Package 'phom'

July 2, 2014

Type Package

Title Persistent Homology in R

Version 1.0.3

Date 2011-11-02

Author Andrew Tausz

Maintainer Andrew Tausz <atausz@stanford.edu>

Description This package computes persistent homology of filtered simplicial complexes, and provides facilities for constructing complexes from geometric data.

License BSD_3_clause + file LICENSE

LazyLoad yes

Depends Rcpp (>= 0.9.7)

LinkingTo Rcpp

NeedsCompilation yes

Repository CRAN

Date/Publication 2014-02-09 21:44:08

R topics documented:

phom-package	*Persistent Homology in R*

Description

This package computes persistent homology of filtered simplicial complexes, and provides facilities for constructing complexes from geometric data. It has been developed by members of the computational topology workgroup at Stanford University. More information may be found at http://comptop.stanford.edu.

Details

Package:	phom
Type:	Package
Version:	1.0.3
Date:	11/5/11
License:	BSD
LazyLoad:	yes

The pipeline for computing the persistent homology consists of two stages: the construction of a filtered simplicial complex, and the computation of the homology of the filtered complex. This package has two methods for creating filtered simplicial complexes from geometric data:

- Vietoris-Rips mode: This uses the Vietoris-Rips construction to build a sequence of simplicial complexes on the underlying point-set. Note that this construction may produce complexes that have a very large number of simplices in them.
- Lazy-Witness mode: This uses the lazy-witness construction to build a filtered simplicial complex on a subset of the points. This construction has the advantage of reducing the noise in the resulting set of intervals, and is considerably faster than the Vietoris-Rips filtration.

The algorithms operate on two types of data:

- Euclidean Data: This assumes that the point set is represented as a matrix, where each point is a row. The number of columns is the dimensionality of the dataset. A wide variety of metrics are available on this sort of data.
- Distance Matrix: For more general metric spaces, one may specify the interpoint distances between all of the points. Note that the algorithms do not check that this indeed specifies a metric.

Author(s)

Andrew Tausz <atausz@stanford.edu>

References

Gunnar Carlsson. Topology and Data. Bull. Amer. Math. Soc. 46 (2009), 255-308.

Examples

```
### Example 1

library(phom)

x <- runif(100)
y <- runif(100)
points <- t(as.matrix(rbind(x, y)))

max_dim <- 2
max_f <- 0.2

intervals <- pHom(points, max_dim, max_f, metric="manhattan")

plotPersistenceDiagram(intervals, max_dim, max_f,
title="Random Points in Cube with l_1 Norm")


### Example 2

library(phom)

t <- 2 * pi * runif(100)
x <- cos(t); y <- sin(t)
X <- t(as.matrix(rbind(x, y)))

max_dim <- 1
max_f <- 0.6

intervals <- pHom(X, max_dim, max_f)

plotPersistenceDiagram(intervals, max_dim, max_f,
title="Random Points on S^1 with Euclidean Norm")
```

---

pHom   *Compute Persistent Homology*

Description

This function computes persistent homology on a given dataset. This is a two-step process which involves 1) creating a filtered simplicial complex on the dataset, and 2) computing the persistent homology of the filtered complex.

It outputs a matrix with three columns. Each row in the output matrix corresponds to a persistence interval. The first column stores the dimension of the interval, the second stores the starting point, and the third stores the ending point.

The method provides two ways to construct a filtered simplicial complex on the data points: the Vietoris-Rips construction, and the lazy-witness construction.

The Vietoris-Rips construction on a metric space, (X, d), is performed as follows.

- For each $u \in X$, $u \in VR(X, 0)$
- For each $u, v \in X$, $[u, v] \in VR(X, r)$ if and only if $d(u, v) \leq r$
- $VR(X, r)$ is maximal among all simpicial complexes containing its 1-skeleton. In other words, $[v_0, \ldots v_n] \in VR(X, r)$ if and only if for all pairs, $[v_i, v_j] \in VR(X, r)$.

Note that the Vietoris-Rips construction can produce complexes that are very large. To circumvent this, the witness construction may be used. This method produces complexes that are significantly smaller.

The lazy-witness construction is outlined as follows. Suppose that we have a metric space, (X, d), and a subset of points, $L \subset X$, which we refer to as the landmark set. The lazy-witness complex, parameterized by ν is defined as follows:

- The vertex set of $LW_\nu(X, L, r)$ is L.
- The edge $[u, v]$ is in $LW_\nu(X, L, r)$ if there exists a witness, z, such that $\max(d(u, z), d(v, z)) \leq r + m(z)$. Here $m(z)$ is the distance from z to the ν-th closest point in the landmark set.
- A higher dimensional simplex is in $LW_\nu(X, L, r)$ if all of its edges are.

The landmark set selection process is performed as follows. Initially a random point is added to L. Inductively, points are added which maximize the minimum distance to the existing landmark set.

$$L_{i+1} = L_i \cup \arg\max_{x \in X}(\min_{y \in L_i} d(x, y))$$

The argument mode is used to select between the two different constructions, by either specififying "vr" or "lw".

Usage

```
pHom(X, dimension, max_filtration_value,
mode = "vr", metric = "euclidean", p = 2,
landmark_set_size = 2 * ceiling(sqrt(length(X))),
maxmin_samples = min(1000, length(X)))
```

Arguments

X   A matrix which has one of the two following interpretations. In the case where metric = "distance_matrix", X is required to be a square matrix whose entries indicate the distance between two points. The number of rows (and columns) equals the number of points in the dataset. In the case where metric is something other than "distance_matrix", X is required to be a matrix in which the rows are points in Euclidean space. The number of columns is the dimensionality of the dataset. Note that for different choices of the metric argument, the points will be endowed with different metrics.

dimension   The maximum dimension to compute persistent homology to.

max_filtration_value

    The maximum filtration value to use in constructing the filtered complex.

mode   This indicates the type of filtration to use. The two possible choices are "vr" (default) and "lw". The choice "vr" indicates that the Vietoris-Rips filtration will be used, and the choice "lw" indicates that the lazy-witness construction will be used. For Vietoris-Rips filtrations, the parameters landmark_set_size and maxmin_samples are ignored.

metric   This indicates the type of metric that will be used. Valid choices include the following: "distance_matrix", "euclidean", "maximum", "manhattan", "canberra", "binary", When "distance_matrix" is specified, the paramter X is interpreted as a distance matrix. Otherwise, X is regarded as a set of points, where each row is one point.

p   This is the value of the power to use in the minkowski metric.

landmark_set_size

    The number of points to include in the landmark set. A sensible value for this is in between 20 and 100. The default value is taken to be $2\sqrt{|X|}$. This parameter is only relevant for the lazy-witness filtration.

maxmin_samples   The number of samples to use when performing the maxmin selection. The default value is taken to be $\min(|X|, 1000)$. This parameter is only relevant for the lazy-witness filtration.

---

plotBarcodeDiagram   *Create a barcode diagram plot from a set of intervals*

Description

This function plots a barcode diagram from a given set of intervals. It creates a slightly different plot from [plotPersistenceDiagram](). As opposed to plotting the starting and ending points as the horizontal and vertical coordinates of points in the plane, this function displays a list of barcodes.

Usage

```
plotBarcodeDiagram(intervals, dimension, max_f, title="Barcode Diagram")
```

Arguments

intervals   A matrix with three columns that specifies the persistence intervals. Entries in the first column indicate the dimension of an interval. The entries in the second and third columns indicate the start and end points of the intervals, respectively. The function [pHom]() produces outputs that are in this form.

dimension   The dimension to plot intervals for. Unlike [plotPersistenceDiagram](), this function only plots intervals for one dimension, and not all of them.

max_f   The maximum filtration value to use in the persistence diagram.

title   The title on the barcode diagram.

plotPersistenceDiagram

Create a persistence diagram plot from a set of intervals

Description

This function plots a persistence diagram from a given set of intervals. A persistence diagram is a set of unordered points $\{(x_i, y_i)|y_i \geq x_i\}$ for each dimension, where x_i indicates the starting point of interval i and y_i indicates the ending point of interval i. This function plots these points as coordinates in the plane, using difference colors and markers for each dimension.

We warn the user that this function may result in two identical points occluding each other. In other words, the multiplicity of each interval is not clearly visible. The function plotBarcodeDiagram does not suffer from this.

Usage

```
plotPersistenceDiagram(intervals, max_dim, max_f, title="Persistence Diagram")
```

Arguments

intervals	A matrix with three columns that specifies the persistence intervals. Entries in the first column indicate the dimension of an interval. The entries in the second and third columns indicate the start and end points of the intervals, respectively. The function pHom produces outputs that are in this form.
max_dim	The maximum dimension to plot. Note that unlike plotBarcodeDiagram, this function plots points (intervals) for all dimensions up to the maximum dimension.
max_f	The maximum filtration value to use in the persistence diagram.
title	The title on the persistence diagram.

Index