Package: SFtools (via r-universe)

August 20, 2024

Type Package

Title Space Filling Based Tools for Data Mining

Version 1.0.0

Author Mohamed Laib and Mikhail Kanevski

Maintainer Mohamed Laib <laib.med@gmail.com>

Description Contains space filling based tools for machine learning and data mining. Some functions offer several computational techniques and deal with the out of memory for large big data by using the ff package.

Imports Biobase, doParallel, parallel, stats

License GPL-3

URL https://mlaib.github.io/

BugReports https://github.com/mlaib/SFtools/issues

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

NeedsCompilation no

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

RemoteUrl https://github.com/mlaib/sftools

RemoteRef HEAD

RemoteSha f96b5d0a365c94eed6bd538658e8b408774c4d86

Contents

SFtools-package *SFtools: Space Filling Based Tools for Data Mining*

Description

Contains space filling based tools for machine learning and data mining. Some functions offer several computational techniques and deal with the out of memory for large big data by using the ff package.

Author(s)

Mohamed Laib <Mohamed.Laib@gmail.com> and

Mikhail Kanevski <Mikhail.Kanevski@unil.ch>,

Maintainer: Mohamed Laib <laib.med@gmail.com>

References

M. Laib, M. Kanevski, A novel filter algorithm for unsupervised feature selection based on a space filling measure. Proceedings of the 26rd European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN), pp. 485-490, Bruges (Belgium), 2018.

M. Laib and M. Kanevski, A new algorithm for redundancy minimisation in geo-environmental data, 2019. Computers & Geosciences, 133 104328.

J. A. Royle, D. Nychka, An algorithm for the construction of spatial coverage designs with implementation in Splus, Computers and Geosciences 24 (1997) p. 479–488.

J. Franco, Planification d'expériences numériques en phase exploratoire pour la simulation des phénomènes complexes, Thesis (2008) 282.

D. Dupuy, C. Helbert, J. Franco (2015). DiceDesign and DiceEval: Two R Packages for Design and Analysis of Computer Experiments. Journal of Statistical Software, 65(11), 1-38. Jstatsoft.

See Also

Useful links:

- https://mlaib.github.io/
- Report bugs at https://github.com/mlaib/SFtools/issues

SimData *Simulated data set*

Description

Generates a simulated data set

Usage

```
SimData(n=1000)
```

Arguments

n	Number of generated data points (by default: n=1000).

Value

A data.frame of simulated data set, with 7 features (4 of them are redundants)

Examples

```
Sim_Data<-SimData(n=1000)
plot(Sim_Data$x1,Sim_Data$x2)

## Not run:

#### Visualisation of the data set (3D) ####
require(rgl)
require(colorRamps)

c <- cut(Sim_Data$z,breaks=100)
cols <- matlab.like(100)[as.numeric(c)]
plot3d(Sim_Data$x1,Sim_Data$x2,Sim_Data$z,radius=0.01, col=cols,
type="s",xlab="x1",ylab="x2",zlab="z",box=F)
grid3d(c("x","y","z"),col="black",lwd=1)


## End(Not run)
```

UfsCov *UfsCov algorithm for unsupervised feature selection*

Description

Applies the UfsCov algorithm based on the space filling concept, by using a sequatial forward search (SFS).

Usage

```
UfsCov(data)
```

Arguments

data	Data of class: matrix or data.frame.

Details

Since the algorithm is based on pairwise distances, and according to the computing power of your machine, large number of data points can take much time and needs more memory.

Value

A list of two elements:

- CovD a vector containing the coverage measure of each step of the SFS.
- IdR a vector containing the added variables during the selection procedure.

Note

The algorithm does not deal with missing values and constant features. Please make sure to remove them.

Author(s)

Mohamed Laib <Mohamed.Laib@unil.ch>

References

M. Laib, M. Kanevski, A novel filter algorithm for unsupervised feature selection based on a space filling measure. Proceedings of the 26rd European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN), pp. 485-490, Bruges (Belgium), 2018.

M. Laib and M. Kanevski, A new algorithm for redundancy minimisation in geo-environmental data, 2019. Computers & Geosciences, 133 104328.

Examples

```
Sim_Data<-SimData(n=800)
Results<- UfsCov(Sim_Data)

cou<-colnames(Sim_Data)
nom<-cou[Results[[2]]]
par(mfrow=c(1,1), mar=c(5,5,2,2))
names(Results[[1]])<-cou[Results[[2]]]
plot(Results[[1]] ,pch=16,cex=1,col="blue", axes = FALSE,
xlab = "Added Features", ylab = "Coverage measure")
lines(Results[[1]] ,cex=2,col="blue")
grid(lwd=1.5,col="gray" )
box()
```

```
axis(2)
axis(1,1:length(nom),nom)
which.min(Results[[1]])

## Not run:

#### UfsCov on the Butterfly dataset ####
require(IDmining)

N <- 1000
raw_dat <- Butterfly(N)
dat<-raw_dat[,-9]

Results<- UfsCov(dat)
cou<-colnames(dat)
nom<-cou[Results[[2]]]
par(mfrow=c(1,1), mar=c(5,5,2,2))
names(Results[[1]])<-cou[Results[[2]]]

plot(Results[[1]] ,pch=16,cex=1,col="blue", axes = FALSE,
xlab = "Added Features", ylab = "Coverage measure")
lines(Results[[1]] ,cex=2,col="blue")
grid(lwd=1.5,col="gray" )
box()
axis(2)
axis(1,1:length(nom),nom)
which.min(Results[[1]])


## End(Not run)
```

`UfsCov_par` *UfsCov algorithm for unsupervised feature selection*

Description

Applies the UfsCov algorithm based on the space filling concept, by using a sequatial forward search (SFS).This function offers a parellel computing.

Usage

```
UfsCov_par(data, ncores=2)
```

Arguments

`data`	Data of class: `matrix` or `data.frame`.
`ncores`	Number of cores to use (by default: `ncores=2`).

Details

Since the algorithm is based on pairwise distances, and according to the computing power of your machine, large number of data points needs more memory.

Value

A list of two elements:

- `CovD` a vector containing the coverage measure of each step of the SFS.
- `IdR` a vector containing the added variables during the selection procedure.

Note

The algorithm does not deal with missing values and constant features. Please make sure to remove them. Note that it is not recommanded to use this function with small data, it takes more time than using the standard `UfsCov` function.

Author(s)

Mohamed Laib <Mohamed.Laib@unil.ch>

References

M. Laib, M. Kanevski, A novel filter algorithm for unsupervised feature selection based on a space filling measure. Proceedings of the 26rd European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN), pp. 485-490, Bruges (Belgium), 2018.

M. Laib and M. Kanevski, A new algorithm for redundancy minimisation in geo-environmental data, 2019. Computers & Geosciences, 133 104328.

Examples

```
N <- 800
dat<-SimData(N)
Results<- UfsCov_par(dat,ncores=2)

cou<-colnames(dat)
nom<-cou[Results[[2]]]
par(mfrow=c(1,1), mar=c(5,5,2,2))
names(Results[[1]])<-cou[Results[[2]]]
plot(Results[[1]] ,pch=16,cex=1,col="blue", axes = FALSE,
xlab = "Added Features", ylab = "Coverage measure")
lines(Results[[1]] ,cex=2,col="blue")
grid(lwd=1.5,col="gray" )
box()
axis(2)
axis(1,1:length(nom),nom)
which.min(Results[[1]])

## Not run:
```

```
N<-5000
dat<-SimData(N)

## Little comparison:
system.time(Uf<-UfsCov(dat))
system.time(Uf.p<-UfsCov_par(dat, ncores = 4))


## End(Not run)
```

Index