Package ‘N2R’

November 19, 2021

Type Package
Title Fast and Scalable Approximate k-Nearest Neighbor Search Methods using 'N2' Library
Version 1.0.0
Description
License Apache License 2.0
Encoding UTF-8
Depends Matrix
Imports Rcpp (>= 1.0.4)
Suggests testthat
LinkingTo Rcpp, RcppSpdlog, RcppEigen
SystemRequirements GNU make
RoxygenNote 7.1.2
URL https://github.com/kharchenkolab/N2R
BugReports https://github.com/kharchenkolab/N2R/issues
NeedsCompilation yes
Author Peter Kharchenko [aut], Viktor Petukhov [aut], Dirk Eddelbuettel [ctb], Evan Biederstedt [cre, aut]
Maintainer Evan Biederstedt <evan.biederstedt@gmail.com>
Repository CRAN
Date/Publication 2021-11-19 20:50:06 UTC
R topics documented:

checkOpenMP ........................................... 2
crossKnn .................................................. 2
Knn ......................................................... 3

Description

boolean to check OpenMP exists

Usage

checkOpenMP()

crossKnn  

Perform fast approximate K-nearest neighbor search of rows input matrix mA in rows of matrix mB.

Description

Perform fast approximate K-nearest neighbor search of rows input matrix mA in rows of matrix mB.

Usage

crossKnn(
    mA,
    mB,
    k,
    nThreads = 10L,
    verbose = TRUE,
    indexType = "angular",
    M = 12L,
    MaxM0 = 24L,
    ef_search_multiplier = 50,
    quiet = FALSE
)
**Knn**

**Arguments**

- **mA**: Input numeric matrix of data
- **mB**: Input numeric matrix of data
- **k**: Integer number of clusters
- **nThreads**: Integer number of threads (default=10)
- **verbose**: Boolean flag for verbose output (default=FALSE)
- **indexType**: Metric distance type, which can be "angular" or "L2" (default="angular")
- **M**: Integer number of connections (default=12) The NSW graph is constructed via consecutive insertion of elements in random order by bidirectionally connecting them to the M closest neighbors from the previously inserted elements.
- **MaxM0**: Integer maximum number of connections that an element can have in the zero layer. (default=24) It is recommended that MaxM0 not exceed 2*M.
- **ef_search_multiplier**: Integer multiplier to calculate candidate nearest neighbors, set to k*ef_search_multiplier (default=50). Refer to the parameters er and efConstruction in Malkov & Yashunin (2020) doi: 10.1109/TPAMI.2018.2889473
- **quiet**: Boolean flag specifically for Rcpp warnings (default=FALSE)

**Value**

clusters per row in sparse Matrix of class "dgCMatrix" of dimensions mB rows by mA rows

**Examples**

```r
data(iris)
iris_df = data.matrix(iris[-5]) ## convert to a numeric matrix
crossKnn(mA=iris_df, mB=head(iris_df, 50), 4)
```

---

**Knn**

Perform fast approximate K-nearest neighbor search on rows of the input matrix m.

**Description**

Perform fast approximate K-nearest neighbor search on rows of the input matrix m.

**Usage**

```r
Knn(
  m,
  k,
  nThreads = 10L,
  verbose = TRUE,
)```
Knn

```r
indexType = "angular",
M = 12L,
MaxM0 = 24L,
ef_search_multiplier = 50,
quiet = FALSE
)
```

**Arguments**

- `m`: Input numeric matrix of data
- `k`: Integer number of clusters
- `nThreads`: Integer number of threads (default=10)
- `verbose`: Boolean flag for verbose output (default=FALSE)
- `indexType`: Metric distance type, which can be "angular" or "L2" (default="angular")
- `M`: Integer number of connections (default=12) The NSW graph is constructed via consecutive insertion of elements in random order by bidirectionally connecting them to the M closest neighbors from the previously inserted elements.
- `MaxM0`: Integer maximum number of connections that an element can have in the zero layer. (default=24) It is recommended that MaxM0 not exceed 2*M.
- `ef_search_multiplier`: Integer multiplier to calculate candidate nearest neighbors, set to k*ef_search_multiplier (default=50). Refer to the parameters er and efConstruction in Malkov & Yashunin (2020) doi: 10.1109/TPAMI.2018.2889473
- `quiet`: Boolean flag specifically for Rcpp warnings (default=FALSE)

**Value**

clusters per row in sparse Matrix of class "dgCMatrix" of dimensions m rows by m rows

**Examples**

```r
data(iris)
iris_df = data.matrix(iris[-5]) ## convert to a numeric matrix
Knn(m=iris_df, 4)
```
Index

checkOpenMP, 2

crossKnn, 2

Knn, 3