

# Package ‘refineMIBC’

April 10, 2021

**Title** What the Package Does (One Line, Title Case)

**Version** 0.0.0.9000

**Description** What the package does (one paragraph).

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.1.1

**Suggests** testthat,  
knitr,  
rmarkdown

**VignetteBuilder** knitr

**Imports** CMScaller,  
varSelRF

## R topics documented:

refineBasal . . . . .	1
refineMIBC . . . . .	2
rfClassifier . . . . .	3
templates . . . . .	3
<b>Index</b>	<b>4</b>

---

refineBasal	<i>Refine basal-like muscle invasive bladder cancer</i>
-------------	---

---

## Description

Refine pre-identified basal-like muscle invasive bladder cancer into either basal-inflamed or basal-noninflamed subtype based on a random forest classifier.

**Usage**

```
refineBasal(
  expr = NULL,
  isBasal = NULL,
  res.path = getwd(),
  res.name = "REFINED_MIBC_BASAL"
)
```

**Arguments**

<code>expr</code>	A numeric expression matrix with row features and sample columns.
<code>isBasal</code>	A logic vector to indicate if the sample is basal-like or not. NULL by default and all samples will be assumed as basal-like.
<code>res.path</code>	A string value to indicate the path for saving the prediction result.
<code>res.name</code>	A string value to indicate the name of the output prediction table.

**Value**

A data.frame storing the random forest prediction results, including `samID` for sample name, `prob` for probability as basal-inflamed, and `basal` for final classifications using 0.5 as cutoff.

---

refineMIBC	<i>Refine muscle-invasive bladder cancer</i>
------------	--

---

**Description**

Refine muscle-invasive bladder cancer into four subtypes based on transcriptome profile by nearest template prediction algorithm.

**Usage**

```
refineMIBC(
  expr = NULL,
  scaleFlag = TRUE,
  centerFlag = TRUE,
  distance = "cosine",
  seed = 123456,
  doPlot = FALSE,
  fig.path = getwd(),
  fig.name = "NTP_PREDICTED_HEATMAP",
  res.path = getwd(),
  res.name = "PREDICTED_MIBC_SUBTYPE"
)
```

**Arguments**

<code>expr</code>	A numeric expression matrix with row features and sample columns.
<code>scaleFlag</code>	A logic value to indicate if the expression data should be further scaled. TRUE by default.

centerFlag	A logic value to indicate if the expression data should be further centered. TRUE by default.
distance	A string value to indicate the distance measurement. Allowed values contain c('cosine', 'pearson', 'spearman', 'kendall'); 'cosine' by default.
seed	An integer value for p-value reproducibility; 123456 by default.
doPlot	A logic value to indicate whether to produce prediction heatmap; FALSE by default.
fig.path	A string value to indicate the output path for storing the nearest template prediction heatmap.
fig.name	A string value to indicate the name of the nearest template prediction heatmap.
res.path	A string value to indicate the path for saving the prediction result.
res.name	A string value to indicate the name of the output prediction table.

### Value

A figure of predictive heatmap by NTP (.pdf) and a data.frame storing the results of nearest template prediction (see [ntp](#)).

### References

Hoshida, Y. (2010). Nearest Template Prediction: A Single-Sample-Based Flexible Class Prediction with Confidence Assessment. PLoS ONE 5, e15543.

---

rfClassifier	<i>Random forest classifier for refining basal-like MIBC</i>
--------------	--

---

### Format

An object of class varSelRF  
contains "rfClassifier" an object of class varSelRF and can be used for external prediction (see [varSelRF](#))

rfClassifier

An R object derived from varSelRF.

datasets

---

templates	<i>Template for classifying MIBC using NTP</i>
-----------	--

---

### Format

A data frame with 120 rows (30 markers for each subtype) and 3 variables.  
contains "templates" the subtype-specific up-regulated markers for MIBC.

templates

A data frame storing the subtype-specific markers for MIBC.

datasets

# Index

ntp, [3](#)

refineBasal, [1](#)

refineMIBC, [2](#)

rfClassifier, [3](#)

templates, [3](#)

varSelRF, [3](#)