

SpatialGEE Tutorial

1. Introduction

This tutorial introduces the **SpatialGEE** package for analyzing spatial transcriptomics data. It includes:

- An example dataset for testing.
- Descriptions and usage of core functions for visualization and statistical analysis.

2. Installation and Loading

To use the **SpatialGEE** package, install it and load it into your R session:

```
if(!requireNamespace("devtools", quietly = TRUE))  
  install.packages("devtools")  
devtools::install_github("yishan03/SpatialGEE", quiet = TRUE)  
library(SpatialGEE)
```

3. Example Dataset

The package includes an example dataset, `example_data`, based on breast cancer spatial transcriptomics data.

3.1. Dataset Description

- The dataset contains metadata and gene expression data for 3168 spots.
- Metadata columns include:
 - **Barcodes**: Spot barcodes.
 - **x, y**: Spatial coordinates.
 - **Pathology.Annotations**: Pathology labels ("Fibrous Tissue" and "Invasive Carcinoma").
- Gene expression data includes counts for three genes: `IGHG2`, `MALAT1`, and `MYH11`.

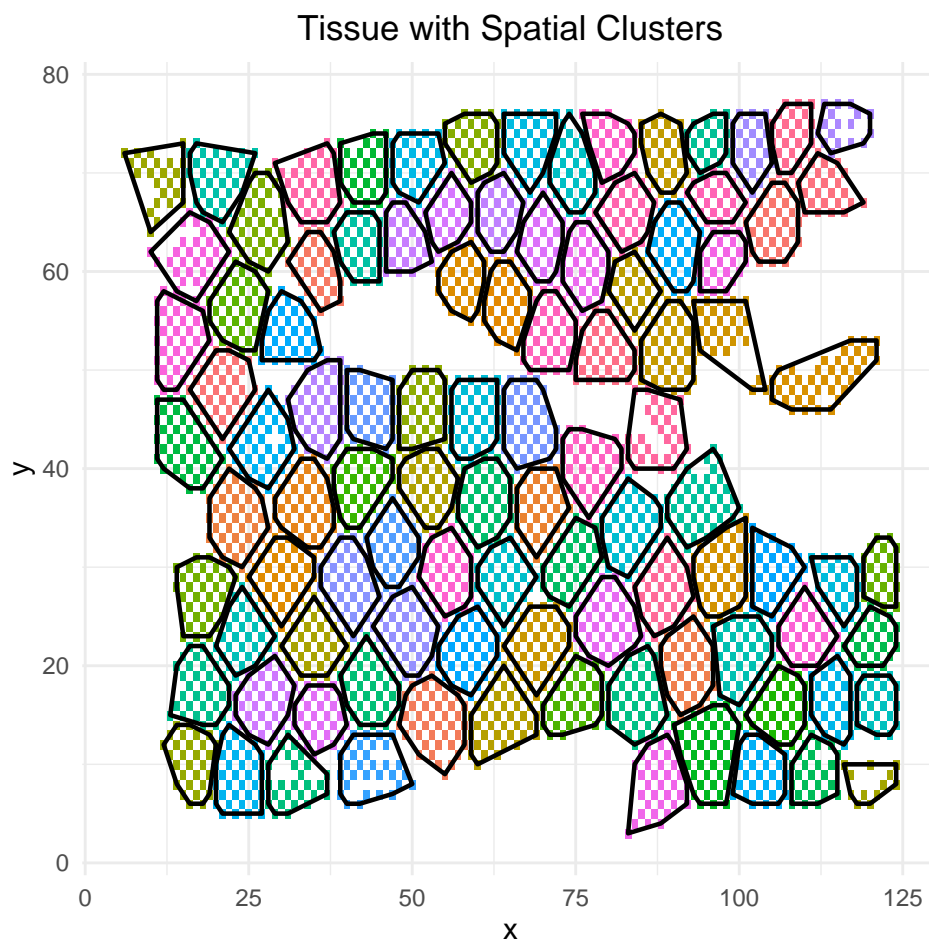
3.2. Dataset Example

```
data(example_data)
head(example_data)
#>      Barcodes  x  y Pathology.Annotations IGHG2 MALAT1 MYH11
#> 1 AAACACCAATAACTGC 20 60 Invasive Carcinoma 0 32 0
#> 2 AAACAGAGCGACTCCT 95 15 Fibrous Tissue 0 43 0
#> 3 AAACAGGGTCTATATT 14 48 Invasive Carcinoma 3 47 0
#> 4 AAACAGTGTTCCTGGG 44 74 Invasive Carcinoma 3 27 1
#> 5 AAACATTTCCTGGATT 98 62 Invasive Carcinoma 0 17 0
#> 6 AAACCTAAGCAGCCGG 84 66 Invasive Carcinoma 1 47 0
```

4. Visualizing Spatial Clusters

The `plot_spatial_clusters` function visualizes spatial transcriptomics data by coloring spots based on k-means clustering.

```
plot <- plot_spatial_clusters(example_data, k = 100)
print(plot)
```



5. Statistical Analysis with SpatialGEE

5.1. Robust Wald Test

The `run_gee_wald` function uses Generalized Estimating Equations (GEE) with the robust Wald test to compare gene expression between two pathology levels.

```
results_wald <- run_gee_wald(  
  data = example_data,  
  compare_levels = c("Fibrous Tissue", "Invasive Carcinoma"),  
  k = 100  
)  
  
results_wald  
#>      gene      p_value  
#> 1  IGHG2 0.7664071263  
#> 2 MALAT1 0.9759877387  
#> 3 MYH11 0.0002168661
```

5.2. Generalized Score Test (GST)

The `run_gee_gst` function performs GEE analysis using the Generalized Score Test (GST) for robust hypothesis testing.

```
results_gst <- run_gee_gst(  
  data = example_data,  
  compare_levels = c("Fibrous Tissue", "Invasive Carcinoma"),  
  k = 100  
)  
  
results_gst  
#>      gene      p_value  
#> 1  IGHG2 0.7708318562  
#> 2 MALAT1 0.9760063165  
#> 3 MYH11 0.0007719343
```

5.3. Independent GEE Analysis

The `run_indep_gee` function treats each spatial location as independent for GEE analysis.

```
results_indep <- run_indep_gee(  
  data = example_data,  
  compare_levels = c("Fibrous Tissue", "Invasive Carcinoma")  
)  
  
results_indep  
#>      gene      p_value  
#> 1  IGHG2 5.312246e-01  
#> 2 MALAT1 9.319693e-01  
#> 3 MYH11 2.544144e-05
```

5.4. Wilcoxon Rank-Sum Test

The `run_wilcoxon` function performs the Wilcoxon rank-sum test to compare gene expression levels between two pathology groups.

```
results_wilcoxon <- run_wilcoxon(  
  data = example_data,  
  compare_levels = c("Fibrous Tissue", "Invasive Carcinoma")  
)  
  
results_wilcoxon  
#>      gene      p_value  
#> 1  IGHG2 0.1884257442  
#> 2  MALAT1 0.5326497220  
#> 3  MYH11 0.0004190615
```