Journal of Computer Allied Intelligence
ISSN: 2584-2676
Vol.02 (01), Feb 2024, pp.8-15

Research Article

Exploring the Power and Practical Applications of K-Nearest Neighbours (KNN) in Machine Learning

Venkateswarlu B[1,*] and Rekha Gangula[2]

[1]Assistant Professor, Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Green Fields, Vaddeswaram, Andhra Pradesh, India.

[2]Assistant Professor, Computer Science and Engineering, Vaagdevi Engineering College, Bollikunta, Warangal, Telangana, India. gangularekha@gmail.com

[*]Corresponding Author: Dr. Venkateswarlu B. Email: bvenki289@gmail.com

Received: 01/02/2024; Accepted: 20/02/2024

DOI: https://doi.org/10.69996/jcai.2024002

Abstract: Artificial intelligence's main component, machine learning, enables systems to learn on their own and improve performance via experience, doing away with the need for explicit programming. This cutting-edge field focuses on equipping computer programs with the ability to access vast datasets and derive intelligent decisions from them. One of the cornerstone algorithms in machine learning, the K-nearest neighbours (KNN) algorithm, is known for its simplicity and effectiveness. KNN leverages the principle of storing all available data points within its training dataset and subsequently classifying new, unclassified cases based on their similarity to the existing dataset. This proximity-based classification approach renders KNN a versatile and intuitive tool with applications spanning diverse domains. This document explores the inner workings of the K-nearest neighbours' algorithm, its practical applications across various domains, and a comprehensive examination of its strengths and limitations. Additionally, it offers insights into practical considerations and best practices for the effective implementation of KNN, illuminating its significance in the continually evolving landscape of machine learning and artificial intelligence.

Keywords: Machine learning; k-nearest neighbours; artificial intelligence; knn; cuting edge field

1 Introduction

An essential component of artificial intelligence (AI) is machine learning, which enables computers to learn on their own and improve performance through experience absorption without explicit programming. At its core, machine learning concentrates on the creation of computer programs that can not only access vast datasets but also adapt and learn from these datasets to make intelligent decisions. One of the fundamental algorithms in machine learning, known for its simplicity and effectiveness, is the K-nearest neighbours (KNN) algorithm. KNN operates on the principle of storing all available data points or cases within its training dataset and subsequently classifying new, unclassified cases based on their similarity to the existing dataset. This approach is rooted in the concept of proximity-based classification, making it a versatile and intuitive method for various applications. In the following sections, we will delve into the mechanics of the K-nearest neighbours' algorithm, explore its applications across different domains, and examine its strengths and limitations. We will also discuss practical considerations and best practices for implementing KNN effectively, shedding light on how it contributes to the ever-evolving landscape of machine learning and artificial intelligence.

2 Related Works

Determine the separation between x and every point in your dataset. Sort the data points in your collection by becoming farther away from x. Assume that the k nearest points have the majority label. Keep in mind that the value of k affects the results; therefore, for better results and a better model, it is recommended to test the model for multiple values of k.

Data

The UCI Glass Identification Database has ten properties, one of which is id. There are seven discrete values in the glass type response.

Attributes

Id: 1 to 214 (removed from CSV file)

RI: refractive index

Na: Sodium (unit measurement: weight percent in corresponding oxide, as are attributes 4-10)

Si: Silicon

K: Potassium

Mg: Magnesium

Ca: Calcium

Ba: Barium

Al: Aluminum

Fe: Iron

Type of glass: (Class Attribute)

1 - building_windows_float_processed

2 - building_windows_non_float_processed

3 - vehicle_windows_float_processed

4 - vehicle_windows_non_float_processed (none in this database)

5 - containers

6 - tableware

7 - headlamps

2.1 Literature Review

The provided work appears to be the beginning of a research paper or article authored by Stefan Securing, focusing on conducting content analysis based on literature reviews related to the identification of glass [1]. However, the text also mentions supply chain management (SCM) literature reviews and discusses the importance of transparent and systematic procedures in research. It's important to note that the context of the text is somewhat unclear, as it transitions from discussing SCM literature reviews to addressing issues with the quality of literature review processes. Glass Identification review [2] provided to discusses the content of a research paper or a data analysis, highlighting the dataset, data preprocessing, and the evaluation methodology employed. It also mentions the use of specific algorithms, C4.5 and K-Means clustering, as well as the handling of missing values in the context of K-Nearest Neighbors (K-NN)[5].Here's

Dataset Description and Preprocessing:

Data preprocessing was conducted to check for missing values in the dataset.The dataset was transformed into ARFF format, a standard representation for datasets with independent, unordered instances.

Evaluation Methodology:

Here we used 10-fold Cross-validation, a common technique for evaluating models. Ten subsets of the dataset were created, with half of them being utilised for training and the other half for testing. The use of Cross-validation helps avoid overlapping test sets. Stratified cross-validation was repeated 10 times to reduce variance and provide an accurate estimate of performance [9].

Algorithms Used:

C4.5: A decision tree algorithm used for classification.

K-Means Clustering: A method for unsupervised learning that groups data according to similarities. It aims to find K clusters in the data.

K-NN (K-Nearest Neighbours): An instance-based learning algorithm that handles missing values differently from C4.5. By comparing the difference between the new instance and the current data points, K-NN addresses missing values [4]. The new instance is assigned the majority class of its nearest K neighbours. An overview of the research's data and methodology is given here, with an emphasis on the preparation of data and the application of machine learning techniques to analysis. Particular-Based Regression by Partitioning Feature Projections (RPFP), a new instance-based learning technique created for regression issues with high-dimensional data, is discussed in By Partitioning Feature Projections[3][7]. This approach seeks to obtain high accuracy on regression issues, beating the popular instance-based method K-Nearest Neighbours (KNN) and traditional eager approaches like MARS, rule-based regression, and regression tree induction systems. RPFP is particularly effective when dealing with domains that have many missing values in the training data.

Challenges in Regression Problems: The text points out that while KNN is popular for classification, it doesn't perform as effectively in regression problems' Introduction: RPFP is introduced as a new instance-based approach that excels in regression tasks. Handling Interactions: RPFP is unique in its ability to handle interactions among features, making it highly adaptable to real-world regression problems. Main Effects vs. Interactions: The text acknowledges that in many regression scenarios, main effects are more prevalent than interactions. RPFP is designed to accommodate these situations effectively. Training Process: RPFP's training process involves storing training data as projections to the features and associating target values with feature dimensions. Instances are sorted based on their feature values for each dimension. Advantage of Local Weights: RPFP assigns lower local weights to features, making it robust when dealing with target values in query locations. Handling Irrelevant Features: The text notes that RPFP is not significantly affected by irrelevant features, in contrast to KNN, which struggles with such features.

3. Methodology

Proposed methodology describes the dataset used in a study involving the classification of types of glass. Here's a breakdown:

3.1 Problem Statement

The problem is to predict the age of abalone (a type of marine mollusk) from physical measurements. Traditionally, abalone age determination is a time-consuming process that involves cutting the shell, staining it, and counting rings under a microscope.

The objective is to predict age using easier-to-obtain measurements, which might include physical characteristics. Additional information, such as weather patterns and location, could potentially aid in solving this problem.

3.2 Dataset Description

A comparison test was conducted to evaluate different approaches, including a rule-based system called BEAGLE, the nearest-neighbour (NN) algorithm, and discriminant analysis (DA).

The test involved classifying glass samples as either "float" glass or not.

The results for the number of incorrect answers are provided for each approach.

The study is motivated by its potential application in criminological investigations, where correctly identifying the type of glass left at a crime scene is crucial evidence.

Attribute Description:

The dataset includes the following attributes:

Id number: An identifier from 1 to 214,RI (refractive index),Na (Sodium, measured in weight percent in corresponding oxide),Mg (Magnesium),Al (Aluminium),Si (Silicon),K (Potassium)

Ca (Calcium),Ba (Barium),Fe (Iron).

Type of glass: This is the class attribute and includes categories such as building_windows_non_float_processed, building_windows_float_processed, vehicle_windows_non_float_processed (not present in the database), vehicle_windows_float_processed, tableware, containers, and headlamps.The dataset appears to be used for classification tasks related to the type of glass, with attributes describing its composition and refractive index.

4. Results and Discussion

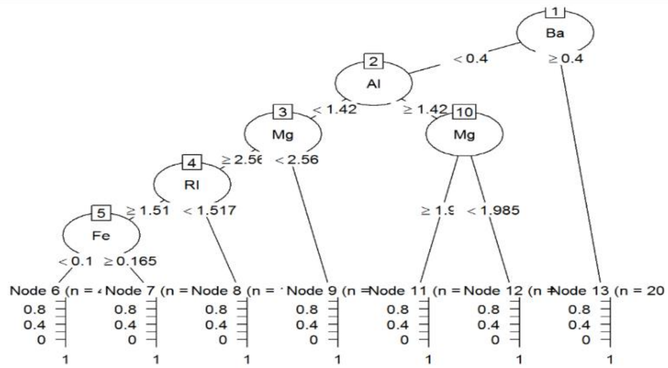
4.1 Pre-processing

```
##  Factor w/ 6 levels "1","2","3","5",..: 1 1 1 1 1 1 1 1 1 1 ...
```

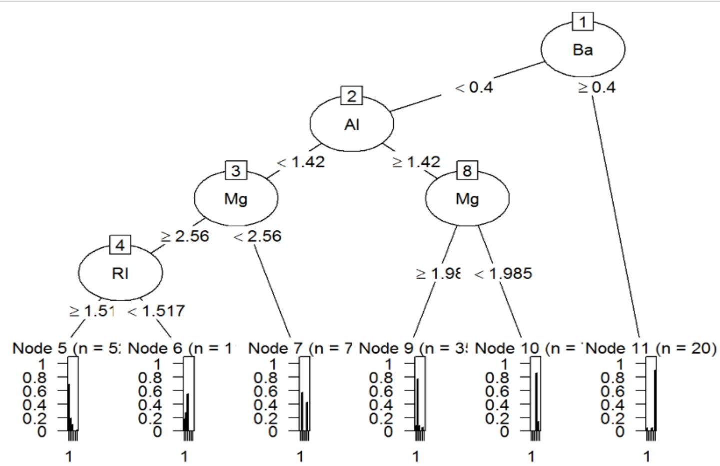
```
set.seed(123)
tree.glass = rpart(Type~., data=train)
print(tree.glass$cptable)
```

```
##           CP nsplit rel error    xerror       xstd
## 1 0.22159091      0 1.0000000 1.0909091 0.05814565
## 2 0.06818182      2 0.5568182 0.6022727 0.06400028
## 3 0.04545455      3 0.4886364 0.6136364 0.06419106
## 4 0.02272727      5 0.3977273 0.4886364 0.06118706
## 5 0.01000000      6 0.3750000 0.5454545 0.06280448
```

```
cp = min(tree.glass$cptable[4,])
prune.tree.glass = prune(tree.glass, cp = cp)
plot(as.party(tree.glass))
```




```
plot(as.party(prune.tree.glass))
```



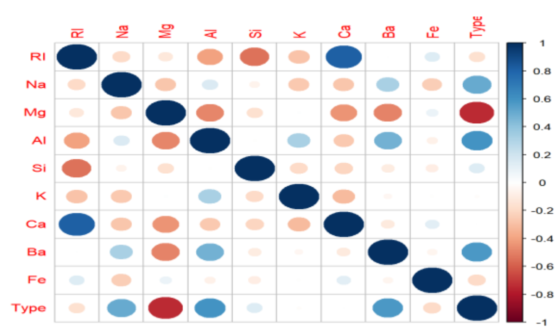
4.2 Data Analysis

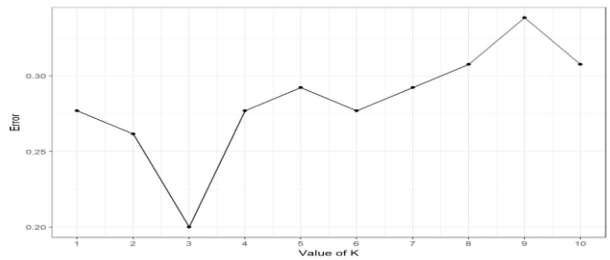
```
> summary(data)
       X              V1            V2              V3               V4               V5               V6
 Min.   :   1   Female:1307   Min.   :0.075   Min.   :0.0550   Min.   :0.0000   Min.   :0.0020   Min.   :0.0010
 1st Qu.:1045   Infant:1342   1st Qu.:0.450   1st Qu.:0.3500   1st Qu.:0.1150   1st Qu.:0.4415   1st Qu.:0.1860
 Median :2089   Male  :1528   Median :0.545   Median :0.4250   Median :0.1400   Median :0.7995   Median :0.3360
 Mean   :2089                 Mean   :0.524   Mean   :0.4079   Mean   :0.1395   Mean   :0.8287   Mean   :0.3594
 3rd Qu.:3133                 3rd Qu.:0.615   3rd Qu.:0.4800   3rd Qu.:0.1650   3rd Qu.:1.1530   3rd Qu.:0.5020
 Max.   :4177                 Max.   :0.815   Max.   :0.6500   Max.   :1.1300   Max.   :2.8255   Max.   :1.4880
       V7               V8             V9
 Min.   :0.0005   Min.   :0.0015   Old  :2406
 1st Qu.:0.0935   1st Qu.:0.1300   Young: 364
 Median :0.1710   Median :0.2340   Adult:1407
 Mean   :0.1806   Mean   :0.2388
 3rd Qu.:0.2530   3rd Qu.:0.3290
 Max.   :0.7600   Max.   :1.0050
```

4.3 Data Splitting

```
> train
     X     V1    V2    V3    V4     V5     V6     V7     V8    V9
2    2   Male 0.350 0.265 0.090 0.2255 0.0995 0.0485 0.0700 Adult
3    3 Female 0.530 0.420 0.135 0.6770 0.2565 0.1415 0.2100   old
4    4   Male 0.440 0.365 0.125 0.5160 0.2155 0.1140 0.1550   old
6    6 Infant 0.425 0.300 0.095 0.3515 0.1410 0.0775 0.1200 Adult
7    7 Female 0.530 0.415 0.150 0.7775 0.2370 0.1415 0.3300 Young
9    9   Male 0.475 0.370 0.125 0.5095 0.2165 0.1125 0.1650   old
10  10 Female 0.550 0.440 0.150 0.8945 0.3145 0.1510 0.3200 Young
11  11 Female 0.525 0.380 0.140 0.6065 0.1940 0.1475 0.2100   old
12  12   Male 0.430 0.350 0.110 0.4060 0.1675 0.0810 0.1350   old
15  15 Female 0.470 0.355 0.100 0.4755 0.1675 0.0805 0.1850   old
17  17 Infant 0.355 0.280 0.085 0.2905 0.0950 0.0395 0.1150 Adult
19  19   Male 0.365 0.295 0.080 0.2555 0.0970 0.0430 0.1000 Adult
22  22 Infant 0.380 0.275 0.100 0.2255 0.0800 0.0490 0.0850   old
26  26 Female 0.560 0.440 0.140 0.9285 0.3825 0.1880 0.3000   old
27  27 Female 0.580 0.450 0.185 0.9955 0.3945 0.2720 0.2850   old
29  29   Male 0.605 0.475 0.180 0.9365 0.3940 0.2190 0.2950 Young
30  30   Male 0.575 0.425 0.140 0.8635 0.3930 0.2270 0.2000   old
31  31   Male 0.580 0.470 0.165 0.9975 0.3935 0.2420 0.3300   old
32  32 Female 0.680 0.560 0.165 1.6390 0.6055 0.2805 0.4600 Young
34  34 Female 0.680 0.550 0.175 1.7980 0.8150 0.3925 0.4550 Young
35  35 Female 0.705 0.550 0.200 1.7095 0.6330 0.4115 0.4900   old
36  36   Male 0.465 0.355 0.105 0.4795 0.2270 0.1240 0.1250 Adult
38  38 Female 0.450 0.355 0.105 0.5225 0.2370 0.1165 0.1450 Adult
39  39 Female 0.575 0.445 0.135 0.8830 0.3810 0.2035 0.2600   old
40  40   Male 0.355 0.290 0.090 0.3275 0.1340 0.0860 0.0900   old
41  41 Female 0.450 0.335 0.105 0.4250 0.1865 0.0910 0.1150   old
42  42 Female 0.550 0.425 0.135 0.8515 0.3620 0.1960 0.2700   old
43  43 Infant 0.240 0.175 0.045 0.0700 0.0315 0.0235 0.0200 Adult
44  44 Infant 0.205 0.150 0.055 0.0420 0.0255 0.0150 0.0120 Adult
45  45 Infant 0.210 0.150 0.050 0.0420 0.0175 0.0125 0.0150 Adult
47  47   Male 0.470 0.370 0.120 0.5795 0.2930 0.2270 0.1400   old
49  49 Infant 0.325 0.245 0.070 0.1610 0.0755 0.0255 0.0450 Adult
50  50 Female 0.525 0.425 0.160 0.8355 0.3545 0.2135 0.2450   old
```

```
> test<-data[-ind,]
> test
     X     V1    V2    V3    V4     V5     V6     V7     V8    V9
1    1   Male 0.455 0.365 0.095 0.5140 0.2245 0.1010 0.1500 Young
5    5 Infant 0.330 0.255 0.080 0.2050 0.0895 0.0395 0.0550 Adult
8    8 Female 0.545 0.425 0.125 0.7680 0.2940 0.1495 0.2600 Young
13  13   Male 0.490 0.380 0.135 0.5415 0.2175 0.0950 0.1900   old
14  14 Female 0.535 0.405 0.145 0.6845 0.2725 0.1710 0.2050   old
16  16   Male 0.500 0.400 0.130 0.6645 0.2580 0.1330 0.2400   old
18  18 Female 0.440 0.340 0.100 0.4510 0.1880 0.0870 0.1300   old
20  20   Male 0.450 0.320 0.100 0.3810 0.1705 0.0750 0.1150   old
21  21   Male 0.355 0.280 0.095 0.2455 0.0955 0.0620 0.0750   old
23  23 Female 0.565 0.440 0.155 0.9395 0.4275 0.2140 0.2700   old
24  24 Female 0.550 0.415 0.135 0.7635 0.3180 0.2100 0.2000   old
25  25 Female 0.615 0.480 0.165 1.1615 0.5130 0.3010 0.3050   old
28  28   Male 0.590 0.445 0.140 0.9310 0.3560 0.2340 0.2800   old
33  33   Male 0.665 0.525 0.165 1.3380 0.5515 0.3575 0.3500 Young
37  37 Female 0.540 0.475 0.155 1.2170 0.5305 0.3075 0.3400 Young
46  46 Infant 0.390 0.295 0.095 0.2030 0.0875 0.0450 0.0750 Adult
48  48 Female 0.460 0.375 0.120 0.4605 0.1775 0.1100 0.1500 Adult
51  51 Infant 0.520 0.410 0.120 0.5950 0.2385 0.1110 0.1900 Adult
53  53   Male 0.485 0.360 0.130 0.5415 0.2595 0.0960 0.1600   old
55  55   Male 0.405 0.310 0.100 0.3850 0.1730 0.0915 0.1100 Adult
60  60 Female 0.505 0.400 0.125 0.5830 0.2460 0.1300 0.1750 Adult
64  64   Male 0.425 0.325 0.095 0.3785 0.1705 0.0800 0.1000 Adult
65  65   Male 0.520 0.400 0.120 0.5800 0.2340 0.1315 0.1850 Adult
66  66   Male 0.475 0.355 0.120 0.4800 0.2340 0.1015 0.1350 Adult
70  70 Infant 0.310 0.235 0.070 0.1510 0.0630 0.0405 0.0450 Adult
72  72 Female 0.400 0.320 0.110 0.3530 0.1405 0.0985 0.1000 Adult
75  75 Female 0.605 0.450 0.195 1.0980 0.4810 0.2895 0.3150   old
76  76 Female 0.600 0.475 0.150 1.0075 0.4425 0.2210 0.2800 Young
77  77   Male 0.595 0.475 0.140 0.9440 0.3625 0.1890 0.3150   old
79  79 Female 0.555 0.425 0.140 0.7880 0.2820 0.1595 0.2850   old
80  80 Female 0.615 0.475 0.170 1.1025 0.4695 0.2355 0.3450   old
```





5. Conclusion and Future Work

5.1 Future Work

The primary goal is to predict the age of abalones without the need for the time-consuming and invasive process of cutting the shell and counting rings under a microscope. Dataset: The dataset used for this project belongs to Marine Research Laboratories (MRL) in Taroona. Age Determination: Traditionally, abalone age is determined by cutting the shell, staining it, and counting the rings. This is a tedious and time-consuming task. Predictive Features: The project aims to find predictability using eight physical measurements. Some of these measurements can be obtained without harming the abalones, such as sex, length, diameter, height, and whole weight. Limitation: Certain "internal data," including the weight of viscera and shell, cannot be obtained without causing harm to the abalones, which is not acceptable. Cultivating Industry: In the abalone cultivation industry, young abalones are typically kept, while adult and old abalones are harvested. Predicting the age based on "internal data" is not practical, as it would require harming the abalones. Data Split: The dataset is divided into two parts:

"External Data": Includes attributes like sex, length, diameter, height, and whole weight, which can be obtained without harming the abalones. “Internal Data": Includes attributes like the weight of shuck, shell, and viscera, which cannot be obtained without harming the abalones.

Results: The papers findings indicate that the "external data" alone is sufficient to predict the age of abalones with nearly the same success rate as using the entire dataset. Using only the "external data" results in a simplified decision tree. The "internal data" does not provide significantly more information. This approach focuses on using non-invasive attributes to predict abalone age, making it more practical and humane for the abalone cultivation industry. This approach simplifies the age prediction process and avoids the need to harm the abalones for data collection.

6. Conclusion

Predicting the age of abalones using a variety of characteristics, such as sex, length, diameter, height, whole weight, shucked weight, viscera weight, shell weight, and the number of rings, is the aim of this paper. The project employs multiple data mining techniques to analyse the data and evaluate their performance. The overarching goal is to leverage historical data to uncover general patterns and enhance the decision-making process. This paper is acknowledged as both interesting and challenging, particularly during the analytical phase.

Acknowledgement: Not Applicable.

Funding Statement*:* The author(s) received no specific funding for this study.
Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

[1] A. Beygelzimer, K. Sham, L. John, Sunil Arya, David Mount *et al*., "FNN: Fast Nearest Neighbor Search Algorithms and Applications," https://CRAN.R-project.org/package=FNN.Bruce, Peter, and Andrew Bruce. 2017. *Practical Statistics for Data Scientists: 50 Essential Concepts.* O'Reilly Media, Inc.pp.1-17, 2017.

[2] P. Cunningham and Sarah Jane Delany. "K-Nearest Neighbour Classifiers." *Multiple Classifier Systems*, vol. 34, no.8, pp. 1-17.

[3] De. Maesschalck, Roy, Delphine Jouan-Rimbaud, and Désiré L Massart. "The Mahalanobis Distance." *Chemometrics and Intelligent Laboratory Systems*, vol.50, no.1, pp. 1–18. 2000.

[4] Friedman, Jerome, Trevor Hastie, and Robert Tibshirani. "The Elements of Statistical Learning. Vol. 1. Springer Series in Statistics New York," NY, USA:Han, Jiawei, Jian Pei, and Micheline Kamber. 2011. *Data Mining: Concepts and Techniques*. Elsevier.

[5] Jiang, Shengyi, P. Guansong, Wu Meilin and Limin Kuang. "An Improved K-Nearest-Neighbor Algorithm for Text Categorization." *Expert Systems with Applications*, vol. 39, no.1,2017.

[6] Mccord, Michael, and M Chuah. "Spam Detection on Twitter Using Traditional Classifiers." *In International Conference on Autonomic and Trusted Computing*, pp.175–86.2011.

[7] Robinson, John T. "The Kdb-Tree: A Search Structure for Large Multidimensional Dynamic Indexes." *In Proceedings of the 1981 Acm Sigmod International Conference on Management of Data*, pp. 10–18. ACM.

[8] Kubat M and Matwin S. "Addressing the curse of imbalanced training sets: one-sided selection," *ICML*, 1997, pp. 179-186.

[9] C. Wang, L. Hu, M. Guo, X. Liu and Q. Zou et al. "an ensemble learning method for imbalanced classification with miRNA data," *Genetics and molecular research.* vol. 14, pp.123, 2015.

[10] NB. Abdel-Hamid, S. ElGhamrawy, AE. Desouky and H. Arafat. "A Dynamic Spark-based Classification Framework for Imbalanced Big Data,' J *Grid Computing*, vol. 16, no.607.2017.