

# Package: scISR (via r-universe)

October 18, 2024

**Type** Package

**Title** Single-Cell Imputation using Subspace Regression

**Version** 0.1.1

**Maintainer** Duc Tran <duct@nevada.unr.edu>

**Description** Provides an imputation pipeline for single-cell RNA sequencing data. The 'scISR' method uses a hypothesis-testing technique to identify zero-valued entries that are most likely affected by dropout events and estimates the dropout values using a subspace regression model (Tran et.al. (2022) <[DOI:10.1038/s41598-022-06500-4](https://doi.org/10.1038/s41598-022-06500-4)>).

**License** LGPL

**Depends** R (>= 3.4)

**Imports** cluster, entropy, stats, utils, parallel, irlba, PINSPlus, matrixStats, markdown

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.1.1

**NeedsCompilation** no

**Suggests** testthat, knitr, mclust

**VignetteBuilder** knitr

**URL** <https://github.com/duct317/scISR>

**BugReports** <https://github.com/duct317/scISR/issues>

**Repository** <https://duct317.r-universe.dev>

**RemoteUrl** <https://github.com/duct317/scisr>

**RemoteRef** HEAD

**RemoteSha** b25eac4ad6b5ee2eaca923d688ffaa49032ac3b1

## Contents

Goolam . . . . .	2
scISR . . . . .	2
<b>Index</b>	<b>5</b>

---

Goolam	<i>Goolam</i>
--------	---------------

---

### Description

Goolam dataset with data and cell types information. The number of genes is reduced to 10,000.

### Usage

Goolam

### Format

An object of class `list` of length 2.

---

scISR	<i>scISR: Single-cell Imputation using Subspace Regression</i>
-------	--

---

### Description

Perform single-cell Imputation using Subspace Regression

### Usage

```
scISR(
  data,
  ncores = 1,
  force_impute = FALSE,
  do_fast = TRUE,
  preprocessing = TRUE,
  batch_impute = FALSE,
  seed = 1
)
```

**Arguments**

data	Input matrix or data frame. Rows represent genes while columns represent samples
ncores	Number of cores that the algorithm should use. Default value is 1.
force_impute	Always perform imputation.
do_fast	Use fast imputation implementation.
preprocessing	Perform preprocessing on original data to filter out low quality features.
batch_impute	Perform imputation in batches to reduce memory consumption.
seed	Seed for reproducibility. Default value is 1.

**Details**

scISR performs imputation for single-cell sequencing data. scISR identifies the true dropout values in the scRNA-seq dataset using hyper-geometric testing approach. Based on the result obtained from hyper-geometric testing, the original dataset is segregated into two subsets including training data and imputable data. Next, training data is used for constructing a generalized linear regression model that is used for imputation on the imputable data.

**Value**

scISR returns an imputed single-cell expression matrix where rows represent genes while columns represent samples.

**Examples**

```
{
# Load the package
library(scISR)
# Load Goolam dataset
data('Goolam');
# Use only 500 random genes for example
set.seed(1)
raw <- Goolam$data[sample(seq_len(nrow(Goolam$data)), 500), ]
label <- Goolam$label

# Perform the imputation
imputed <- scISR(data = raw)

if(requireNamespace('mclust'))
{
  library(mclust)
  # Perform PCA and k-means clustering on raw data
  set.seed(1)
  # Filter genes that have only zeros from raw data
  raw_filer <- raw[rowSums(raw != 0) > 0, ]
  pca_raw <- irlba::prcomp_irlba(t(raw_filer), n = 50)$x
  cluster_raw <- kmeans(pca_raw, length(unique(label)),
    nstart = 2000, iter.max = 2000)$cluster
  print(paste('ARI of clusters using raw data:',
```

```
        round(adjustedRandIndex(cluster_raw, label),3)))

# Perform PCA and k-means clustering on imputed data
set.seed(1)
pca_imputed <- irlba::prcomp_irlba(t(imputed), n = 50)$x
cluster_imputed <- kmeans(pca_imputed, length(unique(label)),
                          nstart = 2000, iter.max = 2000)$cluster
print(paste('ARI of clusters using imputed data:',
            round(adjustedRandIndex(cluster_imputed, label),3)))
}
}
```

# Index

\* **datasets**

Goolam, [2](#)

Goolam, [2](#)

scISR, [2](#)