

**A MINI- PROJECT REPORT ON**

IRIS FLOWER PREDICTION

*Submitted to the partial fulfillment of the requirement*

*for the 15CS331E Data Mining Analytics course and*

*for the award of the degree of*

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

*Under the guidance of*

Mrs. Priyanka Sharma

(Assistant Professor, Department of Computer Science and Engineering)

**Submitted by**

Samarthya Gupta

[Registration No. RA1611003030670]

Suraj Singh

[Registration No. RA1611003030694]

Pranav Kahol

[Registration No. RA1611003030689]



**BONAFIDE CERTIFICATE**

Certified that this project report titled “COLLEGE PREDICTOR" is the bonafide work of **Samarthya Gupta** :RA1611003030670, **Suraj Singh** : RA1611003030694, **Pranav Kahol**: RA1611003030689 who carried out the project work under my supervision and submitted to the partial fulfillment of the requirement for the 15CS331E Data Mining Analytics course and for the award of the degree of Bachelor of Technology in Computer Science and Engineering of SRM Institute of Science and Technology.

|  |  |  |
| --- | --- | --- |
| Mrs. Priyanka Sharma  Supervisor,  Assistant Professor, Department of  Computer Science and Engineering |  | Dr. Rajendra Prasad Mahapatra  Head of the Department,  Professor, Department of  Computer Science and Engineering |

Signature of the Internal Examiner\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

**ABSTRACT**

In Data Mining, we are using semi-automated extraction of knowledge of data for identifying IRIS flower species. Classification is a supervised learning in which the response is categorical that is its values are in finite unordered set. To simply the problem of classification, weka tools has been used. This paper focuses on IRIS flower classification using Machine Learning with weka tools. Here the problem concerns the identification of IRIS flower species on the basis of flowers attribute measurements. Classification of IRIS data set would be discovering patterns from examining petal and sepal size of the IRIS flower and how the prediction was made from analyzing the pattern to from the class of IRIS flower. In this paper we train the machine learning model with data and when unseen data is discovered the predictive model predicts the species using what it has been learnt from the trained data. Keywords: Classification, Logistic Regression, K Nearest Neighbour, Machine Learning.

**1.INTRODUCTION**

The Data Mining is the subfield of computer science, according to Arthur Samuel in 1959 told “computers are having the ability to learn without being explicitly programmed”. Evolved from the study of pattern recognition and computational learning theory in artificial intelligence machine learning explores the study and construction of algorithms that can learn from and make predictions on data such algorithms overcome following strictly static program instructions by making data-driven predictions or decisions, through building a model from sample inputs. Machine learning is employed in a range of computing tasks where designing and programming explicitly algorithms with good performance is difficult or unfeasible; example applications include email filtering, detection of network intruders, learning to rank and computer vision. Machine learning focuses on the development of computer programs that can teach themselves to grow and change when exposed to new data. It is a research field at the intersection of statistics, artificial intelligence and computer science and is also known as predictive analytics or statistical learning. There are two main categories of Machine learning. They are Supervised and Unsupervised learning and here in this, the paper focuses on supervised learning. Supervised learning is a task of inferring a function from labeled training data. The training data consists of set of training examples. In supervised learning, each example is a pair of an input object and desired output value. A supervised learning algorithm analyze the training data and produces an inferred function, which can be used for mapping new examples. Supervised learning problems can be further grouped into regression and classification problems. Classification problem is when the output variable is a category, such as “red” or “blue” or “disease” and “no disease”. Regression problem is when the output variable is a real value, such as “dollars” or “weight”. In this paper a novel method for Identification of Iris flower species is presented. It works in two phases, namely training and testing. During training the training dataset are loaded into Machine Learning Model and Labels are assigned. Further the predictive model, predicts to which species the Iris flower belongs to. Hence, the expected Iris species is labeled. This paper focuses on IRIS flower classification using Machine Learning with weka tool. The problem statement concerns the identification of IRIS flower species on the basic of flower attribute measurements. Classification of IRIS data set would be discovering patterns from examining petal and sepal size of the IRIS flower and how the prediction was made from analyzing the pattern to form the class of IRIS flower. In this paper we train the Machine Learning Model with data and when unseen data is discovered the predictive model predicts the species using what it has learn from trained data.

**2.DATA EXPLORATION**

Many methods have been presented for Identification of Iris Flower Species. Every method employs different strategy.We have used method on KNN ALGORITHM that is K-Nearest Neighbour Algorithm.in which classification is done according to the k nearest neighbours that is how many nearest elements are there from which group. The more the elements from a group the more likely is the point to lie in that group.K is decided by the programmer.

3. Dataset

@relation Iris-weka.filters.unsupervised.instance.Resample-S1-Z85.0

@attribute Id numeric

@attribute SepalLengthCm numeric

@attribute SepalWidthCm numeric

@attribute PetalLengthCm numeric

@attribute PetalWidthCm numeric

@attribute Species {Iris-setosa,Iris-versicolor,Iris-virginica}

@data

136,7.7,3,6.1,2.3,Iris-virginica

89,5.6,3,4.1,1.3,Iris-versicolor

98,6.2,2.9,4.3,1.3,Iris-versicolor

64,6.1,2.9,4.7,1.4,Iris-versicolor

105,6.5,3,5.8,2.2,Iris-virginica

5,5,3.6,1.4,0.2,Iris-setosa

135,6.1,2.6,5.6,1.4,Iris-virginica

107,4.9,2.5,4.5,1.7,Iris-virginica

29,5.2,3.4,1.4,0.2,Iris-setosa

149,6.2,3.4,5.4,2.3,Iris-virginica

20,5.1,3.8,1.5,0.3,Iris-setosa

74,6.1,2.8,4.7,1.2,Iris-versicolor

68,5.8,2.7,4.1,1,Iris-versicolor

64,6.1,2.9,4.7,1.4,Iris-versicolor

13,4.8,3,1.4,0.1,Iris-setosa

35,4.9,3.1,1.5,0.1,Iris-setosa

93,5.8,2.6,4,1.2,Iris-versicolor

13,4.8,3,1.4,0.1,Iris-setosa

97,5.7,2.9,4.2,1.3,Iris-versicolor

90,5.5,2.5,4,1.3,Iris-versicolor

27,5,3.4,1.6,0.4,Iris-setosa

33,5.2,4.1,1.5,0.1,Iris-setosa

11,5.4,3.7,1.5,0.2,Iris-setosa

150,5.9,3,5.1,1.8,Iris-virginica

75,6.4,2.9,4.3,1.3,Iris-versicolor

60,5.2,2.7,3.9,1.4,Iris-versicolor

149,6.2,3.4,5.4,2.3,Iris-virginica

54,5.5,2.3,4,1.3,Iris-versicolor

88,6.3,2.3,4.4,1.3,Iris-versicolor

3,4.7,3.2,1.3,0.2,Iris-setosa

56,5.7,2.8,4.5,1.3,Iris-versicolor

5,5,3.6,1.4,0.2,Iris-setosa

1,5.1,3.5,1.4,0.2,Iris-setosa

107,4.9,2.5,4.5,1.7,Iris-virginica

114,5.7,2.5,5,2,Iris-virginica

106,7.6,3,6.6,2.1,Iris-virginica

40,5.1,3.4,1.5,0.2,Iris-setosa

121,6.9,3.2,5.7,2.3,Iris-virginica

76,6.6,3,4.4,1.4,Iris-versicolor

135,6.1,2.6,5.6,1.4,Iris-virginica

116,6.4,3.2,5.3,2.3,Iris-virginica

111,6.5,3.2,5.1,2,Iris-virginica

128,6.1,3,4.9,1.8,Iris-virginica

88,6.3,2.3,4.4,1.3,Iris-versicolor

28,5.2,3.5,1.5,0.2,Iris-setosa

43,4.4,3.2,1.3,0.2,Iris-setosa

121,6.9,3.2,5.7,2.3,Iris-virginica

139,6,3,4.8,1.8,Iris-virginica

101,6.3,3.3,6,2.5,Iris-virginica

108,7.3,2.9,6.3,1.8,Iris-virginica

53,6.9,3.1,4.9,1.5,Iris-versicolor

34,5.5,4.2,1.4,0.2,Iris-setosa

27,5,3.4,1.6,0.4,Iris-setosa

106,7.6,3,6.6,2.1,Iris-virginica

37,5.5,3.5,1.3,0.2,Iris-setosa

146,6.7,3,5.2,2.3,Iris-virginica

9,4.4,2.9,1.4,0.2,Iris-setosa

80,5.7,2.6,3.5,1,Iris-versicolor

105,6.5,3,5.8,2.2,Iris-virginica

135,6.1,2.6,5.6,1.4,Iris-virginica

1,5.1,3.5,1.4,0.2,Iris-setosa

2,4.9,3,1.4,0.2,Iris-setosa

137,6.3,3.4,5.6,2.4,Iris-virginica

34,5.5,4.2,1.4,0.2,Iris-setosa

149,6.2,3.4,5.4,2.3,Iris-virginica

97,5.7,2.9,4.2,1.3,Iris-versicolor

9,4.4,2.9,1.4,0.2,Iris-setosa

65,5.6,2.9,3.6,1.3,Iris-versicolor

66,6.7,3.1,4.4,1.4,Iris-versicolor

129,6.4,2.8,5.6,2.1,Iris-virginica

110,7.2,3.6,6.1,2.5,Iris-virginica

17,5.4,3.9,1.3,0.4,Iris-setosa

116,6.4,3.2,5.3,2.3,Iris-virginica

59,6.6,2.9,4.6,1.3,Iris-versicolor

73,6.3,2.5,4.9,1.5,Iris-versicolor

88,6.3,2.3,4.4,1.3,Iris-versicolor

43,4.4,3.2,1.3,0.2,Iris-setosa

32,5.4,3.4,1.5,0.4,Iris-setosa

52,6.4,3.2,4.5,1.5,Iris-versicolor

48,4.6,3.2,1.4,0.2,Iris-setosa

39,4.4,3,1.3,0.2,Iris-setosa

13,4.8,3,1.4,0.1,Iris-setosa

42,4.5,2.3,1.3,0.3,Iris-setosa

109,6.7,2.5,5.8,1.8,Iris-virginica

136,7.7,3,6.1,2.3,Iris-virginica

120,6,2.2,5,1.5,Iris-virginica

30,4.7,3.2,1.6,0.2,Iris-setosa

96,5.7,3,4.2,1.2,Iris-versicolor

110,7.2,3.6,6.1,2.5,Iris-virginica

61,5,2,3.5,1,Iris-versicolor

118,7.7,3.8,6.7,2.2,Iris-virginica

127,6.2,2.8,4.8,1.8,Iris-virginica

75,6.4,2.9,4.3,1.3,Iris-versicolor

71,5.9,3.2,4.8,1.8,Iris-versicolor

54,5.5,2.3,4,1.3,Iris-versicolor

126,7.2,3.2,6,1.8,Iris-virginica

2,4.9,3,1.4,0.2,Iris-setosa

2,4.9,3,1.4,0.2,Iris-setosa

87,6.7,3.1,4.7,1.5,Iris-versicolor

1,5.1,3.5,1.4,0.2,Iris-setosa

25,4.8,3.4,1.9,0.2,Iris-setosa

83,5.8,2.7,3.9,1.2,Iris-versicolor

145,6.7,3.3,5.7,2.5,Iris-virginica

26,5,3,1.6,0.2,Iris-setosa

145,6.7,3.3,5.7,2.5,Iris-virginica

25,4.8,3.4,1.9,0.2,Iris-setosa

109,6.7,2.5,5.8,1.8,Iris-virginica

94,5,2.3,3.3,1,Iris-versicolor

4,4.6,3.1,1.5,0.2,Iris-setosa

98,6.2,2.9,4.3,1.3,Iris-versicolor

68,5.8,2.7,4.1,1,Iris-versicolor

131,7.4,2.8,6.1,1.9,Iris-virginica

102,5.8,2.7,5.1,1.9,Iris-virginica

84,6,2.7,5.1,1.6,Iris-versicolor

58,4.9,2.4,3.3,1,Iris-versicolor

17,5.4,3.9,1.3,0.4,Iris-setosa

78,6.7,3,5,1.7,Iris-versicolor

96,5.7,3,4.2,1.2,Iris-versicolor

101,6.3,3.3,6,2.5,Iris-virginica

20,5.1,3.8,1.5,0.3,Iris-setosa

59,6.6,2.9,4.6,1.3,Iris-versicolor

54,5.5,2.3,4,1.3,Iris-versicolor

141,6.7,3.1,5.6,2.4,Iris-virginica

77,6.8,2.8,4.8,1.4,Iris-versicolor

67,5.6,3,4.5,1.5,Iris-versicolor

47,5.1,3.8,1.6,0.2,Iris-setosa

22,5.1,3.7,1.5,0.4,Iris-setosa

OUTPUT OF TRAINING SET WITH K=5

=== Run information ===

Scheme: weka.classifiers.lazy.IBk -K 5 -W 0 -A "weka.core.neighboursearch.LinearNNSearch -A \"weka.core.EuclideanDistance -R first-last\""

Relation: Iris-weka.filters.unsupervised.instance.Resample-S1-Z85.0

Instances: 127

Attributes: 6

Id

SepalLengthCm

SepalWidthCm

PetalLengthCm

PetalWidthCm

Species

Test mode: evaluate on training data

=== Classifier model (full training set) ===

IB1 instance-based classifier

using 5 nearest neighbour(s) for classification

Time taken to build model: 0 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 0.01 seconds

=== Summary ===

Correctly Classified Instances 127 100 %

Incorrectly Classified Instances 0 0 %

Kappa statistic 1

Mean absolute error 0.0062

Root mean squared error 0.0293

Relative absolute error 1.3871 %

Root relative squared error 6.2071 %

Total Number of Instances 127

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

1.000 0.000 1.000 1.000 1.000 1.000 1.000 1.000 Iris-setosa

1.000 0.000 1.000 1.000 1.000 1.000 1.000 1.000 Iris-versicolor

1.000 0.000 1.000 1.000 1.000 1.000 1.000 1.000 Iris-virginica

Weighted Avg. 1.000 0.000 1.000 1.000 1.000 1.000 1.000 1.000

=== Confusion Matrix ===

a b c <-- classified as

42 0 0 | a = Iris-setosa

0 43 0 | b = Iris-versicolor

0 0 42 | c = Iris-virginica

OUTPUT OF TRAINING SET WITH K =1

=== Run information ===

Scheme: weka.classifiers.lazy.IBk -K 1 -W 0 -A "weka.core.neighboursearch.LinearNNSearch -A \"weka.core.EuclideanDistance -R first-last\""

Relation: Iris-weka.filters.unsupervised.instance.Resample-S1-Z85.0

Instances: 127

Attributes: 6

Id

SepalLengthCm

SepalWidthCm

PetalLengthCm

PetalWidthCm

Species

Test mode: evaluate on training data

=== Classifier model (full training set) ===

IB1 instance-based classifier

using 1 nearest neighbour(s) for classification

Time taken to build model: 0 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 0.03 seconds

=== Summary ===

Correctly Classified Instances 127 100 %

Incorrectly Classified Instances 0 0 %

Kappa statistic 1

Mean absolute error 0.0071

Root mean squared error 0.0081

Relative absolute error 1.6064 %

Root relative squared error 1.7279 %

Total Number of Instances 127

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

1.000 0.000 1.000 1.000 1.000 1.000 1.000 1.000 Iris-setosa

1.000 0.000 1.000 1.000 1.000 1.000 1.000 1.000 Iris-versicolor

1.000 0.000 1.000 1.000 1.000 1.000 1.000 1.000 Iris-virginica

Weighted Avg. 1.000 0.000 1.000 1.000 1.000 1.000 1.000 1.000

=== Confusion Matrix ===

a b c <-- classified as

42 0 0 | a = Iris-setosa

0 43 0 | b = Iris-versicolor

0 0 42 | c = Iris-virginica K**=1 IS GIVING LESS ERROR SO CHOOSING K=1.**

**TEST SET WITH WRONG SPECIES NAME GIVEN**

@relation Iris-weka.filters.unsupervised.instance.Resample-S1-Z85.0-weka.filters.unsupervised.instance.Resample-S1-Z85.0-no-replacement-V

@attribute Id numeric

@attribute SepalLengthCm numeric

@attribute SepalWidthCm numeric

@attribute PetalLengthCm numeric

@attribute PetalWidthCm numeric

@attribute Species {Iris-setosa,Iris-versicolor,Iris-virginica}

@data

40,5.1,3.4,1.5,0.2,Iris-setosa

3,4.7,3.2,1.3,0.2,Iris-virginica

145,6.7,3.3,5.7,2.5,Iris-setosa

80,5.7,2.6,3.5,1,Iris-setosa

109,6.7,2.5,5.8,1.8,Iris-virginica

98,6.2,2.9,4.3,1.3,Iris-virginica

1,5.1,3.5,1.4,0.2,Iris-setosa

54,5.5,2.3,4,1.3,Iris-versicolor

29,5.2,3.4,1.4,0.2,Iris-setosa

149,6.2,3.4,5.4,2.3,Iris-virginica

116,6.4,3.2,5.3,2.3,Iris-virginica

74,6.1,2.8,4.7,1.2,Iris-versicolor

68,5.8,2.7,4.1,1,Iris-versicolor

97,5.7,2.9,4.2,1.3,Iris-virginica

13,4.8,3,1.4,0.1,Iris-setosa

127,6.2,2.8,4.8,1.8,Iris-setosa

93,5.8,2.6,4,1.2,Iris-versicolor

78,6.7,3,5,1.7,Iris-versicolor

71,5.9,3.2,4.8,1.8,Iris-versicolor

26,5,3,1.6,0.2,Iris-setosa

58,4.9,2.4,3.3,1,Iris-versicolor

17,5.4,3.9,1.3,0.4,Iris-setosa

78,6.7,3,5,1.7,Iris-versicolor

96,5.7,3,4.2,1.2,Iris-setosa

101,6.3,3.3,6,2.5,Iris-virginica

20,5.1,3.8,1.5,0.3,Iris-setosa

59,6.6,2.9,4.6,1.3,Iris-versicolor

54,5.5,2.3,4,1.3,Iris-versicolor

141,6.7,3.1,5.6,2.4,Iris-virginica

77,6.8,2.8,4.8,1.4,Iris-setosa

67,5.6,3,4.5,1.5,Iris-virginica

47,5.1,3.8,1.6,0.2,Iris-setosa

**TOOL USED**



Weka contains a collection of visualization tools and algorithms for data analysis and predictive modeling, together with graphical user interfaces for easy access to these functions.[1] The original non-Java version of Weka was a Tcl/Tk front-end to (mostly third-party) modeling algorithms implemented in other programming languages, plus data preprocessing utilities in C, and a Makefilebased system for running machine learning experiments. This original version was primarily designed as a tool for analyzing data from agricultural domains,[2][3] but the more recent fully Java-based version (Weka 3), for which development started in 1997, is now used in many different application areas, in particular for educational purposes and research.

Advantages of Weka include:

• Free availability under the GNU General Public License. • Portability, since it is fully implemented in the Java programming language and thus runs on almost any modern computing platform.

• A comprehensive collection of data preprocessing and modeling techniques.

**User interface:** -

• The Preprocess panel has facilities for importing data from a database, a comma-separated values (CSV) file, etc., and for preprocessing this data using a so-called filtering algorithm. These filters can be used to transform the data (e.g., turning numeric attributes into discrete ones) and make it possible to delete instances and attributes according to specific criteria.

• The Classify panel enables applying classification and regression algorithms (indiscriminately called classifiers in Weka) to the resulting dataset, to estimate the accuracy of the resulting predictive model, and to visualize erroneous predictions, receiver operating characteristic (ROC) curves, etc., or the model itself (if the model is amenable to visualization like, e.g., a decision tree).

**OUTPUT WITH THE TEST SET**

=== Run information ===

Scheme: weka.classifiers.lazy.KStar -B 20 -M a

Relation: Iris-weka.filters.unsupervised.instance.Resample-S1-Z85.0

Instances: 127

Attributes: 6

Id

SepalLengthCm

SepalWidthCm

PetalLengthCm

PetalWidthCm

Species

Test mode: user supplied test set: size unknown (reading incrementally)

=== Classifier model (full training set) ===

KStar Beta Verion (0.1b).

Copyright (c) 1995-97 by Len Trigg (trigg@cs.waikato.ac.nz).

Java port to Weka by Abdelaziz Mahoui (am14@cs.waikato.ac.nz).

KStar options : -B 20 -M a

Time taken to build model: 0 seconds

=== Evaluation on test set ===

Time taken to test model on supplied test set: 0.05 seconds

=== Summary ===

Correctly Classified Instances 24 72.7273 %

Incorrectly Classified Instances 9 27.2727 %

Kappa statistic 0.5926

Mean absolute error 0.1818

Root mean squared error 0.4264

Relative absolute error 40.8972 %

Root relative squared error 90.4164 %

Total Number of Instances 33

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.692 0.050 0.900 0.692 0.783 0.683 0.750 0.781 Iris-setosa

1.000 0.261 0.625 1.000 0.769 0.680 0.848 0.674 Iris-versicolor

0.500 0.087 0.714 0.500 0.588 0.464 0.787 0.683 Iris-virginica

Weighted Avg. 0.727 0.125 0.760 0.727 0.720 0.616 0.791 0.719

=== Confusion Matrix ===

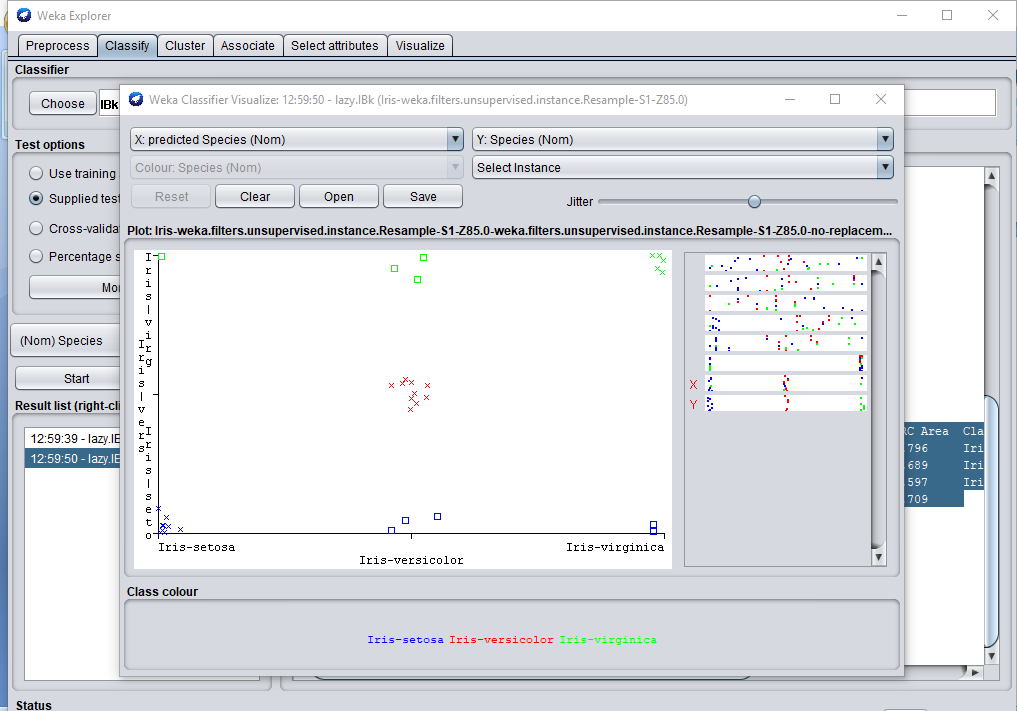
a b c <-- classified as

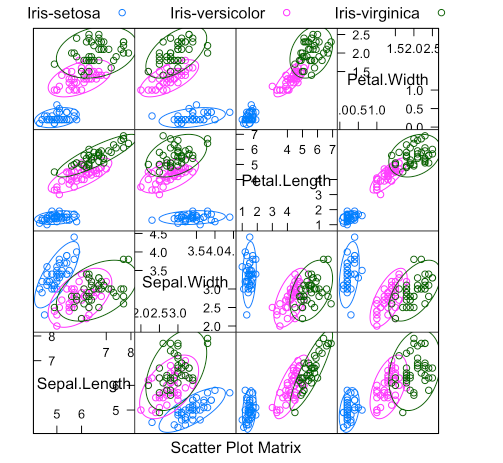
9 2 2 | a = Iris-setosa

0 10 0 | b = Iris-versicolor

1 4 5 | c = Iris-virginica

OUTPUT WITH X=PREDICTED SPECIES AND Y= GIVEN SPECIES(WRONG)





**PREDICTED VALUES**

@relation Iris-weka.filters.unsupervised.instance.Resample-S1-Z85.0-weka.filters.unsupervised.instance.Resample-S1-Z85.0-no-replacement-V\_predicted

@attribute Id numeric

@attribute SepalLengthCm numeric

@attribute SepalWidthCm numeric

@attribute PetalLengthCm numeric

@attribute PetalWidthCm numeric

@attribute 'prediction margin' numeric

@attribute 'predicted Species' {Iris-setosa,Iris-versicolor,Iris-virginica}

@attribute Species {Iris-setosa,Iris-versicolor,Iris-virginica}

@data

40,5.1,3.4,1.5,0.2,0.976923,Iris-setosa,**Iris-setosa**

3,4.7,3.2,1.3,0.2,-0.976923,Iris-setosa**,Iris-virginica**

145,6.7,3.3,5.7,2.5,-0.988327,Iris-virginica,**Iris-setosa**

80,5.7,2.6,3.5,1,-0.976923,Iris-versicolor,**Iris-setosa**

109,6.7,2.5,5.8,1.8,0.988327,Iris-virginica**,Iris-virginica**

98,6.2,2.9,4.3,1.3,-0.988327,Iris-versicolor,**Iris-virginica**

1,5.1,3.5,1.4,0.2,0.992188,Iris-setosa,**Iris-setosa**

54,5.5,2.3,4,1.3,0.992188,Iris-versicolor,**Iris-versicolor**

29,5.2,3.4,1.4,0.2,0.976923,Iris-setosa,**Iris-setosa**

149,6.2,3.4,5.4,2.3,0.992188,Iris-virginica,**Iris-virginica**

116,6.4,3.2,5.3,2.3,0.988327,Iris-virginica,**Iris-virginica**

74,6.1,2.8,4.7,1.2,0.976923,Iris-versicolor,**Iris-versicolor**

68,5.8,2.7,4.1,1,0.988327,Iris-versicolor,**Iris-versicolor**

97,5.7,2.9,4.2,1.3,-0.988327,Iris-versicolor,**Iris-virginica**

13,4.8,3,1.4,0.1,0.992188,Iris-setosa,**Iris-setosa**

127,6.2,2.8,4.8,1.8,-0.976923,Iris-virginica,**Iris-setosa**

93,5.8,2.6,4,1.2,0.976923,Iris-versicolor,**Iris-versicolor**

78,6.7,3,5,1.7,0.976923,Iris-versicolor,**Iris-versicolor**

71,5.9,3.2,4.8,1.8,0.976923,Iris-versicolor,**Iris-versicolor**

26,5,3,1.6,0.2,0.976923,Iris-setosa,**Iris-setosa**

58,4.9,2.4,3.3,1,0.976923,Iris-versicolor,**Iris-versicolor**

17,5.4,3.9,1.3,0.4,0.988327,Iris-setosa,**Iris-setosa**

78,6.7,3,5,1.7,0.976923,Iris-versicolor,**Iris-versicolor**

96,5.7,3,4.2,1.2,-0.988327,Iris-versicolor,**Iris-setosa**

101,6.3,3.3,6,2.5,0.988327,Iris-virginica,**Iris-virginica**

20,5.1,3.8,1.5,0.3,0.988327,Iris-setosa,**Iris-setosa**

59,6.6,2.9,4.6,1.3,0.988327,Iris-versicolor,**Iris-versicolor**

54,5.5,2.3,4,1.3,0.992188,Iris-versicolor,**Iris-versicolor**

141,6.7,3.1,5.6,2.4,0.976923,Iris-virginica,**Iris-virginica**

77,6.8,2.8,4.8,1.4,-0.976923,Iris-versicolor,**Iris-setosa**

67,5.6,3,4.5,1.5,-0.976923,Iris-versicolor,**Iris-virginica**

47,5.1,3.8,1.6,0.2,0.976923,Iris-setosa,**Iris-setosa**

22,5.1,3.7,1.5,0.4,0.976923,Iris-setosa,**Iris-setosa**

**HIGHLIGHTED IS PREDICTED..**

**3.PREDICTION METHODS**

**3.1 K-Nearest Neighbors Algorithm**

The k-Nearest Neighbors algorithm (or kNN for short) is an easy algorithm to understand and to implement, and a powerful tool to have at your disposal. The implementation will be specific for classification problems and will be demonstrated using the Iris flowers classification problem.

**3.1.1 What is k-Nearest Neighbors**

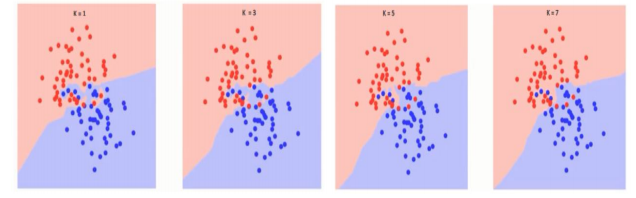
The model for kNN is the entire training dataset. When a prediction is required for a unseen data instance, the kNN algorithm will search through the training dataset for the k-most similar instances. The prediction attribute of the most similar instances is summarized and returned as the prediction for the unseen instance. The similarity measure is dependent on the type of data. For real-valued data, the Euclidean distance can be used. Other types of data such as categorical or binary data, Hamming distance can be used. In the case of regression problems, the average of the predicted attribute may be returned. In the case of classification, the most prevalent class may be returned.

**3.1.2 How does k-Nearest Neighbors**

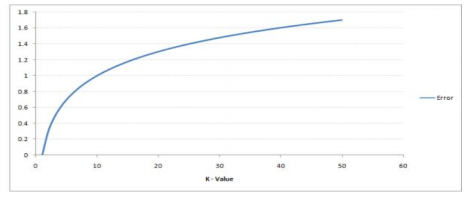
Work The kNN algorithm is belongs to the family of instance-based, competitive learning and lazy learning algorithms. Instance-based algorithms are those algorithms that model the problem using data instances (or rows) in order to make predictive decisions. The kNN algorithm is an extreme form of instance-based methods because all training observations are retained as part of the model. It is a competitive learning algorithm, because it internally uses competition between model elements (data instances) in order to make a predictive decision. The objective similarity measure between data instances causes each data instance to compete to “win” or be most similar to a given unseen data instance and contribute to a prediction. Lazy learning refers to the fact that the algorithm does not build a model until the time that a prediction is required. It is lazy because it only does work at the last second. This has the benefit of only including data relevant to the unseen data, called a localized model. A disadvantage is that it can be computationally expensive to repeat the same or similar searches over larger training datasets. Finally, kNN is powerful because it does not assume anything about the data, other than a distance measure can be calculated consistently between any two instances. As such, it is called non-parametric or non-linear as it does not assume a functional form.

**3.1.3 How do we choose the factor K**

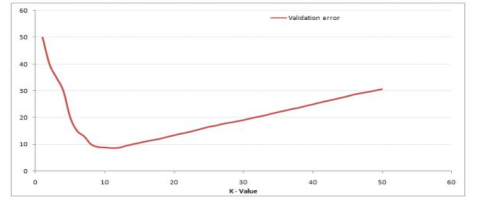
First let us try to understand what exactly does K influence in the algorithm. If we see the last example, given that all the 6 training observation remain constant, with a given K value we can make boundaries of each class. These boundaries will segregate RC from GS. The same way, let’s try to see the effect of value “K” on the class boundaries. Following are the different boundaries separating the two classes with different values of K.



If you watch carefully, you can see that the boundary becomes smoother with increasing value of K. With K increasing to infinity it finally becomes all blue or all red depending on the total majority. The training error rate and the validation error rate are two parameters we need to access on different K-value. Following is the curve for the training error rate with varying value of K :



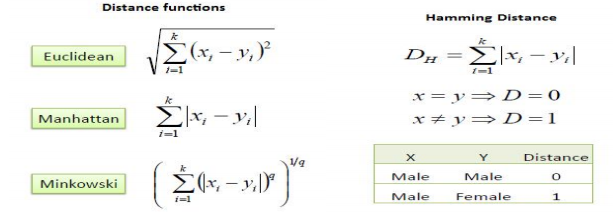
As you can see, the error rate at K=1 is always zero for the training sample. This is because the closest point to any training data point is itself. Hence the prediction is always accurate with K=1. If validation error curve would have been similar, our choice of K would have been 1. Following is the validation error curve with varying value of K:



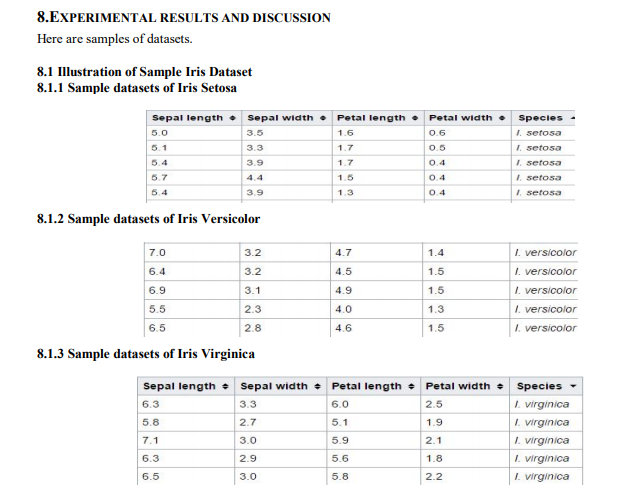
This makes the story more clear. At K=1, we were overfitting the boundaries. Hence, error rate initially decreases and reaches a minima. After the minima point, it then increase with increasing K. To get the optimal value of K, you can segregate the training and validation from the initial dataset. Now plot the validation error curve to get the optimal value of K. This value of K should be used for all predictions.

**3.2 Mathematical Formula**

A case is classified by a majority vote of its neighbors, with the case being assigned to the class most common amongst its K nearest neighbors measured by a distance function. If K = 1, then the case is simply assigned to the class of its nearest neighbor. It should also be noted that all three distance measures are only valid for continuous variables. In the instance of categorical variables the Hamming distance must be used. It also brings up the issue of standardization of the numerical variables between 0 and 1 when there is a mixture of numerical and categorical variables in the dataset.

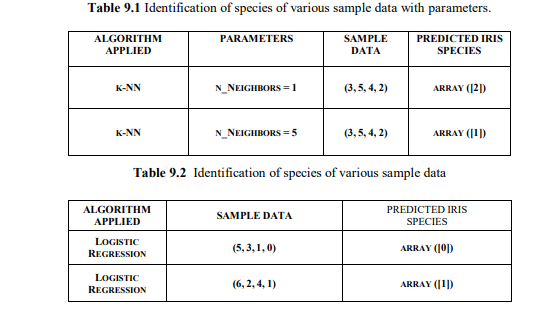


Choosing the optimal value for K is best done by first inspecting the data. In general, a large K value is more precise as it reduces the overall noise but there is no guarantee. Cross-validation is another way to retrospectively determine a good K value by using an independent dataset to validate the K value. Historically, the optimal K for most datasets has been between 3-10. That produces much better results than 1NN.

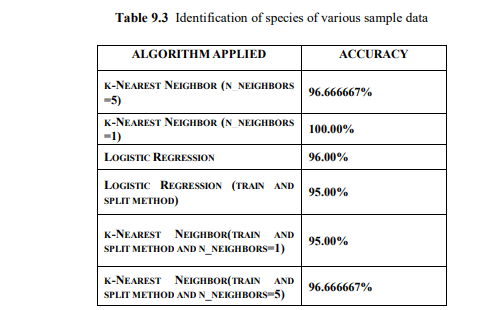


**4. RESULT**

Identification species for testing samples are given Table 8.1.1 and Table 8.1.2 and Table 8.1.3



Identification accuracy for testing samples are given Table 9.3.



It is less competitive than other peers because easy methods have been implemented to get the solution.

**5. CONCLUSION**

The primary goal of supervised learning is to build a model that “generalizes”. Here in this project we make predictions on unseen data which is the data not used to train the model hence the machine learning model built should accurately predicts the species of future flowers rather than accurately predicting the label of already trained data.

Our team has contributed equally to the project and it was a awesome experience in exploring the tools of data mining, we have learnt many thing and in future we will work on a bigger project.

**6. Acknowledgement**

This project is to fulfill the requirements of Data Mining Course at SRMIST , DELHI NCR. We would like to thank our teacher for the guidance of the work done.

**7. References:**

[1] Diptam Dutta, Argha Roy, Kaustav Choudhury, “Training Aritificial Neural Network Using Particle Swarm Optimization Algorithm”, International Journal on Computer Science And Engineering(IJCSE), Volume 3, Issue 3, March 2013.

[2] Poojitha V, Shilpi Jain, “A Collecation of IRIS Flower Using Neural Network CLusterimg tool in MATLAB”, International Journal on Computer Science And Engineering(IJCSE).

[3] Vaishali Arya, R K Rathy, “An Efficient Neura-Fuzzy Approach For Classification of Dataset”, International Conference on Reliability, Optimization and Information Technology, Feb 2014.