Biostatistics, 2025, **26**(1), kxaf012
https://doi.org/10.1093/biostatistics/kxaf012
Article

Addressing the mean–variance relationship in spatially resolved transcriptomics data with *spoon*

Kinnary Shah [1], Boyi Guo [1], Stephanie C. Hicks [1,2,3,4,*]

[1]Department of Biostatistics, Johns Hopkins Bloomberg School of Public Health, 615 N Wolfe Street, Baltimore, MD 21205, United States
[2]Department of Biomedical Engineering, Johns Hopkins School of Medicine, 733 N Broadway, Baltimore, MD 21205, United States
[3]Center for Computational Biology, Johns Hopkins University, 3100 Wyman Park Drive, Baltimore, MD 21211, United States
[4]Malone Center for Engineering in Healthcare, Johns Hopkins University, 3400 N Charles Street, Baltimore, MD 21218, United States

[*]Corresponding author: Department of Biostatistics, Johns Hopkins Bloomberg School of Public Health, 615 NWolfe Street, Baltimore, MD 21205, United States. Email: shicks19@jhu.edu

ABSTRACT

An important task in the analysis of spatially resolved transcriptomics (SRT) data is to identify spatially variable genes (SVGs), or genes that vary in a 2D space. Current approaches rank SVGs based on either *P*-values or an effect size, such as the proportion of spatial variance. However, previous work in the analysis of RNA-sequencing data identified a technical bias with log-transformation, violating the "mean–variance relationship" of gene counts, where highly expressed genes are more likely to have a higher variance in counts but lower variance after log-transformation. Here, we demonstrate the mean–variance relationship in SRT data. Furthermore, we propose *spoon*, a statistical framework using empirical Bayes techniques to remove this bias, leading to more accurate prioritization of SVGs. We demonstrate the performance of *spoon* in both simulated and real SRT data. A software implementation of our method is available at https://bioconductor.org/packages/spoon.

KEYWORDS: empirical Bayes; Gaussian process regression; mean–variance bias; spatial transcriptomics; spatially variable gene.

1. INTRODUCTION

Advances in transcriptomics have led to profiling gene expression in a 2D space using spatially resolved transcriptomics (SRT) technologies (Marx 2021). These technologies have already led to novel biological insights across diverse application areas, including cancer (Deshpande et al. 2023; Jin et al. 2024), developmental biology (Rao et al. 2021; Garcia-Alonso et al. 2022), and neurodegenerative disease (Chen et al. 2023; Vanrobaeys et al. 2023). These emerging data types have also motivated new computational challenges, such as spatially-aware quality control to identify low-quality observations (Totty et al. 2024) and spatially-aware clustering to identify discrete spatial domains (Yuan et al. 2024b). Another common data analysis task with these data is to perform feature selection by identifying a set of spatially variable genes (SVGs) (Svensson et al. 2018; Dries et al. 2021; Hao et al. 2021a; Zhu et al. 2021; Li et al. 2023; Weber et al. 2023b). The

Received: November 18, 2024. **Revised:** April 11, 2025. **Accepted:** April 14, 2025

top SVGs are identified by ranking the genes based on some metric, such as *P*-values or an effect size like the proportion of spatial variance (Svensson et al. 2018). Accurately identifying SVGs is important because the top features are often used for downstream analyses, such as dimensionality reduction or unsupervised clustering (Navarro et al. 2020; Wang et al. 2020; Maynard et al. 2021; Walker et al. 2022; Thompson et al. 2024).

Recently, a computational method to identify SVGs (Weber et al. 2023b) based on a nearest-neighbor Gaussian process (NNGP) regression model (Saha and Datta 2018) was developed. In the paper, the authors identified an important relationship in SRT data. Specifically, they found a relationship between the estimated spatial variation and the overall expression, where genes that have higher overall expression are more likely to be more spatially variable. This phenomenon, known as the "mean–variance relationship," is a well-documented technical bias in genomics (Robinson et al. 2010; Brennecke et al. 2013; Love et al. 2014; Ritchie et al. 2015; Antolović et al. 2017; Eling et al. 2018; Townes et al. 2019; Hao et al. 2021b; Ahlmann-Eltze and Huber 2023). As previously shown in other sequencing-based technologies, the reason for this bias is due to the preprocessing and normalization steps that are often applied to raw gene expression counts, or the number of unique molecular identifiers (UMIs) mapping to each gene. Specifically, the authors used normalized $\log_2$-transformed gene expression as input to the NNGP model (Weber et al. 2023b). These preprocessing techniques are widely used in bulk RNA-seq, scRNA-seq, and SRT data, because these transformations are assumed to enable the use of statistical models based on Gaussian distributions, rather than less tractable count-based distributions (Edsgärd et al. 2018; Svensson et al. 2018; Hafemeister and Satija 2019; Townes et al. 2019; Booeshaghi and Pachter 2021).

However, previous work in the analysis of bulk and scRNA-seq data has also shown that because counts have unequal variances [or larger counts have larger standard deviations compared to smaller counts (Law et al. 2014)] (Fig. S1A), applying these log-transformations is problematic as it can overcorrect (or large logcounts can have a smaller standard deviation than small logcounts) (Fig. S1B). In these settings, it is important to account for the mean–variance relationship. Another way to think about the mean–variance relationship is to describe it as heteroskedasticity (Buettner et al. 2015) in the context of using linear models. In contrast, homoskedasticity, in the case of profiling gene expression, would be if all genes in a sample had the same variance. When applying statistical models that assume homoskedasticity in the data, if we ignore the mean–variance relationship, our results would produce inefficient estimators or even incorrect results (Yang et al. 2019; Sun et al. 2020; Ahlmann-Eltze and Huber 2023). For example, in differential expression analysis, ignoring the mean–variance relationship can produce false positive differentially expressed genes (Love et al. 2014).

To address this technical bias in SRT data, here we introduce the *spoon* framework, which was inspired by the limma-voom method (Law et al. 2014) developed for bulk RNA-seq data. In this way, the name *spoon* incorporates the concepts of both "spatial" and its origin in RNA-seq. Using real and simulated SRT data, we show that *spoon* is able to correct for the mean–variance relationship leading to more accurately prioritizing SVGs. A software implementation of our method is available as an R/Bioconductor package (https://bioconductor.org/packages/spoon).

2. MATERIALS AND METHODS

2.1 An overview of the *spoon* model and methodological framework

The *spoon* model was inspired by the limma-voom method (Law et al. 2014), which estimates the mean–variance relationship to obtain precision weights for each observation to be used as input into a linear regression model to identify differentially expressed genes with bulk RNA-sequencing data (Ritchie et al. 2015). In *spoon*, we use a similar idea. First, we use empirical Bayes techniques to estimate observation- and gene-level weights. However, here we use a Gaussian process regression model, rather than a linear regression model, to model SRT data. Then, we leverage the Delta method to rescale the data and covariates by these weights to address the heteroskedasticity in SRT

data. Briefly, the Gaussian process (GP) regression model is specified as follows (Saha and Datta 2018):

$$y(\boldsymbol{s}) = \boldsymbol{x}(\boldsymbol{s})'\boldsymbol{\beta} + w(\boldsymbol{s}) + \epsilon(\boldsymbol{s}) \tag{1}$$

where $\boldsymbol{s}$ are the spatial locations, $y(\boldsymbol{s})$ is the response at a location, $\boldsymbol{x}(\boldsymbol{s})$ is a vector of explanatory variables, $w(\boldsymbol{s})$ is a function accounting for the spatial dependence, and $\epsilon(\boldsymbol{s}) \sim N(0, \tau^2)$ is noise. $\boldsymbol{\beta}$ is a fixed effect, while w and ϵ are random effects. $w(\boldsymbol{s})$ is modeled with a Gaussian process, $w(\boldsymbol{s}) \sim GP(\mu(\boldsymbol{s}), C(\boldsymbol{\theta}))$, where $\mu(\boldsymbol{s})$ is a mean function and $C(\boldsymbol{\theta})$ is a covariance function with parameters $\theta = (\sigma^2, \phi, ...)$ for the Matérn covariance function:

$$C(\boldsymbol{s_i}, \boldsymbol{s_j}|\boldsymbol{\theta} = (\sigma^2, \phi, \nu)) = \frac{\sigma^2}{2^{\nu-1}\Gamma(\nu)}(||\boldsymbol{s_i} - \boldsymbol{s_j}||\phi)^{\nu} K_{\nu}(||\boldsymbol{s_i} - \boldsymbol{s_j}||\phi); \phi > 0, \nu > 0$$

where σ^2 is the spatial component of variance, ϕ is the decay in spatial correlation, ν is the smoothness parameter, and K_{ν} is the Bessel function of the second kind with order ν. Because we fit these models on a per-gene basis with up to thousands of genes in a given dataset, we use a nearest-neighbor Gaussian process (NNGP) (Datta et al. 2016; Finley et al. 2019) to reduce the computational running time and make *spoon* useful to practitioners. The key idea behind using NNGPs is that instead of conditioning on all of the points in the data, only a subset (a set of nearest neighbors) of the data are used for the conditioning. Conditioning on enough of the closest neighbors provides sufficient estimates of the information needed and improves storage and computational costs. Briefly, a NNGP is fit to the preprocessed expression values for each gene:

$$\boldsymbol{y} \sim N(\boldsymbol{X}\beta, \tilde{\boldsymbol{\Sigma}}(\boldsymbol{\theta}, \tau^2)) \tag{2}$$

where the primary difference between a full GP model (1) and a NNGP (2) is that the NNGP covariance matrix, $\tilde{\boldsymbol{\Sigma}}(\boldsymbol{\theta}, \tau^2)$, is a computationally fast approximation to the covariance matrix from a full GP model, $\Sigma(\boldsymbol{\theta}, \tau^2) = \boldsymbol{C}(\boldsymbol{\theta}) + \tau^2\boldsymbol{I}$. In other words, $\tilde{\boldsymbol{\Sigma}}$ approximates the covariances from both from $w(\boldsymbol{s})$ and $\epsilon(\boldsymbol{s})$. For the kernel, $C(\boldsymbol{\theta}) = [C_{ij}(\boldsymbol{\theta})]$, we assume an exponential covariance function:

$$C_{ij}(\boldsymbol{\theta}) = \sigma^2 \exp\left(\frac{-||\boldsymbol{s_i} - \boldsymbol{s_j}||}{l}\right)$$

where $\boldsymbol{\theta} = (\sigma^2, l)$, and σ^2 is the spatial component of variance of interest. σ^2 is different from the nonspatial component of variance, τ^2, which is also referred to as the nugget. l is the lengthscale parameter, which sets how quickly the correlation decays with distance. $||\boldsymbol{s_i} - \boldsymbol{s_j}||$ is the Euclidean distance between spatial locations. To estimate the parameters in the NNGP model, we use the BRISC R package (Saha and Datta 2018). Using the estimated parameters, we calculate an effect size, the proportion of spatial variance ($\frac{\hat{\sigma}^2}{\hat{\sigma}^2+\hat{\tau}^2}$).

2.2 Calculating observation- and gene-level weights using empirical Bayes techniques

Briefly, we calculate the average $\log_2$ expression values and the standard deviations of the residuals from fitting a NNGP model per gene using BRISC (Fig. 1A). Then, we use splines to fit the gene-wise mean–variance relationship (Fig. 1B). Finally, we use the fitted curve to estimate observation- and gene-level weights (Fig. 1C). Next, we describe each of these steps in greater detail.

2.2.1. Fitting per-gene NNGP models using logCPM values

We start with a counts matrix, transposed so each row is a spot and each column is a gene. There are n spots and G genes in the counts matrix. The UMI counts can be indexed by r_{gi} for spots $i = 1$ to n

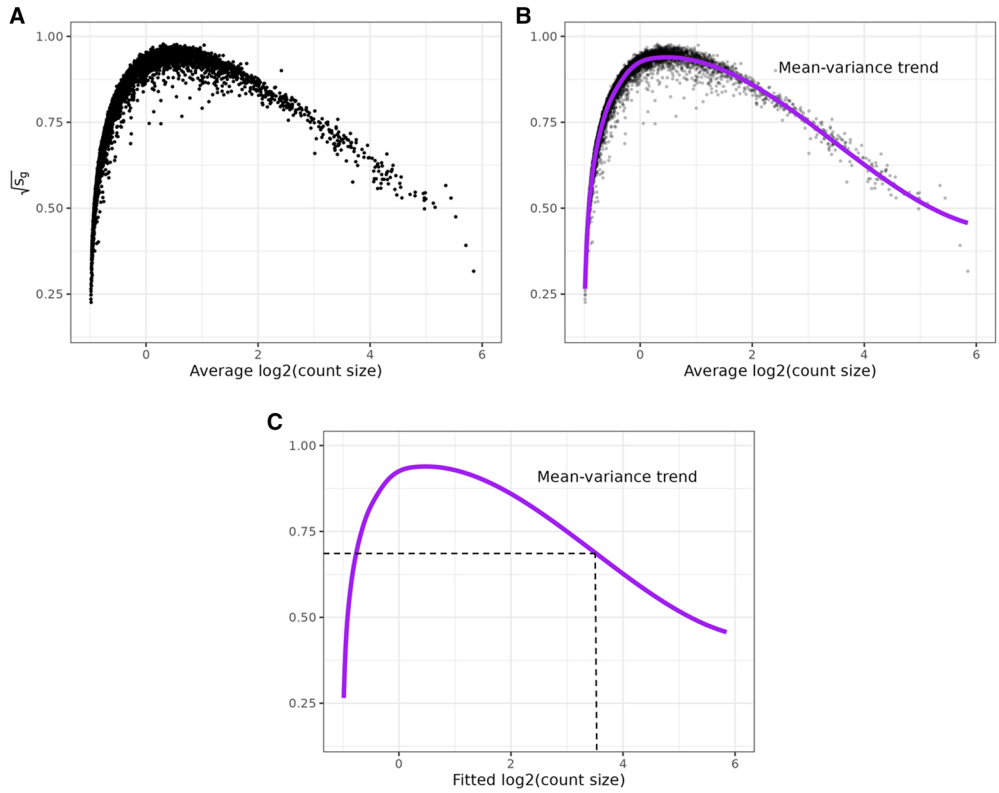

Fig. 1. Calculating precision weights for individual observations. These data are from Invasive Ductal Carcinoma breast tissue analyzed with 10x Genomics Visium (10x Genomics 2022), hereafter referred to as "Ductal Breast." A)–C) The square root of the residual standard deviations estimated using nearest neighbor Gaussian processes [$\sqrt{s_g}$ defined in (3)] are plotted against average logcount ($\tilde{\boldsymbol{r}}$). B) Same as A, except a spline curve is fitted to the data to estimate the gene-wise mean–variance relationship. C) Using the fitted spline curve, each predicted count value ($\hat{\lambda}_{gi}$) is mapped to its corresponding square root standard deviation value using $\texttt{spl}(\hat{\lambda}_{gi})^{-4}$.

and genes $g = 1$ to G. We define the total number of UMIs for sample i as $R_i = \sum_{g=1}^{G} r_{gi}$. Next, we transform r_{gi} to adjust for the total number of UMIs (R_i) by using logcounts per million (logCPM). We use a pseudocount of 0.5 to ensure we do not take the log of 0 and we add a pseudocount of 1 to the library size to make sure $0 < \frac{(r_{gi}+0.5)}{R_i+1} < 1$:

$$y_{gi} = \log_2\left(\frac{r_{gi} + 0.5}{R_i + 1} \times 10^6\right)$$

Using the normalized and $\log_2$-transformed data y_{gi}, we fit a NNGP model (2) per gene with a default of $\boldsymbol{X} = \mathbf{1}_{[N\times 1]}$, corresponding to including an intercept, with β_g representing the overall mean expression level for gene g. Using the observed data y_{gi} and the predicted value $\hat{\mu}_{gi} = x_i^T \hat{\beta}_g$, we can calculate the standard deviation of the residuals between y_{gi} and $\hat{\mu}_{gi}$:

$$s_g = \sqrt{\frac{\sum_{i=1}^{n}(y_{gi} - \hat{\mu}_{gi})^2}{n-1}} \tag{3}$$

The square root of s_g is what we use to represent the "variance" in the mean–variance relationship (see y-axis in Fig. 1A–C). This concept is used in limma-voom as well because the square root of the standard deviations is roughly symmetrically distributed.

2.2.2. *Modeling the mean–variance relationship using $\sqrt{s_g}$ and $\tilde{\boldsymbol{r}}$*

Next, we fit a nonparametric spline curve to model the mean–variance relationship in our data. Instead of using $\bar{y}_g$ directly to represent the "mean" component, we convert $\bar{y}_g$ to average logcount using the geometric mean of library size, $\tilde{\boldsymbol{R}} = \exp\left(\sum_{i=1}^{n} \log(R_i)\right)$. We use the geometric mean to avoid integer overflow:

$$\tilde{\boldsymbol{r}} = \bar{y}_g + \log_2(\tilde{\boldsymbol{R}}) - \log_2(10^6) \tag{4}$$

Then, we use smoothing splines (specifically `smooth.spline()` in the base R `stats` package) to model the mean–variance relationship between $\sqrt{s_g}$ and $\tilde{\boldsymbol{r}}$. We use splines because we found they are a robust way to model the mean–variance relationship seen across multiple datasets. We use the notation `spl()` to denote the fitted curve (Fig. 1B), which represents an estimate of the gene-wise mean–variance relationship.

2.2.3. *Prediction modeling using fitted `spl()` curve*

Similar to (4), we convert the predicted value $\hat{\mu}_{gi}$ (on the logCPM scale) to a predicted count value:

$$\hat{\lambda}_{gi} = \hat{\mu}_{gi} + \log_2(R_i + 1) - \log_2(10^6) \tag{5}$$

The fitted counts values for each observation are used as input to predict the square root residual standard deviation values for each y_{gi} using the spline curve. Figure 1C shows an example of mapping an individual observation to a square-root standard deviation value using its fitted value from the BRISC models.

To avoid extrapolating beyond the range of the function, individual observations that have $\hat{\lambda}_{gi}$ more extreme than the range of $\tilde{\boldsymbol{r}}$ are constrained. If $\hat{\lambda}_{gi}$ is greater than $\max(\bar{\boldsymbol{y}})$, then the predicted square root residual standard deviation value for that observation is constrained to $\texttt{spl}(\max(\bar{\boldsymbol{y}}))$. If $\hat{\lambda}_{gi}$ is less than $\min(\bar{\boldsymbol{y}})$, then the predicted square root residual standard deviation value for that observation is constrained to $\texttt{spl}(\min(\bar{\boldsymbol{y}}))$. The final step is taking the inverse of the squared predicted standard deviation to compute the weight for each individual observation. The weight for each observation is defined as $w_{gi} = \texttt{spl}\,(\hat{\lambda}_{gi})^{-4}$, using the constrained values for observations outside of the range.

2.3 Correct for heteroskedasticity using observation- and gene-level precision weights

If the desired SVG detection method accepts observation- and gene-level weights, then the estimated weights w_{gi} (described in Section 2.2) can be used as input directly into the method. If the desired SVG detection method does not accept weights, then the Delta method is leveraged to rescale the data and covariates by the weights. These scaled data and covariates are used as inputs into the desired SVG detection function.

For example, the SVG detection tool called nearest neighbor SVGs (nnSVG) (Weber et al. 2023b) uses a Gaussian process regression model and can have weights incorporated in the following way. We correct for the heteroskedasticity by adjusting with precision weights, w_{gi} for gene g at spatial location i. If $\boldsymbol{W}$ is a diagonal matrix where each diagonal element is w_{gi}, then we

know: $\boldsymbol{W}\boldsymbol{y} \sim N(\boldsymbol{W}\boldsymbol{X}\beta, \boldsymbol{W}\Sigma\boldsymbol{W})$ where

$$\begin{aligned}\boldsymbol{W}\Sigma\boldsymbol{W} &= \boldsymbol{W}\boldsymbol{C}(\boldsymbol{\theta})\boldsymbol{W} + \tau^2\boldsymbol{W}\boldsymbol{I}\boldsymbol{W} \\ &= \boldsymbol{W}\boldsymbol{C}(\boldsymbol{\theta})\boldsymbol{W} + \tau^2\boldsymbol{W}\boldsymbol{W} \\ &= \boldsymbol{C}(\boldsymbol{\theta}') + \tau'^2\boldsymbol{I}\end{aligned}$$

and the new input data to nnSVG would be $\boldsymbol{W}\boldsymbol{y}$ and $\boldsymbol{W}\boldsymbol{X}$ where $\boldsymbol{X} = \begin{bmatrix} 1 \\ \boldsymbol{X}_{gi} \end{bmatrix}$.

2.4 Data

2.4.1. Real SRT data

Tissues from several regions of the human body analyzed with 10x Genomics Visium were used in the analyses. The datasets and preprocessing steps are further described below:

- DLPFC: This dataset contains two pairs of spatial replicates of human postmortem dorsolateral prefrontal cortex (DLPFC) tissue from three neurotypical adult donors. Only tissue sample 151507 is used for this analysis (Maynard et al. 2021). After preprocessing, this dataset contains 7,343 genes and 4,221 spots.
- Ductal Breast: Invasive Ductal Carcinoma breast tissue data are publicly available from the 10x Genomics website. It contains one tissue sample from one donor with Invasive Ductal Carcinoma (10x Genomics 2022). After preprocessing, this dataset contains 12,321 genes and 4,898 spots.
- ER+ Breast: Estrogen receptor positive (ER+) breast cancer tissue data are publicly available on Zenodo and contains several tissue samples of breast cancer tissue. Only sample CID4290 is used for this analysis (Wu et al. 2021a). After preprocessing, this dataset contains 12,325 genes and 2,419 spots.
- HPC: This dataset contains human postmortem hippocampus (HPC) tissue from several neurotypical adult donors. Each sample was broken up into four Visium slides due to the large size. Only tissue sample V12D07_335, portion D1 is used for this analysis (Thompsonet al. 2024). After preprocessing, this dataset contains 5,348 genes and 4,992 spots.
- LC: This dataset contains human postmortem locus coeruleus (LC) tissue from five neurotypical adult donors. Only tissue sample 2701 is used for this analysis (Weber et al. 2023a). After preprocessing, this dataset contains 1,331 genes and 2,809 spots.
- Lobular Breast: Invasive Lobular Carcinoma breast tissue data are publicly available from the 10x Genomics website. It contains one tissue sample from one donor with Invasive Lobular Carcinoma (10x Genomics 2020). After preprocessing, this dataset contains 12,624 genes and 4,325 spots.
- Ovarian: This dataset contains tissues collected during interval debulking surgery from eight high-grade serous ovarian carcinoma patients undergoing chemotherapy. Only one tissue sample from patient 2 is used for this analysis (Denisenko et al. 2024). After preprocessing, this dataset contains 12,022 genes and 1,935 spots.

Preprocessing was performed as uniformly as possible across the datasets. For datasets that had an annotation for whether or not a spot was in the tissue, spots outside of the tissue were removed. For the ER+ Breast dataset, spots that were classified as artifacts were removed. `nnSVG::filter_genes()` was used to remove genes without enough data, specifically we kept genes with at least two counts in at least 0.2% of spots. For the LC dataset, we used a UMI filter instead of this function to remove genes with less than 80 total UMI counts summed across all spots. `scuttle::logNormCounts()` with default arguments was used to compute log-normalized expression values.

2.4.2. *Simulated SRT data*

To simulate the mean–variance relationship, we simulated raw gene expression counts following a Poisson distribution: $\boldsymbol{c}(\boldsymbol{s})|\lambda(\boldsymbol{s}) \sim Poisson(\lambda(\boldsymbol{s}))$; $\lambda(\boldsymbol{s}) = \exp(\boldsymbol{\beta} + \boldsymbol{C}(\sigma^2))$, where $\boldsymbol{s}$ are spatial locations, $\boldsymbol{\beta}$ is a vector of true mean expression per gene, σ^2 is the spatial component of variance, and $\boldsymbol{C}$ is the covariance function using a Matérn kernel with squared exponential distance. The σ^2 values and $\boldsymbol{\beta}$ values were randomly assigned from ranges of [0.2, 1] and [ln(0.5), ln(1)], respectively. We intentionally simulate σ^2 and β values so they are not correlated. In this way, we ensure we are simulating SVGs at all levels of mean expression. A fixed lengthscale parameter was chosen for all of the genes in a given simulation. Based on the estimated lengthscale distributions for four datasets, we chose to focus our simulations on smaller lengthscales because the majority of estimated lengthscales are between 0 to 0.15 (Fig. S2). For reference, a scaled lengthscale value of 0.15 is interpreted as 15% of the maximum width or height of the tissue area on a standard Visium slide. We simulated 1000 genes in the following simulations.

In addition, we also considered the performance as a function of varying the lengthscale parameter l in $\theta = (\sigma^2, l)$. In the NNGP model, the lengthscale parameter sets how quickly the correlation decays with distance. In the nnSVG SVG detection method (Weber et al. 2023b), a key innovation was using a flexible lengthscale parameter to fit the model for each gene. Genes within the same tissue can spatially vary with different ranges of sizes and patterns, so a flexible lengthscale parameter for each gene enables the discovery of distinct biological processes. For the primary simulation evaluation, a lengthscale of 100 was used. This corresponds to a scaled lengthscale value of roughly 0.02. For supplementary simulation evaluations, 50, 60, 100, and 500 lengthscales were used. These correspond to 0.010, 0.012, 0.020, and 0.100 of the maximum width or height of the tissue area on a standard Visium slide. The spatial coordinates from the example dataset `Visium_DLPFC()` in the `STexampleData` package were used. This dataset contains 4,992 spots. We used the subset of 968 spots with row and column coordinates between 20 to 65 as the spatial coordinates to reduce the amount of time to simulate data.

2.5 Methods to detect SVGs

For Moran's I (Moran 1950), we ranked genes by the Moran's I value. For nnSVG (Weber et al. 2023b), the genes were ranked within the method based on the estimated likelihood ratio test statistic values comparing the fitted model against a classical linear model, assuming the spatial component of variance is zero. For SpaGFT (Chang et al. 2024), the gene ranks were calculated within the method based on decreasing GFTscore, a measure of randomness of gene expression. For SPARK-X (Zhu et al. 2021), adjusted combined *P*-values from multiple covariance matrices and kernels were used to rank genes. For SpatialDE2 (Kats et al. 2021), the genes were ranked by the negative of the fraction of spatial variance for each gene. For SMASH (Seal et al. 2023), the genes are ranked similarly to SPARK-X. For HEARTSVG (Yuan et al. 2024a), the genes are ranked based on combined adjusted *P*-values from Portmanteau tests for significant autocorrelations in time series representations of the data. All of the criteria were ranked using the `ties.method = "first"` option.

1. Moran's I: `Rfast2::moranI()` (Tsagris and Papadakis 2018) was used to compute Moran's I values, and the negative Moran's I value for each gene was ranked.
2. nnSVG: `nnSVG::nnSVG()` (Weber et al. 2023b) was used, and the rank was calculated as part of the output of the function.
3. SpaGFT: `SpaGFT.detect_svg()` (Chang et al. 2024) was implemented in Python, and the rank was calculated as part of the output of the function.
4. SPARK-X: `SPARK::sparkx()` (Zhu et al. 2021) was run with the option of a mixture of various kernels. The combined *P*-value from all the kernels for each gene was ranked.
5. SpatialDE2: `SpatialDE.fit()` (Seal et al. 2023) was implemented in Python to fit the model. The negative of the fraction of spatial variance for each gene was ranked.

6. SMASH: `SMASH.SMASH()` (Kats et al. 2021) was implemented in Python to fit the model. The *P*-value for each gene was ranked.
7. HEARTSVG: `HEARTSVG::heartsvg()` (Yuan et al. 2024a) was used, and the rank was calculated as part of the output of the function.

An intercept-less covariate matrix is required to implement a weighted version of an SVG detection method. To the best of our knowledge, nnSVG is the only SVG detection tool with the option to include a covariate matrix without an intercept term. The weights from *spoon* have the potential to integrate with other methods based on the flexibility of their design.

3. RESULTS

3.1 The mean–variance relationship exists in spatial transcriptomics data

We begin by systematically demonstrating the mean–variance relationship in SRT data. This finding builds upon the initial finding suggested by Weber et al. (2023b). In contrast to investigating this bias in one tissue from one tissue section, here we explore this finding across multiple tissue sections from different regions in the human body, namely DLPFC, Ductal Breast cancer, HPC, LC, and Ovarian cancer. To visualize the mean–variance relationship, we plot the mean logcounts against different components (spatial and non-spatial components) of variance calculated using `nnSVG`. As seen in Fig. 2, the mean–variance relationship is a concern in SRT data, specifically in the nonspatial component of variance, τ^2. Given that τ^2 is used when calculating the proportion of spatial variance, this suggests the way genes are prioritized as spatially variable is dependent on the overall mean expression for the gene.

Next, we further investigated one of these tissues (DLPFC) to ask if the mean–variance relationship was due to differences in the spatial domains of the tissue. The six layers in the human neocortex are transcriptionally quite different from one another (Maynard et al. 2021), so we wanted to show that the mean–variance relationship still exists when stratifying by layer. In order to control for differences in layer domains, the DLPFC data was first separated into Layers I-VI, and white matter and then the mean logcounts were plotted against the components of variance for each layer in the brain. However, we found that the mean–variance relationship was still observed within the different biological domains (Fig. S3).

3.2 The mean-rank relationship exists in other SVG detection methods

Having established that the mean–variance relationship exists in SRT data across different tissues as measured by Gaussian processes in `nnSVG`, we next explored the mean-rank relationship as an extension of the mean–variance relationship. Other SVG detection methods do not separate out the total variance into spatial and nonspatial variance components, so we examine the mean–variance relationship using this proxy.

We examined the mean-rank relationship from several popular SVG detection methods on the DLPFC, Ovarian cancer, and Lobular Breast cancer datasets (Fig. 3). The ranks were calculated for each SVG method (described in Section 2.5). We define high mean expression deciles as deciles containing highly expressed genes (with 10 being the highest), and low mean expression deciles as deciles containing lowly expressed genes (with 1 being the lowest). We also note that low ranked genes are genes with the highest spatial variance. We found that for almost every method, there is a clear relationship between the mean and the rank. Stated another way, the SVG detection methods that we evaluated rank and prioritize genes as SVGs, which is related to the overall mean expression. The modes of the signal distributions in the higher mean expression deciles are lower than the modes of signal distributions of the lower mean expression deciles, illustrating the mean-rank relationship. Because the overall mean expression is likely a technical artifact, we would expect that there should be genes that are highly ranked as SVGs within each mean-level decile. However, what we found is that the mean–variance relationship biases genes towards the higher mean expression deciles. The extreme bias observed in SPARK-X is also noted in a recent benchmarking paper

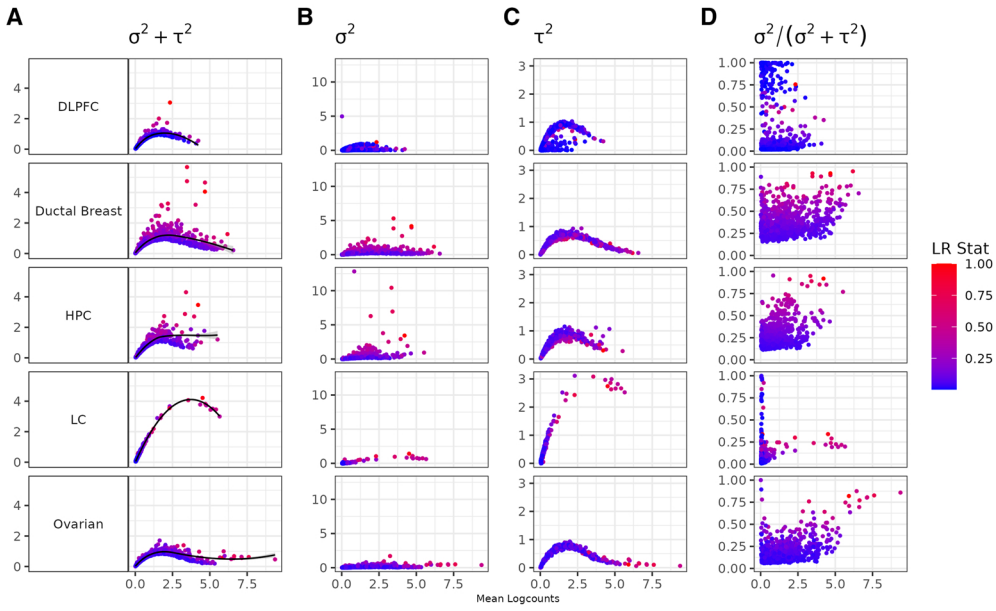

Fig. 2. Mean–variance relationship exists in spatially resolved transcriptomics. Using data from different human tissues, in order from top to bottom: DLPFC (Maynard et al. 2021), Ductal Breast cancer (10x Genomics 2022), HPC (Thompson et al. 2024), LC (Weber et al. 2023a), and Ovarian cancer (Denisenko et al. 2024), we quantified the mean–variance relationship. Each point is a gene colored by the likelihood ratio statistic for a test that compares the fitted model against a classical linear model for the spatial component of variance using a NNGP (Weber et al. 2023b). The likelihood ratio statistics (LR Stat) are scaled by the maximum likelihood ratio statistic for each dataset in order to have more uniform visualization. The *x*-axis is mean logcounts and the *y*-axes represent different components of variance, in order from left to right: A) total variance $\sigma^2 + \tau^2$, B) spatial variance σ^2, C) nonspatial variance τ^2, and D) proportion of spatial variance $\sigma^2/(\sigma^2 + \tau^2)$.

(Chen et al. 2024). These are state of the art methods that perform well in recent benchmarking papers (Li et al. 2023; Chen et al. 2024, 2025), yet they are sorely affected by the mean–variance bias. SPARK-X, SpatialDE2, SMASH, and HEARTSVG are still impacted by the mean–variance relationship despite directly modeling raw counts instead of log-transformed counts.

3.3 Simulation: weighted spatially variable gene evaluation

To address the mean–variance and mean-rank relationships, we began with simulation studies to evaluate the performance of *spoon* under different scenarios. Using simulated raw gene expression counts following a Poisson distribution (Section 2.4.2) with a fixed lengthscale (l=100), we ranked SVGs using nnSVG (Weber et al. 2023b) without weights and with weights estimated via *spoon*. We found a strong mean-rank relationship using the unweighted SVGs (Fig. 4A) compared to the weighted SVGs using *spoon* (Fig. 4B). Stated differently, using observational- and gene-level weights, we can identify highly ranked SVGs even in lower deciles, demonstrating that *spoon* effectively addresses the mean–variance relationship. Because rank is a relational process, moving some of the higher ranked genes from the lower deciles to the higher deciles becomes a reactive process that also shifts some of the lower ranked genes from the higher deciles into lower deciles.

We also explored the false discovery rate (FDR) (Fig. 4C), true negative rate (TNR) (Fig. 4D), and true positive rate (TPR) (Fig. 4E). The red represents weighted `nnSVG` and the blue represents unweighted `nnSVG`. These plots represent the average of each respective rate over five iterations

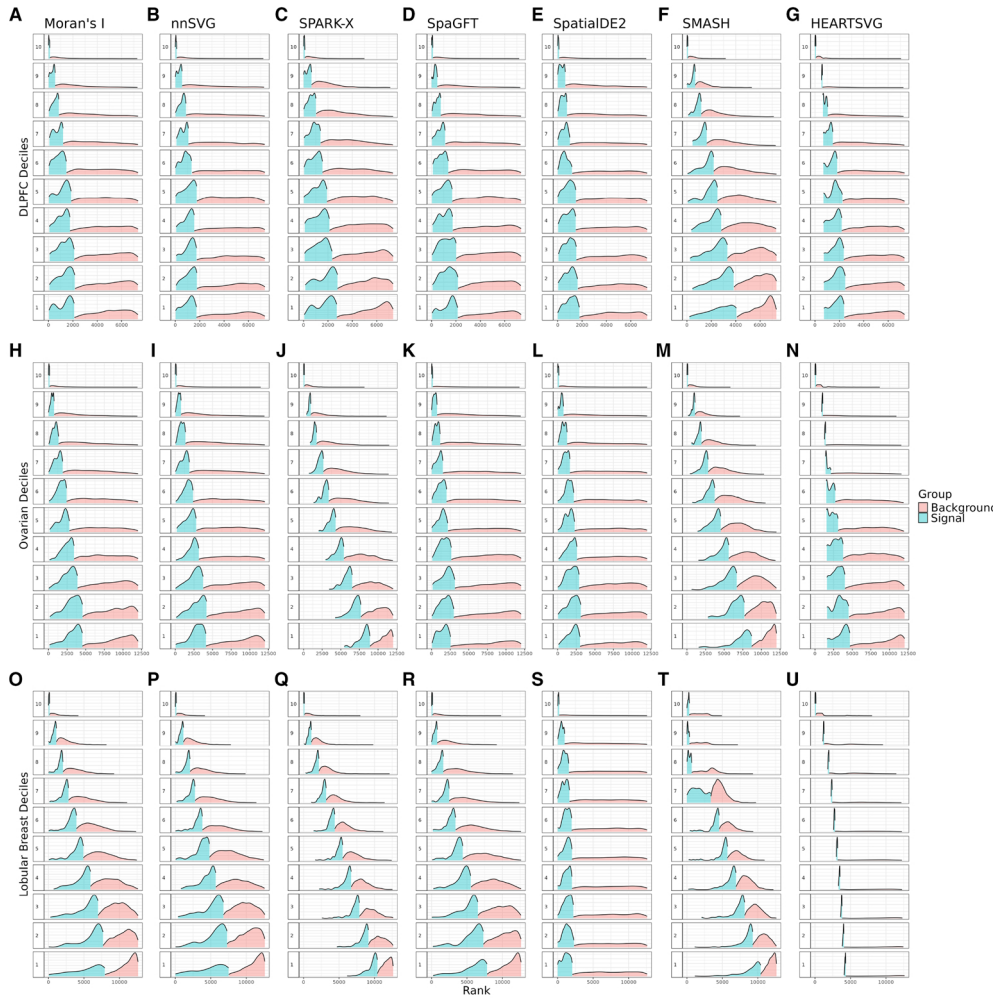

Fig. 3. Mean-rank relationship exists in spatial transcriptomics data. Using three datasets, in order from top to bottom [DLPFC (Maynard et al. 2021), Ovarian cancer (10x Genomics 2022), and Lobular Breast cancer (10x Genomics 2020)], we quantified the mean-rank relationship. The genes were binned into deciles based on mean logcounts. Decile 1 contains the lowest mean expression values. The *x*-axis represents the rank. Within each decile, the density of the top 10% ranks is plotted as the signal in blue, while the density of the remaining ranks is plotted as the background in orange. Each subfigure shows the mean-rank relationship that persists after applying each method, from left to right: A), H), O) Moran's I (Tsagris and Papadakis 2018), B), I), P) nnSVG (Weber et al. 2023b), C), J), Q) SPARK-X (Zhu et al. 2021), D), K), R) SpaGFT (Chang et al. 2024), E), L), S) SpatialDE2 (Kats et al. 2021), F), M), T) SMASH (Seal et al. 2023), and G), N), U) HEARTSVG (Yuan et al. 2024a).

of the same simulation with unique random seeds. The FDR and TNR are similar between the unweighted and weighted methods, with a slight increase in performance observed in the unweighted method. The TPR, however, is very similar for both methods. Finally, we considered other lengthscale values and found that the mean–variance relationship is improved for all values tested (Fig. S4). We found that the weights from *spoon* improve the TPR for smaller lengthscale values, and there are diminishing returns regarding the convergence of the TPR for both the weighted method and unweighted methods at larger lengthscale values.

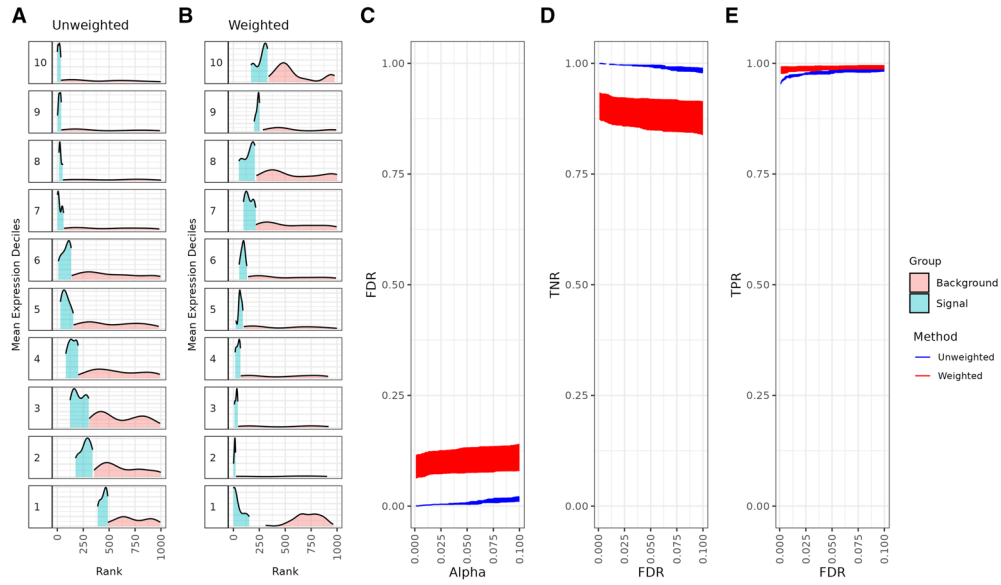

Fig. 4. *Spoon* removes the mean–variance relationship when detecting spatially variable genes. This dataset consists of 1,000 simulated genes across 968 spots using a lengthscale of 100. Separately for unweighted and weighted methods, the genes were binned into deciles based on mean logcounts. Decile 1 contains the lowest mean expression values. Ridge plots for the A) unweighted ranks and B) weighted ranks are shown. Within each decile (y-axis), the density of the top 10% of ranks is plotted as the signal, while the density of the remaining ranks is plotted as the background. C) False discovery rate (FDR) as a function of Type I error (α). As a function of FDR, we show the D) true negative rate (TNR) and E) true positive rate (TPR). The red represents weighted `nnSVG` and the blue represents unweighted `nnSVG`. These plots represent the average performance across five iterations of the same simulation, each with unique random seeds.

3.4 Real data: weighted spatially variable gene evaluation

Next, we evaluated the downstream impact of incorporating weights from *spoon* into SVG detection methods. Here, we aimed to demonstrate the impact of our method on recovering lowly expressed genes that become highly ranked in real biological datasets. We defined small mean gene expression genes as those with means less than the 25th percentile in the dataset. Within the set of small mean gene expression, we identified genes that were in the lowest 10% of ranks before weighting and then increased to the highest 10% of ranks after weighting. In the Ovarian cancer dataset, there are 7 genes that met this criterion. Out of these 7 genes, *TUFT1* and *DDX39B* are known to be implicated in ovarian cancer (Xu et al. 2020; Opławski et al. 2022). These potentially important SVGs were ignored due to their low expression levels and our weighting algorithm can recapture them. Similar analyses were performed for the other three cancer datasets (Fig. 5). The gene lists can be found in the Supplemental Materials.

Then, we explored the improvement in the low lengthscale set of genes. We defined low lengthscale genes as those with lengthscale values between 40 to 90. Within the set of low lengthscale genes, we found genes that were ranked higher after weighting. We also derived the "null distribution"—the underlying total SVGs for each dataset as a point of reference for the proportion of low lengthscale genes that are ranked higher. We found that the differing proportions of low lengthscale genes that become higher ranked after weighting is appropriate based on the "null distribution" of the proportion of unweighted SVGs (Fig. S5). Again, we related the higher-ranked low lengthscale genes to the cancer type of the dataset. In the ER+ Breast dataset, 59 low lengthscale

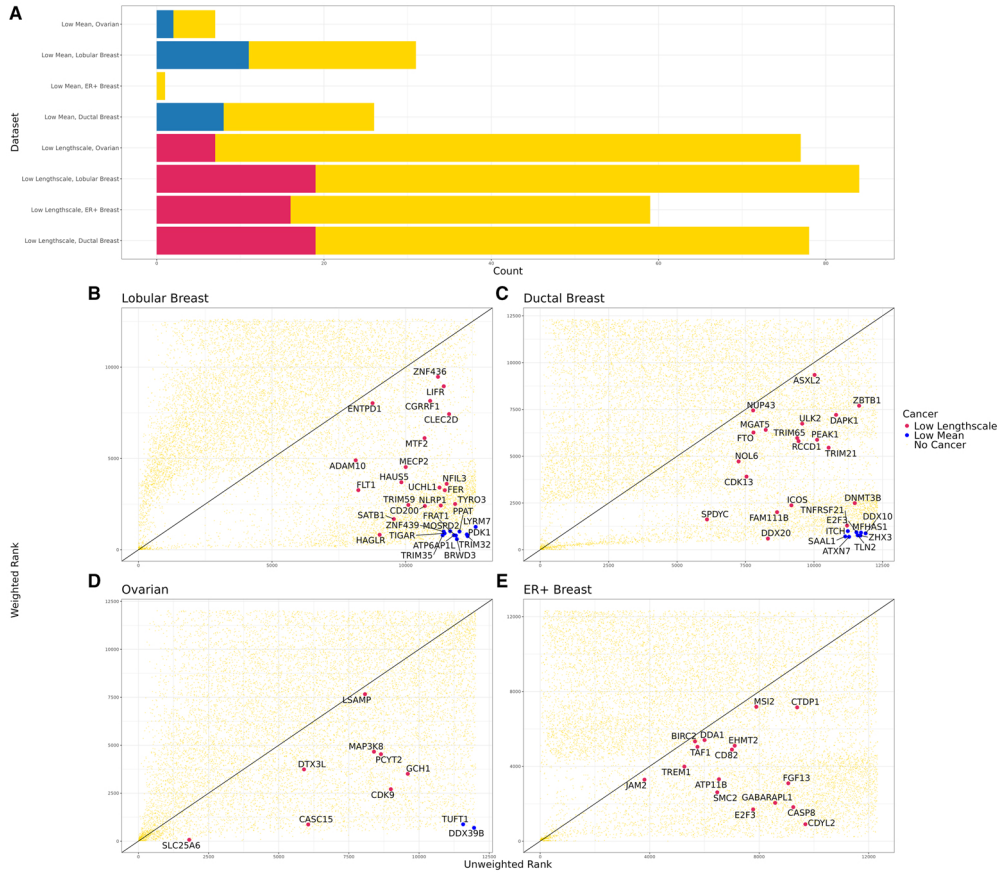

Fig. 5. *Spoon* helps to detect SVGs associated with cancer that are lowly expressed. We used four datasets to evaluate the detection of cancer-related genes: ER+ Breast cancer (Wu et al. 2021a), Ovarian cancer (Denisenko et al. 2024), Lobular Breast cancer (10x Genomics 2020), and Ductal Breast cancer (10x Genomics 2022). A) Each bar contains the intersection of the set of genes of interest with genes within the set associated with cancer. For the first four rows, we defined low mean genes as those with means less than the 25th percentile in the dataset. Within the set of low mean genes, we found genes that were in the lowest 10% of ranks before weighting and then increased to the highest 10% of ranks after weighting. This is the set of genes of interest. The intersection in blue is the number of low mean and higher ranked genes that were found to be associated with the cancer of the dataset. For the last four rows, we defined low lengthscale genes as those with lengthscales between 40 and 90. Within the set of low lengthscale genes, we found genes that were ranked higher after weighting. This is the set of genes of interest. The intersection in pink shows the number of low lengthscale genes that were ranked higher and found to be associated with the cancer type of the dataset. B)–E) Within each dataset, the unweighted rank of each gene is plotted on the *x*-axis and the weighted rank on the *y*-axis. The genes related to cancer are labeled and colored by low lengthscale or low mean.

genes were higher ranked after weighting, with 16 of these genes implicated in breast cancer. Full results are presented in Fig. 5 and gene lists are in Supplementary Materials.

To understand downstream impacts and biological significance, we performed gene set enrichment analysis (GSEA) (Wu et al. 2021b) on the unweighted and weighted SVG sets for the four cancer datasets. We used the Disease Gene Network (DisGeNet) (Pinero et al. 2015) resource of gene-disease associations for this enrichment analysis. We found that the weighted SVG sets show stronger biological relevance to the cancer of the dataset compared to the unweighted SVG sets

(Fig. S6). In particular, the Lobular Breast cancer unweighted gene set finds an Invasive Ductal Breast Carcinoma pathway enriched (which is a different subtype of Breast cancer), while the weighted gene set finds Breast Carcinoma and Malignant neoplasm of breast pathways enriched. The Ductal Breast cancer unweighted gene set finds no Breast cancer pathways enriched, while the weighted gene set finds general carcinogenesis and Noninfiltrating Intraductal Carcinoma pathways enriched. The Ovarian cancer unweighted gene set finds no Ovarian cancer pathways enriched, while the weighted gene set finds a general carcinogenesis pathway enriched. Finally, the ER+ Breast unweighted gene set finds no Breast cancer pathways enriched, while the weighted gene set finds general carcinogenesis and Mammary Neoplasms pathways enriched.

4. DISCUSSION

In our work, we systematically demonstrate the mean–variance and the mean-rank relationships exist in spatially resolved transcriptomics data. Furthermore, we show this is not limited to just one SVG detection method. If researchers fail to adjust for this bias in spatial transcriptomics data, this can lead to false positives and inaccurate rankings of SVGs due to the violation of the homoskedasticity assumption. Here, we show that our method *spoon* is able to correct for this bias. Specifically, our approach uses empirical Bayes techniques to generate weights for downstream analyses to remove the mean–variance relationship, leading to a more informative set of SVGs.

To clarify the methodological advancements, we show that the additional spatial context is required when calculating the empirical Bayes weights to adequately address the mean–variance relationship. We show that just replacing voom weights in our pipeline does not address the mean-rank relationship in SVG detection for the DLPFC dataset and a simulated dataset. We compare this with a panel which shows that the *spoon* weights do address the mean-rank relationship in these datasets (Fig. S7).

In a recent benchmark evaluation of SVG detection methods, Chen et al. (2024) noted a similar bias. NoVaTeST was recently proposed as a method to identify SVGs allowing noise variance to vary with spatial locations (Abrar et al. 2023). This method aims to identify genes that have location-dependent noise variance in SRT data, or genes that have statistically significant heteroskedasticity. This noise variation can be due to technical noise from the mean–variance relationship, variation due to sequencing processes, or underlying biological differences, making it difficult to parse out the mean–variance relationship. Additionally, further analysis of the genes detected by NoVaTest showed that some genes are likely affected by the mean–variance relationship, and the authors suggest using a strong variance-stabilizing transformation.

We recognize there are limitations to our project and aim to address these in future work. Primarily, simulation studies for spatial transcriptomics data are difficult to design and execute due to numerical instability and limitations of parameterization. There is no clear consensus on the definition of an SVG, so we chose to simulate overall SVGs, defined by Yan et al. (2025) as genes that exhibit non-random spatial patterns. To our knowledge, we are not aware of methods to simulate SVGs that include the mean–variance bias. In future work, we aim to refine spatial transcriptomics simulation study design to incorporate the mean–variance relationship and have more flexibility with various parameters, such as mean gene expression, degree of spatial variation, expression strength, and varying effect sizes in the same simulated dataset. We found that our method is most powerful for small lengthscale genes, and we hope to better understand medium and large lengthscale genes in future work as well. Finally, we will consider looking at other SRT platforms to see how the mean–variance relationship impacts other sequencing-based or imaging-based technologies.

In sum, we provide evidence for the mean–variance and mean-rank relationship in SRT data and show that our method *spoon* can mitigate these biases. We offer the software as an easily installable R/Bioconductor package that interfaces with `SpatialExperiment` to make this method broadly accessible to researchers.

ACKNOWLEDGMENTS

We thank members of the Hansen-Hicks lab group and our collaborators at the Lieber Institute for Brain Development for their input and feedback on this project. We also thank maintainers of the Joint High Performance Computing Exchange (JHPCE) computing cluster at Johns Hopkins Bloomberg School of Public Health for computing resources. We thank the reviewers for their thoughtful feedback and suggestions.

SUPPLEMENTARY MATERIAL

Supplementary material is available at *Biostatistics Journal* online. The Supplementary Materials include eight figures and eight tables referenced throughout the Main Manuscript. The figures expand on the results presented in the Main Manuscript, further illustrating the impacts of the mean–variance relationship and providing additional context for our simulations. The tables supplement Fig. 5 by listing citations for all of the genes associated with cancer in each subcategory of the four cancer datasets.

FUNDING

This work was supported by the National Institute on Drug Abuse of the National Institutes of Health [R01DA053581], the National Institute of Mental Health of the National Institutes of Health [R01MH126393], and the National Cancer Institute of the National Institutes of Health [R01CA237170]. This project was also supported by the Chan Zuckerberg Initiative DAF, an advised fund of Silicon Valley Community Foundation [CZF2019-002443]. All funding bodies had no role in the design of the study and collection, analysis, and interpretation of data and in writing the manuscript.

CONFLICT OF INTEREST

No competing interest is declared.

DATA AVAILABILITY

spoon is freely available for use as an R package available from Bioconductor at https://bioconductor.org/packages/spoon. The code to reproduce the analyses in this paper is available on GitHub at https://github.com/kinnaryshah/MeanVarBias. We used *spoon* version 1.1.3 and R version 4.4.1 for the analyses in this manuscript. A schematic of the *spoon* software is available in Fig. S8.

REFERENCES

10x Genomics. 2020. Human breast cancer: whole transcriptome analysis. https://www.10xgenomics.com/datasets/human-breast-cancer-whole-transcriptome-analysis-1-standard-1-2-0

10x Genomics. 2022. Human breast cancer: visium fresh frozen, whole transcriptome. https://www.10xgenomics.com/resources/datasets/human-breast-cancer-visium-fresh-frozen-whole-transcriptome-1-standard

Abrar MA, Kaykobad M, Rahman MS, Samee MAH. 2023. NoVaTeST: identifying genes with location-dependent noise variance in spatial transcriptomics data. Bioinformatics. 39:btad372.

Ahlmann-Eltze C, Huber W. 2023. Comparison of transformations for single-cell RNA-seq data. Nat Methods. 20:665–672.

Antolović V, Miermont A, Corrigan AM, Chubb JR. 2017. Generation of single-cell transcript variability by repression. Curr Biol. 27:1811–1817.e3.

Booeshaghi AS, Pachter L. 2021. Normalization of single-cell RNA-seq counts by. Bioinformatics. 37:2223–2224.

Brennecke P et al. 2013. Accounting for technical noise in single-cell RNA-seq experiments. Nat Methods. 10:1093–1095.

Buettner F et al. 2015. Computational analysis of cell-to-cell heterogeneity in single-cell RNA-sequencing data reveals hidden subpopulations of cells. Nat Biotechnol. 33:155–160.

Chang Y et al. 2024. Graph Fourier transform for spatial omics representation and analyses of complex organs. Nat Commun. 15:7467. https://doi.org/10.1038/s41467-024-51590-5
Chen C, Kim HJ, Yang P. 2024. Evaluating spatially variable gene detection methods for spatial transcriptomics data. Genome Biol. 25:18.
Chen KS et al. 2023. Regional interneuron transcriptional changes reveal pathologic markers of disease progression in a mouse model of Alzheimer's disease. https://doi.org/10.1101/2023.11.01.565165
Chen X et al. 2025. Benchmarking algorithms for spatially variable gene identification in spatial transcriptomics. Bioinformatics. 41:4. Doi:10.1093/bioinformatics/btaf131
Datta A, Banerjee S, Finley AO, Gelfand AE. 2016. Hierarchical nearest-neighbor gaussian process models for large geostatistical datasets. J Am Stat Assoc. 111:800–812.
Denisenko E et al. 2024. Spatial transcriptomics reveals discrete tumour microenvironments and autocrine loops within ovarian cancer subclones. Nat Commun. 15:2860.
Deshpande A et al. 2023. Uncovering the spatial landscape of molecular interactions within the tumor microenvironment through latent spaces. Cell Syst. 14:285–301.e4.
Dries R et al. 2021. Giotto: a toolbox for integrative analysis and visualization of spatial expression data. Genome Biol. 22:78.
Edsgärd D, Johnsson P, Sandberg R. 2018. Identification of spatial expression trends in single-cell gene expression data. Nat Methods. 15:339–342.
Eling N, Richard AC, Richardson S, Marioni JC, Vallejos CA. 2018. Correcting the mean–variance dependency for differential variability testing using single-cell RNA sequencing data. Cell Syst. 7:284–294.e12.
Finley AO et al. 2019. Efficient algorithms for Bayesian nearest neighbor Gaussian processes. J Comput Graph Stat. 28:401–414.
Garcia-Alonso L et al. 2022. Single-cell roadmap of human gonadal development. Nature. 607:540–547.
Hafemeister C, Satija R. 2019. Normalization and variance stabilization of single-cell RNA-seq data using regularized negative binomial regression. Genome Biol. 20:296.
Hao M, Hua K, Zhang X. 2021a. SOMDE: a scalable method for identifying spatially variable genes with self-organizing map. Bioinformatics. 37:4392–4398.
Hao Y et al. 2021b. Integrated analysis of multimodal single-cell data. Cell. 184:3573–3587.e29.
Jin Y et al. 2024. Advances in spatial transcriptomics and its applications in cancer research. Mol Cancer. 23:129.
Kats I, Vento-Tormo R, Stegle O. 2021. SpatialDE2: fast and localized variance component analysis of spatial transcriptomics. https://doi.org/10.1101/2021.10.27.466045
Law CW, Chen Y, Shi W, Smyth GK. 2014. Voom: precision weights unlock linear model analysis tools for RNA-seq read counts. Genome Biol. 15:R29.
Li Z et al. 2023. Benchmarking computational methods to identify spatially variable genes and peaks. https://doi.org/10.1101/2023.12.02.569717
Love MI, Huber W, Anders S. 2014. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. Genome Biol. 15:550.
Marx V. 2021. Method of the year: spatially resolved transcriptomics. Nat Methods. 18:9–14.
Maynard KR et al. 2021. Transcriptome-scale spatial gene expression in the human dorsolateral prefrontal cortex. Nat Neurosci. 24:425–436.
Moran PAP. 1950. Notes on continuous stochastic phenomena. Biometrika. 37:17–23.
Navarro JF et al. 2020. Spatial transcriptomics reveals genes associated with dysregulated mitochondrial functions and stress signaling in alzheimer disease. iScience. 23:101556.
Opławski M et al. 2022. Clinical and molecular evaluation of patients with ovarian cancer in the context of drug resistance to chemotherapy. Front Oncol. 12:954008.
Pinero J et al. 2015. DisGeNET: a discovery platform for the dynamical exploration of human diseases and their genes. Database. 2015:bav028.
Rao A, Barkley D, França GS, Yanai I. 2021. Exploring tissue architecture using spatial transcriptomics. Nature. 596:211–220.
Ritchie ME et al. 2015. Limma powers differential expression analyses for RNA-sequencing and microarray studies. Nucleic Acids Res. 43:e47.
Robinson MD, McCarthy DJ, Smyth GK. 2010. edgeR: a bioconductor package for differential expression analysis of digital gene expression data. Bioinformatics. 26:139–140.
Saha A, Datta A. 2018. BRISC: bootstrap for rapid inference on spatial covariances: rapid bootstrap for spatial covariances. Stat. 7:e184.
Seal S, Bitler BG, Ghosh D. 2023. SMASH: scalable method for analyzing spatial heterogeneity of genes in spatial transcriptomics data. PLOS Genet. 19:e1010983.
Sun S, Zhu J, Zhou X. 2020. Statistical analysis of spatial expression patterns for spatially resolved transcriptomic studies. Nat Methods. 17:193–200.

Svensson V, Teichmann SA, Stegle O. 2018. SpatialDE: identification of spatially variable genes. Nat Methods. 15:343–346.
Thompson JR et al. 2024. An integrated single-nucleus and spatial transcriptomics atlas reveals the molecular landscape of the human hippocampus. https://doi.org/10.1101/2024.04.26.590643
Totty M, Hicks SC, Guo B. 2024. SpotSweeper: spatially-aware quality control for spatial transcriptomics. https://doi.org/10.1101/2024.06.06.597765
Townes FW, Hicks SC, Aryee MJ, Irizarry RA. 2019. Feature selection and dimension reduction for single-cell RNA-Seq based on a multinomial model. Genome Biol. 20:295.
Tsagris M, Papadakis M. 2018. Taking R to its limits: 70+ tips. PeerJ Preprints. 6:e26605v1. https://doi.org/10.7287%2Fpeerj.preprints.26605v1
Vanrobaeys Y et al. 2023. Spatial transcriptomics reveals unique gene expression changes in different brain regions after sleep deprivation. Nat Commun. 14:7095.
Walker BL, Cang Z, Ren H, Bourgain-Chang E, Nie Q. 2022. Deciphering tissue structure and function using spatial transcriptomics. Commun Biol. 5:220–10.
Wang Y, Ma S, Ruzzo WL. 2020. Spatial modeling of prostate cancer metabolic gene expression reveals extensive heterogeneity and selective vulnerabilities. Sci Rep. 10:3490.
Weber LM et al. 2023a. The gene expression landscape of the human locus coeruleus revealed by single-nucleus and spatially-resolved transcriptomics. eLife. 12:RP84628. https://doi.org/10.7554/eLife.84628.3
Weber LM, Saha A, Datta A, Hansen KD, Hicks SC. 2023b. nnSVG for the scalable identification of spatially variable genes using nearest-neighbor Gaussian processes. Nat Commun. 14:4059.
Wu SZ et al. 2021a. A single-cell and spatially resolved atlas of human breast cancers. Nat Genet. 53:1334–1347.
Wu T et al. 2021b. clusterProfiler 4.0: a universal enrichment tool for interpreting omics data. Innovation. 2:100141.
Xu Z et al. 2020. Suppression of DDX39B sensitizes ovarian cancer cells to DNA-damaging chemotherapeutic agents via destabilizing BRCA1 mRNA. Oncogene. 39:7051–7062.
Yan G, Hua SH, Li J. 2025. Categorization of 34 computational methods to detect spatially variable genes from spatially resolved transcriptomics data. Nat Commun. 16:1141. https://doi.org/10.1038/s41467-025-56080-w
Yang K, Tu J, Chen T. 2019. Homoscedasticity: an overlooked critical assumption for linear regression. Gen Psychiatr. 32:e100148.
Yuan X et al. 2024a. HEARTSVG: a fast and accurate method for identifying spatially variable genes in large-scale spatial transcriptomics. Nat Commun. 15:5700.
Yuan Z et al. 2024b. Benchmarking spatial clustering methods with spatially resolved transcriptomics data. Nat Methods. 21:712–722.
Zhu J, Sun S, Zhou X. 2021. SPARK-X: non-parametric modeling enables scalable and robust detection of spatial expression patterns for large spatial transcriptomic studies. Genome Biol. 22:184.