

# Package ‘BigKnn’

March 14, 2023

**Type** Package

**Title** Large Scale K-Nearest Neighbor Classifier using the Lucene Search Engine

**Version** 1.0.2

**Date** 2023-03-14

**Maintainer** Martijn Schuemie <schuemie@ohdsi.org>

**Description** A large scale k-nearest neighbor classifier using the Lucene search engine.

**SystemRequirements** Java version 8 or higher (<https://www.java.com/>)

**Imports** rJava,  
Andromeda (>= 0.3.0),  
dplyr,  
rlang

**Suggests** testthat

**License** Apache License

**RoxygenNote** 7.2.3

**URL** <https://ohdsi.github.io/BigKnn>, <https://github.com/OHDSI/BigKnn>

**BugReports** <https://github.com/OHDSI/BigKnn/issues>

**Encoding** UTF-8

## R topics documented:

buildKnn	2
buildKnnFromPlpData	3
predictKnn	3
predictKnnUsingPlpData	4
<b>Index</b>	<b>6</b>

---

buildKnn	<i>Build a K-nearest neighbor (KNN) classifier</i>
----------	--

---

## Description

buildKnn loads data from two Andromeda tables, and inserts them into a KNN classifier.

## Usage

```
buildKnn(outcomes, covariates, indexFolder, overwrite = TRUE)
```

## Arguments

outcomes	An Andromeda table containing the outcomes with predefined columns (see below).
covariates	An Andromeda table containing the covariates with predefined columns (see below).
indexFolder	Path to a local folder where the KNN classifier index can be stored.
overwrite	Automatically overwrite if an index already exists?

## Details

These columns are expected in the outcome object:

rowId	(integer)	Row ID is used to link multiple covariates (x) to a single outcome (y)
y	(real)	The outcome variable

These columns are expected in the covariates object:

rowId	(integer)	Row ID is used to link multiple covariates (x) to a single outcome (y)
covariateId	(integer)	A numeric identifier of a covariate
covariateValue	(real)	The value of the specified covariate

## Value

Nothing

---

buildKnnFromPlpData	<i>Build a K-nearest neighbor (KNN) classifier from a plpData object</i>
---------------------	--

---

### Description

Build a K-nearest neighbor (KNN) classifier from a plpData object

### Usage

```
buildKnnFromPlpData(  
  plpData,  
  population,  
  indexFolder,  
  overwrite = TRUE,  
  cohortId = NULL,  
  outcomeId = NULL  
)
```

### Arguments

plpData	An object of type plpData.
population	The population.
indexFolder	Path to a local folder where the KNN classifier index can be stored.
overwrite	Automatically overwrite if an index already exists?
cohortId	The ID of the specific cohort for which to fit a model.
outcomeId	The ID of the specific outcome for which to fit a model.

---

predictKnn	<i>Predict using a K-nearest neighbor (KNN) classifier</i>
------------	--

---

### Description

predictKnn uses a KNN classifier to generate predictions.

### Usage

```
predictKnn(  
  cohorts,  
  covariates,  
  indexFolder,  
  k = 1000,  
  weighted = TRUE,  
  threads = 1  
)
```

**Arguments**

cohorts	An Andromeda table containing the cohorts with predefined columns (see below).
covariates	An Andromeda table containing the covariates with predefined columns (see below).
indexFolder	Path to a local folder where the KNN classifier index can be stored.
k	The number of nearest neighbors to use to predict the outcome.
weighted	Should the prediction be weighted by the (inverse of the ) distance metric?
threads	Number of parallel threads to used for the computation.

**Details**

These columns are expected in the covariates object:

rowId	(integer)	Row ID is used to link multiple covariates (x) to a single outcome (y)
covariateId	(integer)	A numeric identifier of a covariate
covariateValue	(real)	The value of the specified covariate

This column is expected in the covariates object:

rowId	(integer)	Row ID is used to link multiple covariates (x) to a single outcome (y)
-------	-----------	--

**Value**

A data.frame with two columns:

rowId	(integer)	Row ID is used to link multiple covariates (x) to a single outcome (y)
prediction	(real)	A number between 0 and 1 representing the probability of the outcome

---

predictKnnUsingPlpData

*Create predictive probabilities using KNN.*

---

**Description**

Create predictive probabilities using KNN.

**Usage**

```
predictKnnUsingPlpData(
  plpData,
  population,
  indexFolder,
  k = 1000,
  weighted = TRUE,
  threads = 10
)
```

**Arguments**

<code>plpData</code>	An object of type <code>plpData</code> as generated using <code>getDbPlpData</code> .
<code>population</code>	The population to predict for.
<code>indexFolder</code>	Path to a local folder where the KNN classifier index is be stored.
<code>k</code>	The number of nearest neighbors to use to predict the outcome.
<code>weighted</code>	Should the prediction be weighed by the (inverse of the ) distance metric?
<code>threads</code>	Number of parallel threads to used for the computation.

**Details**

Generates predictions for the population specified in `plpData`.

**Value**

The value column in the result `data.frame` is: `logistic`: probabilities of the outcome, `poisson`: Poisson rate (per day) of the outcome, `survival`: hazard rate (per day) of the outcome.

# Index

buildKnn, [2](#)  
buildKnnFromPlpData, [3](#)  
predictKnn, [3](#)  
predictKnnUsingPlpData, [4](#)