# Package 'miQC'

June 13, 2021

**Type** Package

**Title** Flexible, probabilistic metrics for quality control of scRNA-seq data

**Version** 1.0.0

**Description** Single-cell RNA-sequencing (scRNA-seq) has made it possible to profile gene expression in tissues at high resolution. An important preprocessing step prior to performing downstream analyses is to identify and remove cells with poor or degraded sample quality using quality control (QC) metrics. Two widely used QC metrics to identify a 'low-quality' cell are (i) if the cell includes a high proportion of reads that map to mitochondrial DNA encoded genes (mtDNA) and (ii) if a small number of genes are detected. miQC is data-driven QC metric that jointly models both the proportion of reads mapping to mtDNA and the number of detected genes with mixture models in a probabilistic framework to predict the low-quality cells in a given dataset.

**URL** https://github.com/greenelab/miQC

**BugReports** https://github.com/greenelab/miQC/issues

**License** BSD_3_clause + file LICENSE

**Imports** SingleCellExperiment, flexmix, ggplot2, splines, BiocParallel

**Suggests** scRNAseq, scater, biomaRt, BiocStyle, knitr, rmarkdown

**biocViews** SingleCell, QualityControl, GeneExpression, Preprocessing, Sequencing

**VignetteBuilder** knitr

**Encoding** UTF-8

**RoxygenNote** 7.1.1.9000

**git_url** https://git.bioconductor.org/packages/miQC

**git_branch** RELEASE_3_13

**git_last_commit** d69caf1

**git_last_commit_date** 2021-05-19

**Date/Publication** 2021-06-13

**Author** Ariel Hippen [aut, cre], Stephanie Hicks [aut]

**Maintainer** Ariel Hippen <ariel.hippen@gmail.com>

# R topics documented:



---

`filterCells` *filterCells*

---

## Description

Find those cells probabilistically determined to be compromised by the mixture model and remove them from the dataset.

## Usage

```
filterCells(sce, model = NULL, posterior_cutoff = 0.75, verbose = TRUE)
```

## Arguments

| | |
|---|---|
| `sce` | (SingleCellExperiment) Input data object. |
| `model` | (flexmix) Output of mixtureModel function, which should be explicitly called first to ensure stability of model parameters. Default = NULL. |
| `posterior_cutoff` | (numeric) The posterior probability of a cell being part of the compromised distribution, a number between 0 and 1. Any cells below the appointed cutoff will be marked to keep. Default = 0.75 |
| `verbose` | (boolean) Whether to report how many cells (columns) are being removed from the SingleCellExperiment object. Default = TRUE |

## Value

Returns a SingleCellExperiment object, the same as the input except with a new column in colData, prob_compromised, and all cells with greater than the set posterior probability removed from the dataset.

## Examples

```
library(scRNAseq)
library(scater)
library(BiocParallel)
sce <- ZeiselBrainData()
mt_genes <- grepl("^mt-",  rownames(sce))
feature_ctrls <- list(mito = rownames(sce)[mt_genes])
```

```
sce <- addPerCellQC(sce, subsets = feature_ctrls, BPPARAM = MulticoreParam())
model <- mixtureModel(sce)
sce <- filterCells(sce, model)
```

---

`mixtureModel` *mixtureModel*

---

## Description

Function to fit a two-distribution mixture model on a SingleCellExperiment object.

## Usage

```
mixtureModel(sce, model_type = "linear")
```

## Arguments

| | |
|---|---|
| `sce` | (SingleCellExperiment) Input data object. |
| `model_type` | (character) What type of model to generate. A linear mixture model ("linear") is recommended, but currently b-spline ("spline") and two-degree polynomial ("polynomial") are also supported Default = "linear". |

## Value

Returns a flexmix object with mixture model parameters, which is used to calculate posterior probability for each cell being compromised and make final filtering decisions.

## Examples

```
library(scRNAseq)
library(scater)
library(BiocParallel)
sce <- ZeiselBrainData()
mt_genes <- grepl("^mt-",  rownames(sce))
feature_ctrls <- list(mito = rownames(sce)[mt_genes])
sce <- addPerCellQC(sce, subsets = feature_ctrls, BPPARAM = MulticoreParam())
model <- mixtureModel(sce)
```

plotFiltering *plotFiltering*

## Description

Function to plot which cells will be kept and removed given their posterior probability of belonging to the compromised distribution.

## Usage

```
plotFiltering(
  sce,
  model = NULL,
  posterior_cutoff = 0.75,
  palette = c("#999999", "#E69F00"),
  detected = "detected",
  subsets_mito_percent = "subsets_mito_percent"
)
```

## Arguments

| | |
|---|---|
| sce | (SingleCellExperiment) Input data object. |
| model | (flexmix) Output of mixtureModel function, which should be explicitly called first to ensure stability of model parameters. Default = NULL. |
| posterior_cutoff | (numeric) The posterior probability of a cell being part of the compromised distribution, a number between 0 and 1. Any cells below the appointed cutoff will be marked to keep. Default = 0.75 |
| palette | (character) Color palette. A vector of length two containing custom colors. Default = c("#999999", "#E69F00"). |
| detected | (character) Column name in sce giving the number of unique genes detected per cell. This name is inherited by default from scater's addPerCellQC() function. |
| subsets_mito_percent | (character) Column name in sce giving the percent of reads mapping to mitochondrial genes. This name is inherited from scater's addPerCellQC() function, provided the subset "mito" with names of all mitochondrial genes is passed in. See examples for details. |

## Value

Returns a ggplot object. Additional plot elements can be added as ggplot elements (e.g. title, customized formatting, etc).

## Examples

```
library(scRNAseq)
library(scater)
library(BiocParallel)
sce <- ZeiselBrainData()
mt_genes <- grepl("^mt-",  rownames(sce))
feature_ctrls <- list(mito = rownames(sce)[mt_genes])
sce <- addPerCellQC(sce, subsets = feature_ctrls, BPPARAM = MulticoreParam())
model <- mixtureModel(sce)
plotFiltering(sce, model)
```

---

`plotMetrics` *plotMetrics*

---

## Description

A function to plot the QC parameters used for a miQC model, number of unique genes expressed and percent mitochondrial reads. This function can be run before calling mixtureModel() to assess if miQC is appropriate given the data distribution. See vignette for examples of cases where miQC is and isn't a good choice for filtering.

## Usage

```
plotMetrics(
  sce,
  model = NULL,
  detected = "detected",
  subsets_mito_percent = "subsets_mito_percent",
  palette = "#33ADFF"
)
```

## Arguments

| | |
|---|---|
| `sce` | (SingleCellExperiment) Input data object. |
| `detected` | (character) Column name in sce giving the number of unique genes detected per cell. This name is inherited by default from scater's addPerCellQC() function. |
| `subsets_mito_percent` | (character) Column name in sce giving the percent of reads mapping to mitochondrial genes. This name is inherited from scater's addPerCellQC() function, provided the subset "mito" with names of all mitochondrial genes is passed in. See examples for details. |
| `palette` | (character) Specifies the color to plot cells as. Default is "#33ADFF". |

## Value

Returns a ggplot object. Additional plot elements can be added as ggplot elements (e.g. title, customized formatting, etc).

## Examples

```
library(scRNAseq)
library(scater)
library(BiocParallel)
sce <- ZeiselBrainData()
mt_genes <- grepl("^mt-",  rownames(sce))
feature_ctrls <- list(mito = rownames(sce)[mt_genes])
sce <- addPerCellQC(sce, subsets = feature_ctrls, BPPARAM = MulticoreParam())
plotMetrics(sce)
```

---

plotModel *plotModel*

---

## Description

Function to plot quality characteristics of cells in dataset, parameters of compromised and intact distributions, and posterior probability of each cell belonging to the compromised distribution.

## Usage

```
plotModel(
  sce,
  model = NULL,
  detected = "detected",
  subsets_mito_percent = "subsets_mito_percent"
)
```

## Arguments

| | |
|---|---|
| sce | (SingleCellExperiment) Input data object. |
| model | (flexmix) Output of mixtureModel function, which should be explicitly called first to ensure stability of model parameters. Default = NULL. |
| detected | (character) Column name in sce giving the number of unique genes detected per cell. This name is inherited by default from scater's addPerCellQC() function. |
| subsets_mito_percent | (character) Column name in sce giving the percent of reads mapping to mitochondrial genes. This name is inherited from scater's addPerCellQC() function, provided the subset "mito" with names of all mitochondrial genes is passed in. See examples for details. |

## Value

Returns a ggplot object. Additional plot elements can be added as ggplot elements (e.g. title, customized formatting, etc).

## Examples

```
library(scRNAseq)
library(scater)
library(BiocParallel)
sce <- ZeiselBrainData()
mt_genes <- grepl("^mt-",  rownames(sce))
feature_ctrls <- list(mito = rownames(sce)[mt_genes])
sce <- addPerCellQC(sce, subsets = feature_ctrls, BPPARAM = MulticoreParam())
model <- mixtureModel(sce)
plotModel(sce, model)
```

# Index