Package 'crisprpred'

December 21, 2016

Title CRISPR/Cas9 sgRNA activity predictions

Version 0.9.9

Author Md Khaledur Rahman

Maintainer Md Khaledur Rahman <khaled.cse.07@gmail.com>

Description This package predicts CRISPR/Cas9 sgRNA activity.

Depends R (>= 3.2.0), DAAG, e1071, h2o, gtools, earth

License GPL (>= 2)

LazyData true

RoxygenNote 5.0.1

R topics documented:

countpattern *Illustration of countpattern This function takes sequence and pattern as input and count how many times a particular pattern is present in the sequence.*

Description

Illustration of countpattern This function takes sequence and pattern as input and count how many times a particular pattern is present in the sequence.

Usage

```
countpattern(sequence, pattern)
```

Arguments

sequence	provided as a list of sequences
pattern	a string

Value

a list of integer indicating frequency of pattern.

Examples

```
sequence = list("ABDEFGHABDAACBBDEBGGGHHH", "ABCBDBEBEBBBDBDBFDFDFGGHHEEFFEECCCD")
pattern = "BD"
feat = featurization(sequence, pattern)
feat
```

crisprpred_main *Explanation of crisprpred_main functions*

Description

This function takes full datasetpath and reads data as a R object (data frame), a list of features and a number to denote cross-validation. It also takes other parameters for different algorithms. Then, it performs Machine Learning algorithms and build prediction models. Then it predicts sgRNA activity based on prediction models.

Usage

```
crisprpred_main(datasetpath, featurelist, kfold, iteration4dl, trees,
  learningrate, samplingrate)
```

Arguments

datasetpath	full path of a csv file
featurelist	provided by user as a list of strings
kfold	used in ML-functions for kfold cross-validation
iteration4dl	number of time dataset will be iterated
trees	number of trees in random forest
learningrate	learning rate of deep learner
samplingrate	sampling rate in random forest

Value

None

Examples

```
setwd('..')
#suppose we have a file as '../crisprpred/data-raw/sample.csv' and current directory is set to '../crisprpred'
dir = getwd()
datasetpath = paste0(dir,'/data-raw/sample.csv')
featurelist = c("X30mer", "Percent.Peptide", "Amino.Acid.Cut.position","predictions")
kfoldCross = 2
crisprpred_main(datasetpath, featurelist, kfoldCross, 3, 4, 0.66)
```

---

dplearning *Deep Learning*

Description

This function takes full filepath, a list of learning features, a value for cross-validation, the number of times data set will be iterated and learning rate. Now, it creates a deep learning model using deeplearning function of h2o package and outputs RMSE based on provided dataset. Note that size of dataset should be enough to choose a suitable value for kfold.

Usage

```
dplearning(featurelist, featuredata, leaveonegene = 0, kfold = 10,
  learningrate = 0.6)
```

Arguments

featurelist	a list of features. last name will indicate the value to be predicted.
featuredata	a sample dataset containing all features
leaveonegene	check for leaveonegeneout cross-validation
kfold	a value for cross validation. Default value is 3.
learningrate	a fractional learning rate for deep learner. Default value is 0.6.

Value

spearman correlation

Examples

```
featurelist = c("Percent.Peptide", "Amino.Acid.Cut.position","predictions")
#suppose we have a file as '../crisprpred/data-raw/sample.csv' and current directory is set to '../crisprpred'
dir = getwd()
filepath = paste0(dir,'/data-raw/sample.csv')
data = read.csv(filepath)
dplearning(featurelist,data)
```

`featureformula` *Making Formula for Learning*

Description

This function takes a list of features to make a suitable formula. For example, it creates a formula Y~X1+X2+X3 from a list of features ('X1','X2','X3','Y')

Usage

```
featureformula(featurelist)
```

Arguments

`featurelist`	a list of strings

Value

a formula

Examples

```
featurelist = c('X1', 'X2', 'X3', 'Y')
formula = featureformula(featurelist)
formula
```

`featurization` *Illustration of Featurization*

Description

This function takes dataset and a list of features as input and produce a features-wise dataset. The number of columns in returned dataset is equal to the number of features in featurelist.

Usage

```
featurization(sequences, string, seq = TRUE, seqorder = 2, pos = TRUE,
  posorder = 2)
```

Arguments

`sequences`	provided as dataframe
`string`	a list of aminoacids or nucleotides
`seq`	sequence based features. by default it is true.
`seqorder`	highest number of sequence which will be considered together
`pos`	position specific features. by default it is true.
`posorder`	highest number of sequence which will be considered together

Value

a featurized dataframe

Examples

```
input = list("ABCDEFGHABDAACBBDEBGGGHHH", "ABCBDBEBEBBBDBDBFDFDFGGHHEEFFEECCCD")
string = c("A", "BD")
featuredata = featurization(input, string, seq = TRUE, pos = FALSE)
featuredata
```

`findposition` *Illustration of findposition This function takes sequence, pattern and position as input and check whether a particulaer pattern is present in position-th place of sequence.*

Description

Illustration of findposition This function takes sequence, pattern and position as input and check whether a particulaer pattern is present in position-th place of sequence.

Usage

```
findposition(sequence, pattern, position)
```

Arguments

sequence	provided as a list of sequences
pattern	a string
position	an integer value

Value

a list of 0/1 indicating present or absent.

Examples

```
sequence = list("ABDEFGHABDAACBBDEBGGGHHH", "ABCBDBEBEBBBDBDBFDFDFGGHHEEFFEECCCD")
pattern = "BD"
position = 2
feat = featurization(sequence, pattern, position)
feat
```

---

lmregression *Linear Regression*

Description

This function takes featurelist, dataset and a value for cross-validation. Now, it creates a formula and outputs RMSE based on provided dataset.

Usage

```
lmregression(featurelist, featuredata, leaveonegene = 0, kfold = 10)
```

Arguments

featurelist	a list of feature
featuredata	provided dataset
leaveonegene	check for leaveonegeneout cross-validation
kfold	a value for cross validation

Value

spearman correlation

Examples

```
featurelist = c("Percent.Peptide", "Amino.Acid.Cut.position","predictions")
dir = getwd()
filepath = paste0(dir,'/data-raw/sample.csv')
data = read.csv(filepath)
lmregression(featurelist,data,0)
```

`mars` *MARS Regression*

Description

This function takes mars regression formula, dataset and a value for cross-validation. Now, it outputs RMSE based on provided dataset.

Usage

```
mars(featurelist, featuredata, leaveonegene = 0, kfold = 10)
```

Arguments

`featurelist`	a list of features
`featuredata`	provided dataset
`leaveonegene`	check for leaveonegeneout cross-validation
`kfold`	a value for cross validation

Value

spearman correlation

Examples

```
featurelist = c("Percent.Peptide", "Amino.Acid.Cut.position","predictions")
dir = getwd()
filepath = paste0(dir,'/data-raw/sample.csv')
data = read.csv(filepath)
mars(featurelist,data,0,2)
```

randomforest *Random Forest*

Description

This function takes full filepath, a list of learning features, a value for cross-validation, the number of times data set will be iterated and learning rate. Now, it creates a deep learning model using deeplearning function of h2o package and outputs RMSE based on provided dataset. Note that size of dataset should be enough to choose a suitable value for kfold.

Usage

```
randomforest(featurelist, featuredata, leaveonegene = 0, kfold = 10,
  trees = 50, learningrate = 0.6)
```

Arguments

`featurelist`	a list of features. last name will indicate the value to be predicted.
`featuredata`	a sample dataset containing all features
`leaveonegene`	check for leaveonegeneout cross-validation
`kfold`	a value for cross validation. Default value is 3.
`trees`	number of trees that will be built.
`learningrate`	a fractional sampling rate in random forest. Default value is 0.6.

Value

spearman correlation

Examples

```
featurelist = c("Percent.Peptide", "Amino.Acid.Cut.position","predictions")
#suppose we have a file as '../crisprpred/data-raw/sample.csv' and current directory is set to '../crisprpred'
dir = getwd()
filepath = paste0(dir,'/data-raw/sample.csv')
data = read.csv(filepath)
randomforest(featurelist,data)
```

randomforest0 *Random Forest*

Description

This function takes full filepath, a list of learning features, a value for cross-validation, the number of times data set will be iterated and learning rate. Now, it creates a deep learning model using deeplearning function of h2o package and outputs RMSE based on provided dataset. Note that size of dataset should be enough to choose a suitable value for kfold.

Usage

```
randomforest0(featurelist, featuredata, leaveonegene = 0, kfold = 10,
  trees = 500)
```

Arguments

featurelist	a list of features. last name will indicate the value to be predicted.
featuredata	a sample dataset containing all features
leaveonegene	check for leaveonegeneout cross-validation
kfold	a value for cross validation. Default value is 3.
trees	number of trees that will be built.

Value

spearman correlation

Examples

```
featurelist = c("Percent.Peptide", "Amino.Acid.Cut.position","predictions")
#suppose we have a file as '../crisprpred/data-raw/sample.csv' and current directory is set to '../crisprpred'
dir = getwd()
filepath = paste0(dir,'/data-raw/sample.csv')
data = read.csv(filepath)
randomforest(featurelist,data,leaveonegene = 0)
```

rmse *Root Mean Square Error*

Description

Return square root of mean squared error.

Usage

```
rmse(error)
```

Arguments

error	a value denoting error

Value

rmse-error

Examples

```
rmse(5)
```

---

svmregression *SMV Regression*

Description

This function takes svm regression formula, dataset and a value for cross-validation. Now, it outputs RMSE based on provided dataset.

Usage

```
svmregression(featurelist, featuredata, leaveonegene = 0, kfold = 10)
```

Arguments

featurelist	a list of features
featuredata	provided dataset
leaveonegene	check for leaveonegeneout cross-validation
kfold	a value for cross validation

Value

spearman correlation

Examples

```
featurelist = c("Percent.Peptide", "Amino.Acid.Cut.position","predictions")
dir = getwd()
filepath = paste0(dir,'/data-raw/sample.csv')
data = read.csv(filepath)
svmregression(featurelist,data,0)
```

viennaRNADataManipulation
Description of viennaRNADataManipulation Function

Description

This function takes a list of sequences as input, manipulates data using rna sequence of viennaRNA and returns a dataframe.

Usage

```
viennaRNADataManipulation(sequences)
```

Arguments

sequences	a list of sequence strings

Value

datafram of extracted features based on input sequences

Examples

```
s = c('AGGCGTGTTAACT','ACGTTTAAGCT')
viennaRNADataManipulation(s)
```

Index