Package: fgga (via r-universe)

November 25, 2026

Type Package

Title Hierarchical ensemble method based on factor graph

Version 1.22.0

Date 2023-11-27

biocViews Software, StatisticalMethod, Classification, Network, NetworkInference, SupportVectorMachine, GraphAndNetwork, GO

Description Package that implements the FGGA algorithm. This package provides a hierarchical ensemble method based ob factor graphs for the consistent cross-ontology annotation of protein coding genes. FGGA embodies elements of predicate logic, communication theory, supervised learning and inference in graphical models.

Depends R (>= 4.3), RBGL

Imports graph, stats, e1071, methods, gRbase, jsonlite, BiocFileCache, curl, igraph

Suggests knitr, rmarkdown, GOstats, GO.db, BiocGenerics, pROC, RUnit, BiocStyle

License GPL-3

URL https://github.com/fspetale/fgga

Encoding UTF-8

LazyData false

VignetteBuilder knitr

NeedsCompilation no

Config/pak/sysreqs libglpk-dev libicu-dev libxml2-dev libssl-dev

Repository Bioconductor 3.24

Date/Publication 2026-11-23 22:47:12 UTC

RemoteUrl https://github.com/bioc/fgga

RemoteRef RELEASE_3_24

RemoteSha 76c619e9f253658515c42e71f6778cedcd6eb7f7

Contents

fgga-package . . . 2
CfData . . . 3
createFolds . . . 4
fgga . . . 5
fgga2bipartite . . . 7
fMeasure . . . 8
maxDistancegraphOnto . . . 9
preCoreFG . . . 10
sumProduct . . . 11
svmOnto . . . 13
svmTrain . . . 15
tableTPG . . . 16
varianceOnto . . . 17

Index **19**

`fgga-package` *FGGA: Factor Graph Gene ontology Annotation.*

Description

FGGA is a graph-based machine learning approach for the automated and consistent GO, PO, HPO and ZFA annotation of protein coding genes. The input is a set of ontoligical-terms annotated protein coding genes previously characterized in terms of a fixed number of user-defined features, including the presence/absence of PFAM domains, physical-chemical properties, presence of signal peptides, among others. The set of ontoligical terms defines the output cross-ontology subgraph. A hierarchical ensemble (SVMs) machine learning model is generated. This model can be used to predict the cross-ontology subgraph annotations of uncharacterized protein coding genes. Individual ontoligical-term annotations are accompanied by maximum a posteriori probability estimates issued by the native message passing algorithm of factor graphs.

Author(s)

Flavio E. Spetale, *Paolo Cacchiarrelli* and *Elizabeth Tapia*

BioInformatics

Cifasis-Conicet

<spetale@cifasis-conicet.gov.ar>

Maintainer: *Flavio E. Spetale*

References

Spetale F.E., et al. **A Factor Graph Approach to Automated GO Annotation**. *PLoS ONE* (2016). https://doi.org/10.1371/journal.pone.0146986.

Spetale Flavio E., et al. **Consistent prediction of GO protein localization**. *Scientific Report* (2018). https://doi.org/10.1038/s41598-018-26041-z.

See Also

fgga, fgga2bipartite, sumProduct, svm

CfData — *A set of characterized protein coding genes from the Cannis familiaris organism annotated to a target GO subgraph considering both experimental and electronic evidence.*

Description

The CfData dataset consists of a list containing the following:

$dxCf: characterizations of *6962* protein coding genes in terms of *72* physico-chemical properties of their amino acid sequences. These sequences, obtained from the Uniprot database, are annotated to *36* GO-terms of the GO Molecular Function (GO-MF) ontology subdomain.

$tableCfGO: a set of *6962* protein coding genes annotated to GO-MF target classes. Genes are identified by their **Uniprot** ID mappings which are obtained with the org.Cf.eg.db annotation package set to work with both experimental and electronic evidence. Additionally, only those GO-MF terms with at least *500* annotated genes were preserved.

$graphCfGO: the target **GO-MF** subgraph obtained with the org.Cf.eg.db annotation package set to work with the set of **GO-MF** target classes.

$indexGO: two arrays of Uniprot ID mappings defining the train-test partition of the set *6962* protein coding genes annotated to **GO-MF** terms.

$nodesGO: labels of the GO-MF subgraph.

$varianceGOs: a vector labeled with the variance of each **GO-MF** term.

Usage

```
data("CfData")
```

Format

A list with five named entries containg:

dxCf A matrix (6962 rows x 72 columns) containing the characterized proteins.

graphCfGO An adjacency binary matrix (36 rows x 36 columns) corresponding to the GO-MF subgraph.

indexGO A list with two named entries: indexTrain and indexTest each containing a numeric vector.

tableCfGO A binary matrix (6962 rows x 36 columns) containing GOs associated with a protein.

nodesGO A numerical vector containing the nodes of the GO-MF subgraph.

Source

Uniprot Taxonomy: 9615

https://www.uniprot.org/uniprot/?query=taxonomy:9615

Package: org.Cf.eg.db - Version: 3.8.2

https://bioconductor.org/packages/org.Cf.eg.db/

Examples

```
data(CfData)

## list objects included
ls(CfData)
# [1] "dxCf"  "graphCfGO" "indexGO"   "nodesGO"   "tableCfGO"

# Physico-chemical properties of each protein
head(CfData[["dxCf"]])

# GO-MF node labels, GO-terms, of each protein
head(CfData[["tableCfGO"]])
```

`createFolds` *Data splitting function useful for binary classification tasks*

Description

createFolds splits binary classification data into k-folds.

Usage

```
createFolds(target, k_fold = 10)
```

Arguments

`target`	A binary vector of a Ontology class
`k_fold`	An integer for the number of folds

Details

A random sampling is performed on binary classification data. A set of k data folds reflecting the original class balance is obtained.

Value

list of row position integers corresponding to the training data

Author(s)

Flavio E. Spetale and Pilar Bulacio <spetale@cifasis-conicet.gov.ar>

References

Hyndman and Athanasopoulos (2013), Forecasting: principles and practice. https://www.otexts.org/fpp

Examples

```
data(CfData)

createFolds(CfData[["tableCfGO"]][ ,"GO:0005515"], k_fold = 2)
```

`fgga` *Factor Graph Cross-Ontology Annotation model*

Description

A hierarchical graph-based machine learning model for the consistent GO, PO, ZFA, HPO annotation of protein coding genes.

Usage

```
fgga(graphOnto, tableOntoTerms, dxCharacterized, dxTestCharacterized,
    kFold, kernelSVM, tmax, epsilon)
```

Arguments

`graphOnto`	A graphNEL graph with 'm' Ontology node labels.
`tableOntoTerms`	A binary matrix with 'n' proteins (rows) by 'm' Ontology node labels (columns).
`dxCharacterized`	A data frame with 'n' proteins (rows) by 'f' features (columns).
`dxTestCharacterized`	A data frame with 'k' proteins (rows) by 'f' features (columns).
`kFold`	An integer for the number of folds.
`kernelSVM`	The kernel used to calculate the variance (default: radial).
`tmax`	An integer indicating the maximum number of iterations (default: 200).
`epsilon`	A real value less than 1 that represents the convergence criteria (default: 0.001).

Details

The **FGGA model** is built in two main steps. In the first step, a core Factor Graph (FG) modeling hidden Ontology-term predictions and relationships is created. In the second step, the FG is enriched with nodes modeling observable Ontology-term predictions issued by **binary SVM classifiers**. In addition, probabilistic constraints modeling learning gaps between hidden and observable Ontology-term predictions are introduced. These gaps are assumed to be independent among Ontology-terms, locally additive with respect to observed predictions, and zero-mean Gaussian. **FGGA predictions** are issued by the native iterative **message passing algorithm** of factor graphs.

Value

A named matrix with 'k' protein coding genes (rows) by 'm' cross-Ontology node labels (columns) where each element indicates a probabilistic prediction value.

Author(s)

Flavio E. Spetale and Elizabeth Tapia <spetale@cifasis-conicet.gov.ar>

References

Spetale F.E., Tapia E., Krsticevic F., Roda F. and Bulacio P. "A Factor Graph Approach to Automated GO Annotation". PLoS ONE 11(1): e0146986, 2016.

Spetale Flavio E., Arce D., Krsticevic F., Bulacio P. and Tapia E. "Consistent prediction of GO protein localization". Scientific Report 7787(8), 2018

See Also

fgga2bipartite, sumProduct, svmOnto

Examples

```
data(CfData)

mygraphGO <- as(CfData[["graphCfGO"]], "graphNEL")

dxCfTestCharacterized <- CfData[["dxCf"]][CfData[["indexGO"]]$indexTest[1:2], ]

myTableGO <- CfData[["tableCfGO"]][
                    CfData[["indexGO"]]$indexTrain[1:300], ]

dataTrain <- CfData[["dxCf"]][
                    CfData[["indexGO"]]$indexTrain[1:300], ]

fggaResults <- fgga(graphOnto = mygraphGO,
                tableOntoTerms = myTableGO, dxCharacterized = dataTrain,
                dxTestCharacterized = dxCfTestCharacterized, kFold = 2,
                tmax = 50, epsilon = 0.05)
```

`fgga2bipartite` *Forney Factor Graph model*

Description

fgga2bipartite builds a Forney Factor Graph from a FGGA model.

Usage

```
fgga2bipartite(graphOnto)
```

Arguments

`graphOnto`	A graphNEL graph with 'm' cross-Ontology node labels.

Details

The **Gene Ontology** (GO) is structured as a directed acyclic graph (DAG) with nodes (GO-terms) representing gene functions and edges characterizing relationships between nodes. A variety of relationships are possible (currently 8). To compute GO-term predictions perfectly aware of GO-term relationships, a Forney Factor Graph is required. Hence, GO-terms are mapped to binary variable nodes, and relationships to logical factor nodes.

Value

A binary matrix with *2*m* rows by *2*m-1* columns where *m* is the quantity of cross-Ontology node labels.

Author(s)

Flavio E. Spetale <spetale@cifasis-conicet.gov.ar>

References

F. Spetale, J. Murillo, E. Tapia, D. Arce, S. Ponce, and P. Bulacio, "Formal modeling of gene ontology annotation predictions based on factor graphs," Journal of Physics: Conference Series, vol. 705, no. 1, p. 012001, 2016.

Spetale F.E., Tapia E., Krsticevic F., Roda F. and Bulacio P. "A Factor Graph Approach to Automated GO Annotation". PLoS ONE 11(1): e0146986, 2016.

Spetale Flavio E., Arce D., Krsticevic F., Bulacio P. and Tapia E. "Consistent prediction of GO protein localization". Scientific Report 7787(8), 2018

Examples

```
data(CfData)

graphGO <- as(CfData$graphCfGO, "graphNEL")
fgga2bipartite(graphGO)
```

fMeasure *Individual and hierarchical F-measures*

Description

Set of functions to compute the individual and hierarchical F-score, precision, recall.

Usage

```
fMeasures(target, predicted, cutoff = 0.5)
fMeasuresByLevel(target, predicted, graphOnto, cutoff = 0.5)
fHierarchicalMeasures(target, predicted, graphOnto, cutoff = 0.5)
```

Arguments

target	A binary matrix with 'n' proteins (rows) by 'm' Ontology node labels (columns) corresponding to the target of ontology terms where 0 stands for negative and 1 for positive.
predicted	A real matrix with 'n' proteins (rows) by 'm' Ontology node labels (columns) corresponding to the predicted terms.
graphOnto	A graphNEL graph with 'm' Ontology node labels.
cutoff	A real value to divide the predicted terms into positive and negative. The predicted values higher than the cutoff will be taken as positive.

Details

fMeasures computes the F-score, precision, recall, specificity and accuracy for each ontological term.

fMeasuresByLevel computes F-score, precision, recall, specificity and accuracy for all ontological terms belongs to graph. The levels are calculated as the maximum distance between two terms of the graph.

fHierarchicalMeasures computes the hierarchical F-score, precision, recall for the predicted terms of a set of proteins.

Value

fMeasures and fMeasuresByLevel returns a list of two elements where the first element is a named vector with six attributes while the second element is an array of 'm' ontological terms by six attributes. The 6 attributes are:

Prec:	Precision
Recall:	Recall
Specif:	Specificity
Fmeasure:	F-score
Acc:	Accuracy

nPositive: Number of positive samples

fHierarchicalMeasures returns a list of five elements:

HP: Hierarchical Precision

HR: Hierarchical Recall

HF: Hierarchical F-score

nSample: Number of proteins evaluated

noEvalSample: Named vector of proteins not evaluated

Author(s)

Flavio E. Spetale <spetale@cifasis-conicet.gov.ar>

References

Verspoor K, Cohn J, Mnizewski S, C J. A categorization approach to automated ontological function annotation. Protein Science. 2006;15:1544–1549.

Examples

```
data(CfData)

predGO <- matrix(runif(360, 0, 1),10,36, dimnames=list(rownames(
        CfData[["tableCfGO"]])[seq_len(10)], colnames(CfData[["tableCfGO"]])))

fMeasures(CfData[["tableCfGO"]][seq_len(10), ], predGO, cutoff = 0.5)

mygraphGO <- as(CfData[["graphCfGO"]], "graphNEL")

fHierarchicalMeasures(CfData[["tableCfGO"]][seq_len(10), ], predGO, mygraphGO,
cutoff = 0.5)
```

---

maxDistancegraphOnto *Maximum distance for a graph*

Description

Computes the maximum distance from any node to the root of the graph

Usage

```
maxDistancegraphOnto(graphOnto)
```

Arguments

graphOnto A graphNEL graph with 'm' Ontology node labels.

Details

This function computes a distance matrix for a graph

Value

Named numeric array containing the distance from any node to the root.

Author(s)

Flavio E. Spetale <spetale@cifasis-conicet.gov.ar>

See Also

fMeasure

Examples

```
data(CfData)

mygraphGO <- as(CfData[["graphCfGO"]], "graphNEL")

maxDistancegraphOnto(mygraphGO)
```

---

preCoreFG *Transitive closure processing of a cross-ontology DAG*

Description

preCoreFG ensures the transitive closure of inference paths -serial concatenation of relationships- in a cross-ontology DAG.

Usage

```
preCoreFG(ontoTerms, domains = "GO")
```

Arguments

ontoTerms	A vector with 'm' cross-ontology node labels
domains	A string that indicates which subdomains or ontologies will be used. Values: "GOBP", "GOMF", "GOCC", "GOCC-PO", "GOCC-ZFA", "GOBP-HPO", "GOMF-HPO", "GOCC-HPO", "GO-PO", "GO-ZFA", "GO-HPO", "GO" (default, "BP-MF-CC")

Details

Non-transitive relationships in cross-ontology DAG's may lead to non-transitive inference paths precluding the free propagation and consistency checking of ontology annotations. A transitive closure screening process over cross-ontology DAG's relationships is required before the construction of Forney Factor Graphs. Serial concatenation of relationships leading to non-transitive inference paths in a cross-ontology DAG are conformed by removing the most specific relationship.

Value

A graphNEL graph with 'm' node labels belong to ontologies used.

Author(s)

Flavio E. Spetale <spetale@cifasis-conicet.gov.ar>

References

Spetale Flavio E., Arce D., Krsticevic F., Bulacio P. and Tapia E. "Consistent prediction of GO protein localization". Scientific Report 7787(8), 2018

See Also

fgga2bipartite

Examples

```
data(CfData)

myGOs <- c(CfData[["nodesGO"]], "GO:1902494", "GO:0032991", "GO:1990234",
           "GO:0005575")


# mygraphGO <- preCoreFG(myGOs, domains = "GOMF")
```

---

sumProduct *Message passing algorithm between nodes of the Forney Factor Graph*

Description

msgFGGA operates in Forney Factor Graphs and computes approximate maximum a posteriori (MAP) estimates of hidden Ontology variable nodes (Ontology-terms).

Usage

```
msgFGGA(matrixFGGA, obsValueOntoTerms, graphOnto, tmax = 200,
        epsilon = 0.001)
```

Arguments

matrixFGGA	A binary matrix with FGGA model of the class 'fgga.'
obsValueOntoTerms	A named vector with 'm' probabilistic prediction values for a protein coding gene.
graphOnto	A graphNEL graph with 'm' Ontology node labels.
tmax	An integer indicating the maximum number of iterations (defualt: 200).
epsilon	An integer that represents the convergence criteria (default: 0.001)

Details

Starting from Ontology-term predictions at observable variable nodes, probability distribution functions modelling the learning noise of individual Ontology-terms, a user-defined number of iterations (maximum 200), a user-defined threshold for the convergence of predictions (maximum 0.001), and the structure of the Forney Factor Graph, the **msgFGGA** delivers approximate maximum a posteriori (MAP) estimates of hidden GO variable nodes (GO-terms).

Value

A named vector with 'm' consistent probabilistic predictions for a protein coding genes.

Author(s)

Flavio E. Spetale and Elizabeth Tapia <spetale@cifasis-conicet.gov.ar>

References

Kschischang FR, Frey BJ, Loeliger H.-A. Factor graphs and the sum-product algorithm. IEEE Trans. Inf. Theor. 47, 498–519 (2001).

Yedidia JS. Message-passing algorithms for inference and optimization. Journal of Statistical Physics 145, 860–890 (2011).

Spetale FE, Tapia E, Krsticevic F, Roda F, Bulacio P (2016). A Factor Graph Approach to Automated GO Annotation. PLOS ONE 11(1): e0146986

See Also

tableTPG

Examples

```
data(CfData)

mygraphGO <- as(CfData[["graphCfGO"]], "graphNEL")

myTableGO <- CfData[["tableCfGO"]][
                    CfData[["indexGO"]]$indexTrain[1:500], ]

modelSVMs <- lapply(CfData[["nodesGO"]], FUN = svmTrain,
```

```
                    tableOntoTerms = myTableGO,
                    dxCharacterized = CfData[["dxCf"]],
                    graphOnto = mygraphGO, kernelSVM = "radial")

rootGO <- leaves(mygraphGO, "in")

varianceGOs <- CfData[["varianceGOs"]]

dxTestCharacterized <- CfData[["dxCf"]][
                                sample(1:dim(CfData[["dxCf"]])[2], 2), ]

matrixGOTest <- svmOnto(svmMoldel = modelSVMs,
                        dxCharacterized = dxTestCharacterized,
                        rootNode = rootGO, varianceSVM = varianceGOs)

modelFGGA <- fgga2bipartite(mygraphGO)

matrixFGGATest <- t(apply(matrixGOTest, MARGIN = 1, FUN = msgFGGA,
                    matrixFGGA = modelFGGA, graphOnto= mygraphGO,
                    tmax = 50, epsilon = 0.1))
```

`svmOnto` *Ontology-term predictions by binary SVM classifiers*

Description

svmOnto delivers soft Ontology-term predictions based on binary SVM classification models.

Usage

```
svmOnto(svmMoldel, dxCharacterized, rootNode, varianceSVM)
```

Arguments

`svmMoldel`	A list of object of class “svm" created by svm.
`dxCharacterized`	A data frame with ‘n’ protein coding genes (rows) by ‘f’ features (columns).
`rootNode`	A character indicating the root of the graph.
`varianceSVM`	A vector named with the variance of cross-Ontology node labels.

Details

Binary SVM predictions are supplemented with their corresponding margins. These margins are used to model the additive zero-mean Gaussian learning noise that corrupts ideal but hidden Ontology-term predictions. These ideal predictions are embedded in hidden variable nodes of the Forney Factor Graph.

Value

svmOnto A named vector of predicted values for a protein sequence.

Author(s)

Flavio E. Spetale, Pilar Bulacio and Javier Murillo <spetale@cifasis-conicet.gov.ar>

References

Chang, Chih-Chung and Lin, Chih-Jen: LIBSVM: a library for Support Vector Machines http://www.csie.ntu.edu.tw/~cjlin/libsvm

Eisner R, Poulin B, Szafron D, Lu P, Greiner R. Improving protein function prediction using the hierarchical structure of the Gene Ontology. In: Proc. IEEE CIBCB; 2005. p. 1–1

Spetale FE, Tapia E, Krsticevic F, Roda F, Bulacio P (2016). A Factor Graph Approach to Automated GO Annotation. PLOS ONE 11(1): e0146986

See Also

svmTrain

Examples

```
data(CfData)

mygraphGO <- as(CfData[["graphCfGO"]], "graphNEL")

modelSVMs <- lapply(CfData[["nodesGO"]][1:4], FUN = svmTrain,
                    tableOntoTerms = CfData[["tableCfGO"]],
                    dxCharacterized = CfData[["dxCf"]],
                    graphOnto = mygraphGO, kernelSVM = "radial")

rootGO <- leaves(mygraphGO, "in")

varianceGOs <- CfData[["varianceGOs"]]

# SVM testing in four GO-terms
dxTestCharacterized <- CfData[["dxCf"]][
                                sample(1:dim(CfData[["dxCf"]])[1], 20), ]

matrixGOTest <- svmOnto(svmMoldel = modelSVMs,
                        dxCharacterized = dxTestCharacterized,
                        rootNode = rootGO, varianceSVM = varianceGOs)
```

svmTrain *Binary SVM classification models for individual Ontology-term predictions*

Description

svmTrain delivers a set of binary SVM classifiers for different Ontology-terms.

Usage

```
svmTrain(nodeGraph, tableOntoTerms, dxCharacterized, graphOnto,
        kernelSVM = "radial")
```

Arguments

nodeGraph	A character indicating a GO node label
tableOntoTerms	A binary matrix with 'n' proteins (rows) by 'm' Ontology node labels (columns).
dxCharacterized	A data frame with 'n' protein coding genes (rows) by 'f' features (columns).
graphOnto	A graphNEL graph with 'm' Ontology node labels.
kernelSVM	The kernel used to calculate the variance (default: radial).

Details

Starting from sets of positively annotated protein sequences to different GO-terms in a GO subgraph, corresponding sets of negatively annotated protein sequences are computed using the inclusive separation policy proposed by Eisner et al. Training datasets for each GO-term are used to train binary Support Vector Machine (SVM) classifiers with a variety of kernel options.

Value

svmTrain	A list of objects of "svm" class containing the fitted model.

Author(s)

Flavio E. Spetale, Pilar Bulacio and Javier Murillo <spetale@cifasis-conicet.gov.ar>

References

Chang, Chih-Chung and Lin, Chih-Jen: LIBSVM: a library for Support Vector Machines http://www.csie.ntu.edu.tw/~cjlin/libsvm

Eisner R, Poulin B, Szafron D, Lu P, Greiner R. Improving protein function prediction using the hierarchical structure of the Gene Ontology. In: Proc. IEEE CIBCB; 2005. p. 1–1

Spetale FE, Tapia E, Krsticevic F, Roda F, Bulacio P (2016). A Factor Graph Approach to Automated GO Annotation. PLOS ONE 11(1): e0146986

See Also

svmOnto

Examples

```
data(CfData)

mygraphGO <- as(CfData[["graphCfGO"]], "graphNEL")

# SVM training in four GO-terms
modelSVMs <- lapply(CfData[["nodesGO"]][1:4], FUN = svmTrain,
                    tableOntoTerms = CfData[["tableCfGO"]],
                    dxCharacterized = CfData[["dxCf"]],
                    graphOnto = mygraphGO, kernelSVM = "radial")
```

---

tableTPG *Valid configurations for hidden variable nodes in a Forney Factor Graph*

Description

tableTPG provides valid configurations of hidden variable nodes at logical function nodes in a Forney Factor Graph under the True Path Graph (TPG) constraint.

Usage

```
tableTPG(att)
```

Arguments

att	An integer indicating the number of cross-Ontology nodes involved

Details

Valid configurations of hidden variable nodes at logical function nodes enable messaging passing across the Forney Factor Graph. The TPG constraint is defined as: "If the child Ontology node describes the protein, then all its parent terms must also apply to that protein; and if a Ontology node does not describe a protein, then all its descendant Ontology nodes must not describe it". The TPG constraint governs the structure of the Ontology-DAG and the inference process in the associated Forney Factor Graph.

Value

A binary matrix with *(n-1)^2+1* rows by *n+1* columns where *n = attr*

Author(s)

Flavio E. Spetale, Pilar Bulacio and Javier Murillo <spetale@cifasis-conicet.gov.ar>

References

Tanoue J, Yoshikawa M, Uemura S (2012). The Gene Around GO viewer. Bioinformatics 18(12): 1705–1706.

Spetale FE, Tapia E, Krsticevic F, Roda F, Bulacio P (2016). A Factor Graph Approach to Automated GO Annotation. PLOS ONE 11(1): e0146986

Examples

```
tableTPG(3)
```

`varianceOnto` *The variance of the gaussian learning noise at individual Ontology-terms*

Description

varianceOnto estimates the variance of gaussian distributions modeling the additive learning noise that corrupts ideal Ontology-term predictions.

Usage

```
varianceOnto(tableOntoTerms, dxCharacterized, kFold, graphOnto, rootNode,
            kernelSVM = "radial")
```

Arguments

`tableOntoTerms`	A binary matrix with 'n' protein coding genes (rows) by 'm' cross-Ontology node labels (columns).
`dxCharacterized`	A data frame with 'n' protein coding genes (rows) by 'f' features (columns).
`kFold`	An integer for the number of folds.
`graphOnto`	A graphNEL graph with 'm' cross-Ontology node labels.
`rootNode`	A character indicating the root of the graph.
`kernelSVM`	The kernel used to calculate the variance (default: radial).

Details

Under the assumption of symmetrical (Gaussian) conditional probability distributions for observable variable node predictions y_i over a hidden variable node annotations x_i, variances η_i can be estimated using a validation dataset of positively annotated samples. Let *D* be a validation dataset with L^+ positively annotated samples

$$\hat{\eta_i} = 1/(L^+ - 1) * \sum_{i=1}^{L} (x_i - y_i)$$

where $x_i = 1$ is the positive annotation of the l-th data sample to the ith Ontology-term and y_i is the corresponding real-valued classifier (SVM) prediction.

Value

A vector named with the variance of each cross-Ontology node.

Author(s)

Flavio E. Spetale <spetale@cifasis-conicet.gov.ar>

References

Spetale FE, Tapia E, Krsticevic F, Roda F, Bulacio P (2016). A Factor Graph Approach to Automated GO Annotation. PLOS ONE 11(1): e0146986

Examples

```
data(CfData)

mygraphGO <- as(CfData[["graphCfGO"]], "graphNEL")

rootGO <- leaves(mygraphGO, "in")

mygraphGO <- subGraph(c("GO:0140110", "GO:0098772", "GO:0003674"), mygraphGO)

myTableGO <- CfData[["tableCfGO"]][
                   CfData[["indexGO"]]$indexTrain,
                   c("GO:0140110", "GO:0098772", "GO:0003674")]

varianceGOs <- varianceOnto(tableOntoTerms = myTableGO,
                            dxCharacterized = CfData[["dxCf"]],
                            kFold = 2, graphOnto = mygraphGO,
                            rootNode = rootGO, kernelSVM = "radial")
```

Index

∗ **TPG**
    tableTPG, 16
∗ **datasets**
    CfData, 3
∗ **fgga2bipartite**
    fgga, 5
∗ **msgFFGA**
    preCoreFG, 10
∗ **msgFGGA**
    fgga, 5
    sumProduct, 11
∗ **package**
    fgga-package, 2
∗ **svm**
    svmOnto, 13
    svmTrain, 15
∗ **variance**
    varianceOnto, 17

CfData, 3
createFolds, 4

fgga, *3*, 5
fgga-package, 2
fgga2bipartite, *3*, *6*, 7, *11*
fHierarchicalMeasures *(*fMeasure*)*, 8
fMeasure, 8, *10*
fMeasures *(*fMeasure*)*, 8
fMeasuresByLevel *(*fMeasure*)*, 8

maxDistancegraphOnto, 9
msgFGGA *(*sumProduct*)*, 11

preCoreFG, 10

sumProduct, *3*, *6*, 11
svm, *3*
svmOnto, *6*, 13, *16*
svmTrain, *14*, 15

tableTPG, *12*, 16

varianceOnto, 17