ANALOGY OF K-NN AND NAIVE BAYES CLASSIFER USING WEKA TOOL

SHEEBA M
UG STUDENT
DEPARTMENT OF COMPUTER SCIENCE AND APPLICATIONS
SRI KRISHNA ARTS AND SCIENCE
COLLEGE,COIMBATORE

RESHMA A
UG STUDENT,
DEPARTMENT OF COMPUTER SCIENCE AND APPLICATIONS,
SRI KRISHNA ARTS AND SCIENCE
COLLEGE,COIMBATORE

PRANAV SHANKAR V
UG STUDENT,
DEPARTMENT OF COMPUTER SCIENCE AND APPLICATIONS,
SRI KRISHNA ARTS AND SCIENCE
COLLEGE,COIMBATORE

Abstract— ***Materialization of contemporary techniques for scientific data compilation has resulted in major accumulation of data related to varied fields. Conventional database querying methods are scarce to dig up useful information from huge data banks. The progress of data-mining applications such as classification and clustering has exposed the need for machine learning algorithms to be functional to large scale data. In this paper we present the analogy of distinctive classification and clustering techniques using WEKA. The algorithm or methods tested are DBSCAN, BAYES NETWORK CLASSIFIER classification algorithms.***

Keywords— DBSCAN, Bayes, clustering, classification, analogy

I. INTRODUCTION

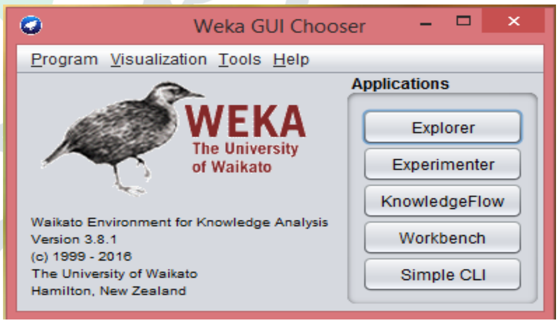
This paper shows the comparative study of density based clustering and Naive Bayes algorithms. Classification and clustering are vital data mining techniques that panels objects into evocative disjoint subgroups.

Clustering is done to build groups which has elements, which are akin to others within the group but very disparate to elements of other groups. Classification is a two step process. In the first step, training data are analyzed by a classification algorithm. In the second step, test data are used to estimate the accuracy of the classification rules. If the accuracy is considered satisfactory, the rules can be practiced to the classification of new data.

This paper assesses the consummation of algorithms based on certainty, timelessness and flaw rates.

II. WEKA

WEKA (Waikato Environment for Knowledge Analysis) is an open source, platform liberated and accessible to use data mining tool circulated under GNU General Public License. It show up with Graphical User Interface (GUI) and encompass assortment of data preprocessing and modeling techniques. Tools for data pre-processing, classification, regression, clustering, association rules and visualization inclusive of appropriate contemporary machine learning schemes are afforded in the package. It is convenient since it is fully enforced in the Java programming language and thus runs on virtually any modernized computing platform.



Waikato Environment for Knowledge Analysis
Version 3.8.1
(c) 1999 - 2016
The University of Waikato
Hamilton, New Zealand

User interfaces

Weka's prime user interface is the Explorer, but substantially the monotonous functionality can be pervaded through the component-based Knowledge Flow as well as the command line interface (CLI). There is likewise the Experimenter, which grants the methodical analogy of the predictive accomplishment of Weka's machine learning algorithms on assortment of datasets.

The Explorer interface features assorted panels contingent upon approach to the main peripherals of the workbench:

- The Preprocess panel has competence for importing data from a database, a csv or an arff file, etc., and for preprocessing this data uses a so-called filtering algorithm. These filters can be used to mutate the data from numeric to discrete, to evacuate missing instances; too aptly embrace missing values and reorganizing csv file to arff and vice versa.

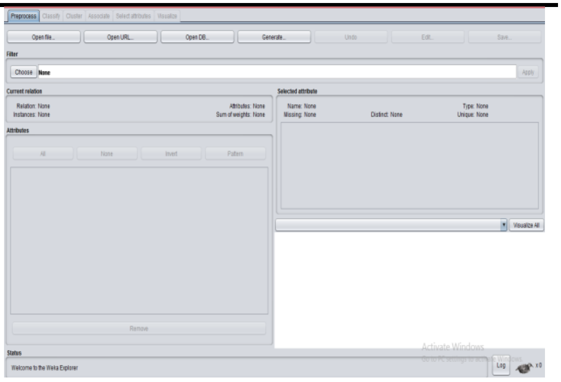
- The Classify panel empowers the user to spread classification and regression algorithms to the emanate dataset, to estimate the veracity of an emanate predictive model, and to envision errors. There are numerous types of classification algorithms like rule based, decision tree, naïve Bayesian, lazy, mi, misc etc.

- The Associate panel quest to analyse all paramount interdependence between facet in the data with the advice of association pupil like apriori, filtered associator, predictive apriori etc.

- The Cluster panel gives approach to the clustering techniques in Weka, e.g., the simple k-means, cobweb, DBSCAN, CLOPE algorithm to cater peculiar kind of clustering's for distinct situations and custom of their consequence.

- The Select attributes panel caters algorithms for diagnose the most predictive attributes in a dataset.

- The *Visualize* panel displays a scatter plot matrix, where sole scatter plots can be selected and intensified, and scrutinize further using several selection operators.



Filter: None

Relation: None
Instances: None
Attributes: None
Sum of weights: None

Name: None
Missing: None
Distinct: None
Type: None
Unique: None

Welcome to the Weka Explorer

Extension packages

In version 3.7.2 of weka, a package manager was combined to pursuit the easier installation of extension packages. Much functionality has come in weka through unceasing extension and refurbish to make it more mature.

III. DENSITY BASED CLUSTER

Density based clustering algorithm has play a crucial part in verdict non linear shapes design based on the density. Density-Based Spatial Clustering of Applications with Noise (DBSCAN) is most extensively used density based algorithm. It uses the approach of density **reachability** and **density connectivity**.

Density Reachability - A point "p" is aforesaid to be density reachable from a point "q" provided that point "p" is inside ε radius from point "q" and "q" has ample number of points in its acquaintance which are within distance ε.

Density Connectivity - A point "p" and "q" are said to be density akin if there prevail a point "r" which has tolerable number of points in its acquaintance and both the points "p" and "q" are in reach to the ε radius. This is dynamic action. So, if "q" is acquaintance of "r", "r" is acquaintance of "s", "s" is acquaintance of "t" which in turn is acquaintance of "p" signify that "q" is acquaintance of "p".

Parameters:

The DBSCAN algorithm essentially needs 2 parameters:
Eps: the minimal distance between two points. It means that if the distance between two points is curtailed or equal to this value (eps), these points are treated acquaintance.
Midpoints: the minimal number of points to mode an opaque region. For example, if we set the midpoints parameter as 5, then we demand partially 5 points to form an opaque region.

Parameter estimation:

The parameter assessment is a dilemma for every data mining task. To embrace good parameters we need to discern how they are worn and have at least a basic preceding knowledge about the data set that will be used.

eps: if the eps value preferred is too paltry, a hefty part of the data will not be clustered. It will be considered outliers because don't placate the number of points to create an opaque region. On the other hand, if the value that was preferred is too lofty, clusters will meld and the majority of objects will be in the alike cluster. The eps should be preferred based on the distance of the dataset (we can use a k-distance graph to find it), but in common paltry eps values are favored.

MinPoints: As a common rule, a minimal MinPoints can be derived from a number of dimensions (D) in the data set, as MinPoints ≥ D + 1. Lofty values are usually exceptional for data sets with noise and will form more momentous clusters. The minimal value for the MinPoints must be 3, but the bigger the data set, the bigger the MinPoints value that should be preferred.

Algorithmic steps for DBSCAN clustering

The DBSCAN algorithm should be used to find clubs and framework in data that are tough to find manually but that can be pertinent and favorable to find patterns and envision trends.

1) Start with a random origin point that has not been hit.
2) Excerpt the region of this point using ε (All points which are in reach to the ε distance are region).
3) If there are tolerable region over this point then clustering process begins and point is noted as visited else this point is designated as noise (Later this point can become the part of the cluster).
4) If a point is found to be a section of the cluster then its ε region is also the section of the cluster and the above action from step 2 is redone for all ε region points. This is redone until all points in the cluster are resolved.
5) A new unvisited point is fetched and treated, leading to the revelation of a further cluster or noise.
6) This process goes on as far as all points are notable as visited.

Advantages

1) Does not lack a-nunnery specification of number of clusters.
2) Able to diagnose noise data while clustering.
3) DBSCAN algorithm is able to find forthwith size and forthwith shaped clusters.
4) Designed for accelerate region queries.
5) MinPts and eps can be set by a domain expert.
6) Has a notion of noise and is robust to outliers.
7) Requires just two parameters and is mostly insensitive to the ordering of points in the database.

Disadvantages

1) DBSCAN algorithm declines in case of fluctuating density clusters.
2) Declines in the event of neck type of dataset.
3) Reduces its work in case of high dimensional data.
4) DBSCAN is not entirely deterministic: Border points that are responsible from more than one cluster, depending on the order the data are processed.
5) The quality of DBSCAN depends on distance measure used in the function region query.
6) If the data and scale are not well understood, choosing a meaningful distance threshold can be difficult

IV. NAVIE BAYES

Bayes' Theorem

Bayes' Theorem finds the probability of an event transpire given the probability of another event that has previously occurred. Bayes' theorem is declared mathematically as the following equation:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Here A and B are events

- Basically, we are strenuous to find probability of event A, given the event B is true. Event B is also described as **evidence**.
- P (A) is the **preceding** of A (the preceding probability, i.e. Probability of event ahead evidence is spotted). The evidence is an attribute value of an exotic instance (here, it is event B).

- P (A|B) is a hind probability of B, i.e. probability of event subsequently evidence is spotted.

Naive assumption

Now, it's time to stick a naive assumption to the Bayes' theorem, which is, **independence** in the middle of the features. So now, we rupture evidence into the independent parts.

If any two events A and B are liberated, then,

$$P (A, B) = P (A) P (B)$$

A. Naive Bayes Classifier

Naive Bayes classifiers are an assortment of classification algorithms based on **Bayes' Theorem**. It is not a single algorithm but a folk of algorithms where all of them stake a trivial principle, i.e. every couple of features being classified is liberated of each other.

Naive Bayes is a classification algorithm for twofold (two-class) and multi-class classification obstacles. The technique is easiest to understand when described using binary or emphatic input values.

It is also called as an *idiot Bayes* because the computation of the probabilities for each thesis is abridged to make their calculation acquiescent. Rather than solicit to calculate the values of every attribute value P (d1, d2, d3|h), they are feigned to be conditionally independent given the objective value and calculated as P (d1|h) * P (d2|H) and so on.

This is a very tenacious assumption that is most unlikely in original data, i.e. that the attributes do not mesh. Though, the way accomplish notably well on data where this assumption does not grip.

<u>Representation Used By Naive Bayes Models</u>

The depiction for naive Bayes is probabilities.

Records of probabilities are stocked to file for an accomplished naive Bayes model. This includes:

Class Probabilities: The probabilities of every class in the training dataset.

Conditional Probabilities: The conditional probabilities of every input value accustomed every class value.

<u>Advantages</u>

If the NB conditional independence premise literally grasp, a Naive Bayes classifier will mingle agile than discerning replica like logistic regression, so we demand fewer training data. And even if the NB premise doesn't grasp, a NB classifier still frequently does a great job in routine. A good bet if want something agile and effortless that carryout pretty well. Its main issue is that it can't learn synergy between features.

<u>Disadvantages</u>

1)Issue 1: Fragmentary training data

Remembrance that in order to appliance it, we demand to figure out assorted conditional probabilities. Pointedly, the class conditional probability, which outlooks the probability that a characteristic presume a particular value, given the result or feedback class.

2)Issue 2: Continuous variables

When a characteristic is endless, estimate the probabilities by the classic method of recurrence counts is not viable. In this case we would either demand to convert the facet to a distinct variable or use probability density actions to figure out probability densities (not substantial probabilities!). Most classic fulfilment undoubtedly account for nominal and unceasing facet so the user does not need to woe about these metamorphosis.

3)Issue 3: Facet independence

This is by far the most paramount frailty and something which lack a morsel of ancillary effort. In the forecast of consequence probabilities using the traditional Bayes theorem, the tacit premise is that all the facet are jointly independent. This grants us to proliferate the class conditional probabilities in order to figure out the fallout probability.

EXPERIMENTAL RESULTS

The classification algorithms that are enforced one by one on the dataset are Naive Bayes and KNN with lazy IBK. The preeminent objective is to figure out the finest classifier whose accuracy is excelling than the rest of the classifiers Ensuing tables demonstrate terse of consequence of implementation of disparate classifiers using WEKA tool. Table I demonstrates the confusion matrix of KNN classifier. Table II demonstrates the confusion matrix of Naïve Bayes classifier. Table III demonstrates the results of time taken by classifiers for classifying accustomed datasets.

Table I – Confusion Matrix of KNN classifier

a	b	c	d	E	classified as
2	4	13	0	5	a = TN 99
3	3	10	0	9	b= TN 66
6	3	12	0	8	c= TN 55
0	0	1	0	0	d= TN 66
4	7	11	0	4	e= TN 65

Table II – Confusion Matrix of Naive Bayes classifier

a	b	c	d	E	classified as
0	0	16	0	8	a= TN 99
0	0	13	0	12	b= TN 66
0	0	18	0	11	c= TN 55
0	0	1	0	0	d= TN 66
0	7	14	0	5	e= TN 65

Table III – Result of Analogy of Classifiers

Classifier	Time Taken	Correct	Incorrect
Naive Bayes	0.02	20 %	80 %
KNN	0.00	21.9048 %	78.952%

CONCLUSION

As in the analysis, the result it is erect that K-NN classification method works as excelling classifier when implementation is accomplished on the basis of accuracy and classification available. When these 2 classification ways square measure implemented on alike knowledge sets to fetch the optimum result pageant that K-NN classification technique offers higher accuracy as analyzed to naïve Bayes classification method . The comparative analysis has depicted that each algorithm has its own benefits and difficulties. Not any algorithm can suffice all stifle and criteria. Rely upon utilization and necessity, distinct algorithm can be preferred.

REFERENCES

[1]Brijain R. Patel and Kushik K.Rana, “A Survey on Decision Tree Algorithm for Classification”, International Journal of Engineering Development and Research, 2014.

[2] BhaveshPatankar and Dr. Vijay Chavda, “A Comparative Study of Decision Tree, Naive Bayesian and k-nn Classifiers in Data Mining”, International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 4, Issue 12, December 2014.

[3] Meenakshi and Geetika, “Survey on Classification Methods using WEKA", International Journal of Computer Applications, Vol. 86, No.18, January 2014.

[4] H. Bhavsar and A. Ganatra, “A Comparative Study of Training Algorithms for Supervised Machine Learning”, International Journal of Soft Computing and Engineering (IJSCE), Vol. 2, Issue. 4, September 2012

[5] S.Archana and Dr. K.Elangovan, “Survey of Classification Techniques in Data Mining”, International Journal of Computer Science and Mobile Applications, Vol. 2 Issue. 2, February 2014.

[6] T. M. Cover and P. E. Hart, “Nearest Neighbor Pattern Classification”, IEEE Transactions on Information Theory, vol. 13, No. 1, pp. 21-27, 1967.

[7] Sagar S. Nikam, “A Comparative Study of Classification Techniques in Data Mining Algorithms”, Oriental Journal of Computer Science & Technology, Vol. 8, April 2015.

[8] S. B. Kotsiantis, “Supervised Machine Learning: A Review of Classification Techniques”, Informatica, vol. 31, pp. 249-268, 2007.

[9]M. Soundarya and R. Balakrishnan, “Survey on Classification Techniques in Data mining”, International Journal of Advanced Research in Computer and Communication Engineering, Vol. 3, Issue 7, July 2014.

[10] K. P. Soman, “Insight into Data Mining Theory and Practice”, New Delhi: PHI, 2006.

[11] J. Han and M. Kamber, “Data Mining Concepts and
Techniques”, Elevier, 2011.
[12]R. Duda, and P. Hart, "Pattern Classification and Scene
Analysis", John Wiley and Sons, New York, 1973.
[13]N. Friedman, D. Geiger, and Goldazmidt, “Bayesian
Network Classifiers”, Machine Learning, vol. 29, pp.
131-163, 1997.
[14] “Decision tree learning” pdf.
[15]Matthew N. Anyanwu and Sajjan G. Shiva, “Comparative Analysis of Serial Decision Tree ClassificationAlgorithms”, Researchgate, January 2009.