Package 'RaPKod'

February 5, 2018

Type Package

Title Random Projection Kernel Outlier Detector

Version 0.9

Date 2018-01-30

Author Jeremie Kellner

Maintainer Jeremie Kellner <jeremie.kellner@ed.univ-lille1.fr>

Description Kernel method that performs online outlier detection through random low-dimensional projections in a kernel space. Controls the probability of false alarm error. See Kellner J., ``Gaussian models and kernel methods" (2016), PhD thesis for reference <https://ori-nuxeo.univ-lille1.fr/nuxeo/site/esupversions/789d119a-6763-4205-972f-b318cd4fdb27>.

License GPL (>= 2.0)

Imports Rcpp (>= 0.12.15)

LinkingTo Rcpp, RcppArmadillo

Depends proxy, kernlab, MASS

NeedsCompilation yes

Repository CRAN

Date/Publication 2018-02-05 18:18:21 UTC

R topics documented:

RaPKod-package *Random Projection Kernel Outlier Detector*

Description

The RaPKod package implements a kernel method made for outlier detection. Namely, given a data set of reference typical observation (non-outliers or inliers), it tests each new observation in an online way to determine whether it is an outlier or not. This method uses random low-dimensional projections in a kernel space to build a test statistic whose asymptotic null-distribution (ie when the tested observation is not an outlier) is known. The RaPKod method has two parameters: gamma - the hyperparameter of the (Gaussian) kernel used - and p - the dimensionality of the random projection in the kernel space.

Details

The package consists of two functions: the main function "rapkod" and the auxilary function "od.opt.param" which computes optimal parameters values in RaPKod.

Author(s)

Jeremie Kellner

Maintainer: Jeremie Kellner <jeremie.kellner@ed.univ-lille1.fr>

References

Kellner J., "Gaussian Models and Kernel Methods", PhD thesis, Universite des Sciences et Technologies de Lille (defended on December 1st, 2016)

See Also

rapkod, od.opt.param

od.opt.param *Optimal Parameter Values In RaPKod*

Description

Uses a heuristic formula to set optimal values for gamma and p.

Usage

```
od.opt.param(X, K1 = 6, K2 = 50, which.estim = "Gauss", RATIO = 0.1,
             randomize = TRUE, sub.n = floor(nrow(X)))
```

Arguments

X	a data frame or an n x d matrix.
K1	universal constant used in the heuristic formula of the optimal parameter gamma.
K2	universal constant used in the heuristic formula of the optimal parameter p.
which.estim	specifies the estimation method of the parameters: either "Gauss"(default) or "general".
RATIO	optional parameter used in estimation method "Gauss"
randomize	optional parameter used in the estimation method "general".
sub.n	optional parameter used in the estimation method "general" if randomize=TRUE.

Details

This function uses a heuristic formula to determine the optimal parameter values gamma and p, in the case when a Gaussian kernel is used. This formula is of the form $gamma = K1 * |f|_2^{2/(d+2)} * n^{1/(d+2)}$ and $p = ceil(K2 * |f|_2^{2/(d+2)} * n^{2/(d+2)})$, where $|f|_2$ is the L2-norm of the density function of non-outliers f and $ceil(x)$ denotes the smallest integer larger than x.

Two methods are proposed to estimate $|f|_2$ and are specified by the argument which.estim: "Gauss" and "general".

If which.estim="Gauss", the estimation is done as though f was a Gaussian density, which yields $|f|_2^{2/(d+2)}) = (4 * pi)^{-0.5} * exp(0.5 * mean(log(1/ev)))$, where ev are the covariance eigenvalues of the non-outlier distribution. Note that the eigenvalues smaller than $ev[1] * RATIO$ (where $ev[1]$ is the largest eigenvalue) are discarded to avoid numerical issues.

If which.estim="general", $|f|_2$ is estimated without any assumption on f. However this method may fail in very high dimensions because of the dimensionality curse, since it relies on an estimation of the derivative of F at 0 where F is the cdf of the pairwise distance between two non-outliers. . Besides, to shorten the computation time, the optional argument 'randomize' can be set as TRUE, so that only a subset of size sub.n of the data is considered to estimate the cdf F.

Value

gamma.opt	optimal value for gamma.
p.opt	optimal value for p.
est.f2.pw	estimation of $|f|_2^{2/(d+2)}$.

See Also

rapkod

Examples

```
data(iris)

##Define data frame with non-outliers
inliers = iris[sample(which(iris$Species!="setosa"), 100, replace=FALSE),
                                                  -which(names(iris)=="Species")]
```

```
param <- od.opt.param(inliers)

#display optimal gamma
param$gamma.opt
#display optimal p
param$p.opt
```

---

rapkod *RaPKod: Random Projections Kernel Outlier Detection*

Description

RaPKod is a kernel method for detecting outliers in a given dataset on the basis of a reference set of non-outliers. To do so, it 'transforms' a tested observation into some kernel space (through a 'feature map') and then projects it onto a random low-dimensional subspace of this kernel space. Since the distribution of this projection is known in the case of a non-outlier, it allows RaPKod to control the probability of false alarm error (ie labelling a non-outlier as an outlier).

Usage

```
  rapkod(X, given.kern = FALSE, ref.n=NULL, gamma=NULL,  p=NULL, alpha = 0.05,
use.tested.inlier = FALSE, lowrank = "No", r.lowrk = ceiling(sqrt(nrow(X))),
K1 = 6, K2 = 50)
```

Arguments

X	either a data frame or an n x d matrix (if given.kern=FALSE), otherwise an n x n kernel matrix (if given.kern=TRUE). In the former case, a Gaussian kernel is used by default.
given.kern	If FALSE (default), each row of X is an observation. Otherwise X is a kernel matrix (in this case, gamma and p must be user-specified).
ref.n	the size of the reference non-outlier dataset. Must be smaller than n.
gamma	the hyperparameter of the Gaussian kernel $k(x, y) = exp(-gamma*\|\|x-y\|\|^2)$. Set automatically by the program if not specified and given.kern=FALSE.
p	the number of dimensions of the projection made in the kernel space. Set automatically by the program if not specified and given.kern=FALSE.
alpha	the prescribed probability of false alarm error.
use.tested.inlier	If TRUE, each tested observation that is labelled as a non-outlier is appended to the reference dataset of non-outliers (the 'oldest' reference non-outlier is discarded). Set to FALSE by default.
lowrank	if lowrank="No" (default), the full kernel matrix is used. Otherwise, a low-rank approximation of the kernel matrix is computed: if "Nyst", it is approximated through Nystrom method; if "RKS", it is approximated by random Kitchen Sinks (in this case, X must be a dataset matrix, not a kernel matrix)

r.lowrk	if lowrank="Nyst" or "RKS", specifies the (low) rank of the approximated kernel matrix.
K1	universal constant used in the heuristic formula of the optimal parameter gamma.
K2	universal constant used in the heuristic formula of the optimal parameter p.

Details

If given.kern = FALSE, X is a dataset matrix whose first ref.n rows corresponds to the reference dataset of non-outliers. The (n - ref.n) other observations will be tested one by one by RaPKod to determine whether they are outliers or not.

If given.kern = TRUE, X must be a n x n Gram matrix. The kernel used to compute this Gram matrix should be of the form $k(x, y) = K(gamma * ||x - y||^2)$ where K is a positive function. Also note that in this case, the parameters gamma and p must be specified by the user.

Value

stats	a vector of length (n - ref.n) containing the test statistics for each tested observation.
flag	a vector of length (n - ref.n) indicating which observations have been labelled as an outlier (TRUE in this case).
pv	a vector of length (n - ref.n) containing p-values for each tested observation.
gamma	the optimal value of gamma determined by the program (or the value provided by the user if it was user-specified).
p	the optimal value of p determined by the program (or the value provided by the user if it was user-specified).

See Also

od.opt.param

Examples

```
data(iris)

##Define data frame with non-outliers
inliers = iris[sample(which(iris$Species!="setosa"), 100, replace=FALSE),
                                                  -which(names(iris)=="Species")]
##Define data frame with outliers
outliers = iris[which(iris$Species=="setosa"),-which(names(iris)=="Species")]


X = rbind(inliers, outliers)

ref.n = 50
result <- rapkod(X, ref.n = ref.n, use.tested.inlier = FALSE, alpha = 0.05)


##False alarm error ratio obtained on tested non-outliers (should be close to 0.05)
mean(result$pv[1:(nrow(inliers)-ref.n)]<0.05, na.rm = TRUE)
```

```
##Missed detection error ratio obtained on tested outliers (should be close to 0)
mean(result$pv[-(1:(nrow(inliers)-ref.n))]>0.05, na.rm = TRUE)
```

Index