UCLA Statistics

Categorization of 31 computational methods to detect spatially variable genes (SVGs) from spatial transcriptomics data

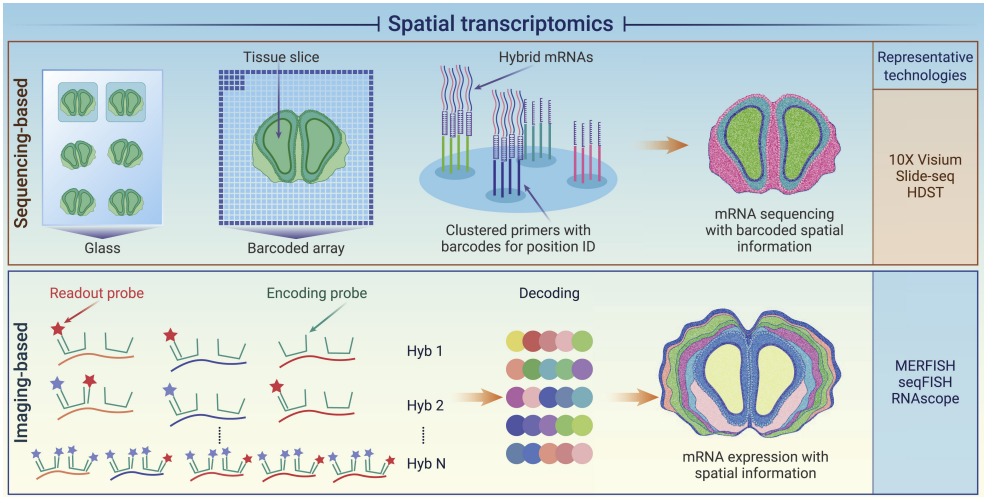
Jingyi Jessica Li

Department of Statistics and Data Science
University of California, Los Angeles

http://jsb.ucla.edu

joint work with Guan'ao Yan and Shuo Harper Hua

Spatially Transcriptomics Technologies



metacellular
$\sim 10^4$ genes

subcellular
$\sim 10^2$ genes

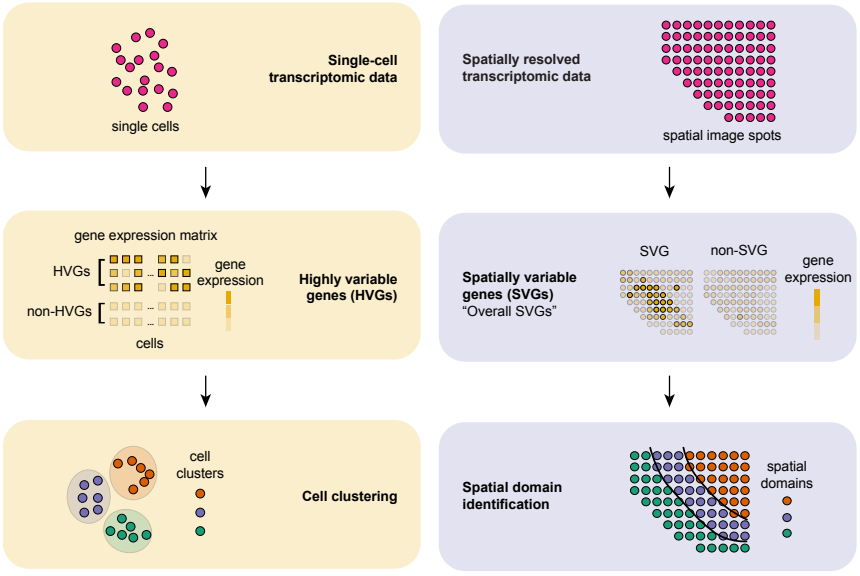
Lu Wen, Guoqiang Li, Tao Huang, et al., Single-cell technologies: From research to application, The Innovation, Volume 3, Issue 6, 2022, 100342.
https://doi.org/10.1016/j.xinn.2022.100342

Highly Variable Genes (HVGs) vs. Spatially Variable Genes (SVGs)

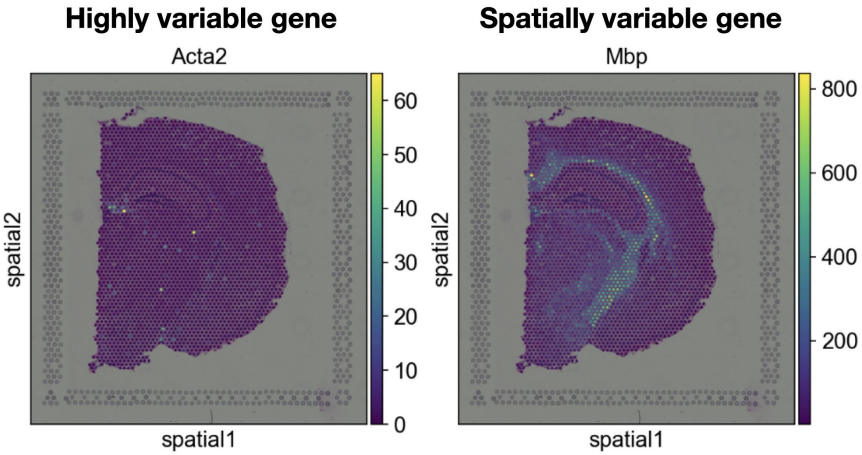
Informative features to screen for before linear dimension reduction and Euclidean distance calculation

- **HVG detection**
  - Used in single-cell transcriptomics data analysis
  - Identifies genes with high expression variability across single cells
  - Helps in clustering cells and identifying subpopulations
- **SVG detection**
  - Used in spatial transcriptomics data analysis
  - Identifies genes with high expression variability across spatial locations
  - Helps in identifying spatial patterns and regions with distinct molecular signatures

Highly Variable Genes (HVGs) vs. Spatially Variable Genes (SVGs)

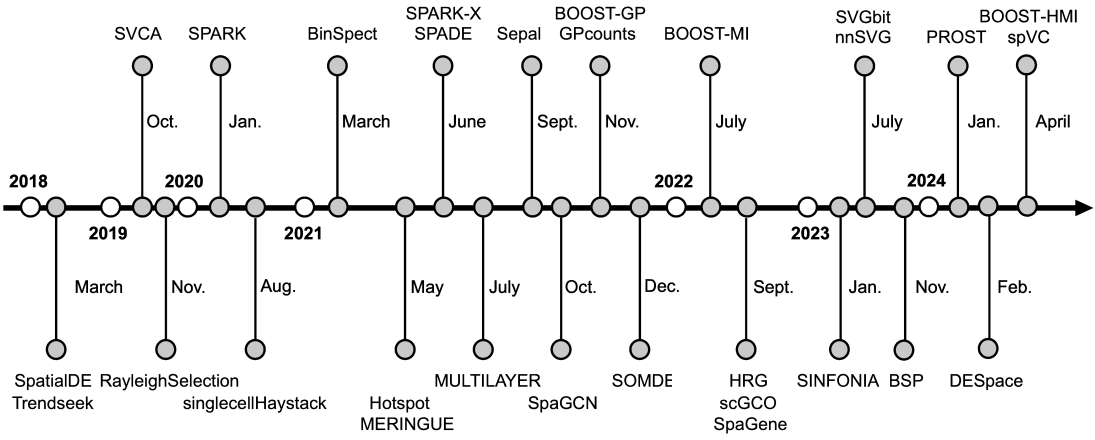


Highly Variable Genes (HVGs) vs. Spatially Variable Genes (SVGs)



https://www.sc-best-practices.org/_images/svg.jpeg

31 SVG Detection Methods



There is no consensus in SVG definitions

Existing Review and Benchmark Studies

Review

- Adhikari et al., *Computational and Structural Biotechnology Journal*, 2024 (19 methods)

Benchmark studies

- Charitakis et al., *Genome Biology*, 2023 (6 methods)
- Chen et al., *Genome Biology*, 2024 (7 methods)
- Li et al., *bioRxiv*, 2023 (14 methods)

Categorization of SVG definitions is not the focus

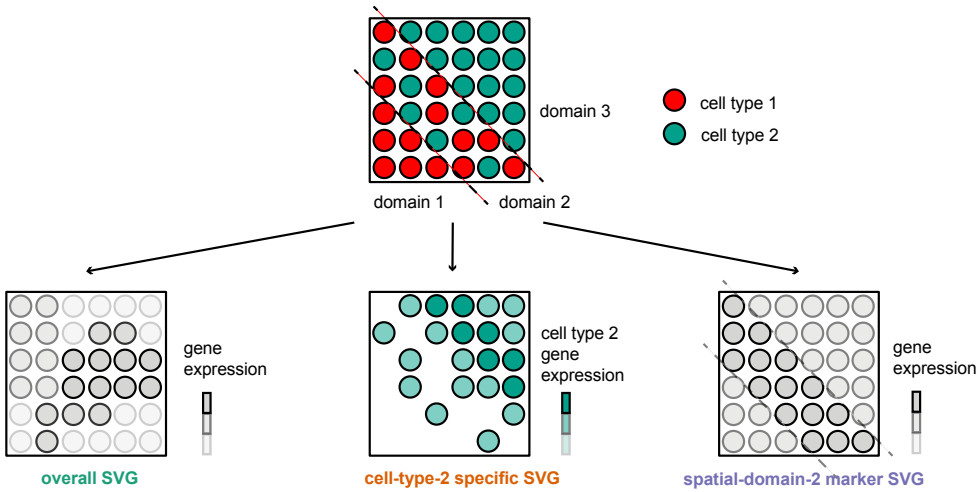
Proposal: Three Categories of SVGs

1. **Overall SVGs**:
   - Informative genes for downstream analysis (e.g., spatial domain identification)
2. **Cell-type-specific SVGs**:
   - Revealing spatial variation within a cell type $\Longrightarrow$ cell subpopulations or states
3. **Spatial-domain-marker SVGs**:
   - Marker genes to annotate and interpret spatial domains already detected

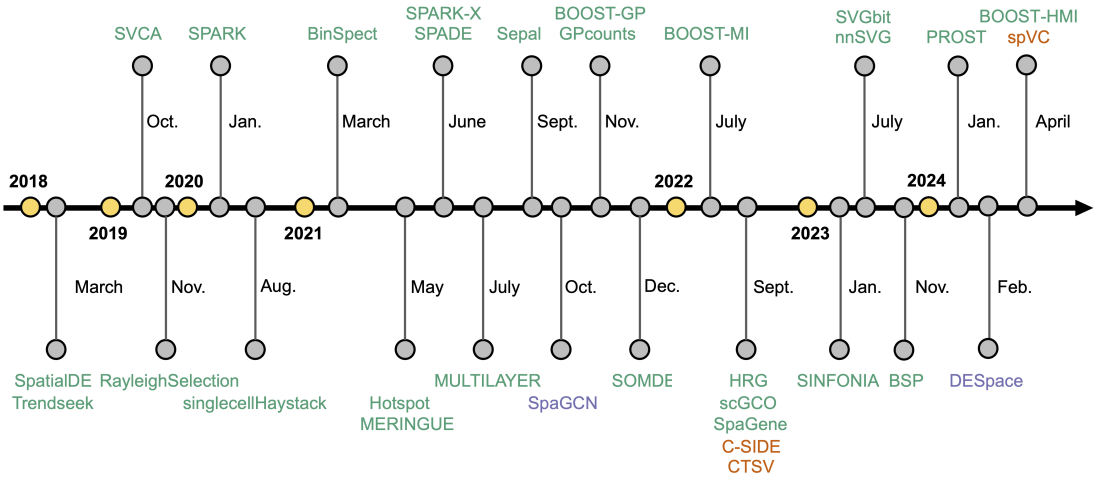
Relationships among the three categories depends on

- Detection methods' null and alternative hypotheses

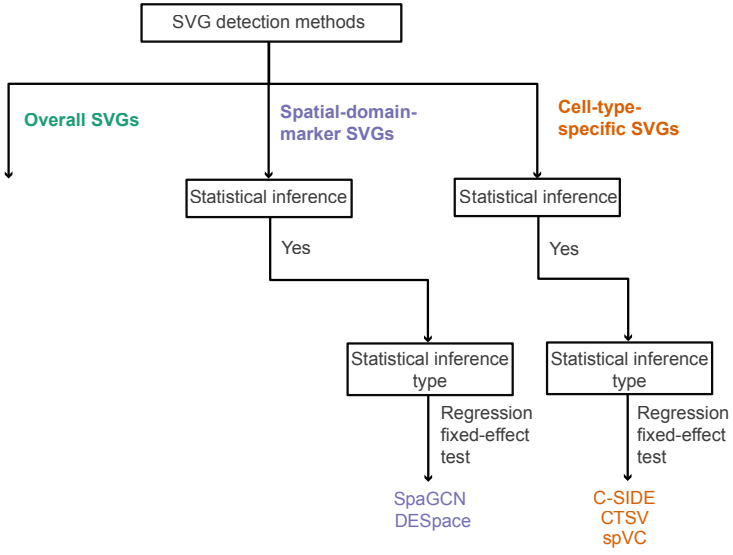
SVG Categories: Overall, Cell-type-specific, and Spatial-domain-marker SVGs



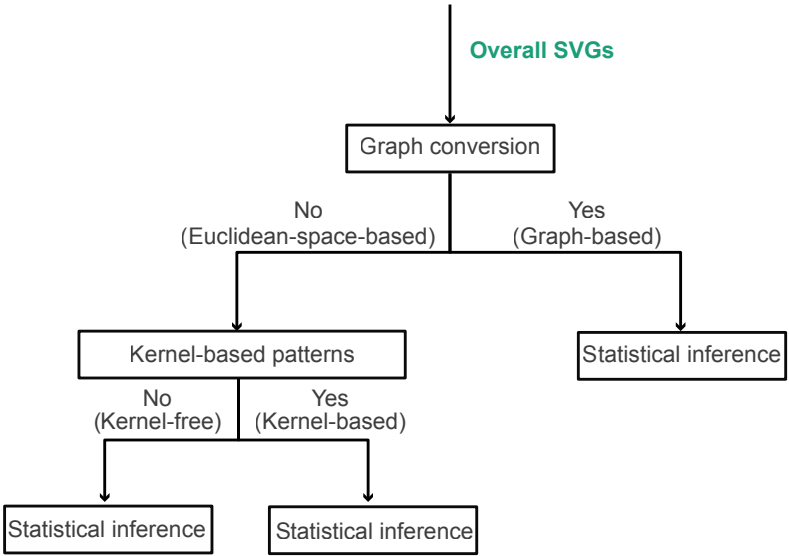
Categorization of 31 SVG Detection Methods



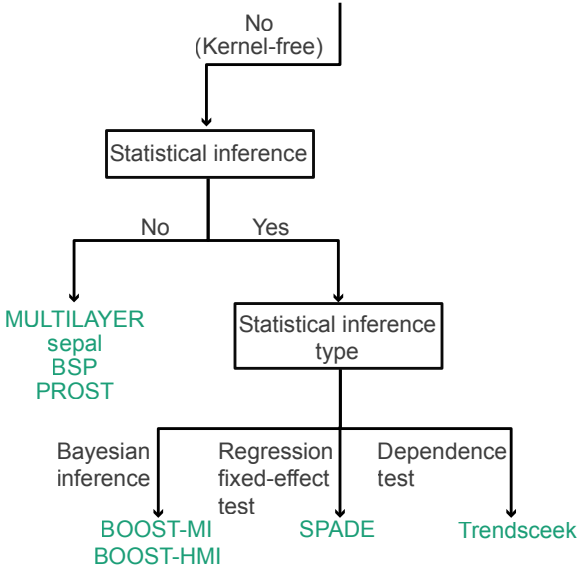
Hierarchy of 31 SVG Detection Methods (Part 1: Three Categories)



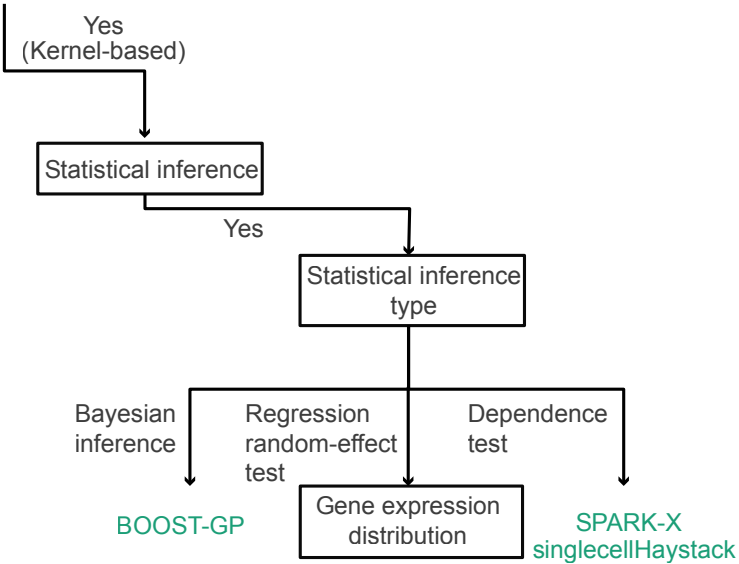
Hierarchy of 31 SVG Detection Methods (Part 2: Overall SVGs)



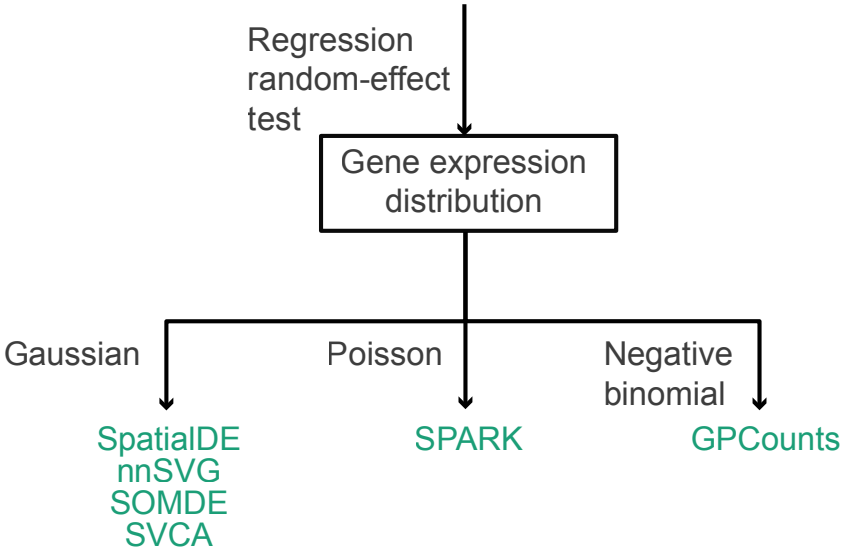
Hierarchy of 31 SVG Detection Methods (Part 3: Kernel-free Methods)



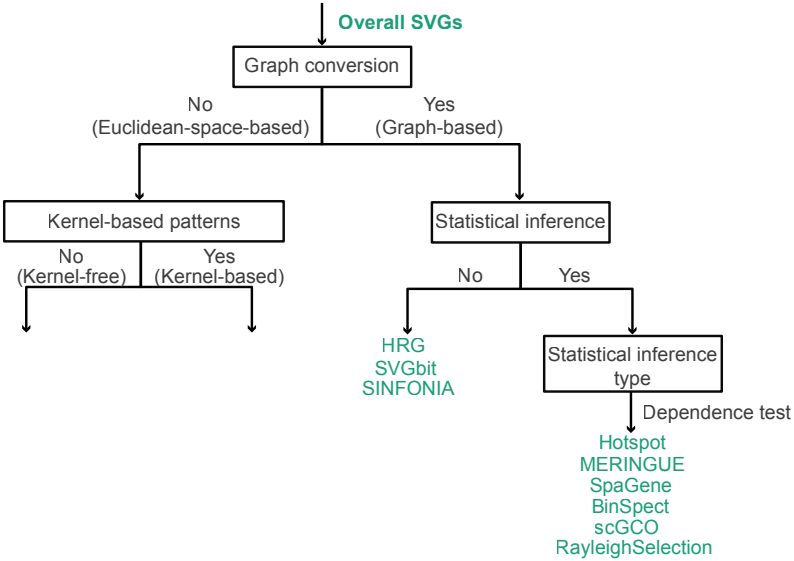
Hierarchy of 31 SVG Detection Methods (Part 4: Kernel-based Methods)



Hierarchy of 31 SVG Detection Methods (Part 5: Kernel-based Methods)



Hierarchy of 31 SVG Detection Methods (Part 6: Graph-based Methods)



Notations for SVG Detection (Per Gene)

For a given gene with expression levels measured at n spatial spots

Observed variables at spot $i = 1, \ldots, n$

- Gene expression level
  - $y_i \in \mathbb{R}$
  - $Y_i \in \mathbb{R}$: random variable notation
- 2D spatial location
  - $\mathbf{s}_i = (s_{i1}, s_{i2})^\top \in \mathbb{R}^2$
  - $\mathbf{s} = [\mathbf{s}_1, \ldots, \mathbf{s}_n]^\top \in \mathbb{R}^{n \times 2}$: spatial location matrix

Inferred variables at spot $i = 1, \ldots, n$

- Spatial-domain indicator vector
  - $\mathbf{d}_i = (d_{i1}, \ldots, d_{iL})^\top \in \{0, 1\}^L$, with $\sum_{l=1}^{L} d_{il} = 1$
- Cell-type proportion vector
  - $\mathbf{c}_i = (c_{i1}, \ldots, c_{iK})^\top \in [0, 1]^K$, with $\sum_{k=1}^{K} c_{ik} = 1$

Hypothesis Tests Used for SVG Detection

Among the 31 SVG detection methods, 21 use frequentist inference to detect SVGs:

- Define a test statistic
- Derive the test statistic's null distribution
- Convert the test statistic value to a p-value

Types of null hypotheses:

- **Dependence tests:** a gene's expression level is independent of spatial location
- **Regression-based tests:** spatial location has no "effect" on a gene's expression level
  - **Fixed-effect tests**
  - **Random-effect tests (variance component tests)**

Dependence Tests

Null hypothesis:

$$H_0 : Y \perp \mathbf{S}$$

Assume that $(y_1, \mathbf{s}_1), \ldots, (y_n, \mathbf{s}_n)$ are independently sampled from the distribution of $(Y, \mathbf{S})$

If H_0 is rejected, the gene is detected as an overall SVG

Nine methods adopt the dependence test formulation:

- **Conventional test statistics** (with theoretical null distribution):
  SPARK-X, Hotspot, MERINGUE, BinSpect, scGCO
- **Unconventional test statistics** (with permutation-based null distribution):
  Trendsceek, singlecellHaystack, RayleighSelection, SpaGene

SPARK-X (Zhu et al., Genome Biology, 2021)

SPARK-X: a non-parametric test that compares two $n \times n$ spot similarity matrices:

- Matrix 1 based on the gene's expression levels at the n spots
- Matrix 2 based on the kernel-transformed spatial locations of the n spots

SPARK-X (Zhu et al., Genome Biology, 2021)

SPARK-X: a non-parametric test that compares two $n \times n$ spot similarity matrices:

- Matrix 1 based on the gene's expression levels at the n spots
- Matrix 2 based on the kernel-transformed spatial locations of the n spots

To detect diverse spatial patterns, SPARK-X transforms the spatial locations $\mathbf{s}_i = (s_{i1}, s_{i2})$, $i = 1, \ldots, n$, using two kernel-based functions:

- Gaussian transformation $s'_{il} = \exp\left(\frac{-s_{il}^2}{2\sigma_l^2}\right)$, $l = 1, 2$, to detect clustered or focal patterns
- Cosine transformation $s'_{il} = \cos\left(\frac{2\pi s_{il}}{\phi_l}\right)$, $l = 1, 2$, to detect periodic patterns

where σ_1, σ_2, ϕ_1, and ϕ_2 are tuning parameters

SPARK-X (Zhu et al., Genome Biology, 2021)

SPARK-X: a non-parametric test that compares two $n \times n$ spot similarity matrices:

- Matrix 1 based on the gene's expression levels at the n spots
- Matrix 2 based on the kernel-transformed spatial locations of the n spots

To detect diverse spatial patterns, SPARK-X transforms the spatial locations $\mathbf{s}_i = (s_{i1}, s_{i2})$, $i = 1, \ldots, n$, using two kernel-based functions:

- Gaussian transformation $s'_{il} = \exp\left(\frac{-s_{il}^2}{2\sigma_l^2}\right)$, $l = 1, 2$, to detect clustered or focal patterns
- Cosine transformation $s'_{il} = \cos\left(\frac{2\pi s_{il}}{\phi_l}\right)$, $l = 1, 2$, to detect periodic patterns

where σ_1, σ_2, ϕ_1, and ϕ_2 are tuning parameters

Test statistic: Pearson correlation of the two matrices

Theoretical null: mixture chi-square distribution

singlecellHaystack (Vandenbon and Diez, Nature Communications, 2020)

singlecellHaystack: a unconventional test involves two pre-processing steps:

- Binarize the gene's expression levels at spots into **two states**: detected and undetected
- Divide the 2D Euclidean space into **grid points** as coarse spatial coordinates

singlecellHaystack (Vandenbon and Diez, Nature Communications, 2020)

singlecellHaystack: a unconventional test involves two pre-processing steps:

- Binarize the gene's expression levels at spots into **two states**: detected and undetected
- Divide the 2D Euclidean space into **grid points** as coarse spatial coordinates

singlecellHaystack uses a 2D independent Gaussian kernel, assuming independence of the two dimensions of the Euclidean space, to define **three distributions** of grid points:

- A reference distribution based on all grid points
- A conditional distribution based on grid points in the detected state
- Another conditional distribution based on grid points in the undetected state

singlecellHaystack (Vandenbon and Diez, Nature Communications, 2020)

singlecellHaystack: a unconventional test involves two pre-processing steps:

- Binarize the gene's expression levels at spots into **two states**: detected and undetected
- Divide the 2D Euclidean space into **grid points** as coarse spatial coordinates

singlecellHaystack uses a 2D independent Gaussian kernel, assuming independence of the two dimensions of the Euclidean space, to define **three distributions** of grid points:

- A reference distribution based on all grid points
- A conditional distribution based on grid points in the detected state
- Another conditional distribution based on grid points in the undetected state

Test statistic: sum of Kullback-Leibler divergences of the two conditional distributions from the reference distribution

Permutation null

Regression-based Tests

Two types: **fixed-effect tests** and **random-effect tests**

Linear mixed-effect model (LMM) for a given gene:

$$Y_i = \beta_0 + \mathbf{x}_i^\top \boldsymbol{\beta} + \mathbf{z}_i^\top \boldsymbol{\gamma} + \epsilon_i$$

- Y_i: a gene's expression level at spot i (response variable)
- β_0: (fixed) intercept
- $\mathbf{x}_i \in \mathbb{R}^p$: fixed-effect covariates of spot i
- $\boldsymbol{\beta} \in \mathbb{R}^p$: fixed effects

Regression-based Tests

Two types: **fixed-effect tests** and **random-effect tests**

Linear mixed-effect model (LMM) for a given gene:

$$Y_i = \beta_0 + \mathbf{x}_i^\top \boldsymbol{\beta} + \mathbf{z}_i^\top \boldsymbol{\gamma} + \epsilon_i$$

- Y_i: a gene's expression level at spot i (response variable)
- β_0: (fixed) intercept
- $\mathbf{x}_i \in \mathbb{R}^p$: fixed-effect covariates of spot i
- $\boldsymbol{\beta} \in \mathbb{R}^p$: fixed effects
- $\mathbf{z}_i \in \mathbb{R}^q$: random-effect covariates of spot i
- $\boldsymbol{\gamma} \in \mathbb{R}^q$: random effects with zero means $\mathbb{E}[\boldsymbol{\gamma}] = 0$ and covariance matrix

$$\mathrm{Cov}(\boldsymbol{\gamma}) \in \mathbb{R}^{q \times q}$$

- ϵ_i: independent random error at spot i with $\mathbb{E}[\epsilon_i] = 0$
- $\boldsymbol{\gamma} \perp \boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_n)^\top$

Fixed-effect Tests

$$Y_i = \beta_0 + \mathbf{x}_i^\top \boldsymbol{\beta} + \mathbf{z}_i^\top \boldsymbol{\gamma} + \epsilon_i$$

Fixed-effect tests examine whether $\mathbf{x}_i$ contribute to $\mathbb{E}[Y_i]$

If $\mathbf{x}_i$ makes no contribution, then $\mathbb{E}[Y_i|\mathbf{x}_i] = \mathbb{E}[Y_i]$

Fixed-effect Tests

$$Y_i = \beta_0 + \mathbf{x}_i^\top \boldsymbol{\beta} + \mathbf{z}_i^\top \boldsymbol{\gamma} + \epsilon_i$$

Fixed-effect tests examine whether $\mathbf{x}_i$ contribute to $\mathbb{E}[Y_i]$

If $\mathbf{x}_i$ makes no contribution, then $\mathbb{E}[Y_i|\mathbf{x}_i] = \mathbb{E}[Y_i]$

Null hypothesis

$$H_0 : \boldsymbol{\beta} = \mathbf{0}$$

implies $\mathbb{E}[Y_i|\mathbf{x}_i] = \mathbb{E}[Y_i]$, $i = 1, \ldots, n$

Random-effect Tests

$$Y_i = \beta_0 + \mathbf{x}_i^\top \boldsymbol{\beta} + \mathbf{z}_i^\top \boldsymbol{\gamma} + \epsilon_i$$

Random-effect tests examine whether $\mathbf{z}_i$ contribute to $\mathrm{Var}(Y_i)$:

$$\mathrm{Var}(Y_i) = \mathrm{Var}(\mathbb{E}[Y_i|\mathbf{z}_i]) + \mathbb{E}[\mathrm{Var}(Y_i|\mathbf{z}_i)] = \mathbf{z}_i^\top \mathrm{Cov}(\boldsymbol{\gamma})\mathbf{z}_i + \mathrm{Var}(\epsilon_i)$$

If $\mathbf{z}_i$ makes no contribution, then $\mathrm{Var}(\mathbb{E}[Y_i|\mathbf{z}_i]) = 0$

Random-effect Tests

$$Y_i = \beta_0 + \mathbf{x}_i^\top \boldsymbol{\beta} + \mathbf{z}_i^\top \boldsymbol{\gamma} + \epsilon_i$$

Random-effect tests examine whether $\mathbf{z}_i$ contribute to $\mathrm{Var}(Y_i)$:

$$\mathrm{Var}(Y_i) = \mathrm{Var}(\mathbb{E}[Y_i|\mathbf{z}_i]) + \mathbb{E}[\mathrm{Var}(Y_i|\mathbf{z}_i)] = \mathbf{z}_i^\top \mathrm{Cov}(\boldsymbol{\gamma})\mathbf{z}_i + \mathrm{Var}(\epsilon_i)$$

If $\mathbf{z}_i$ makes no contribution, then $\mathrm{Var}(\mathbb{E}[Y_i|\mathbf{z}_i]) = 0$

Null hypothesis

$$H_0 : \mathrm{Cov}(\boldsymbol{\gamma}) = \mathbf{0}$$

implies $\mathrm{Var}(\mathbb{E}[Y_i|\mathbf{z}_i]) = 0$, $i = 1, \ldots, n$

Generalization of LMM

Assume $\epsilon_i \overset{\text{iid}}{\sim} N(0, \sigma^2)$ and $\boldsymbol{\gamma} \perp \boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_n)^\top$

$$Y_i = \beta_0 + \mathbf{x}_i^\top \boldsymbol{\beta} + \mathbf{z}_i^\top \boldsymbol{\gamma} + \epsilon_i \quad \Longleftrightarrow \quad \begin{cases} Y_i \mid \mu_i \overset{\text{ind}}{\sim} N(\mu_i, \sigma^2) \\ \mu_i = \beta_0 + \mathbf{x}_i^\top \boldsymbol{\beta} + \mathbf{z}_i^\top \boldsymbol{\gamma} \end{cases}$$

Generalization of LMM

Assume $\epsilon_i \overset{\text{iid}}{\sim} N(0, \sigma^2)$ and $\boldsymbol{\gamma} \perp \boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_n)^\top$

$$Y_i = \beta_0 + \mathbf{x}_i^\top \boldsymbol{\beta} + \mathbf{z}_i^\top \boldsymbol{\gamma} + \epsilon_i \quad \Longleftrightarrow \quad \begin{cases} Y_i \mid \mu_i \overset{\text{ind}}{\sim} N(\mu_i, \sigma^2) \\ \mu_i = \beta_0 + \mathbf{x}_i^\top \boldsymbol{\beta} + \mathbf{z}_i^\top \boldsymbol{\gamma} \end{cases}$$

Generalized LMM (GLMM): The distribution of Y_i can be non-Gaussian

$$\text{e.g.,} \quad \begin{cases} Y_i \mid \mu_i \overset{\text{ind}}{\sim} \text{Poisson}(\mu_i) \\ \log(\mu_i) = \beta_0 + \mathbf{x}_i^\top \boldsymbol{\beta} + \mathbf{z}_i^\top \boldsymbol{\gamma} \end{cases}$$

Generalization of LMM

Assume $\epsilon_i \overset{\text{iid}}{\sim} N(0, \sigma^2)$ and $\boldsymbol{\gamma} \perp \boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_n)^\top$

$$Y_i = \beta_0 + \mathbf{x}_i^\top \boldsymbol{\beta} + \mathbf{z}_i^\top \boldsymbol{\gamma} + \epsilon_i \quad \Longleftrightarrow \quad \begin{cases} Y_i \mid \mu_i \overset{\text{ind}}{\sim} N(\mu_i, \sigma^2) \\ \mu_i = \beta_0 + \mathbf{x}_i^\top \boldsymbol{\beta} + \mathbf{z}_i^\top \boldsymbol{\gamma} \end{cases}$$

Generalized LMM (GLMM): The distribution of Y_i can be non-Gaussian

$$\text{e.g.,} \quad \begin{cases} Y_i \mid \mu_i \overset{\text{ind}}{\sim} \text{Poisson}(\mu_i) \\ \log(\mu_i) = \beta_0 + \mathbf{x}_i^\top \boldsymbol{\beta} + \mathbf{z}_i^\top \boldsymbol{\gamma} \end{cases}$$

Generalized non-parametric mixed-effect model:

The effects of $\mathbf{x}_i$ is modeled as non-parametric:

$$\text{e.g.,} \quad \log(\mu_i) = \beta_0 + f(\mathbf{x}_i) + \mathbf{z}_i^\top \boldsymbol{\gamma}$$

Generalization of LMM

Assume $\epsilon_i \overset{\text{iid}}{\sim} N(0, \sigma^2)$ and $\boldsymbol{\gamma} \perp \boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_n)^\top$

$$Y_i = \beta_0 + \mathbf{x}_i^\top \boldsymbol{\beta} + \mathbf{z}_i^\top \boldsymbol{\gamma} + \epsilon_i \quad \Longleftrightarrow \quad \begin{cases} Y_i \mid \mu_i \overset{\text{ind}}{\sim} N(\mu_i, \sigma^2) \\ \mu_i = \beta_0 + \mathbf{x}_i^\top \boldsymbol{\beta} + \mathbf{z}_i^\top \boldsymbol{\gamma} \end{cases}$$

Generalized LMM (GLMM): The distribution of Y_i can be non-Gaussian

$$\text{e.g.,} \quad \begin{cases} Y_i \mid \mu_i \overset{\text{ind}}{\sim} \text{Poisson}(\mu_i) \\ \log(\mu_i) = \beta_0 + \mathbf{x}_i^\top \boldsymbol{\beta} + \mathbf{z}_i^\top \boldsymbol{\gamma} \end{cases}$$

Generalized non-parametric mixed-effect model:

The effects of $\mathbf{x}_i$ is modeled as non-parametric:

$$\text{e.g.,} \quad \log(\mu_i) = \beta_0 + f(\mathbf{x}_i) + \mathbf{z}_i^\top \boldsymbol{\gamma}$$

Q: Is spatial location $\mathbf{s}_i$ modeled as $\mathbf{x}_i$ or $\mathbf{z}_i$?

Fixed-effect Tests for SVG Detection

Six methods use regression fixed-effect tests, covering all three SVG categories:

- **Overall SVGs:** SPADE
  - $\mathbf{x}_i$ includes $\mathbf{s}_i$
- **Cell-type-specific SVGs:** C-SIDE, CTSV, and spCV
  - $\mathbf{x}_i$ includes $\mathbf{s}_i$ and $\mathbf{c}_i$ (cell-type proportion vector)
- **Spatial-domain-marker SVGs:** SpaGCN and DESpace
  - $\mathbf{x}_i$ includes $\mathbf{s}_i$ and $\mathbf{d}_i$ (spatial-domain indicator vector)

SPADE (Bae et al., Nucleic Acids Research, 2021)

SPADE: linear-model fixed-effect test that detects **overall SVGs**:

$$\mu_i = \beta_0 + \mathbf{x}_i(\mathbf{s})^\top \boldsymbol{\beta}$$

- $\mathbf{x}_i(\mathbf{s})$: principal components of 512 features from a pre-trained convolutional neural network applied to the n spots' spatial locations $\mathbf{s}$ in an H&E image

SPADE (Bae et al., Nucleic Acids Research, 2021)

SPADE: linear-model fixed-effect test that detects **overall SVGs**:

$$\mu_i = \beta_0 + \mathbf{x}_i(\mathbf{s})^\top \boldsymbol{\beta}$$

- $\mathbf{x}_i(\mathbf{s})$: principal components of 512 features from a pre-trained convolutional neural network applied to the n spots' spatial locations $\mathbf{s}$ in an H&E image

Null hypothesis:

$$H_0 : \boldsymbol{\beta} = \mathbf{0}$$

If H_0 is rejected, the gene is detected as an overall SVG

SPADE (Bae et al., Nucleic Acids Research, 2021)

SPADE: linear-model fixed-effect test that detects **overall SVGs**:

$$\mu_i = \beta_0 + \mathbf{x}_i(\mathbf{s})^\top \boldsymbol{\beta}$$

- $\mathbf{x}_i(\mathbf{s})$: principal components of 512 features from a pre-trained convolutional neural network applied to the n spots' spatial locations $\mathbf{s}$ in an H&E image

Null hypothesis:

$$H_0 : \boldsymbol{\beta} = \mathbf{0}$$

If H_0 is rejected, the gene is detected as an overall SVG

Test: R package `limma`
(Smyth, G. K., 2005 $\Rightarrow$ Ritchie et al., *Nucleic Acids Research*, 2015)

spVC (Yu and Li, Genome Biology, 2024)

spVC: fixed-effect test that detects **cell-type-specific SVGs**

Assume

$$Y_i \mid \mu_i \overset{\text{ind}}{\sim} \text{Poisson}(\mu_i)$$

Two-step procedure:

1. A reduced model without interactive effects between $\mathbf{c}_i$ and $\mathbf{s}_i$:

$$\log(\mu_i) = \beta_0 + \sum_{k=1}^{K} c_{ik}\beta_k + f_0(\mathbf{s}_i)$$

It tests two null hypotheses:

- $H_0 : \boldsymbol{\beta} = (\beta_1, \ldots, \beta_K)^\top = \mathbf{0}$ using the likelihood ratio test
- $H_0 : f_0(\cdot) = 0$ using the Wald test

If both null hypotheses are rejected, it proceeds to the second step

2. A full model with interactive effects between $\mathbf{c}_i$ and $\mathbf{s}_i$:

$$\log(\mu_i) = \beta_0 + \sum_{k=1}^{K} c_{ik}\beta_k + f_0(\mathbf{s}_i) + \sum_{k=1}^{K} c_{ik} f_k(\mathbf{s}_i)$$

It tests if any of the interactive effects $f_1(\cdot), \ldots, f_K(\cdot)$ are zero using the likelihood ratio test

2. A full model with interactive effects between $\mathbf{c}_i$ and $\mathbf{s}_i$:

$$\log(\mu_i) = \beta_0 + \sum_{k=1}^{K} c_{ik}\beta_k + f_0(\mathbf{s}_i) + \sum_{k=1}^{K} c_{ik} f_k(\mathbf{s}_i)$$

It tests if any of the interactive effects $f_1(\cdot), \ldots, f_K(\cdot)$ are zero using the likelihood ratio test

If

$$H_0 : f_k(\cdot) = 0$$

is rejected, the gene is detected as a SVG specific to cell type k

DESpace (Cai et al., Bioinformatics, 2024)

DESpace: fixed-effect test that detects **spatial-domain-marker SVGs**

Assume

$$Y_i \mid \mu_i \overset{\text{ind}}{\sim} \text{NegativeBinomial}(\mu_i, \phi)$$

$$\log(\mu_i) = \beta_0 + \sum_{l=1}^{L} d_{il}\beta_l$$

where β_l indicates the effect of spatial domain l

DESpace (Cai et al., Bioinformatics, 2024)

DESpace: fixed-effect test that detects **spatial-domain-marker SVGs**

Assume

$$Y_i \mid \mu_i \overset{\text{ind}}{\sim} \text{NegativeBinomial}(\mu_i, \phi)$$

$$\log(\mu_i) = \beta_0 + \sum_{l=1}^{L} d_{il}\beta_l$$

where β_l indicates the effect of spatial domain l

If

$$H_0 : \beta_l = 0$$

is rejected, the gene is detected as a marker SVG of spatial domain l

Random-effect Tests for SVG Detection

Six methods use regression random-effect tests to detect **overall SVGs**:

SpatialDE, nnSVG, SOMDE, SVCA, SPARK, and GPcounts

$$Y_i = \beta_0 + \mathbf{x}_i^\top \boldsymbol{\beta} + \mathbf{z}_i^\top \boldsymbol{\gamma}(\mathbf{s}) + \epsilon_i$$

Random-effect Tests for SVG Detection

Six methods use regression random-effect tests to detect **overall SVGs**:

SpatialDE, nnSVG, SOMDE, SVCA, SPARK, and GPcounts

$$Y_i = \beta_0 + \mathbf{x}_i^\top \boldsymbol{\beta} + \mathbf{z}_i^\top \boldsymbol{\gamma}(\mathbf{s}) + \epsilon_i$$

With n spots, $\mathbf{z}_i = (z_{i1}, \ldots, z_{in})^\top \in \{0,1\}^n$ is a binary indicator vector for spot i s.t.

$$z_{ii} = 1\,; \quad z_{ij} = 0 \text{ if } j \neq i$$

Random-effect Tests for SVG Detection

Six methods use regression random-effect tests to detect **overall SVGs**:

SpatialDE, nnSVG, SOMDE, SVCA, SPARK, and GPcounts

$$Y_i = \beta_0 + \mathbf{x}_i^\top \boldsymbol{\beta} + \mathbf{z}_i^\top \boldsymbol{\gamma}(\mathbf{s}) + \epsilon_i$$

With n spots, $\mathbf{z}_i = (z_{i1}, \ldots, z_{in})^\top \in \{0,1\}^n$ is a binary indicator vector for spot i s.t.

$$z_{ii} = 1\,; \quad z_{ij} = 0 \text{ if } j \neq i$$

Random-effect vector $\boldsymbol{\gamma}(\mathbf{s}) = (\gamma_1(\mathbf{s}_1), \ldots, \gamma_n(\mathbf{s}_n))^\top \in \mathbb{R}^n$ has

$\gamma_i(\mathbf{s}_i)$ indicating the random effect of $\mathbf{s}_i$

$\mathrm{Cov}(\boldsymbol{\gamma}(\mathbf{s}))$ is assumed to depend on the spatial proximity of $\mathbf{s}_1, \ldots, \mathbf{s}_n$ via a kernel

Random-effect Tests for SVG Detection

Six methods use regression random-effect tests to detect **overall SVGs**:

SpatialDE, nnSVG, SOMDE, SVCA, SPARK, and GPcounts

$$Y_i = \beta_0 + \mathbf{x}_i^\top \boldsymbol{\beta} + \mathbf{z}_i^\top \boldsymbol{\gamma}(\mathbf{s}) + \epsilon_i$$

With n spots, $\mathbf{z}_i = (z_{i1}, \ldots, z_{in})^\top \in \{0,1\}^n$ is a binary indicator vector for spot i s.t.

$$z_{ii} = 1\,; \quad z_{ij} = 0 \text{ if } j \neq i$$

Random-effect vector $\boldsymbol{\gamma}(\mathbf{s}) = (\gamma_1(\mathbf{s}_1), \ldots, \gamma_n(\mathbf{s}_n))^\top \in \mathbb{R}^n$ has

$\gamma_i(\mathbf{s}_i)$ indicating the random effect of $\mathbf{s}_i$

$\mathrm{Cov}(\boldsymbol{\gamma}(\mathbf{s}))$ is assumed to depend on the spatial proximity of $\mathbf{s}_1, \ldots, \mathbf{s}_n$ via a kernel

If

$$H_0 : \mathrm{Cov}(\boldsymbol{\gamma}(\mathbf{s})) = \mathbf{0}$$

is rejected, the gene is detected as an overall SVG

SpatialDE (Svensson et al., Nature Methods, 2018)

SpatialDE: a linear random-effect model:

$$Y_i = \beta_0 + \mathbf{z}_i^\top \boldsymbol{\gamma}(\mathbf{s}) + \epsilon_i$$

- The random errors $\epsilon_1, \ldots, \epsilon_n \overset{\text{iid}}{\sim} N(0, \delta)$
- The random effects $\boldsymbol{\gamma}(\mathbf{s}) \sim \text{MVN}(\mathbf{0}, \sigma_s^2 \cdot \mathbf{K}(\mathbf{s}))$
  The kernel matrix $\mathbf{K}(\mathbf{s}) = [K(\mathbf{s}_i, \mathbf{s}_j)]_{n \times n}$ is specified by a kernel function $K(\cdot, \cdot)$

This model is essentially a Gaussian process

If

$$H_0 : \sigma_s^2 = 0$$

is rejected, the gene is detected as an overall SVG

Discussion: Power vs. Specificity Trade-off

26 methods for detecting overall SVGs:

9 kernel-based methods vs. 17 other methods (kernel-free or graph-based)

Kernel-based methods have

- Higher specificity for targeted patterns
- Lower overall power for other patterns

Discussion: Challenges in Detecting Non-Global Expression Patterns

1. **Small regions of interests (ROIs)**
   - Spatial-domain-marker SVGs by first identifying ROIs as spatial domains (e.g., SpaGCN)
2. **Spatial-Domain-Specific SVGs**
   - Genes with spatial patterns in small ROIs but not marker genes
   - No existing methods
3. **Cell-Type-Specific SVGs**
   - Easily missed if cell types have small proportions
   - Existing methods' model goodness-of-fit
4. **Sharp Expression Changes**
   - Genes with sharp changes at tissue layer boundaries (e.g., Belayer)
   - Adding H&E image can help refine tissue boundaries

Discussion: Challenges in Detecting Non-Global Expression Patterns

1. **Small regions of interests (ROIs)**
   - Spatial-domain-marker SVGs by first identifying ROIs as spatial domains (e.g., SpaGCN)
2. **Spatial-Domain-Specific SVGs**
   - Genes with spatial patterns in small ROIs but not marker genes
   - No existing methods
3. **Cell-Type-Specific SVGs**
   - Easily missed if cell types have small proportions
   - Existing methods' model goodness-of-fit
4. **Sharp Expression Changes**
   - Genes with sharp changes at tissue layer boundaries (e.g., Belayer)
   - Adding H&E image can help refine tissue boundaries

Future direction: Incorporate knowledge on "interesting genes" to improve specificity

Discussion: Scalability

1. Calculate a summary statistic for each gene.
2. Convert the summary statistic to a p-value (frequentist methods only)

Summary Statistic Calulation (n: number of spatial spots)

- Gaussian process: $O(n^3)$ in SpatialDE and SPARK
- Nearest-neighbor Gaussian process approximation: $O(n)$ in nnSVG

p-value Conversion

- Fast if closed-form null distribution is available (conventional statistics)
- Computationally intensive if by permutation (unconventional statistics)

Improving Scalability

- Use approximation algorithms to speed up summary statistic calculation
- Reduce number of permutations in the p-value conversion step

Future Direction 1: Accommodating Technological Differences

Two Key Differences:

- **Spatial Resolution**
  - Imaging-based Technologies: Single-cell or subcellular resolution
  - Sequencing-based Technologies: Multicellular level, coarser resolution
- **Positional Randomness**
  - Structured grids (e.g., Spatial Transcriptomics, 10x Visium)
  - Unstructured spots (e.g., Slide-seq, MERFISH, SeqFISH)

Future Direction 1: Accommodating Technological Differences

Two Key Differences:

- **Spatial Resolution**
  - Imaging-based Technologies: Single-cell or subcellular resolution
  - Sequencing-based Technologies: Multicellular level, coarser resolution
- **Positional Randomness**
  - Structured grids (e.g., Spatial Transcriptomics, 10x Visium)
  - Unstructured spots (e.g., Slide-seq, MERFISH, SeqFISH)

Current Limitations:

- Most SVG detection methods lack consideration of these technological differences
- Lack of consensus on pre-processing and modeling SRT data

Future Direction 2: Enhancing Statistical Rigor and Method Benchmarking

Challenges:

- Double-dipping: Same data analyzed more than once, leading to confirmation bias
- Example: Spatial-domain-marker SVG detection

Strategies:

- Use *in silico* negative control data to remove spurious discoveries (e.g., ClusterDE)
- Develop fast visualization tools for interpreting top-detected SVGs

Method Benchmarking:

- Benchmarking requires well-annotated datasets with SVG ground truths
- Synthetic datasets and realistic simulators (e.g., SRTsim, scDesign3)
- No method is optimal in every aspect; benchmarking should be specific to data characteristics and align with biological questions

Preprint

Yan, G., Hua, S. H., & Li, J. J. (2024). Categorization of 31 computational methods to detect spatially variable genes from spatially resolved transcriptomics data. *arXiv*. https://arxiv.org/abs/2405.18779