.8.10`, along with `sklearn 1.0.2` (Pedregosa et al., 2011), `numpy 1.21.5` (Harris et al., 2020), and `matplotlib 3.5.1` (Hunter, 2007).

4.2 Simulation study

To generate a realistic validation dataset with ground truth marker genes *(Simulation I)*, we simulated read counts based on the Tasic et al. (2018) read counts X_{cg} as follows. For each gene g, we computed the average expression fractions $p_g = \sum_c X_{cg} / \sum_{cg} X_{cg}$. For a set G of 14 well-known brain cell type marker genes (Tasic et al., 2016) and each of the 133 clusters in the Tasic et al. (2018) data, we computed the within-cluster fraction p_{ig}, where i is the cluster index. For each cell c, we computed its total read counts $n_c = \sum_g X_{cg}$ and assumed total UMI counts per cell $n_c^{\mathrm{UMI}} = n_c/100$. We then generated UMI counts as $k_{cg} \sim \mathrm{NB}(\mu_{cg}^{\mathrm{UMI}}, \theta = 100)$, where

$$\mu_{cg}^{\mathrm{UMI}} = \begin{cases} n_c^{\mathrm{UMI}} \cdot p_{i(c)g} & \text{for } g \in G \\ n_c^{\mathrm{UMI}} \cdot p_g & \text{for } g \notin G \end{cases}, \tag{17}$$

where $i(c)$ denotes cluster assignments of cell c. In words, only the marker genes from G were allowed to differ between clusters. We then simulated the amplification of each UMI by drawing copy numbers from the shifted geometric distribution $z_i \sim \mathrm{Z} = \mathrm{Geom}_+(\mu = 100)$, which corresponds to amplification with $\mathbb{E}[Z] = 100$ and $\alpha_Z = 199$ (see below). We finally summed the copy numbers for each gene and cell to obtain read counts $X_{cg} = \sum_{i=1}^{k_{cg}} z_i$ (Equation 5). After filtering out genes with less than 5 cells as above, *Simulation I* yielded 23 822 cells and 30 652 genes.

To obtain a second validation dataset with a richer cluster structure and ground truth cell types *(Simulation II)*, we used the same simulation setup as above, but allowed all genes to have cluster-specific fractions, i.e.

$$\mu_{cg}^{\mathrm{UMI}} = n_c^{\mathrm{UMI}} \cdot p_{i(c)g}. \tag{18}$$

After filtering as above, *Simulation II* yielded 23 822 cells and 30 576 genes.

Both simulations generated copy numbers z_i from the shifted geometric distribution $Z = \mathrm{Geom}_+(\mu = 100)$, which is equivalent to $Z = \mathrm{NB}(\mu = 99, \theta = 1) + 1$, with $z_i \in \mathbb{N}_+$ being positive integers. The variance of the negative binomial is equal to $99 + 99^2/1 = 9900$ and $\mathbb{E}[Z] = 100$, so the Fano factor is 99, leading to $\alpha_Z = \mathbb{E}[Z] + \mathbb{FF}[Z] = 199$.

4.3 Mathematical details of the compound negative binomial model

We use the term *compound Poisson/NB distribution* to describe a discrete random variable that is constructed as a sum over a random number of i.i.d. terms. A compound model has an 'inner' and an 'outer' distribution: The inner distribution generates the i.i.d. summation terms (Equation 6), while the outer distribution governs the number of terms to be summed (Equation 7). This setup is known under various names: Johnson et al. (2005) uses the term *stopped-sum* distribution. When the outer distribution is the Poisson distribution, the compound model is known as compound Poisson (Adelson, 1966), stuttering Poisson (Kemp, 1967; Moothathu and Kumar, 1995), or generalized Poisson (Feller, 1943) distribution.

Note that the term *compound distribution* can also have a different meaning: for example, in their work on qUMI normalization, Townes and Irizarry (2020) used the term 'compound Poisson model' to describe a Poisson model with rate parameter λ governed by another distribution.

The expectation of a compound random variable $X = \sum_{i=1}^{k} z_i$ with inner distribution $z_i \sim Z$ and outer distribution $k \sim K$ can be obtained as follows:

$$E[X] = \mathbb{E}_K\big[\mathbb{E}_X[X \mid K]\big] = \mathbb{E}_K\big[\mathbb{E}_X[z_1 + z_2 + \cdots + z_k \mid K]\big] = \mathbb{E}_K\big[k \cdot \mathbb{E}_X[Z]\big] = \mathbb{E}[K] \cdot \mathbb{E}[Z]. \tag{19}$$

The variance can be computed similarly:

$$\begin{aligned} \mathrm{Var}[X] &= \mathbb{E}_K\big[\mathrm{Var}[X \mid K]\big] + \mathrm{Var}_K\big[\mathbb{E}[X \mid K]\big] && (20)\\ &= \mathbb{E}_K\big[\mathrm{Var}[z_1 + z_2 + \cdots + z_k \mid K]\big] + \mathrm{Var}_K\big[\mathbb{E}[z_1 + z_2 + \cdots + z_k \mid K]\big] && (21)\\ &= \mathbb{E}_K\big[k \cdot \mathrm{Var}[Z]\big] + \mathrm{Var}_K\big[k \cdot \mathbb{E}[Z]\big] && (22)\\ &= \mathbb{E}[K] \cdot \mathrm{Var}[Z] + \mathrm{Var}[K] \cdot \mathbb{E}[Z]^2. && (23) \end{aligned}$$

Together, this leads to the following mean-variance relationship:

$$\mathrm{Var}[X] = \mathbb{E}[X] \cdot \frac{\mathrm{Var}[Z]}{\mathbb{E}[Z]} + \frac{\mathrm{Var}[K]}{\mathbb{E}[K]^2} \cdot \mathbb{E}[X]^2. \tag{24}$$

We use the negative binomial (NB) distribution as outer distribution in our compound model. The probability mass function for the NB distribution can be parametrized in several different ways. We use

$$p_{\mathrm{NB}}(k, \mu, \theta) = \frac{\Gamma(k+\theta)}{k!\,\Gamma(\theta)} \left(\frac{\mu}{\mu+\theta}\right)^k \left(\frac{\theta}{\theta+\mu}\right)^\theta, \tag{25}$$

where μ is the mean and θ is the overdispersion parameter. The variance is then given by $\mathrm{Var}[K] = \mathbb{E}[K] + \mathbb{E}[K]^2/\theta = \mu + \mu^2/\theta$.

Plugging the mean-variance relationship of K into the mean-variance relationship of X, we finally get

$$\begin{aligned}\mathrm{Var}[X] &= \mathbb{E}[X] \cdot \frac{\mathrm{Var}[Z]}{\mathbb{E}[Z]} + \frac{\mathbb{E}[K] + \mathbb{E}[K]^2/\theta}{\mathbb{E}[K]^2} \cdot \mathbb{E}[X]^2 && (26)\\ &= \mathbb{E}[X] \cdot \frac{\mathrm{Var}[Z]}{\mathbb{E}[Z]} + \frac{\mathbb{E}[X]^2}{\mathbb{E}[K]} + \frac{\mathbb{E}[X]^2}{\theta} && (27)\\ &= \mathbb{E}[X] \cdot \underbrace{\left(\frac{\mathrm{Var}[Z]}{\mathbb{E}[Z]} + \mathbb{E}[Z]\right)}_{\alpha_Z} + \frac{E[X]^2}{\theta}, && (28)\end{aligned}$$

where we used the fact that $\mathbb{E}[X] = \mathbb{E}[Z] \cdot \mathbb{E}[K]$. The relationship in Equation 28 yields the lines shown in Figure 2a.

From here we can obtain the relationship between the mean of X and the Fano factor of X:

$$\mathbb{FF}[X] = \frac{\mathrm{Var}[X]}{\mathbb{E}[X]} = \alpha_Z + \frac{\mathbb{E}[X]}{\theta}, \tag{29}$$

shown as lines in Figure 2b. For $E[X] \ll \theta$, this reduces to $\mathbb{FF}[X] \approx \alpha_Z$.

To derive the relationship between the mean of X and the fraction of zero counts, we note that the inner distribution in our model is strictly positive ($z \geq 1$). Any zero count $X = 0$ must thus originate from a $k = 0$ from the outer NB distribution. As a result, we can derive the fraction of zero counts from the NB probability mass function (Equation 25):

$$P(X=0) = p_{\mathrm{NB}}(K=0) = \left(\frac{\theta}{\theta + \mathbb{E}[K]}\right)^\theta = \left(\frac{\theta}{\theta + \mathbb{E}[X]/\mathbb{E}[Z]}\right)^\theta. \tag{30}$$

This relationship is shown as lines in Figure 2c. For $\theta \to \infty$, this converges to the Poisson case $P(X = 0) = e^{-\mathbb{E}[K]} = e^{-\mathbb{E}[X]/\mathbb{E}[Z]}$.

4.4 Compound Pearson residuals

For gene selection with compound Pearson residuals, we computed $\hat{\mathbb{E}}[X_{cg}]$ from the filtered read count matrix X (Equation 10) and then obtained residuals using Equation 13. We selected 3 000 highly variable genes (HVGs) with the highest residual variance. To normalize, we then subset the raw count data matrix to the HVGs, and computed compound Pearson residuals again on that subset, and used these re-computed residuals for further analysis (consistent with our previous work, Lause et al. (2021)). Using the residuals computed from the full data matrix and subsetting them to HVGs led to very similar results.

Unless otherwise stated, we used $\alpha_Z = 50$ and $\theta = 100$ for computing residuals, and clipped residuals to $\sqrt{n}$ where n is the number of cells, following Hafemeister and Satija (2019) (see Lause et al. (2021) for a motivation for this heuristic).

4.5 Census counts and qUMIs

We obtained both Census counts and qUMIs via their official `R` implementations using `R 4.1.3`. To obtain Census counts, we used `bioconductor-monocle 2.22.0` (Huber et al., 2015; Qiu et al., 2017).

To obtain qUMIs, we used `quminorm 0.1.0` from `http://github.com/willtownes/quminorm/` (Townes and Irizarry, 2020). As both methods expect TPMs as input, we subset the (Tasic et al., 2018) data to the 27 841 genes for which length annotations were available (see above), and computed TPM from read counts X_{cg} and gene lengths l_g (in kilobase) as

$$\text{TPM}_{cg} = \frac{X_{cg}/l_g}{\sum_g X_{cg}/l_g} \cdot 1\,000\,000 \tag{31}$$

Running Census on the full matrix was very slow (>24 h), so we split the TPM matrix into batches of 1000 cells. This substantially sped up the computation. Filtering Census counts and qUMIs for genes with at least 5 cells yielded 23 822 cells and 25 248 genes.

4.6 t-SNE visualizations

As basis for all t-SNE embeddings, we computed the first 1 000 principal components (PCs) of the HVGs residuals with `sklearn 1.0.2` (Pedregosa et al., 2011). For all t-SNE embeddings, we used `openTSNE 0.6.0` (Poličar et al., 2019) with default settings unless otherwise stated. To ensure comparability between the t-SNE embeddings in Supplementary Figure S6, we used the first two PCs of the HVG residuals computed with $\alpha_Z = 10$ (panel C) as shared initialization after scaling them with `openTSNE.initialization.rescale()`.

To visualize expression strength of a given gene across the t-SNE map(Figure 3c), we used square-root-transformed depth-normalized counts

$$S_{cg} = \sqrt{m \cdot \frac{X_{cg}}{\sum_i X_{ci}}}, \tag{32}$$

where m is the median row sum of X.

4.7 Fitting zero-inflated negative binomial (ZINB) models to single genes

To obtain per-gene estimates of overdispersion $\hat{\theta}_g$ and zero inflation $\hat{\psi}_g$ in the absence of biological variability, we fitted a ZINB model to the raw read counts of each gene in the *VISp Penk Col27a1* cluster ($n = 1\,049$ cells). We used the `ZeroInflatedNegativeBinomialP.fit_regularized()` function from `statsmodels 0.13.2` (Seabold and Perktold, 2010) with default parameters. Only the 11 549 genes with within-cluster mean expression >5 were included because low-expression genes suffered from unstable parameter estimates. All genes with fitting warnings ($n = 2\,064$), fitting errors ($n = 22$) or invalid resulting estimates $\hat{\psi}_g > 1$ ($n = 2\,359$) were excluded, such that 7 104 genes with valid converged estimates remained for further analysis and are shown in Figures 6h–i.

We applied the same fitting procedure to the simulated read counts shown as lines in Figure 6 (see below for simulation details). Here, 62 out of 100 simulated genes had a mean expression >5 and were used for fitting. 5 of them resulted in invalid $\hat{\psi}_g > 1$ values and were excluded, such that 57 simulated genes remained for plotting and analysis.

4.8 Sampling copy numbers from the broken zeta model to simulate compound model read counts

The broken zeta model we describe in Equation 15 is the discrete version of the broken power law. A continuous probability distribution with probability density following a power law is called the Pareto distribution. Its discrete analogue is known under various names including Riemann zeta distribution (or simply zeta distribution), discrete Pareto distribution, and Zipf distribution(Johnson et al., 2005). We therefore refer to the discrete broken power law distribution as *broken zeta distribution.* While continuous broken power law distributions are commonly used in astrophysics (Jóhannesson et al., 2006), we are not aware of any prior use of discrete broken power law distributions for statistical modeling.

For the simulations in Figures 5 and 6, we sampled copy numbers from a given broken zeta distribution. For that, we computed the approximate PMF for a limited support $z \in \{1, 2, \ldots, 10^5\}$ with Equation 15, and normalized the resulting probabilities to sum to 1. This way we did not need to compute the normalization constant in Equation 15.

For the simulations of the Drop-seq copy number distribution in Figure 5b–e we used $n = 2\,506\,244$ samples per parameter set, which is the number of UMIs in the Drop-seq A dataset. We extended the

$\mathbb{E}[Z]$		α_Z				
target	observed	target	observed	a_1	a_2	b
1	1.0	1	1.0	–	–	–
20	20.1	50	50.4	0.96	15.1	91
30	30.2	50	51.4	0.36	5.1	56
40	37.9	50	50.5	0.01	17.6	71

Table 2: Broken zeta parameters used for compound model read count simulations. Each row in the table corresponds to one of the four compound model simulations shown in Figure 6. Observed values refer to the obtained sample mean and sample variance ($n \approx 2$ billion, see text for simulation details). The model with $\alpha_Z = 1$ corresponds to the UMI case with constant copy number $z_i = 1$, and thus has no broken zeta parameters.

support until 10^6 for this particular simulation, because we observed $\max(z_i) \approx 10^5$ for some of the more extreme parameter combinations.

In order to sample realistic read counts from four compound models with different combinations of mean copy number $\mathbb{E}[Z]$ and α_Z (Figure 6), we used a grid search over the broken zeta parameters a_1, a_2, and b to find parameter combinations that best matched the required values. Table 2 lists the parameters and amplification statistics of the chosen models. Across all parameter sets shown in Figure 6, we observed $\max(z_i) = 7\,179$ which was far below the end of the support.

We first sampled unique sequenced molecules k_{cg} from a negative binomial with $k_{cg} \sim \mathrm{NB}(p_g n_c, \theta)$ for 25 genes over a log-spaced range of expression fractions $p_g \in [10^{-8}, 10^{-1}]$. We did this for 10^5 cells, each with fixed sequencing depth of $n_c = 10^5$. We used overdispersion parameter $\theta = 10$ for all genes. This led to a total of $\sum k_{cg} = 2\,047\,196\,087$ simulated UMIs. Then, for each of them, we sampled a copy number z_i from the broken zeta model as described above, and summed over copy numbers for the same cell and gene to obtain a read count $X_{cg} = \sum_{i=1}^{k_{cg}} z_i$. We used the same set of simulated UMIs for the four compound model simulations shown in Figure 6 and Table 2.

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Availability of data and materials

The datasets generated and/or analysed during the current study are publicly available as described in Methods (Section 4.1). All analysis code is available under the GNU General Public License v3.0 at `https://github.com/berenslab/read-normalization`, and is archived in the Zenodo repository `https://zenodo.org/doi/10.5281/zenodo.12806891`.

Competing interests

The authors declare that they have no competing interests.

Funding

The work was funded by the German Science Foundation (BE5601/8-1, BE5601/6-1, EXC 2064 "Machine Learning — New Perspectives for Science") and the European Union (ERC, "NextMechMod", ref. 101039115). Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency. Neither the European Union nor the granting authority can be held responsible for them. Additional support comes from the National Institute of Mental Health and National Institute of Neurological Disorders And Stroke under Award Number UM1MH130981. CZ is supported by the Swedish Research Council (2022-01471).

The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

Authors' contributions

Conception of the method: JL, PB, DK. Data acquisition and preprocessing: CZ, LH. Data analysis and interpretation: JL, DK, CZ. Drafted the manuscript: JL. Revised the manuscript: All authors.

Acknowledgements

We would like to thank Rickard Sandberg for discussions.

References

RM Adelson. Compound Poisson distributions. *Journal of the Operational Research Society*, 17(1): 73–75, 1966.

Constantin Ahlmann-Eltze and Wolfgang Huber. Comparison of transformations for single-cell RNA-seq data. *Nature Methods*, pages 1–8, 2023.

Tallulah S Andrews, Vladimir Yu Kiselev, Davis McCarthy, and Martin Hemberg. Tutorial: guidelines for the computational analysis of single-cell RNA sequencing data. *Nature Protocols*, 16(1):1–9, 2021.

Rhonda Bacher, Li-Fang Chu, Ning Leng, Audrey P Gasch, James A Thomson, Ron M Stewart, Michael Newton, and Christina Kendziorski. SCnorm: robust normalization of single-cell RNA-seq data. *Nature Methods*, 14(6):584–586, 2017.

Katharine Best, Theres Oakes, James M Heather, John Shawe-Taylor, and Benny Chain. Computational analysis of stochastic heterogeneity in PCR amplification efficiency revealed by single molecule barcoding. *Scientific Reports*, 5(1):1–13, 2015.

Jérémie Breda, Mihaela Zavolan, and Erik van Nimwegen. Bayesian inference of gene expression states from single-cell RNA-seq data. *Nature Biotechnology*, 39(8):1008–1016, 2021.

Philip Brennecke, Simon Anders, Jong Kyoung Kim, Aleksandra A Kołodziejczyk, Xiuwei Zhang, Valentina Proserpio, Bianka Baying, Vladimir Benes, Sarah A Teichmann, John C Marioni, et al. Accounting for technical noise in single-cell RNA-seq experiments. *Nature Methods*, 10(11):1093–1095, 2013.

Yingying Cao, Simo Kitanovski, Ralf Küppers, and Daniel Hoffmann. UMI or not UMI, that is the question for scRNA-seq zero-inflation. *Nature Biotechnology*, 39(2):158–159, 2021.

Wenan Chen, Yan Li, John Easton, David Finkelstein, Gang Wu, and Xiang Chen. UMI-count modeling and differential expression analysis for single-cell RNA sequencing. *Genome Biology*, 19(1):1–17, 2018.

Saket Choudhary and Rahul Satija. Comparison and evaluation of statistical error models for scRNA-seq. *bioRxiv*, 2021.

Aaron Clauset, Cosma Rohilla Shalizi, and Mark EJ Newman. Power-law distributions in empirical data. *SIAM Review*, 51(4):661–703, 2009.

Fiona Cunningham, James E Allen, Jamie Allen, Jorge Alvarez-Jarreta, M Ridwan Amode, Irina M Armean, Olanrewaju Austine-Orimoloye, Andrey G Azov, If Barnes, Ruth Bennett, et al. Ensembl 2022. *Nucleic Acids Research*, 50(D1):D988–D995, 2022.

Jiarui Ding, Xian Adiconis, Sean K Simmons, Monika S Kowalczyk, Cynthia C Hession, Nemanja D Marjanovic, Travis K Hughes, Marc H Wadsworth, Tyler Burks, Lan T Nguyen, et al. Systematic comparison of single-cell and single-nucleus RNA-sequencing methods. *Nature Biotechnology*, 38(6): 737–746, 2020.

Alexander Dobin, Carrie A Davis, Felix Schlesinger, Jorg Drenkow, Chris Zaleski, Sonali Jha, Philippe Batut, Mark Chaisson, and Thomas R Gingeras. STAR: ultrafast universal RNA-seq aligner. *Bioinformatics*, 29(1):15–21, 2013.

William Feller. On a general class of 'contagious' distributions. *The Annals of Mathematical Statistics*, 14(4):389–400, 1943.

Huijuan Feng, Daniel F Moakley, Shuonan Chen, Melissa G McKenzie, Vilas Menon, and Chaolin Zhang. Complexity and graded regulation of neuronal cell-type–specific alternative splicing revealed by single-cell RNA sequencing. *Proceedings of the National Academy of Sciences*, 118(10):e2013056118, 2021.

Dominic Grün. Revealing dynamics of gene expression variability in cell state space. *Nature Methods*, 17(1):45–49, 2020.

Dominic Grün, Lennart Kester, and Alexander Van Oudenaarden. Validation of noise models for single-cell transcriptomics. *Nature Methods*, 11(6):637–640, 2014.

Christoph Hafemeister and Rahul Satija. Normalization and variance stabilization of single-cell RNA-seq data using regularized negative binomial regression. *Genome Biology*, 20(1):1–15, 2019.

Michael Hagemann-Jensen, Christoph Ziegenhain, Ping Chen, Daniel Ramsköld, Gert-Jan Hendriks, Anton JM Larsson, Omid R Faridani, and Rickard Sandberg. Single-cell RNA counting at allele and isoform resolution using Smart-seq3. *Nature Biotechnology*, 38(6):708–714, 2020.

Michael Hagemann-Jensen, Christoph Ziegenhain, and Rickard Sandberg. Scalable single-cell RNA sequencing from full transcripts with Smart-seq3xpress. *Nature Biotechnology*, 40(10):1452–1457, 2022.

Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020.

Lukas Heumos, Anna C Schaar, Christopher Lance, Anastasia Litinetskaya, Felix Drost, Luke Zappia, Malte D Lücken, Daniel C Strobl, Juan Henao, Fabiola Curion, et al. Best practices for single-cell analysis across modalities. *Nature Reviews Genetics*, pages 1–23, 2023.

Lauren L Hsu and Aedín C Culhane. Correspondence analysis for dimension reduction, batch integration, and visualization of single-cell RNA-seq data. *Scientific Reports*, 13(1):1–17, 2023.

Wolfgang Huber, Vincent J Carey, Robert Gentleman, Simon Anders, Marc Carlson, Benilton S Carvalho, Hector Corrada Bravo, Sean Davis, Laurent Gatto, Thomas Girke, et al. Orchestrating high-throughput genomic analysis with bioconductor. *Nature methods*, 12(2):115–121, 2015.

J. D. Hunter. Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007.

Rafael Irizarry. R package with methods for small counts stored in a sparse matrix. `https://github.com/rafalab/smallcount`, 2021.

Saiful Islam, Amit Zeisel, Simon Joost, Gioele La Manno, Pawel Zajac, Maria Kasper, Peter Lönnerberg, and Sten Linnarsson. Quantitative single-cell RNA-seq with unique molecular identifiers. *Nature Methods*, 11(2):163–166, 2014.

Gudlaugur Jóhannesson, Gunnlaugur Björnsson, and Einar H Gudmundsson. Afterglow light curves and broken power laws: a statistical study. *The Astrophysical Journal*, 640(1):L5, 2006.

Norman L Johnson, Adrienne W Kemp, and Samuel Kotz. *Univariate discrete distributions*, volume 444. John Wiley & Sons, 2005.

Per Johnsson, Christoph Ziegenhain, Leonard Hartmanis, Gert-Jan Hendriks, Michael Hagemann-Jensen, Björn Reinius, and Rickard Sandberg. Transcriptional kinetics and molecular functions of long non-coding RNAs. *Nature Genetics*, 54(3):306–317, 2022.

C. D. Kemp. 'Stuttering-Poisson' distributions. *Journal of the Statistical and Social Inquiry Society of Ireland*, XXI(V):151–157, May 1967.

Jan Lause, Philipp Berens, and Dmitry Kobak. Analytic Pearson residuals for normalization of single-cell RNA-seq UMI data. *Genome Biology*, 22(1):1–20, 2021.

Bo Li and Colin N Dewey. RSEM: accurate transcript quantification from RNA-seq data with or without a reference genome. *BMC Bioinformatics*, 12:1–16, 2011.

Marcela Lipovsek, Cedric Bardy, Cathryn R Cadwell, Kristen Hadley, Dmitry Kobak, and Shreejoy J Tripathy. Patch-seq: Past, present, and future. *Journal of Neuroscience*, 41(5):937–946, 2021.

Romain Lopez, Jeffrey Regier, Michael B Cole, Michael I Jordan, and Nir Yosef. Deep generative modeling for single-cell transcriptomics. *Nature Methods*, 15(12):1053–1058, 2018.

Malte D Luecken and Fabian J Theis. Current best practices in single-cell RNA-seq analysis: a tutorial. *Molecular Systems Biology*, 15(6):e8746, 2019.

Aaron TL Lun. Overcoming systematic errors caused by log-transformation of normalized single-cell RNA sequencing data. *bioRxiv*, page 404962, 2018.

Aaron TL Lun, Karsten Bach, and John C Marioni. Pooling across cells to normalize single-cell RNA sequencing data with many zero counts. *Genome Biology*, 17(1):1–14, 2016.

Aaron TL Lun, Fernando J Calero-Nieto, Liora Haim-Vilmovsky, Berthold Göttgens, and John C Marioni. Assessing the reliability of spike-in normalization for analyses of single-cell RNA sequencing data. *Genome research*, 27(11):1795–1806, 2017.

TS Moothathu and C Satheesh Kumar. Some properties of the stuttering Poisson distribution. *Calcutta Statistical Association Bulletin*, 45(1-2):125–130, 1995.

Ali Mortazavi, Brian A Williams, Kenneth McCue, Lorian Schaeffer, and Barbara Wold. Mapping and quantifying mammalian transcriptomes by RNA-seq. *Nature Methods*, 5(7):621–628, 2008.

Anna Neufeld, Joshua Popp, Lucy L Gao, Alexis Battle, and Daniela Witten. Negative binomial count splitting for single-cell RNA sequencing data. *arXiv*, 2023.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in Python. *The Journal of Machine Learning Research*, 12:2825–2830, 2011.

Belinda Phipson, Luke Zappia, and Alicia Oshlack. Gene length and detection bias in single cell RNA sequencing protocols. *F1000Research*, 6, 2017.

Simone Picelli, Åsa K Björklund, Omid R Faridani, Sven Sagasser, Gösta Winberg, and Rickard Sandberg. Smart-seq2 for sensitive full-length transcriptome profiling in single cells. *Nature Methods*, 10 (11):1096–1098, 2013.

Emma Pierson and Christopher Yau. ZIFA: Dimensionality reduction for zero-inflated single-cell gene expression analysis. *Genome Biology*, 16(1):1–10, 2015.

Pavlin G Poličar, Martin Stražar, and Blaž Zupan. openTSNE: a modular python library for t-SNE dimensionality reduction and embedding. *bioRxiv*, page 731877, 2019.

Xiaojie Qiu, Andrew Hill, Jonathan Packer, Dejun Lin, Yi-An Ma, and Cole Trapnell. Single-cell mRNA quantification and differential analysis with Census. *Nature Methods*, 14(3):309–315, 2017.

Davide Risso, Fanny Perraudeau, Svetlana Gribkova, Sandrine Dudoit, and Jean-Philippe Vert. A general and flexible method for signal extraction from single-cell RNA-seq data. *Nature Communications*, 9 (1):284, 2018.

Reyna Edith Rosales-Alvarez, Jasmin Rettkowski, Josip Stefan Herman, Gabrijela Dumbović, Nina Cabezas-Wallscheid, and Dominic Grün. VarID2 quantifies gene expression noise dynamics and unveils functional heterogeneity of ageing hematopoietic stem cells. *Genome Biology*, 24(1):1–30, 2023.

Abhishek Sarkar and Matthew Stephens. Separating measurement and expression models clarifies confusion in single-cell RNA sequencing analysis. *Nature Genetics*, 53(6):770–777, 2021.

Rahul Satija, Jeffrey A Farrell, David Gennert, Alexander F Schier, and Aviv Regev. Spatial reconstruction of single-cell gene expression data. *Nature Biotechnology*, 33(5):495–502, 2015.

Skipper Seabold and Josef Perktold. Statsmodels: Econometric and statistical modeling with Python. In *9th Python in Science Conference*, 2010.

Shaked Slovin, Annamaria Carissimo, Francesco Panariello, Antonio Grimaldi, Valentina Bouché, Gennaro Gambardella, and Davide Cacchiarelli. Single-cell RNA sequencing analysis: a step-by-step overview. *RNA Bioinformatics*, pages 343–365, 2021.

Valentine Svensson. Droplet scRNA-seq is not zero-inflated. *Nature Biotechnology*, 38(2):147–150, 2020.

Valentine Svensson, Eduardo da Veiga Beltrame, and Lior Pachter. A curated database reveals trends in single-cell transcriptomics. *Database*, 2020.

Wenhao Tang, François Bertaux, Philipp Thomas, Claire Stefanelli, Malika Saint, Samuel Marguerat, and Vahid Shahrezaei. bayNorm: Bayesian gene expression recovery, imputation and normalization for single-cell RNA-sequencing data. *Bioinformatics*, 36(4):1174–1181, 2020.

Bosiljka Tasic, Vilas Menon, Thuc Nghi Nguyen, Tae Kyung Kim, Tim Jarsky, Zizhen Yao, Boaz Levi, Lucas T Gray, Staci A Sorensen, Tim Dolbeare, et al. Adult mouse cortical cell taxonomy revealed by single cell transcriptomics. *Nature Neuroscience*, 19(2):335–346, 2016.

Bosiljka Tasic, Zizhen Yao, Lucas T Graybuck, Kimberly A Smith, Thuc Nghi Nguyen, Darren Bertagnolli, Jeff Goldy, Emma Garren, Michael N Economo, Sarada Viswanathan, et al. Shared and distinct transcriptomic cell types across neocortical areas. *Nature*, 563(7729):72–78, 2018.

F William Townes and Rafael A Irizarry. Quantile normalization of single-cell RNA-seq read counts without unique molecular identifiers. *Genome Biology*, 21(1):1–17, 2020.

F William Townes, Stephanie C Hicks, Martin J Aryee, and Rafael A Irizarry. Feature selection and dimension reduction for single-cell RNA-Seq based on a multinomial model. *Genome Biology*, 20:295, 2019.

Catalina A Vallejos, Davide Risso, Antonio Scialdone, Sandrine Dudoit, and John C Marioni. Normalizing single-cell RNA sequencing data: challenges and opportunities. *Nature Methods*, 14(6):565–571, 2017.

Isaac Virshup, Sergei Rybakov, Fabian J Theis, Philipp Angerer, and F Alexander Wolf. anndata: Annotated data. *bioRxiv*, pages 2021–12, 2021.

F Alexander Wolf, Philipp Angerer, and Fabian J Theis. SCANPY: large-scale single-cell gene expression data analysis. *Genome Biology*, 19:1–5, 2018.

Chunlei Wu, Ian MacLeod, and Andrew I Su. BioGPS and MyGene.info: organizing online, gene-centric information. *Nucleic Acids Research*, 41(D1):D561–D565, 2013.

Zizhen Yao, Hanqing Liu, Fangming Xie, Stephan Fischer, Ricky S Adkins, Andrew I Aldridge, Seth A Ament, Anna Bartlett, M Margarita Behrens, Koen Van den Berge, et al. A transcriptomic and epigenomic cell atlas of the mouse primary motor cortex. *Nature*, 598(7879):103–110, 2021.

Luke Zappia, Belinda Phipson, and Alicia Oshlack. Splatter: simulation of single-cell RNA sequencing data. *Genome Biology*, 18(1):174, 2017.

Christoph Ziegenhain, Beate Vieth, Swati Parekh, Björn Reinius, Amy Guillaumet-Adkins, Martha Smets, Heinrich Leonhardt, Holger Heyn, Ines Hellmann, and Wolfgang Enard. Comparative analysis of single-cell RNA sequencing methods. *Molecular Cell*, 65(4):631–643, 2017.

Christoph Ziegenhain, Gert-Jan Hendriks, Michael Hagemann-Jensen, and Rickard Sandberg. Molecular spikes: a gold standard for single-cell RNA counting. *Nature Methods*, 19(5):560–566, 2022.

Supplementary Figures

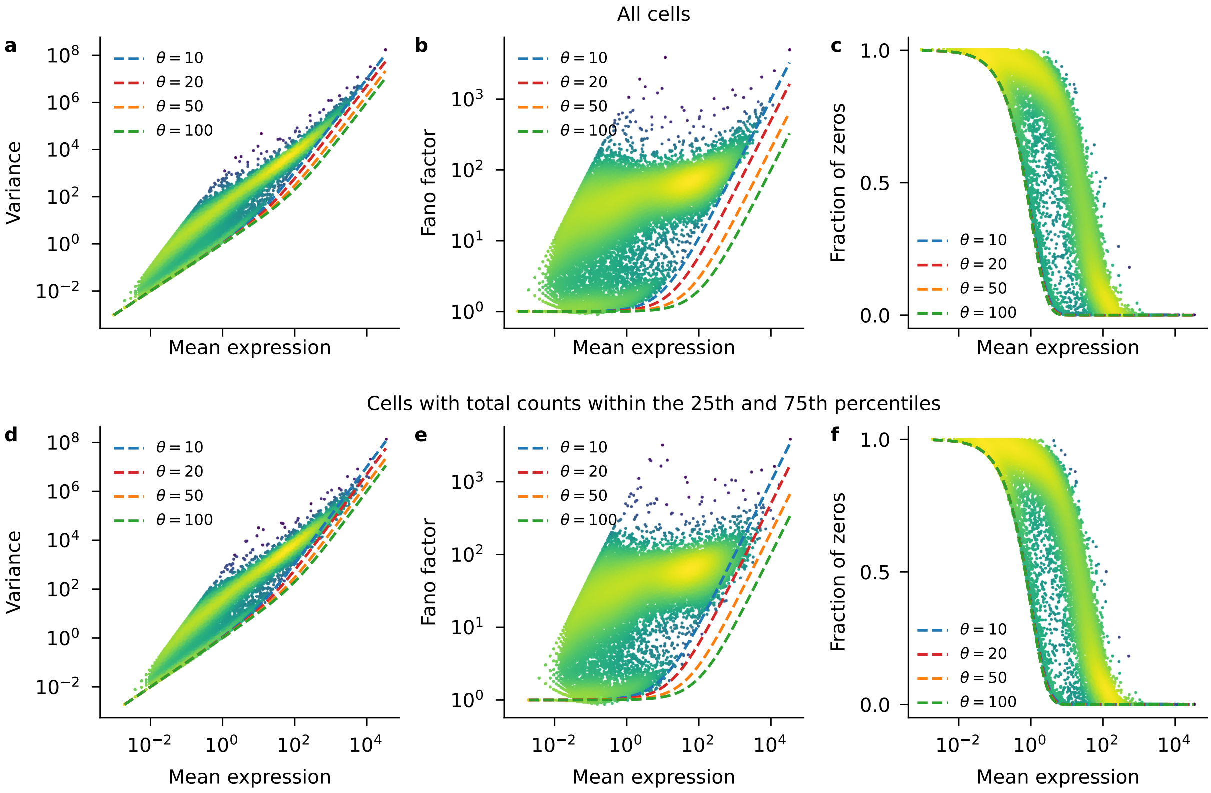

Figure S1: Sequencing depth variation increases the apparent overdispersion. Both rows are a reproduction of Figure 2, but showing pure NB models ($\alpha_Z = 1$) that differ in their overdispersion parameter θ. **a–c:** Plots based on all 1 049 cells as shown in Figure 2 with total counts per cell ranging from ∼680 000 (1^{st} percentile) to ∼2.84 million (99^{th} percentile). Note that the NB model with $\theta = 10$ fits the boundary of the data distributions. **d–f:** Plots based on a subset of the 523 cells with total counts within the 25th and 75th percentiles (from ∼1.54 million to ∼1.90 million reads). Now the NB model with $\theta = 20$ fits the boundary of the data distributions better than with $\theta = 10$. In other words, overdispersion is smaller when controlling for sequencing depth variation.

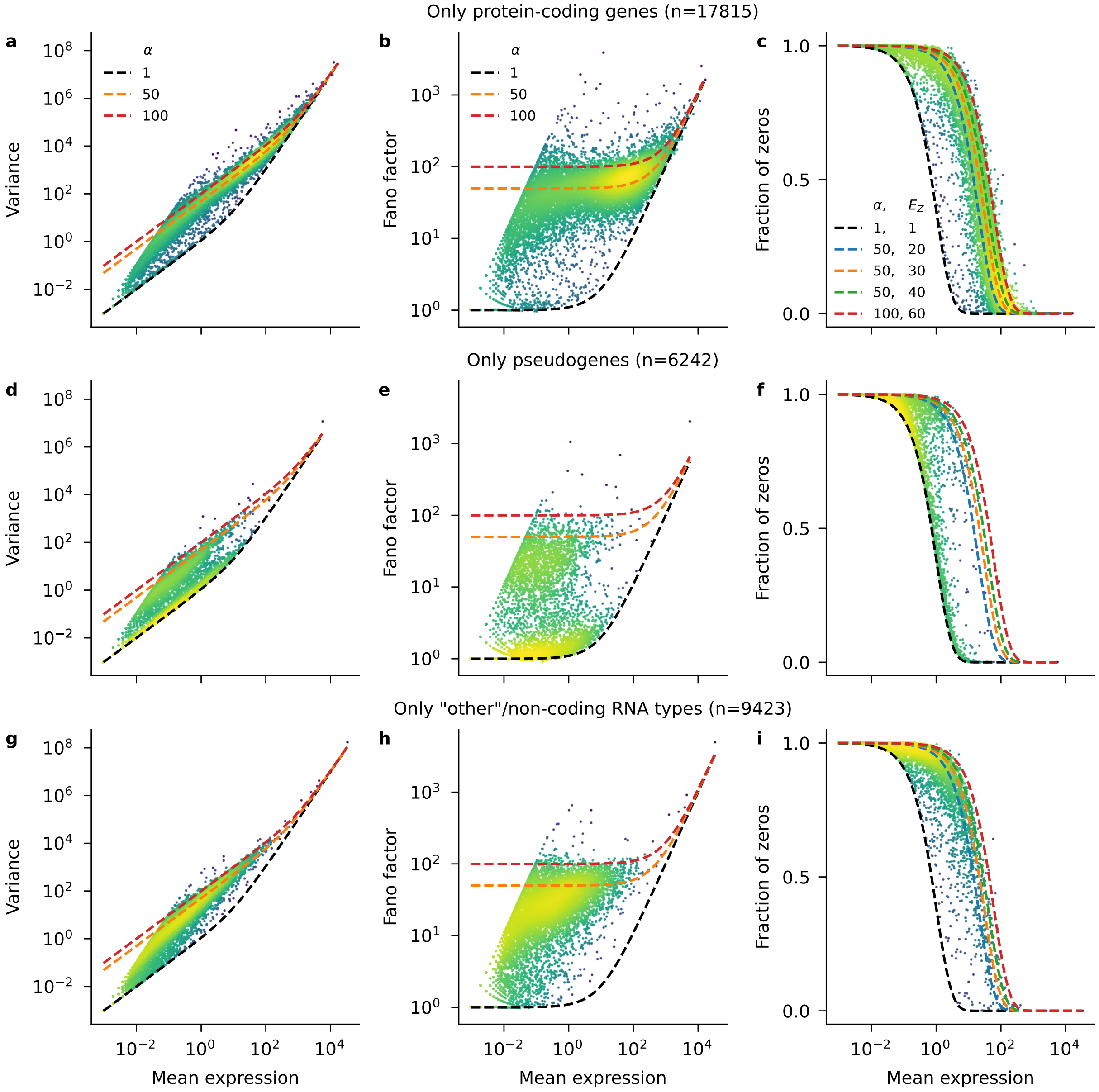

Figure S2: Non-amplified genes are mostly pseudogenes. Each row is a reproduction of Figure 2, but showing only genes from a certain category as obtained by the `mygene.info` annotation service. 434 genes without available annotation are not shown. **a–c:** Protein-coding genes. **d–f:** Pseudogenes. Many of them appear ‘non-amplified’ and do not follow the compound model, but rather the UMI model without amplification (black). **g–i:** Other, non-coding RNA species. Note that the bulk of these low-expression genes did not follow the compound model either.

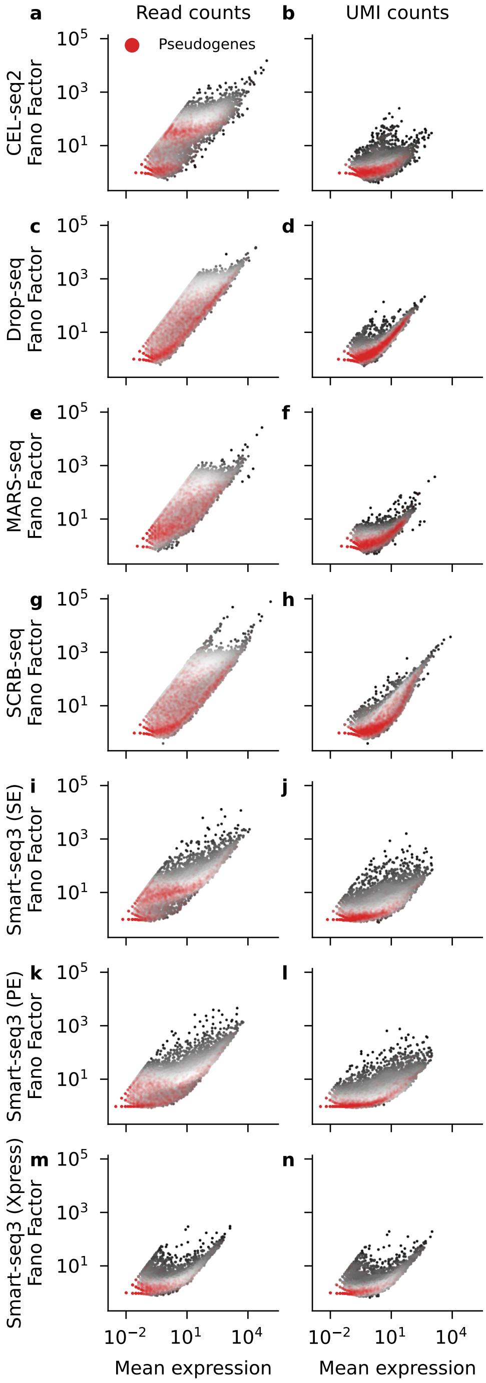

Figure S3: Pseudogenes have low Fano factor of read counts across protocols. Each panel shows the relationship between the mean and the Fano factor for all genes in each dataset. Higher density of dots is shown in brighter gray. Red dots with transparency show pseudogenes. Each row of panels shows a homogeneous dataset sequenced with a different protocol. All left-column panels are based on read counts, all right-column panels are based on UMI counts from the same dataset. Same data as in Figure 4. **a–h**: Mouse embryonic stem cells sequenced with various UMI protocols (Ziegenhain et al., 2017). For all protocols, only run A is shown. **i–l:** Smart-seq3 data from mouse fibroblasts (Hagemann-Jensen et al., 2020). SE: Single-end run. PE: Paired-end run. **m–n:** Smart-seq3 Xpress data from HEK293 cells (Hagemann-Jensen et al., 2022).

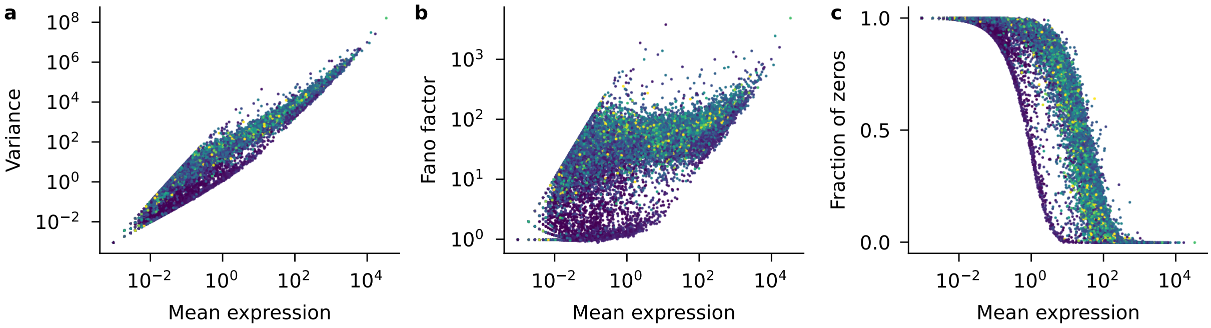

Figure S4: Pseudogenes have lower maximum transcript lengths. Reproduction of Figure 2, but showing each gene colored by the maximum length across all of its transcripts present in the `Ensembl` mouse gene database. 11 097 genes without transcript length annotation are not shown. Maximum lengths were clipped to the 98th percentile (9 849 bp) before plotting.

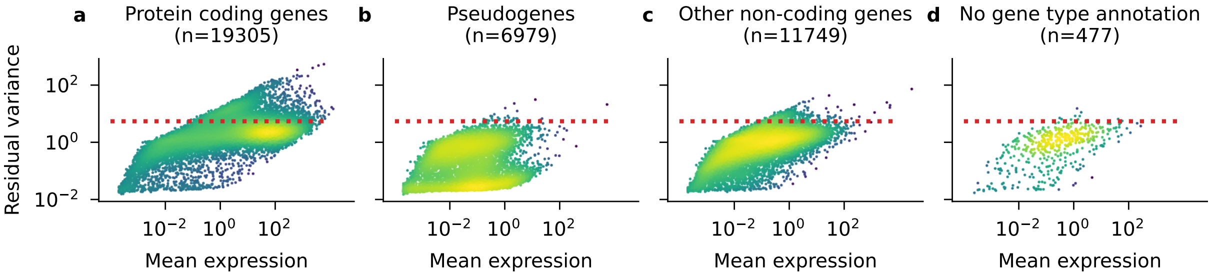

Figure S5: Genes with residual variance $\ll 1$ are mostly pseudogenes. Each panel is a reproduction of Figure 3a, showing only a certain category of genes, as in Figure S2. Each dot represents a gene and shows its mean and residual variance in the full mouse visual cortex dataset (Tasic et al., 2018). Brighter color indicates higher density of points. Red line shows cutoff for selecting 3 000 HVGs among all genes. Gene type annotations taken from the `mygene.info` service. **a:** Protein-coding genes. **b:** Pseudogenes. **c:** Other, non-coding RNA species. **d:** Genes without available annotation.

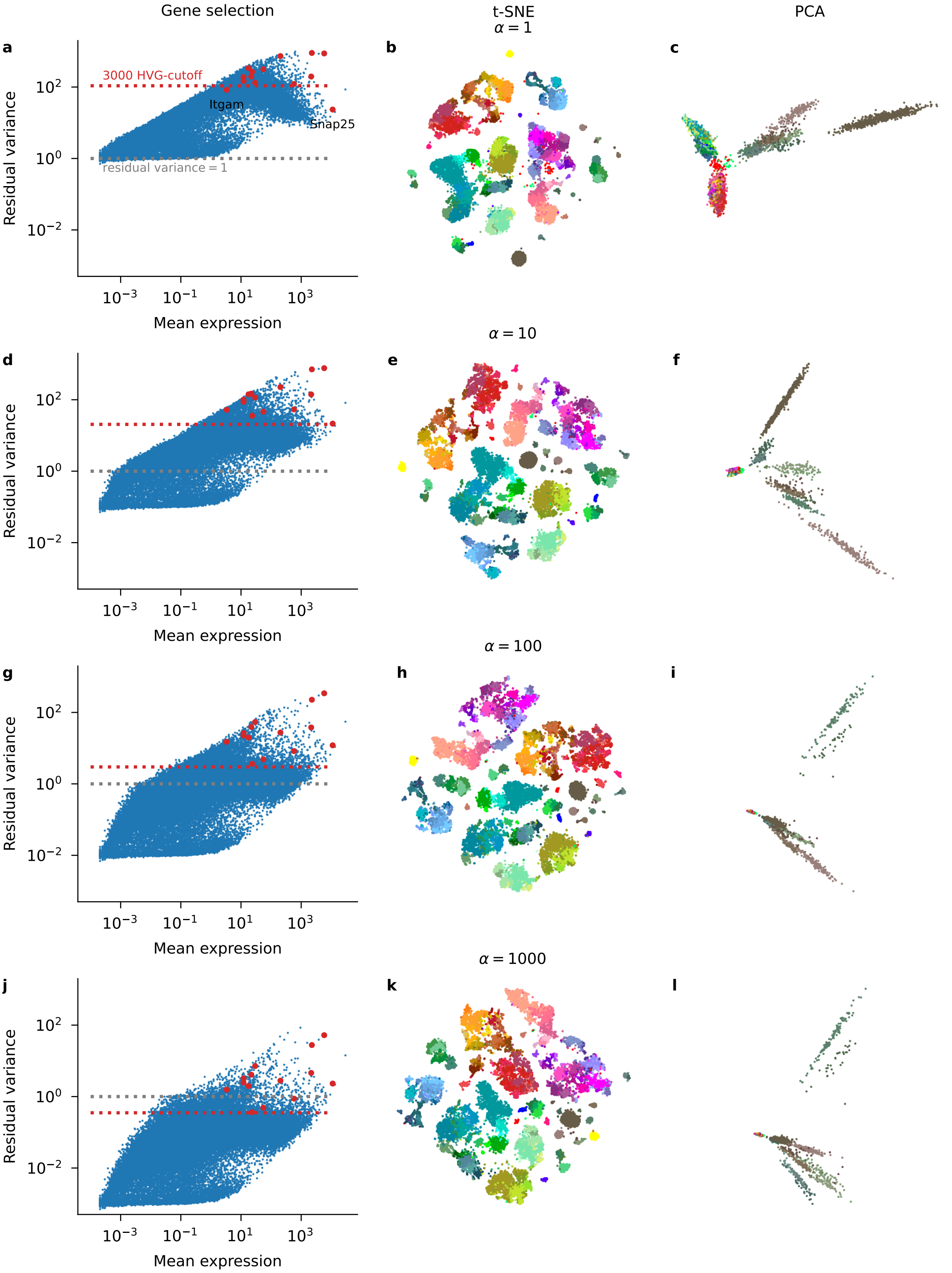

Figure S6: Influence of the amplification parameter α. Each row contains a reproduction of Figure 3a–b for various values of α_Z. The first row corresponds to the NB model without amplification, used for UMI data (Lause et al., 2021). Gray line: indicates residual variance = 1, where most non-differentially expressed genes should lie if the model is correct. All t-SNEs used the same shared initialization (see Methods). Right column shows the first two principal components (PCs) of the compound Pearson residuals.

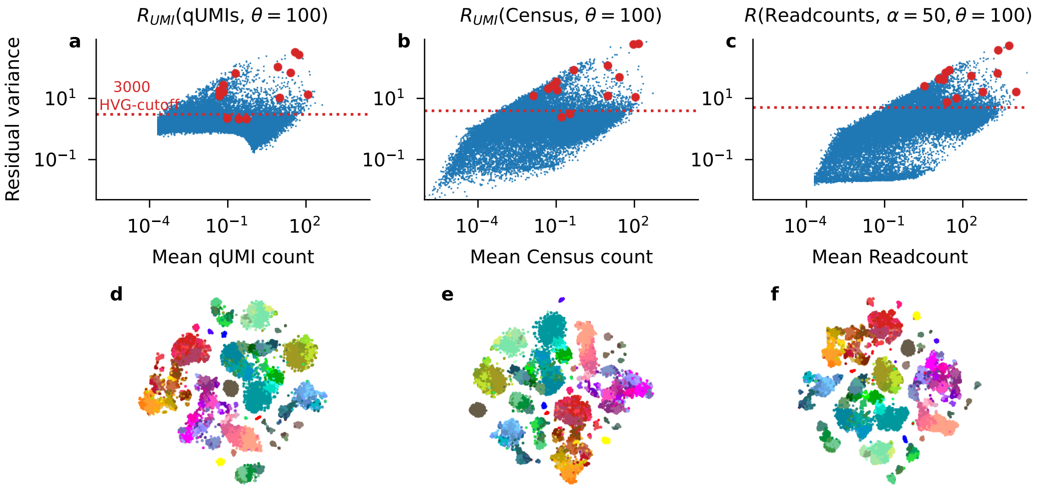

Figure S7: Comparing compound Pearson residuals to Census and qUMI. The same analysis as shown in Figure 3a–b for read counts from Tasic et al. (2018) processed with qUMIs (Townes and Irizarry, 2020) followed by UMI Pearson residuals (a, d); Census counts (Qiu et al., 2017) followed by UMI Pearson residuals (b, e); and compound Pearson residuals (our method) applied to the same set of genes (see Methods) (c, f).

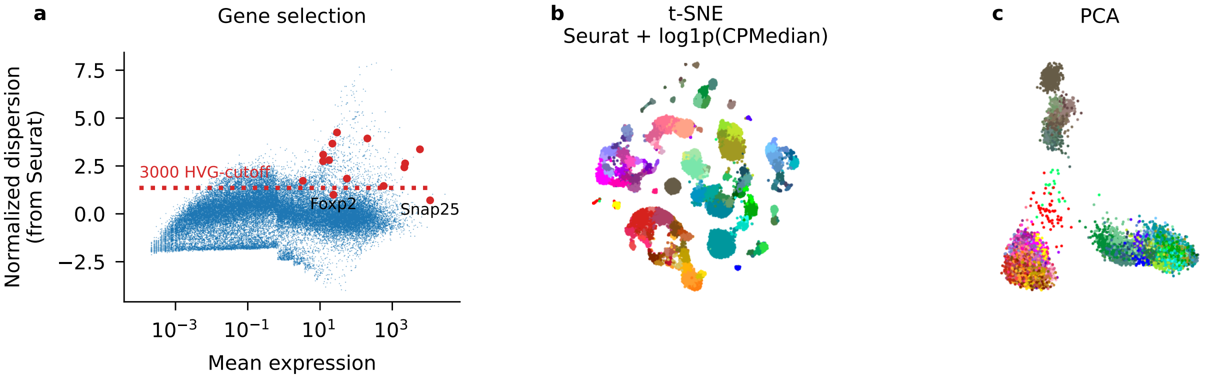

Figure S8: Pre-processing Smart-seq2 data with Scanpy default settings. The same dataset as in Figure S6, processed with `scanpy 1.9.0` defaults for normalization (counts per median normalization with `normalize_total()`, followed by `log1p()` transform) and gene selection (`flavor='seurat'`, Satija et al. (2015)) and PCA to 1 000 PCs. **a:** Seurat gene selection based on normalized dispersion. Note that two known markers (*Foxp2*, *Snap25*) are not among the top 3 000 genes selected by this method. The genes with highest normalized dispersion are markers of small, non-neural populations. **b:** t-SNE embedding of the pre-processed data. **c:** PCA embedding of the same, pre-processed data.

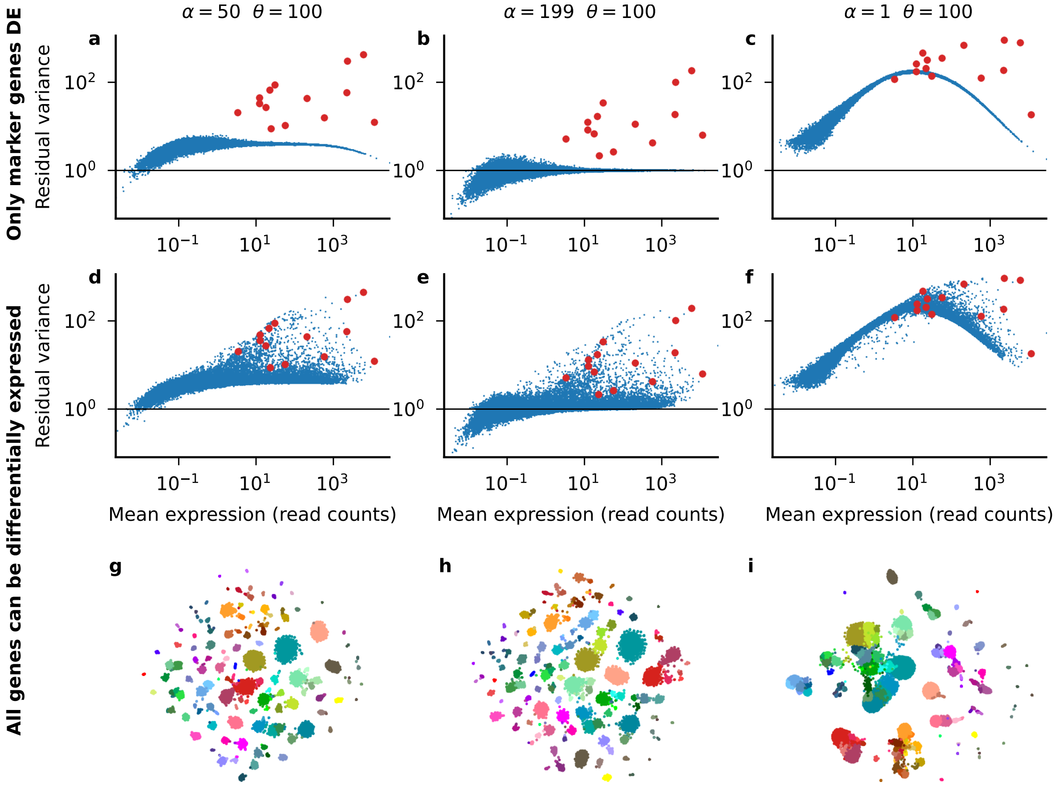

Figure S9: Compound Pearson residuals recover ground truth in realistic simulations. The same analysis as shown in Figure 3 for two simulated read count datasets that mirror the Tasic et al. (2018) cluster structure. Both simulated datasets were processed by compound Pearson residuals with three different settings: $\alpha_Z = 50$ as in Figure 3 (left); $\alpha_Z = 199$ which is the ground truth amplification factor used in this simulation (middle); $alpha_Z = 1$ corresponding to UMI Pearson residuals (right). **a–c:** Simulation I. Marker genes (red) were simulated with cluster-specific expression strengths from the Tasic et al. (2018) data, all other genes with their average expression strength across the whole dataset. Horizontal line indicates unit residual variance, expected for genes without differential expression. **d–i:** Simulation II. All genes were simulated with their cluster-specific expression strengths.

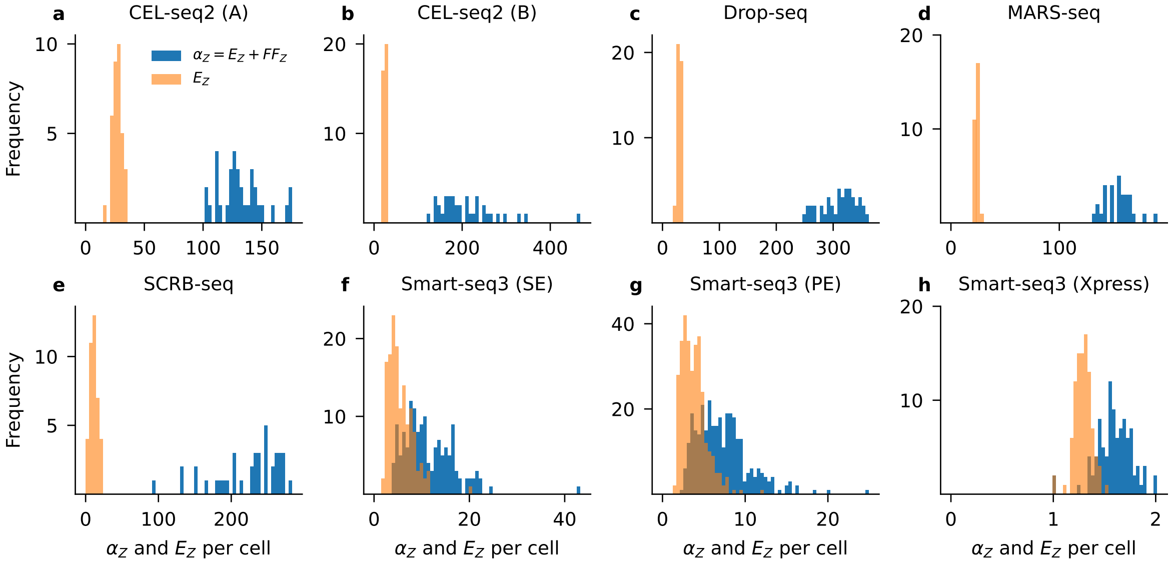

Figure S10: Cell-to-cell variability in α_Z and $\mathbb{E}[Z]$. Each panel shows amplification statistics computed per cell for all sequencing platforms listed in Table 1. Only run A is shown unless otherwise indicated.

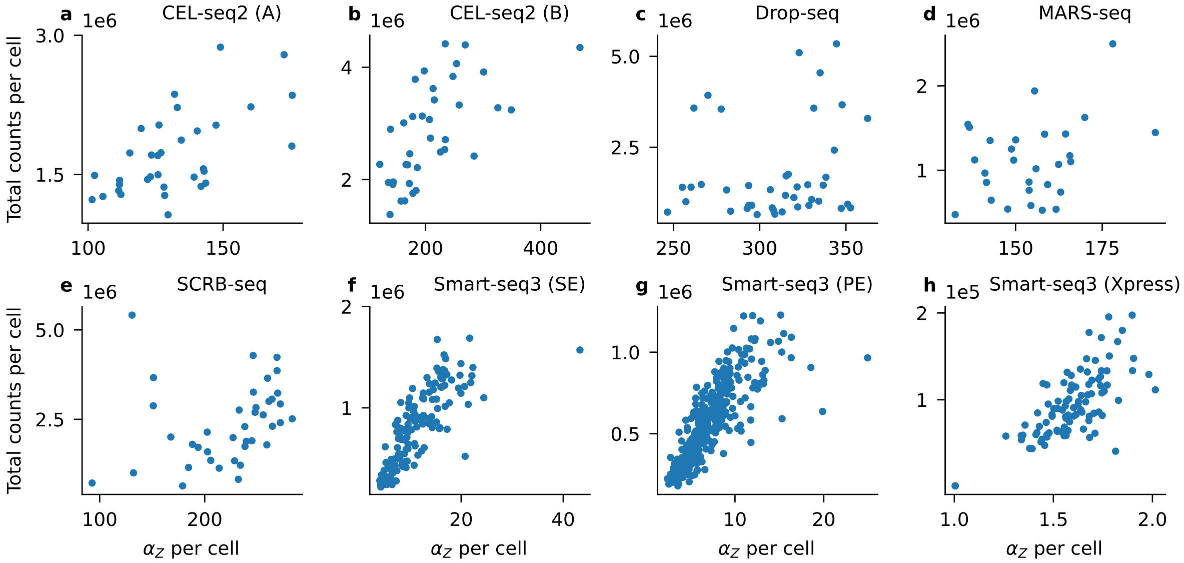

Figure S11: Total read counts are correlated with α_Z. Each panel corresponds to one of the sequencing platforms listed in Table 1; each dot is a cell. Only run A is shown unless otherwise indicated.