

Package ‘CancerInSilico’

October 14, 2021

Type Package

Title An R interface for computational modeling of tumor progression

Version 2.12.0

Date 2018-04-06

Author Thomas D. Sherman,
Raymond Cheng,
Elana J. Fertig

Maintainer Thomas D. Sherman <tomsherman159@gmail.com>,
Elana J. Fertig <ejfertig@jhmi.edu>

Description The CancerInSilico package provides an R interface for running mathematical models of tumor progression and generating gene expression data from the results. This package has the underlying models implemented in C++ and the output and analysis features implemented in R.

License GPL-2

Depends R (>= 3.4), Rcpp

Imports methods, utils, graphics, stats

LinkingTo Rcpp, BH

Suggests testthat, knitr, rmarkdown, BiocStyle, Rtsne, viridis, rgl,
gplots

VignetteBuilder knitr

biocViews ImmunoOncology, MathematicalBiology, SystemsBiology,
CellBiology, BiomedicalInformatics, GeneExpression, RNASeq,
SingleCell

RoxygenNote 6.1.1

Encoding UTF-8

Collate 'CellSimulation.R' 'GeneExpressionSimulation.R'
'RcppExports.R' 'class-CellModel.R' 'class-CellType.R'
'class-OffLatticeModel.R' 'class-DrasdoHohmeModel.R'
'class-Drug.R' 'class-GeneExpressionParams.R' 'class-Pathway.R'
'documentation.R'

git_url <https://git.bioconductor.org/packages/CancerInSilico>

git_branch RELEASE_3_13

git_last_commit 1336eb2

git_last_commit_date 2021-05-19

Date/Publication 2021-10-14

R topics documented:

calibratePathway	3
CancerInSilico	3
CellModel-class	4
cellSummary	4
CellType-class	5
checkDataSet	5
DrasdoHohmeModel-class	6
Drug-class	6
GeneExpressionParams-class	6
getAxisAngle	7
getAxisLength	8
getCellDistance	8
getCellPhase	9
getCellType	10
getCoordinates	10
getCycleLength	11
getDensity	12
getLocalDensity	12
getNumberOfCells	13
getRadius	14
getTrialAcceptRate	14
initialize,CellModel-method	15
initialize,DrasdoHohmeModel-method	16
initialize,OffLatticeModel-method	17
inSilicoCellModel	17
inSilicoGeneExpression	18
inSilicoPathways	19
interactivePlot	19
modCellTypes	20
modDefault	20
modDrugs	20
modHighDensity	21
modLargeRun	21
modLongRun	21
OffLatticeModel-class	21
Pathway-class	22
plotCells	23
pwContactInhibition	23
pwGrowth	24
pwMitosis	24

pwySPhase	24
referenceGeneExpression	24
run	25

Index 26

calibratePathway *calibrate pathway with data*

Description

sets the min and max values for each gene in a pathway based on a data set

Usage

```
calibratePathway(pathway, dataSet)
```

Arguments

pathway	a 'Pathway' object
dataSet	reference data set

Value

pathway with min/max values for expression based on data set

CancerInSilico *CancerInSilico*

Description

Package:	CancerInSilico
Type:	Package
Version:	1.99.0
Date:	2017-06-24
License:	LGPL

Author(s)

Maintainer: Elana J. Fertig <ejfertig@jhmi.edu>, Thomas D. Sherman <tsherma4@jhu.edu>

CellModel-class	<i>CellModel</i>
-----------------	------------------

Description

The top-level CellModel class. All other cell model classes inherit from this in some way

Slots

`cells` A list object that describes the state of the cells at each time. The state representation depends on the type of model run, and is accessed by the function designed for each model type.

`initialNum` number of cells at time 0

`runTime` number of model hours to run the simulation

`density` initial density of cells

`boundary` keep cells within circular boundary

`syncCycles` start all cells in the beginning of interphase

`randSeed` random seed used for both R and C++ functions

`outputIncrement` how often simulation info is displayed

`recordIncrement` how often cell info is recorded (controls size of resulting CellModel object)

`timeIncrement` controls how fine the model timestep is

`cellTypes` list of CellType objects used in the model

`cellTypeInitFreq` initial frequency of cell types among cells

`drugs` list of Drug objects used in the model

cellSummary	<i>summary of cell model at a given time</i>
-------------	--

Description

summary of cell model at a given time

Usage

```
cellSummary(model, time)
```

```
## S4 method for signature 'CellModel'
cellSummary(model, time)
```

Arguments

`model` cell model object

`time` hour of the model to query

Value

string containing summary of model

Examples

```
data(SampleModels)
cellSummary(modDefault, modDefault@runTime)
```

CellType-class	<i>CellType</i>
----------------	-----------------

Description

The properties of a cell type

Slots

name the name of the cell type
 size the relative size (volume) of the cell
 minCycle minimum possible cell cycle length
 cycleLength function that returns sample from distribution of cycle lengths

checkDataSet	<i>verify gene expression data set is valid for this package</i>
--------------	--

Description

checks a data set before it is used to calibrate the pathway values for min/max expression

Usage

```
checkDataSet(dataSet, genes)
```

Arguments

dataSet	matrix of gene expression data where row names are genes
genes	names of all genes being simulated

Value

no value is return, but errors/warnings are thrown related to potential problems in the data set

Examples

```
data(referenceGeneExpression)
```

DrasdoHohmeModel-class

DrasdoHohmeModel

Description

Implementation of an off-lattice cell-based model based on the work in Drasdo, Hohme (2003)

Slots

nG number of monte carlo steps between each growth trial

epsilon constant that controls the probability trails are accepted

delta controls distance over which short range interactions occur

Drug-class

Drug

Description

describes the properties of a drug

Slots

name name of drug

timeAdded the time at which this drug is added to the simulation

cycleLengthEffect effect this drug has on cell cycle length

GeneExpressionParams-class

GeneExpressionParams

Description

Parameters for simulating gene expression

Slots

sampleFreq how often to generate data
 RNAseq generate RNA-seq data
 singleCell generate single cell data
 nCells number of cells to sample at each time point
 nDummyGenes number of dummy genes
 dummyDist function to determine expression of dummy genes
 combineFUN function used to combine gene expression data
 randSeed random seed
 perError error for normal error model
 bcvCommon error for voom error model
 bcvDF degrees of freedom for voom error model
 dropoutPresent whether to simulate dropout in single cell data
 dropoutMid parameter for dropout distribution
 dropoutShape parameter for dropout distribution

getAxisAngle	<i>get cell axis angle at a given time</i>
--------------	--

Description

get cell axis angle at a given time

Usage

```

getAxisAngle(model, time, cell)

## S4 method for signature 'OffLatticeModel'
getAxisAngle(model, time, cell)

```

Arguments

model	cell model object
time	hour of the model to query
cell	id of cell to query

Value

axis angle

Examples

```

data(SampleModels)
getAxisAngle(modDefault, modDefault@runTime, 1)

```

getAxisLength	<i>get cell axis length at a given time</i>
---------------	---

Description

get cell axis length at a given time

Usage

```
getAxisLength(model, time, cell)

## S4 method for signature 'OffLatticeModel'
getAxisLength(model, time, cell)
```

Arguments

model	cell model object
time	hour of the model to query
cell	id of cell to query

Value

axis length

Examples

```
data(SampleModels)
getAxisLength(modDefault, modDefault@runTime, 1)
```

getCellDistance	<i>get distance between two cells</i>
-----------------	---------------------------------------

Description

get distance between two cells

Usage

```
getCellDistance(model, time, cellA, cellB)

## S4 method for signature 'OffLatticeModel'
getCellDistance(model, time, cellA, cellB)
```

Arguments

model	cell model object
time	hour of the model to query
cellA	id of cell to query
cellB	id of cell to query

Value

distance between cellA and cellB

Examples

```
data(SampleModels)
getCellDistance(modDefault, modDefault@runTime, 1, 2)
```

<i>getCellPhase</i>	<i>get phase of a cell at a given time</i>
---------------------	--

Description

get phase of a cell at a given time

Usage

```
getCellPhase(model, time, cell)

## S4 method for signature 'OffLatticeModel'
getCellPhase(model, time, cell)
```

Arguments

model	cell model object
time	hour of the model to query
cell	id of cell to query

Value

cell phase

Examples

```
data(SampleModels)
getCellPhase(modDefault, modDefault@runTime, 1)
```

getCellType	<i>get type of a cell at a given time</i>
-------------	---

Description

get type of a cell at a given time

Usage

```
getCellType(model, time, cell)

## S4 method for signature 'OffLatticeModel'
getCellType(model, time, cell)
```

Arguments

model	cell model object
time	hour of the model to query
cell	id of cell to query

Value

cell type

Examples

```
data(SampleModels)
getCellType(modDefault, modDefault@runTime, 1)
```

getCoordinates	<i>get coordinates of a cell at a given time</i>
----------------	--

Description

get coordinates of a cell at a given time

Usage

```
getCoordinates(model, time, cell)

## S4 method for signature 'OffLatticeModel'
getCoordinates(model, time, cell)
```

Arguments

model	cell model object
time	hour of the model to query
cell	id of cell to query

Value

pair of (x,y) coordinates

Examples

```
data(SampleModels)
getCoordinates(modDefault, modDefault@runTime, 1)
```

<code>getCycleLength</code>	<i>get cycle length of a cell at a given time</i>
-----------------------------	---

Description

get cycle length of a cell at a given time

Usage

```
getCycleLength(model, time, cell)

## S4 method for signature 'OffLatticeModel'
getCycleLength(model, time, cell)
```

Arguments

model	cell model object
time	hour of the model to query
cell	id of cell to query

Value

cycle length in hours

Examples

```
data(SampleModels)
getCycleLength(modDefault, modDefault@runTime, 1)
```

getDensity	<i>get density of the cell population at a given time</i>
------------	---

Description

get density of the cell population at a given time

Usage

```
getDensity(model, time)
```

```
## S4 method for signature 'OffLatticeModel'
getDensity(model, time)
```

Arguments

model	cell model object
time	hour of the model to query

Value

density

Examples

```
data(SampleModels)
getDensity(modDefault, modDefault@runTime)
```

getLocalDensity	<i>get neighborhood density around a cell at a given time</i>
-----------------	---

Description

get neighborhood density around a cell at a given time

Usage

```
getLocalDensity(model, time, cell, radius)
```

```
## S4 method for signature 'OffLatticeModel'
getLocalDensity(model, time, cell, radius)
```

Arguments

model	cell model object
time	hour of the model to query
cell	id of cell to query
radius	distance to search for neighboring cells

Value

density

Examples

```
data(SampleModels)
getLocalDensity(modDefault, modDefault@runTime, 1, 3.3)
```

`getNumberOfCells` *get number of cells in the model at a given time*

Description

get number of cells in the model at a given time

Usage

```
getNumberOfCells(model, time)

## S4 method for signature 'OffLatticeModel'
getNumberOfCells(model, time)
```

Arguments

model	cell model object
time	hour of the model to query

Value

number of cells

Examples

```
data(SampleModels)
getNumberOfCells(modDefault, modDefault@runTime)
```

getRadius	<i>get cell radius at a given time</i>
-----------	--

Description

get cell radius at a given time

Usage

```
getRadius(model, time, cell)

## S4 method for signature 'OffLatticeModel'
getRadius(model, time, cell)
```

Arguments

model	cell model object
time	hour of the model to query
cell	id of cell to query

Value

radius of cell

Examples

```
data(SampleModels)
getRadius(modDefault, modDefault@runTime, 1)
```

getTrialAcceptRate	<i>get rate of growth acceptance of a given cell at a given time</i>
--------------------	--

Description

get rate of growth acceptance of a given cell at a given time

Usage

```
getTrialAcceptRate(model, time, cell)

## S4 method for signature 'OffLatticeModel'
getTrialAcceptRate(model, time, cell)
```

Arguments

model	cell model object
time	hour of the model to query
cell	id of cell to query

Value

acceptance rate

Examples

```
data(SampleModels)
getTrialAcceptRate(modDefault, modDefault@runTime, 1)
```

```
initialize,CellModel-method
```

Constructor for CellModel

Description

Constructor for CellModel

Usage

```
## S4 method for signature 'CellModel'
initialize(.Object, initialNum, runTime, density,
  boundary = 1, syncCycles = FALSE, randSeed = 0,
  outputIncrement = 4, recordIncrement = 0.1, timeIncrement = 0.001,
  cellTypes = c(new("CellType", name = "DEFAULT")),
  cellTypeInitFreq = c(1), drugs = list(), ...)
```

Arguments

.Object	CellModel object
initialNum	initial number of cells
runTime	run time of the model in hours
density	initial density of the cell population
boundary	impose a physical boundary on the cells
syncCycles	synchronization all cells to the same point in the cycle
randSeed	random seed
outputIncrement	how often (model hours) to print simulation status
recordIncrement	how often (model hours) to record cell information

timeIncrement	internal time step (model hours) used by the model
cellTypes	list of CellType objects
cellTypeInitFreq	initial proportions of all cell types
drugs	list of Drug objects
...	model specific parameters

Value

initialized cell model object

initialize,DrasdoHohmeModel-method
DrasdoHohmeModel Constructor

Description

DrasdoHohmeModel Constructor

Usage

```
## S4 method for signature 'DrasdoHohmeModel'
initialize(.Object, nG = 28, epsilon = 10,
  delta = 0.2, ...)
```

Arguments

.Object	DrasdoHohmeModel object
nG	number of trials per growth trial
epsilon	model parameter
delta	small scale interaction distance
...	extra parameters

Value

initialized cell model

```
initialize,OffLatticeModel-method
      Off-Lattice Model Constructor
```

Description

Off-Lattice Model Constructor

Usage

```
## S4 method for signature 'OffLatticeModel'
initialize(.Object, maxTranslation = 0.1,
          maxRotation = 0.3, ...)
```

Arguments

.Object	OffLatticeModel object
maxTranslation	maximum movement of cell
maxRotation	maximim rotation of mitosis cell
...	model specific parameters

Value

initialized cell model

```
inSilicoCellModel      Simulates Cell Model
```

Description

Simulates Cell Model

Usage

```
inSilicoCellModel(initialNum, runTime, density,
                  modelType = "DrasdoHohme", ...)
```

Arguments

initialNum	how many cells initially
runTime	how long the simulation runs in real cellular time (hours)
density	initial density of cell population
modelType	the name of the cell-based model to use
...	model specific parameters (depends on modelType)

Details

This function provides a centralized R interface to run c++ code for cell-based models implemented in this package. Standard parameters, as well as model-specific parameters, are passed in to this function along with a model name. This function then runs the model and returns a CellModel object containing all of the information from the model. This object can then be accessed with various functions designed to interact with the class. To see a list of available functions, there is a show() command implemented for CellModel objects.

Value

A CellModel containing all info from the model run

Examples

```
inSilicoCellModel(initialNum=1, runTime=8, density=0.1)
```

inSilicoGeneExpression

simulate gene expression data

Description

simulate gene expression data for a set of pathways, using the behavior of a CellModel as the basis for the simulation

Usage

```
inSilicoGeneExpression(model, pathways,  
  params = new("GeneExpressionParams"))
```

Arguments

model	a CellModel object
pathways	list of genes pathways
params	GeneExpressionParams object

Value

list of pathway activity and gene expression

inSilicoPathways	<i>This data contains the list of gene targets of modeled signaling pathways</i>
------------------	--

Description

This data contains the list of gene targets of modeled signaling pathways

Usage

```
inSilicoPathways
```

Details

List of gene targets for GtoM, GtoS, and Growth factor pathways modeled in CancerInSilico. GtoM represents genes associated with the G to M checkpoint of the cell cycle collected from PID pathways PI3KCI_AKT, TNF, TGFBR, and RB1 in MSigDB v5.1. GtoS represented genes associated with the G to S checkpoint of the cell cycle collected from E2F target genes in PID and hallmark genesets from MSigDb v5.1 and experimentally validated targets in TRANSFAC professional 2014. Growth represents the targets of transcription factors resulting from EGFR activation (STAT1, STAT3, STAT5, Elk-1, Myc, NF-kappaB, and RelA) from the TRANSFAC professional database 2014 as described in Fertig et al, 2012.

Value

List of gene targets for GtoM, GtoS, and Growth factor pathways.

References

Subramanian, Tamayo, et al, PNAS, 102:155545, 2005; Matys et al, Nuc Ac Res, 34:D108, 2006; and Fertig et al, BMC Genomics, 13:160, 2012.

interactivePlot	<i>plot the cell population and interactively scroll through time points</i>
-----------------	--

Description

plot the cell population and interactively scroll through time points

Usage

```
interactivePlot(model)

## S4 method for signature 'CellModel'
interactivePlot(model)
```

Arguments

model cell model object

Value

plot

modCellTypes *sample cell model*

Description

sample cell model

Usage

modCellTypes

modDefault *sample cell model*

Description

sample cell model

Usage

modDefault

modDrugs *sample cell model*

Description

sample cell model

Usage

modDrugs

modHighDensity	<i>sample cell model</i>
----------------	--------------------------

Description

sample cell model

Usage

modHighDensity

modLargeRun	<i>sample cell model</i>
-------------	--------------------------

Description

sample cell model

Usage

modLargeRun

modLongRun	<i>sample cell model</i>
------------	--------------------------

Description

sample cell model

Usage

modLongRun

OffLatticeModel-class	<i>OffLatticeModel</i>
-----------------------	------------------------

Description

General description of an off-lattice cell-based model. not quite a full implementation, but contains much of the necessary structure for models of this type

Slots

maxTranslation the largest distance the center of a cell can move

maxRotation the largest angle a cell can rotate

 Pathway-class

Pathway Class

Description

Describes the basic properties of a gene pathway

Usage

```
## S4 method for signature 'Pathway'
initialize(.Object, ...)
```

Arguments

.Object	Pathway object
...	extra arguments for constructor

Details

expressionScale is a function that accepts three arguments: model, cell, and time. It should return a number in [0,1] that describes how active the genes are in this pathway for a given cell in the model at a given time. In bulk data, the pathway activity is averaged and transformed by $1 / (1 + \exp(-k * (x - M)))$ where $k = \text{transformSlope}$ and $M = \text{transformMidpoint}$. The scale determines how expressed genes in this pathway are. i.e. and scale of 0 means all genes will have minExpression value and a scale of 1 means all genes will have maxExpression value. In between these values the gene expression scales linearly.

Value

constructed object

Slots

genes names of genes in the pathway
 expressionScale function describing how this pathway is affected by the state of the model
 minExpression minimum expression value for each gene (vector)
 maxExpression maximum expression value for each gene (vector)
 transformSlope parameter for transforming bulk data
 transformMidpoint parameter for transforming bulk data

plotCells *plot cell population at a given time*

Description

plot cell population at a given time

Usage

```
plotCells(model, time)

## S4 method for signature 'OffLatticeModel'
plotCells(model, time)
```

Arguments

model	cell model object
time	hour of the model to query

Value

plot

Examples

```
data(SampleModels)
plotCells(modDefault, modDefault@runTime)
```

pwyContactInhibition *sample pathway*

Description

sample pathway

Usage

```
pwyContactInhibition
```

pwyGrowth *sample pathway*

Description

sample pathway

Usage

pwyGrowth

pwyMitosis *sample pathway*

Description

sample pathway

Usage

pwyMitosis

pwySPhase *sample pathway*

Description

sample pathway

Usage

pwySPhase

referenceGeneExpression
gene expression data for genes in sample pathways

Description

gene expression data for genes in sample pathways

Usage

referenceGeneExpression

run	<i>run a cell model</i>
-----	-------------------------

Description

run a cell model

Usage

```
run(model)
```

```
## S4 method for signature 'DrasdoHohmeModel'  
run(model)
```

Arguments

model cell model object

Value

cell model object with simulation info

Examples

```
data(SampleModels)  
run(modDefault)
```

Index

calibratePathway, 3
CancerInSilico, 3
CancerInSilico-package
 (CancerInSilico), 3
CellModel-class, 4
cellSummary, 4
cellSummary, CellModel-method
 (cellSummary), 4
CellType-class, 5
checkDataSet, 5

DrasdoHohmeModel-class, 6
Drug-class, 6

GeneExpressionParams-class, 6
getAxisAngle, 7
getAxisAngle, OffLatticeModel-method
 (getAxisAngle), 7
getAxisLength, 8
getAxisLength, OffLatticeModel-method
 (getAxisLength), 8
getCellDistance, 8
getCellDistance, OffLatticeModel-method
 (getCellDistance), 8
getCellPhase, 9
getCellPhase, OffLatticeModel-method
 (getCellPhase), 9
getCellType, 10
getCellType, OffLatticeModel-method
 (getCellType), 10
getCoordinates, 10
getCoordinates, OffLatticeModel-method
 (getCoordinates), 10
getCycleLength, 11
getCycleLength, OffLatticeModel-method
 (getCycleLength), 11
getDensity, 12
getDensity, OffLatticeModel-method
 (getDensity), 12
getLocalDensity, 12
getLocalDensity, OffLatticeModel-method
 (getLocalDensity), 12
getNumberOfCells, 13
getNumberOfCells, OffLatticeModel-method
 (getNumberOfCells), 13
getRadius, 14
getRadius, OffLatticeModel-method
 (getRadius), 14
getTrialAcceptRate, 14
getTrialAcceptRate, OffLatticeModel-method
 (getTrialAcceptRate), 14

initialize, CellModel-method, 15
initialize, DrasdoHohmeModel-method, 16
initialize, OffLatticeModel-method, 17
initialize, Pathway-method
 (Pathway-class), 22
inSilicoCellModel, 17
inSilicoGeneExpression, 18
inSilicoPathways, 19
interactivePlot, 19
interactivePlot, CellModel-method
 (interactivePlot), 19

modCellTypes, 20
modDefault, 20
modDrugs, 20
modHighDensity, 21
modLargeRun, 21
modLongRun, 21

OffLatticeModel-class, 21

Pathway-class, 22
plotCells, 23
plotCells, OffLatticeModel-method
 (plotCells), 23
pwyContactInhibition, 23
pwyGrowth, 24
pwyMitosis, 24

`pwySPhase`, [24](#)

`referenceGeneExpression`, [24](#)

`run`, [25](#)

`run,DrasdoHohmeModel-method (run)`, [25](#)