

# Package: oCELLoc (via r-universe)

May 22, 2026

**Title** Predicts Suitable Cell Types in Spatial Transcriptomics and scRNA-seq Data

**Version** 1.0.0

**Description** Picks the suitable cell types in spatial and scRNA-seq data using shrinkage methods. The package includes curated reference gene expression profiles for human and mouse cell types, facilitating immediate application to common spatial transcriptomics or scRNA datasets. Additionally, users can input custom reference data to support tissue- or experiment-specific analyses.

**License** MIT + file LICENSE

**URL** <https://doi.org/10.64898/2025.12.11.693812>

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.3.3

**Imports** glmnet, utils, stats, ggplot2, rlang, reshape2,

**Suggests** testthat (>= 3.0.0), knitr, rmarkdown

**LazyData** true

**Depends** R (>= 3.5.0)

**Config/pak/sysreqs** libicu-dev

**Repository** <https://afeefa-zainab.r-universe.dev>

**Date/Publication** 2025-12-17 06:21:45 UTC

**RemoteUrl** <https://github.com/afeefa-zainab/ocelloc>

**RemoteRef** HEAD

**RemoteSha** f4b22bb7e46a22377a636e8158733339f6a8df92

## Contents

human_ref . . . . .	2
mouse_ref . . . . .	2
oCELLoc . . . . .	3
predict_cell_types . . . . .	3

**Index**[7](#)


---

human_ref	<i>Human Cell Type Reference Data</i>
-----------	---------------------------------------

---

**Description**

A reference dataset containing gene expression profiles for various human cell types. Used as a reference for the `predict_cell_types` function to predict cell type proportions in spatial transcriptomics data.

**Usage**

```
human_ref
```

**Format**

A data frame with rows corresponding to cell types and columns corresponding to genes. The data frame has a 'cell\_type' column that identifies the cell type, and numerous gene columns with expression values.

**Source**

Generated from reference single-cell RNA sequencing datasets

---

mouse_ref	<i>Mouse Cell Type Reference Data</i>
-----------	---------------------------------------

---

**Description**

A reference dataset containing gene expression profiles for various mouse cell types. Used as a reference for the `predict_cell_types` function to predict cell type proportions in spatial transcriptomics data.

**Usage**

```
mouse_ref
```

**Format**

A data frame with rows corresponding to cell types and columns corresponding to genes. The data frame has a 'cell\_type' column that identifies the cell type, and numerous gene columns with expression values.

**Source**

Generated from reference single-cell RNA sequencing datasets

---

`oCELLoc`*oCELLoc: Spatial Transcriptomics Cell Type Prediction*

---

### Description

Predicts average cell type proportions for a spatial transcriptomics sample using Lasso regression on average spot expression. Applies specific lambda selection rules and normalizes output proportions.

---

`predict_cell_types`*Predict Average Cell Type Proportions for a Sample*

---

### Description

This function takes spatial transcriptomics data for a single sample (potentially across multiple spots), calculates the average expression, and predicts average cell type proportions using Lasso regression against a reference dataset. It applies an exponential transformation to input data, uses a specific rule for lambda selection (seeking 3-14 non-zero coefficients), filters coefficients, and normalizes the final proportions to sum to 1.

### Usage

```
predict_cell_types(  
  spatial_data,  
  reference,  
  sample_name = NULL,  
  nfolds = 5,  
  transform_input = TRUE,  
  normalize_reference = TRUE,  
  lambda_selection_rule = "auto",  
  alpha = 1,  
  lambda_min = 0.001,  
  lambda_max = 1,  
  lambda_n = 100,  
  min_nonzero = 3,  
  max_nonzero = 14,  
  keep_top_n = 14,  
  nonzero_threshold = 0.001,  
  generate_plots = TRUE,  
  verbose = TRUE  
)
```

**Arguments**

<code>spatial_data</code>	A data.frame or matrix containing spatial gene expression data. Genes should be in row names, and columns should represent spots/barcodes. Assumes expression values are log-transformed (e.g., $\log(\text{CPM}+1)$ or $\log(\text{TPM}+1)$ ).
<code>reference</code>	A data.frame or matrix containing reference expression data. Genes should be in row names, cell types should be in column names. Alternatively, a character string specifying a built-in reference ("human" or "mouse").
<code>sample_name</code>	Optional name for the sample (used in plot titles). If NULL, uses "Sample".
<code>nfolds</code>	Number of folds for cross-validation in <code>cv.glmnet</code> . (Default: 5)
<code>transform_input</code>	Logical, whether to apply $\exp(\text{data}) - 1$ transformation to the input spatial data. Set to FALSE if data is already in linear scale (e.g., counts, CPM). (Default: TRUE)
<code>normalize_reference</code>	Logical, whether to normalize each cell type in the reference to have the same total expression. (Default: TRUE)
<code>lambda_selection_rule</code>	Character, method for lambda selection. Options are: "auto" (use <code>glmnet</code> 's default lambda sequence) or "custom" (use custom lambda range). (Default: "auto")
<code>alpha</code>	The elasticnet mixing parameter, where $\alpha=1$ is the lasso (default) and $\alpha=0$ is ridge.
<code>lambda_min</code>	Minimum lambda value for custom lambda sequence (only used when <code>lambda_selection_rule="custom"</code> ). (Default: 0.001)
<code>lambda_max</code>	Maximum lambda value for custom lambda sequence (only used when <code>lambda_selection_rule="custom"</code> ). (Default: 1.0)
<code>lambda_n</code>	Number of lambda values in custom sequence (only used when <code>lambda_selection_rule="custom"</code> ). (Default: 100)
<code>min_nonzero</code>	Minimum number of desired non-zero coefficients for lambda selection. (Default: 3)
<code>max_nonzero</code>	Maximum number of desired non-zero coefficients for lambda selection. (Default: 14)
<code>keep_top_n</code>	Maximum number of positive coefficients to retain after filtering. If more coefficients are positive, only the top <code>keep_top_n</code> are kept. Set to Inf to disable. (Default: 14)
<code>nonzero_threshold</code>	Threshold below which coefficients are considered zero during lambda selection and final filtering. (Default: $1e-3$ )
<code>generate_plots</code>	Logical, whether to generate CV and coefficient path plots. (Default: TRUE)
<code>verbose</code>	Logical, whether to print progress messages. (Default: TRUE)

**Value**

A list containing:

- proportions: Data frame with columns 'Cell\_Type' and 'Proportion'
- nonzero\_celltypes: Vector of cell type names with non-zero proportions
- selected\_lambda: The lambda value selected by the algorithm
- selection\_rule: Whether lambda was selected by "3-14\_rule\_glmnet", "3-14\_rule\_custom", or "fallback"
- common\_genes: Vector of genes used in the analysis
- cv\_plot: Function to generate cross-validation ggplot (if generate\_plots=TRUE)
- coef\_plot: Function to generate coefficient path ggplot (if generate\_plots=TRUE)

Returns NULL if processing fails.

**Examples**

```
# Example 1: Using built-in human reference with glmnet lambda sequence
# Load example human average expression data
load(system.file("extdata", "human_avg_expression.rda", package = "oCELLoc"))

# Run with built-in human reference and glmnet lambda sequence
results_human <- predict_cell_types(
  spatial_data = human_avg_expression,
  reference = "human",
  sample_name = "Human_Example",
  lambda_selection_rule = "auto"
)

# View top results
print(head(results_human$proportions, 10))
print(results_human$nonzero_celltypes)

# Example 2: Using built-in mouse reference with custom lambda sequence
# Load example mouse average expression data
load(system.file("extdata", "mouse_avg_expression.rda", package = "oCELLoc"))

# Run with built-in mouse reference and custom lambda sequence
results_mouse <- predict_cell_types(
  spatial_data = mouse_avg_expression,
  reference = "mouse",
  sample_name = "Mouse_Example",
  lambda_selection_rule = "custom",
  lambda_min = 0.001,
  lambda_max = 0.5,
  lambda_n = 50
)

# View top results
print(head(results_mouse$proportions, 10))
```

```
print(results_mouse$nonzero_celltypes)
```

# Index

## \* datasets

human\_ref, [2](#)

mouse\_ref, [2](#)

human\_ref, [2](#)

mouse\_ref, [2](#)

oCELLoc, [3](#)

predict\_cell\_types, [3](#)