Package 'DuoClustering2018'

November 10, 2026

Type Package

Title Data, Clustering Results and Visualization Functions From Duò et al (2018)

Version 1.30.0

Author Angelo Duò, Charlotte Soneson

Maintainer Angelo Duò <angelo.duo@icloud.com>

Description Preprocessed experimental and simulated scRNA-seq data sets used for evaluation of clustering methods for scRNA-seq data in Duò et al (2018). Also contains results from applying several clustering methods to each of the data sets, and functions for plotting method performance.

License GPL (>=2)

Encoding UTF-8

LazyData true

biocViews SingleCellData, ExperimentData

Imports ExperimentHub, utils, magrittr, dplyr, tidyr, mclust, ggplot2, purrr, reshape2, viridis, ggthemes, stats, methods

Suggests knitr, rmarkdown, BiocStyle, iSEE, scater, SingleCellExperiment, SummarizedExperiment, plyr

VignetteBuilder knitr

RoxygenNote 6.1.0

git_url https://git.bioconductor.org/packages/DuoClustering2018

git_branch RELEASE_3_23

git_last_commit bc2c9ec

git_last_commit_date 2026-04-28

Repository Bioconductor 3.23

Date/Publication 2026-11-10

Contents

ari_df . . . 2
clustering_summary_filteredExpr10_Koh_v1 . . . 3
clustering_summary_filteredExpr10_Koh_v2 . . . 4
DuoClustering2018 . . . 5
duo_clustering_all_parameter_settings_v1 . . . 5
duo_clustering_all_parameter_settings_v2 . . . 6
plot_entropy . . . 7
plot_k_diff . . . 8
plot_performance . . . 8
plot_stability . . . 9
plot_timing . . . 10
sce_full_Koh . . . 10
sce_full_Kumar . . . 13
sce_full_SimKumar4easy . . . 15
sce_full_Trapnell . . . 17
sce_full_Zhengmix4eq . . . 19
shannon_entropy . . . 21

Index **22**

`ari_df` *Help function for computing ARI*

Description

Help function for computing ARI

Usage

```
ari_df(x)
```

Arguments

`x`	A `data.frame` with clustering results.

Value

a `data.frame` with ARI values for each pair of runs.

clustering_summary_filteredExpr10_Koh_v1
Clustering summaries

Description

Clustering results for the performance evaluation of clustering methods for scRNA-seq data, corresponding to v1 of Duò et al. (2018).

Usage

```
clustering_summary_filteredExpr10_Koh_v1(metadata = FALSE)
```

Arguments

metadata	Logical, whether only metadata should be returned

Details

These objects contain clustering results from the performance evaluation of clustering methods for scRNA-seq data. The clustering results are provided as a `data.frame` object containing 10 variables (columns) named `dataset`, `method`, `cell`, `run`, `k`, `resolution`, `cluster`, `trueclass`, `est_k` and `elapsed`. For further information see Duò et al. (2018).

Value

Returns a `data.frame`.

References

Duò A, Robinson MD. and Soneson C. (2018). *A systematic performance evaluation of clustering methods for single-cell RNA-seq data.* F1000Res., 7:1141.

Examples

```
clustering_summary_filteredExpr10_Koh_v1()
```

clustering_summary_filteredExpr10_Koh_v2 *Clustering summaries*

Description

Clustering results for the performance evaluation of clustering methods for scRNA-seq data, corresponding to v2 of Duò et al. (2018).

Usage

```
clustering_summary_filteredExpr10_Koh_v2(metadata = FALSE)
```

Arguments

metadata	Logical, whether only metadata should be returned

Details

These objects contain clustering results from the performance evaluation of clustering methods for scRNA-seq data. The clustering results are provided as a `data.frame` object containing 10 variables (columns) named `dataset`, `method`, `cell`, `run`, `k`, `resolution`, `cluster`, `trueclass`, `est_k` and `elapsed`. For further information see Duò et al. (2018).

Value

Returns a `data.frame`.

References

Duò A, Robinson MD. and Soneson C. (2018). *A systematic performance evaluation of clustering methods for single-cell RNA-seq data.* F1000Res., 7:1141.

Examples

```
clustering_summary_filteredExpr10_Koh_v2()
```

DuoClustering2018 *DuoClustering2018*

Description

Data package containing scRNA-seq data sets, clustering results and functions for summarizing the performance of different scRNA-seq clustering methods.

Details

This package contains publicly available scRNA-seq data sets and the accompanying results from clustering using general-purpose methods and scRNA-seq clustering methods. Several real data sets a well as simulated data sets are provided. The data sets have been used to evaluate the performance of clustering algorithms in our previous work and publication (Duò et al., F1000Research 2018). The data sets are available as `SingleCellExperiment` objects. For additional details on the data sets, see the help files for the respective data sets.

Additionally, the clustering results from the evaluation as well as functions for summarization and visualization of the clustering results are provided.

A description of the basic usage of the package for retrieving data sets and clustering results, and how to construct various plots summarizing the performance of different methods is outlined in the package vignettes.

Author(s)

Angelo Duò and Charlotte Soneson

References

Duò, A., Robinson, M.D., and Soneson, C. (2018). *A systematic performance evaluation of clustering methods for single-cell RNA-seq data.* F1000Research, 7:1141.

`duo_clustering_all_parameter_settings_v1` *Hyperparameter values*

Description

Hyperparameter values for all clustering algorithms and data sets in v1 of Duo et al (F1000Research 2018)

Usage

```
duo_clustering_all_parameter_settings_v1(metadata = FALSE)
```

Arguments

metadata	Logical, whether only metadata should be returned

Details

List of hyperparameter values used for all clustering algorithms and data sets in v1 of Duò et al (F1000Research 2018).

Value

Returns a list with hyperparameter values for all data sets and methods.

References

Duò, A., Robinson, M.D., and Soneson, C. (2018). *A systematic performance evaluation of clustering methods for single-cell RNA-seq data.* F1000Research, 7:1141.

Examples

```
duo_clustering_all_parameter_settings_v1()
```

---

duo_clustering_all_parameter_settings_v2
Hyperparameter values

Description

Hyperparameter values for all clustering algorithms and data sets in v2 of Duo et al (F1000Research 2018)

Usage

```
duo_clustering_all_parameter_settings_v2(metadata = FALSE)
```

Arguments

metadata	Logical, whether only metadata should be returned

Details

List of hyperparameter values used for all clustering algorithms and data sets in v2 of Duò et al (F1000Research 2018).

Value

Returns a list with hyperparameter values for all data sets and methods.

References

Duò, A., Robinson, M.D., and Soneson, C. (2018). *A systematic performance evaluation of clustering methods for single-cell RNA-seq data.* F1000Research, 7:1141.

Examples

```
duo_clustering_all_parameter_settings_v2()
```

`plot_entropy` *Plot entropy of cluster assignments*

Description

Plot entropy of cluster assignments

Usage

```
plot_entropy(res, method_colors = NULL)
```

Arguments

`res`	A data.frame with clustering results.
`method_colors`	A named vector with colors to use for the different clustering methods. Can be NULL, in which case colors are chosen automatically.

Value

A named list of ggplot2 objects

Author(s)

Angelo Duo, Charlotte Soneson

Examples

```
res <- clustering_summary_filteredExpr10_Koh_v2()
plots <- plot_entropy(res)
```

plot_k_diff *Plot differences between optimal, estimated and true number of clusters*

Description

Plot differences between optimal, estimated and true number of clusters

Usage

```
plot_k_diff(res, method_colors = NULL)
```

Arguments

res	A data.frame with clustering results.
method_colors	A named vector with colors to use for the different clustering methods. Can be NULL, in which case colors are chosen automatically.

Value

A named list of ggplot2 objects

Author(s)

Angelo Duo, Charlotte Soneson

Examples

```
res <- clustering_summary_filteredExpr10_Koh_v1()
plots <- plot_k_diff(res)
```

plot_performance *Plot performance of clustering methods*

Description

Generate various plots of the agreement between each clustering and the true partitioning of the cells, quantified by the adjusted Rand index (ARI).

Usage

```
plot_performance(res, method_colors = NULL)
```

Arguments

res	A data.frame with clustering results.
method_colors	A named vector with colors to use for the different clustering methods. Can be NULL, in which case colors are chosen automatically.

Value

A named list of ggplot2 objects

Author(s)

Angelo Duo, Charlotte Soneson

Examples

```
res <- clustering_summary_filteredExpr10_Koh_v1()
plots <- plot_performance(res)
```

---

plot_stability *Plot stability of methods*

Description

Plot the stability of the clusterings obtained for each method

Usage

```
plot_stability(res, method_colors = NULL)
```

Arguments

res	A data.frame with clustering results.
method_colors	A named vector with colors to use for the different clustering methods. Can be NULL, in which case colors are chosen automatically.

Value

A named list of ggplot2 objects

Author(s)

Angelo Duo, Charlotte Soneson

Examples

```
res <- clustering_summary_filteredExpr10_Koh_v1()
plots <- plot_stability(res)
```

plot_timing *Plot timing of methods*

Description

Plot the elapsed time for each clustering method

Usage

```
plot_timing(res, method_colors = NULL, scaleMethod = NULL)
```

Arguments

res	A data.frame with clustering results.
method_colors	A named vector with colors to use for the different clustering methods. Can be NULL, in which case colors are chosen automatically.
scaleMethod	Either NULL or one of the clustering methods in the result data.frame. If not NULL, a plot will be generated where all elapsed times are normalized by dividing with the time for scaleMethod. If NULL, this plot will not be generated.

Value

A named list of ggplot2 objects

Author(s)

Angelo Duo, Charlotte Soneson

Examples

```
res <- clustering_summary_filteredExpr10_Koh_v1()
plots <- plot_timing(res)
```

sce_full_Koh *Koh data sets*

Description

Gene or TCC counts for a scRNA-seq data set from Koh et al. (2016), consisting of in vitro cultured H7 embryonic stem cells (WiCell) and H7-derived downstream early mesoderm progenitors.

Usage

```
sce_full_Koh(metadata = FALSE)
sce_filteredExpr10_Koh(metadata = FALSE)
sce_filteredHVG10_Koh(metadata = FALSE)
sce_filteredM3Drop10_Koh(metadata = FALSE)
sce_full_KohTCC(metadata = FALSE)
sce_filteredExpr10_KohTCC(metadata = FALSE)
sce_filteredHVG10_KohTCC(metadata = FALSE)
sce_filteredM3Drop10_KohTCC(metadata = FALSE)
```

Arguments

metadata	Logical, whether only metadata should be returned

Format

SingleCellExperiment

Details

This is a scRNA-seq data set originally from Koh et al. (2016). The data set consists of gene-level read counts or TCCs (transcript compatibility counts) of in vitro cultured human H7 embryonic stem cells (WiCell) and H7-derived downstream early mesoderm progenitors. It contains 9 subpopulations, defined by the cell phenotype given by the authors' annotations. The data sets have been used to evaluate the performance of clustering algorithms in Duò et al. (2018).

For the `sce_full_Koh` data set, all genes except those with zero counts across all cells are retained. The gene counts are gene-level length-scaled TPM values derived from Salmon (Patro et al. (2017)) quantifications (see Soneson and Robinson (2018)). For the TCC data set we estimated transcripts compatibility counts using `kallisto` as an alternative to the gene-level count matrix (Bray et al. (2016), Ntranos et al. (2016)).

The `scater` package was used to perform quality control of the data sets (McCarthy et al. (2017)). Features with zero counts across all cells, as well as all cells with total count or total number of detected features more than 3 median absolute deviations (MADs) below the median across all cells (on the log scale), were excluded.

The `sce_full_Koh` data set consists of 531 cells and 48,981 features, and the `sce_full_KohTCC` data set of 531 cells and 811,938 features. The `filteredExpr`, `filteredHVG` and `filteredM3Drop10` are further reduced data sets. For each of the filtering methods, we retained 10 percent of the number of genes (with a non-zero count in at least one cell) in the original data sets.

For the `filteredExpr` data sets, only the genes/TCCs with the highest average expression (log-normalized count) value across all cells were retained. Using the `Seurat` package (Satija et al. (2015)), the `filteredHVG` data sets were filtered on the variability of the features and only the most highly variable ones were retained. Finally, the `M3Drop` package was used to model the dropout rate of the genes as a function of the mean expression level using the Michaelis-Menten equation and select variables to retain for the `filteredM3Drop10` data sets (Andrews and Hemberg (2018)).

The `scater` package was used to normalize the count values, based on normalization factors calculated by the deconvolution method from the `scran` package (Lun et al. (2016)).

This data set is provided as a `SingleCellExperiment` object (Lun and Risso (2017)). Raw data files for the original data set (SRP073808) are available from https://www.ncbi.nlm.nih.gov/sra?term=SRP073808.

Value

Returns a `SingleCellExperiment` object.

References

Andrews, T.S., and Hemberg, M. (2018). *Dropout-based feature selection for scRNASeq*. bioRxiv doi:https://doi.org/10.1101/065094.

Bray, N.L., Pimentel, H., Melsted, P., and Pachter, L. (2016). *Near-optimal probabilistic RNA-seq quantification*. Nat. Biotechnol. 34: 525–527.

Duò, A., Robinson, M.D., and Soneson, C. (2018). *A systematic performance evaluation of clustering methods for single-cell RNA-seq data.* F1000Res. 7:1141.

Koh, P.W., Sinha, R., Barkal, A.A., Morganti R.M., Chen, A., Weissman, I.L., Ang, L.T., Kundaje, A., and Loh, K.M. (2016). *An atlas of transcriptional, chromatin accessibility, and surface marker changes in human mesoderm development.* Scientific Data 3:160109.

Lun, A.T.L., Bach, K., and Marioni, J.C. (2016) *Pooling across cells to normalize single-cell RNA sequencing data with many zero counts.* Genome Biol. 17(1): 75.

Lun, A.T.L., and Risso, D. (2017). *SingleCellExperiment: S4 Classes for Single Cell Data*. R package version 1.0.0.

McCarthy, D.J., Campbell, K.R., Lun, A.T.L., and Wills, Q.F. (2017): *Scater: pre-processing, quality control, normalization and visualization of single-cell RNA-seq data in R.* Bioinformatics 33(8): 1179-1186.

Ntranos, V., Kamath, G.M., Zhang, J.M., Pachter, L., and Tse, D.N. (2016): *Fast and accurate single-cell RNA-seq analysis by clustering of transcript-compatibility counts.* Genome Biol. 17:112.

Patro, R., Duggal, G., Love, M.I., Irizarry, R.A., and Kingsford, C. (2017): *Salmon provides fast and bias-aware quantification of transcript expression.* Nat. Methods 14:417-419.

Satija, R., Farrell, J.A., Gennert, D., Schier, A.F., and Regev, A. (2015). *Spatial reconstruction of single-cell gene expression data.* Nat. Biotechnol. 33(5): 495–502.

Soneson, C., and Robinson, M.D. (2018). *Bias, robustness and scalability in single-cell differential expression analysis.* Nat. Methods, 15(4): 255-261.

Examples

```
sce_filteredHVG10_Koh()
```

sce_full_Kumar *Kumar data sets*

Description

Gene or TCC counts for scRNA-seq data set from Kumar et al. (2014), consisting of mESCs with various genetic perturbations which are cultured in different media.

Usage

```
sce_full_Kumar(metadata = FALSE)
sce_filteredExpr10_Kumar(metadata = FALSE)
sce_filteredHVG10_Kumar(metadata = FALSE)
sce_filteredM3Drop10_Kumar(metadata = FALSE)
sce_full_KumarTCC(metadata = FALSE)
sce_filteredExpr10_KumarTCC(metadata = FALSE)
sce_filteredHVG10_KumarTCC(metadata = FALSE)
sce_filteredM3Drop10_KumarTCC(metadata = FALSE)
```

Arguments

metadata	Logical, whether only metadata should be returned

Format

SingleCellExperiment

Details

This is a scRNA-seq data set originally from Kumar et al. (2014). The data set consists of gene-level read counts or TCCs (transcript compatibility counts) for mESCs from *Mus musculus* with various genetic perturbations which are cultured in different media. It contains 3 subpopulations, defined by the cell phenotype given by the authors' annotations. The data sets have been used to evaluate the performance of clustering algorithms in Duò et al. (2018).

For the sce_full_Kumar data set, all genes except those with zero counts across all cells are retained. The gene counts are gene-level length-scaled TPM values derived from Salmon (Patro et al. (2017)) quantifications (see Soneson and Robinson (2018)). For the TCC data set we estimated transcripts compatibility counts using kallisto as an alternative to the gene-level count matrix (Bray et al. (2016), Ntranos et al. (2016)).

The scater package was used to perform quality control of the data sets (McCarthy et al. (2017)). Features with zero counts across all cells, as well as all cells with total count or total number of detected features more than 3 median absolute deviations (MADs) below the median across all cells (on the log scale), were excluded. Additionally, cells with a large fraction of ERCC reads were filtered out.

The sce_full_Kumar data set consists of 246 cells and 45,159 features, the sce_full_KumarTCC data set of 246 cells and 803,405 features, respectively. The filteredExpr, filteredHVG and

`filteredM3Drop10` are further reduced data sets. For each filtering method, we retained 10 percent of the original number of genes (with a non-zero count in at least one cell) in the original data sets.

For the `filteredExpr` data sets, only the genes/TCCs with the highest average expression (log-normalized count) value across all cells were retained. Using the `Seurat` package, the `filteredHVG` data sets were filtered on the variability of the features and only the most highly variable ones were retained (Satija et al. (2015)). Finally, the `M3Drop` package was used to model the dropout rate of the genes as a function of the mean expression level using the Michaelis-Menten equation and select variables to retain for the `filteredM3Drop10` data sets (Andrews and Hemberg (2018)).

The `scater` package was used to normalize the count values, based on normalization factors calculated by the deconvolution method from the `scran` package (Lun et al. (2016)). This data set is provided as a `SingleCellExperiment` object (Lun and Risso (2017)). For further information on the `SingleCellExperiment` class, see the corresponding manual. Raw data files for the original data set (GSE60749) are available from https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE60749.

Value

Returns a `SingleCellExperiment` object.

References

Andrews, T.S., and Hemberg, M. (2018). *Dropout-based feature selection for scRNASeq*. bioRxiv doi:https://doi.org/10.1101/065094.

Bray, N.L., Pimentel, H., Melsted, P., and Pachter, L. (2016). *Near-optimal probabilistic RNA-seq quantification*. Nat. Biotechnol. 34: 525–527.

Duò, A., Robinson, M.D., and Soneson, C. (2018). *A systematic performance evaluation of clustering methods for single-cell RNA-seq data.* F1000Res. 7:1141.

Kumar R.M., Cahan P., Shalek A.K., Satija R., DaleyKeyser A.J., Li H., Zhang J., Pardee K., Gennert D., Trombetta J.J., Ferrante T.C., Regev A., Daley G.Q., and Collins J.J. (2014) *Deconstructing transcriptional heterogeneity in pluripotent stem cells.* Nature 516(7529): 56–61.

Lun, A.T.L., Bach, K., and Marioni, J.C. (2016) *Pooling across cells to normalize single-cell RNA sequencing data with many zero counts.* Genome Biol. 17(1): 75.

Lun, A.T.L., and Risso, D. (2017). *SingleCellExperiment: S4 Classes for Single Cell Data*. R package version 1.0.0.

McCarthy, D.J., Campbell, K.R., Lun, A.T.L., and Wills, Q.F. (2017): *Scater: pre-processing, quality control, normalization and visualization of single-cell RNA-seq data in R.* Bioinformatics 33(8): 1179-1186.

Ntranos, V., Kamath, G.M., Zhang, J.M., Pachter, L., and Tse, D.N. (2016): *Fast and accurate single-cell RNA-seq analysis by clustering of transcript-compatibility counts.* Genome Biol. 17:112.

Patro, R., Duggal, G., Love, M.I., Irizarry, R.A., and Kingsford, C. (2017): *Salmon provides fast and bias-aware quantification of transcript expression.* Nat. Methods 14:417-419.

Satija, R., Farrell, J.A., Gennert, D., Schier, A.F., and Regev, A. (2015). *Spatial reconstruction of single-cell gene expression data.* Nat. Biotechnol. 33(5): 495–502.

Soneson, C., and Robinson, M.D. (2018). *Bias, robustness and scalability in single-cell differential expression analysis.* Nat. Methods 15(4): 255-261.

Examples

```
sce_filteredExpr10_Kumar()
```

`sce_full_SimKumar4easy` *SimKumar data sets*

Description

Gene counts for scRNA-seq data sets simulated with the `splatter` package.

Usage

```
sce_full_SimKumar4easy(metadata = FALSE)
sce_filteredExpr10_SimKumar4easy(metadata = FALSE)
sce_filteredHVG10_SimKumar4easy(metadata = FALSE)
sce_filteredM3Drop10_SimKumar4easy(metadata = FALSE)
sce_full_SimKumar4hard(metadata = FALSE)
sce_filteredExpr10_SimKumar4hard(metadata = FALSE)
sce_filteredHVG10_SimKumar4hard(metadata = FALSE)
sce_filteredM3Drop10_SimKumar4hard(metadata = FALSE)
sce_full_SimKumar8hard(metadata = FALSE)
sce_filteredExpr10_SimKumar8hard(metadata = FALSE)
sce_filteredHVG10_SimKumar8hard(metadata = FALSE)
sce_filteredM3Drop10_SimKumar8hard(metadata = FALSE)
```

Arguments

`metadata`	Logical, whether only metadata should be returned

Format

SingleCellExperiment

Details

Using one subpopulation of the `sce_full_Kumar` data set as input, scRNA-seq data with known group structure was simulated with the `splatter` package from Zappia et al. (2017). The simulated data have been used to evaluate the performance of clustering algorithms in Duò et al. (2018).

Three data sets have been generated, each consisting of 500 cells and approximately 43,000 features, with varying degree of cluster separability. The `sce_full_SimKumar4easy` data set consists of 4 subpopulations with relative abundances 0.1, 0.15, 0.5 and 0.25, and probabilities of differential expression set to 0.05, 0.1, 0.2 and 0.4 for the four subpopulations, respectively. The `sce_full_SimKumar4hard` data set consists of 4 subpopulations with relative abundances 0.2, 0.15, 0.4 and 0.25, and probabilities of differential expression 0.01, 0.05, 0.05 and 0.08. Finally, the `sce_full_SimKumar8hard` data set consists of 8 subpopulations with relative abundances 0.13,

0.07, 0.1, 0.05, 0.4, 0.1, 0.1 and 0.05, and probabilities of differential expression equal to 0.03, 0.03, 0.03, 0.05, 0.05, 0.07, 0.08 and 0.1, respectively.

The `scater` package was used to perform quality control of the data sets (McCarthy et al. (2017)). Features with zero counts across all cells, as well as cells with total count or total number of detected features more than 3 median absolute deviations (MADs) below the median across all cells (on the log scale), were excluded. The `filteredExpr`, `filteredHVG` and `filteredM3Drop10` are further reduced data sets. For each of the filtering method, we retained 10 percent of the original number of genes (with a non-zero count in at least one cell) in the original data sets.

For the `filteredExpr` data sets, only the genes with the highest average expression (log-normalized count) value across all cells were retained. Using the `Seurat` package, the `filteredHVG` data sets were filtered on the variability of the features and only the most highly variable ones were retained (Satija et al. (2015)). Finally, the `M3Drop` package was used to model the dropout rate of the genes as a function of the mean expression level using the Michaelis-Menten equation and select variables to retain for the `filteredM3Drop10` data sets (Andrews and Hemberg (2018)).

The `scater` package was used to normalize the count values, based on normalization factors calculated by the deconvolution method from the `scran` package (Lun et al. (2016)). This data set is provided as a `SingleCellExperiment` object (Lun and Risso (2017)). For further information on the `SingleCellExperiment` class, see the corresponding manual.

Value

Returns a `SingleCellExperiment` object.

References

Andrews, T.S., and Hemberg, M. (2018). *Dropout-based feature selection for scRNASeq*. bioRxiv doi:https://doi.org/10.1101/065094.

Duò, A., Robinson, M.D., and Soneson, C. (2018). *A systematic performance evaluation of clustering methods for single-cell RNA-seq data.* F1000Res. 7:1141.

Lun, A.T.L., Bach, K., and Marioni, J.C. (2016) *Pooling across cells to normalize single-cell RNA sequencing data with many zero counts.* Genome Biol. 17(1): 75.

Lun, A.T.L., and Risso, D. (2017). *SingleCellExperiment: S4 Classes for Single Cell Data*. R package version 1.0.0.

McCarthy, D.J., Campbell, K.R., Lun, A.T.L., and Wills, Q.F. (2017): *Scater: pre-processing, quality control, normalization and visualization of single-cell RNA-seq data in R.* Bioinformatics 33(8): 1179-1186.

Satija, R., Farrell, J.A., Gennert, D., Schier, A.F., and Regev, A. (2015). *Spatial reconstruction of single-cell gene expression data.* Nat. Biotechnol. 33(5): 495–502.

Zappia, L., Phipson, B., and Oshlack, A. (2017). *Splatter: simulation of single-cell RNA sequencing data.* Genome Biol. 18(1): 174.

Examples

```
sce_filteredExpr10_SimKumar4easy()
```

`sce_full_Trapnell` *Trapnell data sets*

Description

Gene or TCC counts for scRNA-seq data set from Trapnell et al. (2014), consisting of primary myoblasts over a time course of serum-induced differentiation.

Usage

```
sce_full_Trapnell(metadata = FALSE)
sce_filteredExpr10_Trapnell(metadata = FALSE)
sce_filteredHVG10_Trapnell(metadata = FALSE)
sce_filteredM3Drop10_Trapnell(metadata = FALSE)
sce_full_TrapnellTCC(metadata = FALSE)
sce_filteredExpr10_TrapnellTCC(metadata = FALSE)
sce_filteredHVG10_TrapnellTCC(metadata = FALSE)
sce_filteredM3Drop10_TrapnellTCC(metadata = FALSE)
```

Arguments

`metadata`	Logical, whether only metadata should be returned

Format

SingleCellExperiment

Details

This is a scRNA-seq data set originally from Trapnell et al. (2014). The data set consists of gene-level read counts or TCCs (transcript compatibility counts) from human primary myoblasts over a time course of serum-induced differentiation. It contains 3 subpopulations, defined by the cell phenotype given by the authors' annotations. The data sets have been used to evaluate the performance of clustering algorithms in Duò et al. (2018).

For the `sce_full_Trapnell` data set, all genes except those with zero counts across all cells are retained. The gene counts are gene-level length-scaled TPM values derived from Salmon (Patro et al. (2017)) quantifications (see Soneson and Robinson (2018)). For the TCC data set we estimated transcripts compatibility counts using `kallisto` as an alternative to the gene-level count matrix (Bray et al. (2016), Ntranos et al. (2016)).

The `scater` package was used to perform quality control of the data sets (McCarthy et al. (2017)). Features with zero counts across all cells, as well as all cells with total count or total number of detected features more than 3 median absolute deviations (MADs) below the median across all cells (on the log scale), were excluded. Additionally, cells that were classified as doublets or debris were filtered out.

The `sce_full_Trapnell` data set consists of 222 cells and 41,111 features, the `sce_full_TrapnellTCC` data set of 227 cells and 684,953 features, respectively. The `filteredExpr`, `filteredHVG` and

`filteredM3Drop10` are further reduced data sets. For each of the filtering method, we retained 10 percent of the original number of genes (with a non-zero count in at least one cell) in the original data sets.

For the `filteredExpr` data sets, only the genes/TCCs with the highest average expression (log-normalized count) value across all cells were retained. Using the `Seurat` package, the `filteredHVG` data sets were filtered on the variability of the features and only the most highly variable ones were retained (Satija et al. (2015)). Finally, the `M3Drop` package was used to model the dropout rate of the genes as a function of the mean expression level using the Michaelis-Menten equation and select variables to retain for the `filteredM3Drop10` data sets (Andrews and Hemberg (2018)).

The `scater` package was used to normalize the count values, based on normalization factors calculated by the deconvolution method from the `scran` package (Lun et al. (2016)).

This data set is provided as a `SingleCellExperiment` object (Lun and Risso (2017)). For further information on the `SingleCellExperiment` class, see the corresponding manual. Raw data files for the original data set (GSE52529) are available from https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE52529.

Value

Returns a `SingleCellExperiment` object.

References

Andrews, T.S., and Hemberg, M. (2018). *Dropout-based feature selection for scRNASeq*. bioRxiv doi:https://doi.org/10.1101/065094.

Bray, N.L., Pimentel, H., Melsted, P., and Pachter, L. (2016). *Near-optimal probabilistic RNA-seq quantification*. Nat. Biotechnol. 34: 525–527.

Duò, A., Robinson, M.D., and Soneson, C. (2018). *A systematic performance evaluation of clustering methods for single-cell RNA-seq data.* F1000Res. 7:1141.

Lun, A.T.L., Bach, K., and Marioni, J.C. (2016) *Pooling across cells to normalize single-cell RNA sequencing data with many zero counts.* Genome Biol. 17(1): 75.

Lun, A.T.L., and Risso, D. (2017). *SingleCellExperiment: S4 Classes for Single Cell Data*. R package version 1.0.0.

McCarthy, D.J., Campbell, K.R., Lun, A.T.L., and Wills, Q.F. (2017): *Scater: pre-processing, quality control, normalization and visualization of single-cell RNA-seq data in R.* Bioinformatics 33(8): 1179-1186.

Ntranos, V., Kamath, G.M., Zhang, J.M., Pachter, L., and Tse, D.N. (2016): *Fast and accurate single-cell RNA-seq analysis by clustering of transcript-compatibility counts.* Genome Biol. 17:112.

Patro, R., Duggal, G., Love, M.I., Irizarry, R.A., and Kingsford, C. (2017): *Salmon provides fast and bias-aware quantification of transcript expression.* Nat. Methods 14:417-419.

Satija, R., Farrell, J.A., Gennert, D., Schier, A.F., and Regev, A. (2015). *Spatial reconstruction of single-cell gene expression data.* Nat. Biotechnol. 33(5): 495–502.

Soneson, C., and Robinson, M.D. (2018). *Bias, robustness and scalability in single-cell differential expression analysis.* Nat. Methods, 15(4): 255-261.

Trapnell, C., Cacchiarelli, D., Grimsby, J., Pokharel, P., Li, S., Morse, M., Lennon, N.J., Livak, K.J., Mikkelsen, T.S., and Rinn, J.L. (2014). *The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells.* Nat. Biotechnol. 32(4): 381–386.

Examples

```
sce_filteredExpr10_Trapnell()
```

`sce_full_Zhengmix4eq` *Zheng data sets*

Description

Gene counts for scRNA-seq data sets from Zheng et al. (2017), consisting of pre-sorted cell types combined into three artificial data sets with different cell proportions.

Usage

```
sce_full_Zhengmix4eq(metadata = FALSE)
sce_filteredExpr10_Zhengmix4eq(metadata = FALSE)
sce_filteredHVG10_Zhengmix4eq(metadata = FALSE)
sce_filteredM3Drop10_Zhengmix4eq(metadata = FALSE)
sce_full_Zhengmix4uneq(metadata = FALSE)
sce_filteredExpr10_Zhengmix4uneq(metadata = FALSE)
sce_filteredHVG10_Zhengmix4uneq(metadata = FALSE)
sce_filteredM3Drop10_Zhengmix4uneq(metadata = FALSE)
sce_full_Zhengmix8eq(metadata = FALSE)
sce_filteredExpr10_Zhengmix8eq(metadata = FALSE)
sce_filteredHVG10_Zhengmix8eq(metadata = FALSE)
sce_filteredM3Drop10_Zhengmix8eq(metadata = FALSE)
```

Arguments

`metadata`	Logical, whether only metadata should be returned

Format

SingleCellExperiment

Details

This is a scRNA-seq data set originally from Zheng et al. (2017). The data set consists of eight pre-sorted cell types (B-cells, naive cytotoxic T-cells, CD14 monocytes, regulatory T-cells, CD56 NK cells, memory T-cells, CD4 T-helper cells and naive T-cells) from *Homo sapiens* combined into three artificial data sets with different cell proportions. The annotated cell type (obtained by pre-sorting of the cells) is used as the true cell label. The data sets have been used to evaluate the performance of clustering algorithms in Duò et al. (2018).

For the `Zhengmix4eq` data set, randomly selected B-cells, CD14 monocytes, naive cytotoxic T-cells and regulatory T-cells were combined in equal proportions (1,000 cells per subpopulation). The `Zhengmix4uneq` data set consists of four cell types, combined in unequal proportions (1,000 B-cells, 500 naive cytotoxic T-cells, 2,000 CD14 monocytes and 3,000 regulatory T-cells). For the

Zhengmix8eq data set, all eight populations were combined in approximately equal proportions (400–600 cells per population).

For the sce_full_Zhengmix4eq, sce_full_Zhengmix4uneq and sce_full_Zhengmix8eq data set, all genes except those with zero counts across all cells are retained. The gene counts are unique molecular identifiers (UMIs) counts. The scater package was used to perform quality control of the data (McCarthy et al. (2017)). Features with zero counts across all cells, as well as all cells with total count or total number of detected features more than 3 median absolute deviations (MADs) below the median across all cells (on the log scale), were excluded.

The sce_full_Zhengmix4eq data set consists of 3,994 cells and 15,568 features, the sce_full_Zhengmix4uneq data set of 6,498 cells and 16,443 features and the sce_full_Zhengmix8eq of 3,994 cells and 16,443 features, respectively. The filteredExpr, filteredHVG and filteredM3Drop10 are further reduced data sets. For each of the filtering method, we retained 10 percent of the original number of genes (with a non-zero count in at least one cell) in the original data sets.

For the filteredExpr data sets, only the genes with the highest average expression (log-normalized count) value across all cells were retained. Using the Seurat package, the filteredHVG data sets were filtered on the variability of the features and only the most highly variable ones were retained (Satija et al. (2015)). Finally, the M3Drop package was used to model the dropout rate of the genes as a function of the mean expression level using the Michaelis-Menten equation and select variables to retain for the filteredM3Drop10 data sets (Andrews and Hemberg (2018)).

The scater package was used to normalize the count values, based on normalization factors calculated by the deconvolution method from the scran package (Lun et al. (2016)). This data set is provided as a SingleCellExperiment object (Lun and Risso (2017)). For further information on the SingleCellExperiment class, see the corresponding manual. Raw data files or the original data sets are available from https://support.10xgenomics.com/single-cell-gene-expression/datasets.

Value

Returns a SingleCellExperiment object.

References

Andrews, T.S., and Hemberg, M. (2018). *Dropout-based feature selection for scRNASeq*. bioRxiv doi:https://doi.org/10.1101/065094.

Duò, A., Robinson, M.D., and Soneson, C. (2018). *A systematic performance evaluation of clustering methods for single-cell RNA-seq data.* F1000Res. 7:1141.

Lun, A.T.L., Bach, K., and Marioni, J.C. (2016) *Pooling across cells to normalize single-cell RNA sequencing data with many zero counts.* Genome Biol. 17(1): 75.

Lun, A.T.L., and Risso, D. (2017). *SingleCellExperiment: S4 Classes for Single Cell Data*. R package version 1.0.0.

McCarthy, D.J., Campbell, K.R., Lun, A.T.L., and Wills, Q.F. (2017): *Scater: pre-processing, quality control, normalization and visualization of single-cell RNA-seq data in R.* Bioinformatics 33(8): 1179-1186.

Satija, R., Farrell, J.A., Gennert, D., Schier, A.F., and Regev, A. (2015). *Spatial reconstruction of single-cell gene expression data.* Nat. Biotechnol. 33(5): 495–502.

Zheng, G.X., Terry, J.M., Belgrader P., Ryvkin, Pl, Bent, Z.W., Wilson, R., Ziraldo, S.B., Wheeler, T.D., McDermott, G.P., Zhu, J., Gregory, M.T., Shuga, J., Montesclaros, L., Underwood, J.G.,

Masquelier, D.A., Nishimura, S.Y., Schnall-Levin, M., Wyatt, P.W., Hindson, C.M., Bharadwaj, R., Wong, A., Ness, K.D., Beppu, L.W., Deeg, H.J., McFarland, C., Loeb, K.R., Valente, W.J., Ericson, N.G., Stevens, E.A., Radich, J.P., Mikkelsen, T.S., Hindson, B.J., and Bielas, J.H. (2017). *Massively parallel digital transcriptional profiling of single cells.* Nat. Commun. 8:14049.

Examples

```
sce_filteredExpr10_Zhengmix4eq()
```

`shannon_entropy` *Calculate Shannon entropy*

Description

Calculate Shannon entropy

Usage

```
shannon_entropy(cluster_assignments)
```

Arguments

`cluster_assignments`
: A vector with cluster assignments

Value

The Shannon entropy of the assignment vector

Index

∗ **datasets**
clustering_summary_filteredExpr10_Koh_v1, 3
clustering_summary_filteredExpr10_Koh_v2, 4
duo_clustering_all_parameter_settings_v1, 5
duo_clustering_all_parameter_settings_v2, 6
sce_full_Koh, 10
sce_full_Kumar, 13
sce_full_SimKumar4easy, 15
sce_full_Trapnell, 17
sce_full_Zhengmix4eq, 19

ari_df, 2

clustering_summary_filteredExpr10_Koh_v1, 3
clustering_summary_filteredExpr10_Koh_v2, 4
clustering_summary_filteredExpr10_KohTCC_v1 *(clustering_summary_filteredExpr10_Koh_v1)*, 3
clustering_summary_filteredExpr10_KohTCC_v2 *(clustering_summary_filteredExpr10_Koh_v2)*, 4
clustering_summary_filteredExpr10_Kumar_v1 *(clustering_summary_filteredExpr10_Koh_v1)*, 3
clustering_summary_filteredExpr10_Kumar_v2 *(clustering_summary_filteredExpr10_Koh_v2)*, 4
clustering_summary_filteredExpr10_KumarTCC_v1 *(clustering_summary_filteredExpr10_Koh_v1)*, 3
clustering_summary_filteredExpr10_KumarTCC_v2 *(clustering_summary_filteredExpr10_Koh_v2)*, 4
clustering_summary_filteredExpr10_SimKumar4easy_v1 *(clustering_summary_filteredExpr10_Koh_v1)*, 3
clustering_summary_filteredExpr10_SimKumar4easy_v2 *(clustering_summary_filteredExpr10_Koh_v2)*, 4
clustering_summary_filteredExpr10_SimKumar4hard_v1 *(clustering_summary_filteredExpr10_Koh_v1)*, 3
clustering_summary_filteredExpr10_SimKumar4hard_v2 *(clustering_summary_filteredExpr10_Koh_v2)*, 4
clustering_summary_filteredExpr10_SimKumar8hard_v1 *(clustering_summary_filteredExpr10_Koh_v1)*, 3
clustering_summary_filteredExpr10_SimKumar8hard_v2 *(clustering_summary_filteredExpr10_Koh_v2)*, 4
clustering_summary_filteredExpr10_Trapnell_v1 *(clustering_summary_filteredExpr10_Koh_v1)*, 3
clustering_summary_filteredExpr10_Trapnell_v2 *(clustering_summary_filteredExpr10_Koh_v2)*, 4
clustering_summary_filteredExpr10_TrapnellTCC_v1 *(clustering_summary_filteredExpr10_Koh_v1)*, 3
clustering_summary_filteredExpr10_TrapnellTCC_v2 *(clustering_summary_filteredExpr10_Koh_v2)*, 4
clustering_summary_filteredExpr10_Zhengmix4eq_v1 *(clustering_summary_filteredExpr10_Koh_v1)*, 3
clustering_summary_filteredExpr10_Zhengmix4eq_v2 *(clustering_summary_filteredExpr10_Koh_v2)*, 4
clustering_summary_filteredExpr10_Zhengmix4uneq_v1 *(clustering_summary_filteredExpr10_Koh_v1)*, 3

clustering_summary_filteredExpr10_Zhengmix4uneq_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredExpr10_Zhengmix8eq_v1 (clustering_summary_filteredExpr10_Koh_v1), 3
clustering_summary_filteredExpr10_Zhengmix8eq_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredHVG10_Koh_v1 (clustering_summary_filteredExpr10_Koh_v1), 3
clustering_summary_filteredHVG10_Koh_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredHVG10_KohTCC_v1 (clustering_summary_filteredExpr10_Koh_v1), 3
clustering_summary_filteredHVG10_KohTCC_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredHVG10_Kumar_v1 (clustering_summary_filteredExpr10_Koh_v1), 3
clustering_summary_filteredHVG10_Kumar_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredHVG10_KumarTCC_v1 (clustering_summary_filteredExpr10_Koh_v1), 3
clustering_summary_filteredHVG10_KumarTCC_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredHVG10_SimKumar4easy_v1 (clustering_summary_filteredExpr10_Koh_v1), 3
clustering_summary_filteredHVG10_SimKumar4easy_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredHVG10_SimKumar4hard_v1 (clustering_summary_filteredExpr10_Koh_v1), 3
clustering_summary_filteredHVG10_SimKumar4hard_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredHVG10_SimKumar8hard_v1 (clustering_summary_filteredExpr10_Koh_v1), 3
clustering_summary_filteredHVG10_SimKumar8hard_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredHVG10_Trapnell_v1 (clustering_summary_filteredExpr10_Koh_v1), 3
clustering_summary_filteredHVG10_Trapnell_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredHVG10_TrapnellTCC_v1 (clustering_summary_filteredExpr10_Koh_v1), 3
clustering_summary_filteredHVG10_TrapnellTCC_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredHVG10_Zhengmix4eq_v1 (clustering_summary_filteredExpr10_Koh_v1), 3
clustering_summary_filteredHVG10_Zhengmix4eq_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredHVG10_Zhengmix4uneq_v1 (clustering_summary_filteredExpr10_Koh_v1), 3
clustering_summary_filteredHVG10_Zhengmix4uneq_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredHVG10_Zhengmix8eq_v1 (clustering_summary_filteredExpr10_Koh_v1), 3
clustering_summary_filteredHVG10_Zhengmix8eq_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredM3Drop10_Koh_v1 (clustering_summary_filteredExpr10_Koh_v1), 3
clustering_summary_filteredM3Drop10_Koh_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredM3Drop10_KohTCC_v1 (clustering_summary_filteredExpr10_Koh_v1), 3
clustering_summary_filteredM3Drop10_KohTCC_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredM3Drop10_Kumar_v1 (clustering_summary_filteredExpr10_Koh_v1), 3

clustering_summary_filteredM3Drop10_Kumar_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredM3Drop10_KumarTCC_v1 (clustering_summary_filteredExpr10_Koh_v1), 3
clustering_summary_filteredM3Drop10_KumarTCC_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredM3Drop10_SimKumar4easy_v1 (clustering_summary_filteredExpr10_Koh_v1), 3
clustering_summary_filteredM3Drop10_SimKumar4easy_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredM3Drop10_SimKumar4hard_v1 (clustering_summary_filteredExpr10_Koh_v1), 3
clustering_summary_filteredM3Drop10_SimKumar4hard_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredM3Drop10_SimKumar8hard_v1 (clustering_summary_filteredExpr10_Koh_v1), 3
clustering_summary_filteredM3Drop10_SimKumar8hard_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredM3Drop10_Trapnell_v1 (clustering_summary_filteredExpr10_Koh_v1), 3
clustering_summary_filteredM3Drop10_Trapnell_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredM3Drop10_TrapnellTCC_v1 (clustering_summary_filteredExpr10_Koh_v1), 3
clustering_summary_filteredM3Drop10_TrapnellTCC_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredM3Drop10_Zhengmix4eq_v1 (clustering_summary_filteredExpr10_Koh_v1), 3
clustering_summary_filteredM3Drop10_Zhengmix4eq_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredM3Drop10_Zhengmix4uneq_v1 (clustering_summary_filteredExpr10_Koh_v1), 3
clustering_summary_filteredM3Drop10_Zhengmix4uneq_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
clustering_summary_filteredM3Drop10_Zhengmix8eq_v1 (clustering_summary_filteredExpr10_Koh_v1), 3
clustering_summary_filteredM3Drop10_Zhengmix8eq_v2 (clustering_summary_filteredExpr10_Koh_v2), 4
duo_clustering_all_parameter_settings_v1, 5
duo_clustering_all_parameter_settings_v2, 6
DuoClustering2018, 5
plot_entropy, 7
plot_k_diff, 8
plot_performance, 8
plot_stability, 9
plot_timing, 10
sce_filteredExpr10_Koh (sce_full_Koh), 10
sce_filteredExpr10_KohTCC (sce_full_Koh), 10
sce_filteredExpr10_Kumar (sce_full_Kumar), 13
sce_filteredExpr10_KumarTCC (sce_full_Kumar), 13
sce_filteredExpr10_SimKumar4easy (sce_full_SimKumar4easy), 15
sce_filteredExpr10_SimKumar4hard (sce_full_SimKumar4easy), 15
sce_filteredExpr10_SimKumar8hard (sce_full_SimKumar4easy), 15
sce_filteredExpr10_Trapnell (sce_full_Trapnell), 17
sce_filteredExpr10_TrapnellTCC (sce_full_Trapnell), 17
sce_filteredExpr10_Zhengmix4eq (sce_full_Zhengmix4eq), 19
sce_filteredExpr10_Zhengmix4uneq (sce_full_Zhengmix4eq), 19
sce_filteredExpr10_Zhengmix8eq (sce_full_Zhengmix4eq), 19
sce_filteredHVG10_Koh (sce_full_Koh), 10
sce_filteredHVG10_KohTCC (sce_full_Koh), 10

sce_filteredHVG10_Kumar
(sce_full_Kumar), 13
sce_filteredHVG10_KumarTCC
(sce_full_Kumar), 13
sce_filteredHVG10_SimKumar4easy
(sce_full_SimKumar4easy), 15
sce_filteredHVG10_SimKumar4hard
(sce_full_SimKumar4easy), 15
sce_filteredHVG10_SimKumar8hard
(sce_full_SimKumar4easy), 15
sce_filteredHVG10_Trapnell
(sce_full_Trapnell), 17
sce_filteredHVG10_TrapnellTCC
(sce_full_Trapnell), 17
sce_filteredHVG10_Zhengmix4eq
(sce_full_Zhengmix4eq), 19
sce_filteredHVG10_Zhengmix4uneq
(sce_full_Zhengmix4eq), 19
sce_filteredHVG10_Zhengmix8eq
(sce_full_Zhengmix4eq), 19
sce_filteredM3Drop10_Koh
(sce_full_Koh), 10
sce_filteredM3Drop10_KohTCC
(sce_full_Koh), 10
sce_filteredM3Drop10_Kumar
(sce_full_Kumar), 13
sce_filteredM3Drop10_KumarTCC
(sce_full_Kumar), 13
sce_filteredM3Drop10_SimKumar4easy
(sce_full_SimKumar4easy), 15
sce_filteredM3Drop10_SimKumar4hard
(sce_full_SimKumar4easy), 15
sce_filteredM3Drop10_SimKumar8hard
(sce_full_SimKumar4easy), 15
sce_filteredM3Drop10_Trapnell
(sce_full_Trapnell), 17
sce_filteredM3Drop10_TrapnellTCC
(sce_full_Trapnell), 17
sce_filteredM3Drop10_Zhengmix4eq
(sce_full_Zhengmix4eq), 19
sce_filteredM3Drop10_Zhengmix4uneq
(sce_full_Zhengmix4eq), 19
sce_filteredM3Drop10_Zhengmix8eq
(sce_full_Zhengmix4eq), 19
sce_full_Koh, 10
sce_full_KohTCC (sce_full_Koh), 10
sce_full_Kumar, 13
sce_full_KumarTCC (sce_full_Kumar), 13
sce_full_SimKumar4easy, 15
sce_full_SimKumar4hard
(sce_full_SimKumar4easy), 15
sce_full_SimKumar8hard
(sce_full_SimKumar4easy), 15
sce_full_Trapnell, 17
sce_full_TrapnellTCC
(sce_full_Trapnell), 17
sce_full_Zhengmix4eq, 19
sce_full_Zhengmix4uneq
(sce_full_Zhengmix4eq), 19
sce_full_Zhengmix8eq
(sce_full_Zhengmix4eq), 19
shannon_entropy, 21