

# Package ‘ksNN’

October 13, 2022

**Encoding** UTF-8

**Type** Package

**Title** K\* Nearest Neighbors Algorithm

**Version** 0.1.2

**Description** Prediction with k\* nearest neighbor algorithm  
based on a publication by Anava and Levy (2016) <[arXiv:1701.07266](https://arxiv.org/abs/1701.07266)>.

**License** GPL (>= 2)

**Author** Kei Nakagawa [aut, cre] (<<https://orcid.org/0000-0001-5046-8128>>),  
Shingo Sashida [aut]

**Maintainer** Kei Nakagawa <[kei.nak.0315@gmail.com](mailto:kei.nak.0315@gmail.com)>

**LazyData** TRUE

**Imports** Rcpp

**Depends** R(>= 3.0.2)

**LinkingTo** Rcpp(>= 0.10.6)

**RoxygenNote** 6.1.1

**NeedsCompilation** yes

**Repository** CRAN

**Date/Publication** 2019-04-11 12:02:37 UTC

## R topics documented:

ksNN . . . . .	2
rcpp_ksNN . . . . .	3
<b>Index</b>	<b>5</b>

---

ksNN	<i>This function calculates the prediction value of k* nearest neighbors algorithm.</i>
------	---

---

**Description**

This function calculates the prediction value of k\* nearest neighbors algorithm.

**Usage**

```
ksNN(Label, Distance, L_C = 1)
```

**Arguments**

Label	vectors of the known labels of the samples.
Distance	vectors of the distance between the target sample we want to predict and the other samples.
L_C	parameter of k* nearest neighbors algorithm.

**Value**

the prediction value(pred) and the weight of the samples(alpha).

**Note**

This algorithm is based on Anava and Levy(2017).

**Examples**

```
library(ksNN)
set.seed(1)

#make the nonlinear regression problem
X<-runif(100)
Y<-X^6-3*X^3+5*X^2+2

suffle<-order(rnorm(length(X)))
X<-X[suffle]
Y<-Y[suffle]

test_X<-X[1]
test_Y<-Y[1]

train_X<-X[-1]
train_Y<-Y[-1]

Label<-train_Y
Distance<-sqrt((test_X-train_X)^2)
```

```

pred_ksNN<-ksNN(Label,Distance,L_C=1)

#the predicted value with k*NN
pred_ksNN$pred

#the 'true' value
test_Y

```

---

rcpp_ksNN	<i>This function calculates the prediction value of k* nearest neighbors algorithm.</i>
-----------	---

---

### Description

This function calculates the prediction value of k\* nearest neighbors algorithm.

### Usage

```
rcpp_ksNN(Label, Distance, L_C = 1)
```

### Arguments

Label	vectors of the known labels of the samples.
Distance	vectors of the distance between the target sample we want to predict and the other samples.
L_C	parameter of k* nearest neighbors algorithm.

### Value

the prediction value(pred) and the weight of the samples(alpha).

### Note

This algorithm is based on Anava and Levy(2017).

### Examples

```

library(ksNN)
set.seed(1)

#make the nonlinear regression problem
X<-runif(100)
Y<-X^6-3*X^3+5*X^2+2

suffle<-order(rnorm(length(X)))
X<-X[suffle]
Y<-Y[suffle]

test_X<-X[1]

```

```
test_Y<-Y[1]

train_X<-X[-1]
train_Y<-Y[-1]

Label<-train_Y
Distance<-sqrt((test_X-train_X)^2)

pred_ksNN<-rcpp_ksNN(Label,Distance,L_C=1)

#the predicted value with k*NN
pred_ksNN$pred

#the 'true' value
test_Y
```

# Index

ksNN, [2](#)

rcpp\_ksNN, [3](#)