Package: PredCRG (via r-universe)

September 2, 2024

Type Package

Title Computational Prediction of Proteins Encoded by Circadian Genes

Version 1.0.2

Date 2020-12-11

Author Prabina Kumar Meher <meherprabin@yahoo.com>

Maintainer Prabina Kumar Meher <meherprabin@yahoo.com>

Depends R(>= 3.3.3)

Imports Biostrings, protr,Peptides, kernlab, e1071

LazyData TRUE

Description A computational model for predicting proteins encoded by circadian genes. The support vector machine has been employed with Laplace kernel for prediction of circadian proteins, where compositional, transitional and physico-chemical features were utilized as numeric features. User can predict for the test dataset using the proposed computational model. Besides, the user can also build their own training model using their training dataset, followed by prediction for the test set.

License GPL (>= 2)

NeedsCompilation no

Date/Publication 2020-12-11 16:20:03 UTC

Repository https://prabinameher.r-universe.dev

RemoteUrl https://github.com/cran/PredCRG

RemoteRef HEAD

RemoteSha 359bbf52e88f5306c21e0403f6247fcbfbf4632e

Contents

`model1` *Trained model with the Q1 dataset.*

Description

The `model1` is the trained model with the Q1 dataset using the developed approach.

Usage

```
data("model1")
```

Details

Here, 1558 sequences of pos_Q1 and neg_Q1 datasets were used for training. For prediction, support vector machine with Laplace kernel has been trained in which compositionsl, transitional and physico-chemical features are utilized.

See Also

PredCRG, PredCRG_Enc, PredCRG_training

Examples

```
library(kernlab)
data(test)
nam <- names(test)

#encoding of test set using compositional, transitional and physico-chemical features
enc <- PredCRG_Enc(test)

#predicting test set using model1 as CRG or non-CRG
pred <- predict(model1, newdata=enc[1:10,], type="response")

#predicting probabilities of the test sequences using model1
pred1 <- predict(model1, newdata=enc[1:10,], type="probabilities")

#combining predicted labels and probabilities
result <- data.frame(seq_name=nam[1:10], predicted_label=as.character(pred)
,predicted_probability=pred1[,"CRG"])

print(result)
```

model2 *Trained model with the Q2 dataset.*

Description

The model2 is the trained model with the Q2 dataset using the developed approach.

Usage

```
data("model2")
```

Details

Here, 1596 sequences of pos_Q2 and neg_Q2 datasets were used for training. For prediction, support vector machine with Laplace kernel has been trained in which compositionsl, transitional and physico-chemical features are utilized.

See Also

PredCRG, PredCRG_Enc, PredCRG_training

Examples

```
library(kernlab)
data(test)
nam <- names(test)

#encoding of test set using compositional, transitional and physico-chemical features
enc <- PredCRG_Enc(test)

#predicting test set using model2 as CRG or non-CRG
pred <- predict(model2, newdata=enc[1:10,], type="response")

#predicting probabilities of the test sequences using model2
pred1 <- predict(model2, newdata=enc[1:10,], type="probabilities")

#combining predicted labels and probabilities
result <- data.frame(seq_name=nam[1:10], predicted_label=as.character(pred)
,predicted_probability=pred1[,"CRG"])

print(result)
```

model3 *Trained model with the Q3 dataset.*

Description

The model3 is the trained model with the Q3 dataset using the developed approach.

Usage

```
data("model3")
```

Details

Here, 1593 sequences of pos_Q3 and neg_Q3 datasets were used for training. For prediction, support vector machine with Laplace kernel has been trained in which compositionsl, transitional and physico-chemical features are utilized.

See Also

PredCRG, PredCRG_Enc, PredCRG_training

Examples

```
library(kernlab)
data(test)
nam <- names(test)

#encoding of test set using compositional, transitional and physico-chemical features
enc <- PredCRG_Enc(test)

#predicting test set using model3 as CRG or non-CRG
pred <- predict(model3, newdata=enc[1:10,], type="response")

#predicting probabilities of the test sequences using model3
pred1 <- predict(model3, newdata=enc[1:10,], type="probabilities")

#combining predicted labels and probabilities
result <- data.frame(seq_name=nam[1:10], predicted_label=as.character(pred)
,predicted_probability=pred1[,"CRG"])

print(result)
```

model4 *Trained model with the Q4 dataset.*

Description

The model4 is the trained model with the Q4 dataset using the developed approach.

Usage

```
data("model4")
```

Details

Here, 1365 sequences of pos_Q4 and neg_Q4 datasets were used for training. For prediction, support vector machine with Laplace kernel has been trained in which compositionsl, transitional and physico-chemical features are utilized.

See Also

PredCRG, PredCRG_Enc, PredCRG_training

Examples

```
library(kernlab)
data(test)
nam <- names(test)

#encoding of test set using compositional, transitional and physico-chemical features
enc <- PredCRG_Enc(test)

#predicting test set using model4 as CRG or non-CRG
pred <- predict(model4, newdata=enc[1:10,], type="response")

#predicting probabilities of the test sequences using model4
pred1 <- predict(model4, newdata=enc[1:10,], type="probabilities")

#combining predicted labels and probabilities
result <- data.frame(seq_name=nam[1:10], predicted_label=as.character(pred)
,predicted_probability=pred1[,"CRG"])

print(result)
```

PredCRG *Prediction of circadian proteins using the proposed PredCRG model.*

Description

The user can predict the protein sequences as CRG (circadian protein) or non-CRG (non-circadian protein) with certain probability by supplying the test sequences.

Usage

```
PredCRG(seq_data)
```

Arguments

seq_data	Sequence dataset in FASTA format consisting of protein sequences with standard amino acid residues only. It must be an object of class AAStringSet which can be obtained by reading sequences with readAAStringSet available in Biostrings package.

Details

The user has to supply only the seq_data for which the prediction is to be made.

Value

A dataframe with three columns consisting of sequence name, predicted labels of sequences (CRG or non-CRG) and probabilities of prediction.

Author(s)

Prabina Kumar Meher, ICAR-Indian Agricultural Statsitics Research Institute, New Delhi-110012, INDIA

See Also

PredCRG_Enc, PredCRG_training,model1, model2,model3,model4

Examples

```
data(test)
tst <- test[1:10]
PredCRG(seq_data=tst)
```

PredCRG_data *Training dataset of the PredCRG model.*

Description

The dataset that has been used to train the PredCRG model contains four sub-datasets (Q1, Q2, Q3 and Q4) which are prepared based on the homogeneity of sequence length. The positive sets of the sub-datasets are denoted as pos_Q1, pos_Q2, pos_Q3 and pos_Q4 respectively, whereas the negative sets as neg_Q1, neg_Q2, neg_Q3 and neq_Q4 respectively. Further, same number of sequences are there in both positive and negative sets in each sub-dataset. More clearly, 1588, 1596, 1593 and 1365 sequences are present for both positive and negative sets for Q1, Q2, Q3 and Q4 sub-datasets respectively. Further, the range of the length of the sequences for pos_Q1, pos_Q2, pos_Q3 and pos_Q4 are 39-221, 221-363, 363-538, 538-1000 amino acids respectively, and the range of the length of the sequences for neg_Q1, neg_Q2, neg_Q3 and neg_Q4 are 43-407, 407-485, 485-607 and 607-1000 amino acids respectively. In this dataset, only the Q1 sub-dataset is available due to constraint of space in CRAN. However, one can get all the four sub-datasets from GitHub repository (https://github.com/meher861982/PredCRG_dataset).

Usage

```
data("PredCRG_data")
```

Format

The datasets are in AAStringSet format, which can be obtained by reading the FASTA file using readAAStringSet function availbale in Biostrings package.

Details

The protein sequences encoded by the circadian genes contitutes the positive datasets, whereas a randomly selected dataset from the **Uniprot** for the clad *Viridi plantae* constitutes the negative dataset.

Source

The circadian gene sequecnces are collected from the circadian gene database accessible at http://cgdb.biocuckoo.org/ .

See Also

PredCRG, PredCRG_Enc, PredCRG_training,model1, model2,model3,model4

Examples

```
data(PredCRG_data)

pos_Q1 <- PredCRG_data$pos_Q1 #positive set of Q1 dataset
neg_Q1 <- PredCRG_data$neg_Q1 #negative set of Q1 dataset
```

PredCRG_Enc *Encoding of protein sequence data in to numeric feature vector based on PredCRG features.*

Description

Before using the protein sequences for prediction using the proposed model, the sequences must be transformed into numeric feature vectors. The function PredCRG_Enc will transform each protein sequnces to a numeric vector of 62 observations, based on the compositional, physico-chemical and transitional features used in the PredCRG model.

Usage

```
PredCRG_Enc(prot_seq)
```

Arguments

prot_seq	Sequence dataset to be supplied as input, must be an object of class AAStringSet

Details

The dataset must contains the protein sequences having standard amino acid residues only. The clas AAStringSet can be obtained by reading the FASTA file using readAAStringSet available in bioconductor package Biostrings.

Value

A matrix of dimension n*62, for n number of sequences.

Author(s)

Prabina Kumar Meher, ICAR-Indian Agricultural Statistics Research Institute, New Delhi-110012, INDIA

See Also

PredCRG, PredCRG_training, model1, model2,model3,model4

Examples

```
data(test)
enc <- PredCRG_Enc(test)#encoding of test sequence data
enc[1:5,1:5]
```

PredCRG_training *Training of the PredCRG model using the user supplied sequence dataset.*

Description

User can build their own PredCRG model by using their own training dataset. User has to supply the protein sequence dataset of both positive and negative classes having standard amino acid residues only.

Usage

```
PredCRG_training(pos_seq, neg_seq, kern)
```

Arguments

pos_seq	circadian protein sequence dataset (also called positive dataset), must be an object of class AAStringSet.
neg_seq	non-circadian protein sequence dataset (also called negative dataset), must be an object of class AAStringSet.
kern	Type of kernel to be used. It may be laplace, linear, polynomial or RBF.

Details

The sequences must of AAStringSet type can be obtained by reading the FASTA file of the sequences using function readAAStringSet available in Biostrings package.

Value

Support Vector Machine object of class ksvm

Author(s)

Prabina Kumar Meher, ICAR-Indian Agricultural Statistics Research Institute, New Delhi-110012, INDIA

See Also

PredCRG, PredCRG_Enc, model1, model2,model3,model4

Examples

```
library(kernlab)
pos_Q1 <- PredCRG_data$pos_Q1
neg_Q1 <- PredCRG_data$neg_Q1

#training of the model using laplace kernel.
user_model <- PredCRG_training(pos_seq=pos_Q1[1:100], neg_seq=neg_Q1[1:100], kern="laplace")
```

```
data(test)
tst_enc <- PredCRG_Enc(test[1:10])#encoding of the test set
predict(user_model, tst_enc, type="response") #predicting the label of the test instances
predict(user_model, tst_enc, type="probabilities")#predicting the probability of the test instances


library(e1071)
#training of the model using RBF kernel.
user_model <- PredCRG_training(pos_seq=pos_Q1[1:100], neg_seq=neg_Q1[1:100], kern="RBF")
predict(user_model, tst_enc, probability=TRUE) #Predicting probability
predict(user_model, tst_enc) #Predicting labels
```

`test` *Test dataset.*

Description

A test dataset containing 54 circadian protein sequences collected from literature. This dataset has been used as an independent test dataset for assessing the predition accuracy of `PredCRG` model.

Usage

```
data("test")
```

See Also

PredCRG, PredCRG_Enc, PredCRG_data

Examples

```
data(test)
PredCRG(test[1:10])
```

Index