sDmat

February 12, 2015

`sDmat` *Function to calculate distance matrix in high-dimensional input space but according to neighborhood relationships in 2D output space*

Description

`sDmat` is supposed to calculate distance (measured in high-dimensional input space) to neighbors (defined by based on 2D output space) for each of hexagons/rectangles

Usage

```
sDmat(sMap, which_neigh = 1, distMeasure = c("median", "mean", "min",
"max"))
```

Arguments

`sMap`	an object of class "sMap"
`which_neigh`	which neighbors in 2D output space are used for the calculation. By default, it sets to "1" for direct neighbors, and "2" for neighbors within neighbors no more than 2, and so on
`distMeasure`	distance measure used to calculate distances in high-dimensional input space

Value

- `dMat`: a vector with the length of nHex. It stores the distance a hexaon/rectangle is away from its output-space-defined neighbors in high-dimensional input space

Note

"which_neigh" is defined in output 2D space, but "distMeasure" is defined in high-dimensional input space

See Also

`sNeighAny`

Examples

```
# 1) generate an iid normal random matrix of 100x10
data <- matrix( rnorm(100*10,mean=0,sd=1), nrow=100, ncol=10)

# 2) get trained using by default setup
sMap <- sPipeline(data=data)

# 3) calculate "median" distances in INPUT space to different neighbors in 2D OUTPUT space
# 3a) using direct neighbors in 2D OUTPUT space
dMat <- sDmat(sMap=sMap, which_neigh=1, distMeasure="median")
# 3b) using no more than 2-topological neighbors in 2D OUTPUT space
# dMat <- sDmat(sMap=sMap, which_neigh=2, distMeasure="median")
```