Nicolas Indelicato

Methodology Description

Executive Summary

There are times, that when data sets are very similar, classification becomes difficult. The ability to classify the different instances can either be accomplished with by examining the data closely with the naked eye and descriptive statistics, or by using an algorithm. There are two sets of data that offer a problem to the classification of the entities into distinct subsets.

The wine and the iris datasets both make classification difficult and it is recommended that an algorithm be applied to expedite the classification process. The wine dataset includes thirteen attributes and three classes of wine, while the iris dataset includes four attributes and three classes of flowers. The algorithm that will be used to classify these datasets is the k-Nearest Neighbor classification algorithm. This algorithm will first require a minor training period and will then be able to classify the distinct classes within the datasets quickly, efficiently, and with little error.

When the k-Nearest Neighbor classification algorithm was applied to the iris dataset with all non-normalized attributes, a misclassification rate of only 6% was achieved, while normalization produced a misclassification rate of 7.33%. When the k-Nearest Neighbor classification algorithm was applied to the iris dataset with the non-normalized attributes of petal length and petal width, a misclassification rate of only 4.67% was achieved, while normalization produced a misclassification rate of 7.33%. The iris dataset, therefore, was most accurately classified, 95.33% correctly, using the non-normalized attributes of petal length and petal width.

When the k-Nearest Neighbor classification algorithm was applied to the wine dataset with all non-normalized attributes, a misclassification rate of 27.45% was achieved, while normalization produced a misclassification rate of only 5.88%. When the k-Nearest Neighbor classification algorithm was applied to the wine dataset with the non-normalized attributes of Total Phenols, Flavanoids, and OD280/OD315 of Diluted Wines, a misclassification rate of 20.92% was achieved, while normalization produced a misclassification rate of only 20.92%. The wine dataset, therefore, was most accurately classified, 94.12% correctly, using all normalized attributes.

Problem Description

There are times, that when data sets are very similar, classification becomes difficult. The ability to classify the different instances can either be accomplished with by examining the data closely with the naked eye and descriptive statistics, or by using an algorithm. There are two sets of data that offer a problem to the classification of the entities into distinct subsets.

The wine and the iris datasets both make classification difficult and it is recommended that an algorithm be applied to expedite the classification process. The wine dataset includes thirteen attributes and three classes of wine, while the iris dataset includes four attributes and three classes of flowers. The algorithm that will be used to classify these datasets is the k-Nearest Neighbor classification algorithm. This algorithm will first require a minor training period and will then be able to classify the distinct classes within the datasets quickly, efficiently, and with little error.

Analysis Technique

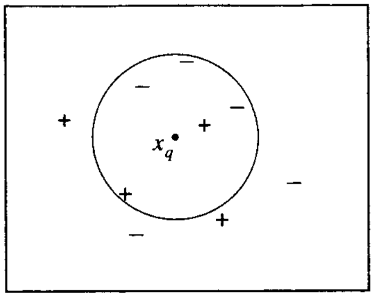
In order to classify the data presented in the wine and the iris datasets, the k-Nearest Neighbor classification algorithm will be applied. This algorithm will be very beneficial in the accurate classification of the data provided. Not only is this algorithm an instance-based learning algorithm with a clever method of classifying data, but is has also been extensively used in classification and pattern recognition since the 1960s (Witten & Frank, 2000). Instance-based learning is favorable in the fact that the algorithm will classify the data it has been given by each new instance it encounters. Each instance will be classified by comparing its location with the neighbors' locations. As Giudici (2003) states, "k is a fixed constant that specifies the number of elements to be included in each neighborhood" (p. 120). The constant "k" can be any constant and will be set to seven nearest neighbors to compare against a relatively large neighborhood without being too large and encompassing of points that may cause the classification to become erroneous.

When the algorithm is applied, each new instance will undergo a distance calculation from its location to the locations of the data provided in the training set. The training set is first read into the algorithm to give the algorithm a basis from which to classify the incoming instances of the chosen dataset. The distance is calculated using standard Euclidean distance in n space, with n being the number of attributes applied to the algorithm in order to classify the dataset.

Standard Euclidean distance in n space, with $a_r(x)$ defined to be the r^{th} attribute of instance x; and x_i and x_j representing two separate instances of the dataset is calculated simply as the square root of the sum of the squares of the distances between x_i and x_j for each attribute in x, represented by the following equation:

$$d(x_i, x_j) \equiv \sqrt{\sum_{r=1}^{n}(a_r(x_i) - a_r(x_j))^2} \quad \text{(Mitchell, 1997, p. 232)}$$

The instance will then be placed into the class from which the majority of the k nearest neighbors is classified as. This will make the classifications very accurate with every new instance presented depending on the value of k used in the algorithm. For example, in the graph below, x_q would be classified as positive if the algorithm used the one nearest neighbor, but it would be classified as negative if the algorithm used the five nearest neighbors.



(Mitchell, 1997, p. 233)

According to Giudici (2003), the model is “applied to the whole data set, but the statistical analysis is split into separate local analyses,” which allows the model to more closely fit the data (p. 119).

The training set that is applied to the algorithm is taken verbatim (Witten & Frank, 2000). The algorithm does not change the training set. It will read in the training data and not make any modifications to it. Once the training set is read into the algorithm, the algorithm has been trained. It is not necessary to train the algorithm to fit the entire test dataset like other classification algorithms would require. Many times training an algorithm will take an extensive period of time, while the dataset it must classify is repeatedly fed into the algorithm to make the algorithm’s boundaries more closely fit the data.

It is recommended that the training set for the iris dataset include ten random entities from each of the three categories of flowers. The training set for the wine dataset is recommended to include ten random entities from each of the three categories of wine. These training sets should theoretically include enough instances of the dataset that the algorithm should be able to differentiate between the three classes, with great accuracy, in each dataset.

The algorithm will complete one instance at a time, by comparing the distance between the test entity and the entire training set. It will completely finish when it has examined all of the instances in the dataset. It will then print to an output file each individual entity and its classification resulting from analysis by the algorithm.

Assumptions

The use of the k-Nearest Neighbor classification algorithm will assume that the least relevant attributes of the dataset will be excluded from use in the algorithm. The attributes with the least correlation between the attribute and the appropriate class must be discarded in order to keep the algorithm accurate. If these attributes, which do not assist in the correct classification of an instance, are not removed, two very similar instances may end up plotting on completely opposite sides of the n space graph, resulting in a Euclidean distance measurement that is highly inaccurate.

With the removal of these unnecessary attributes, the algorithm will produce a truer and more accurate classification of each of the instances. This also assumes that there are no missing values in the datasets and that all values are numerical, or quantitative, not text based, or qualitative. It is important that the values also be normalized, so that the attributes in the dataset all have equal weights (Witten & Frank, 2000). Missing values, qualitative values, and non-normalized values can cause the algorithm to drastically misclassify instances.

It must also be assumed that the datasets being used with the algorithm are relatively small in size because of the large processing overhead that exists in the algorithm. One final, essential assumption is that the training set must be assumed to have the preferred and correct classification for each item (Dunham 2003).

Results

After classification with the algorithm has been accomplished, the advantages and disadvantages of the k-Nearest Neighbors classification algorithm have become apparent. The algorithm was run a total of eight times on two different datasets. Each dataset was run through the algorithm once normalized and once non-normalized, once with all attributes and once with all relevant attributes. The relevant attributes were determined to be those attributes with the highest correlation, positive or negative, between the attribute and the class. The two attributes with the highest correlation were used for the iris dataset, while the three attributes with the highest correlation were used for the wine dataset. Normalization was accomplished for each attribute by taking the maximum value for that attribute and dividing each entity's attribute by that value.

When the k-Nearest Neighbor classification algorithm was applied to the iris dataset with all non-normalized attributes, a misclassification rate of only 6% was achieved. None of the setosa or versicolor were misclassified, and only nine of the virginica were misclassified. When these attributes were normalized, a misclassification rate of 7.33% was achieved. None of the setosa was misclassified, one of the versicolor was misclassified, and ten of the virginica were misclassified.

When the k-Nearest Neighbor classification algorithm was applied to the iris dataset with the non-normalized attributes of petal length and petal width, a misclassification rate of only 4.67% was achieved. None of the setosa or versicolor were misclassified, and only seven of the virginica were misclassified. When these attributes were normalized, a misclassification rate of 7.33% was achieved. None of the setosa or versicolor was misclassified, and eleven of the virginica were misclassified.

When the k-Nearest Neighbor classification algorithm was applied to the wine dataset with all non-normalized attributes, a misclassification rate of 27.45% was achieved. Seven instances of class one wine were misclassified, twenty-three instances of class two wine were misclassified, and twelve instances of class three wine were misclassified. When these attributes were normalized, a misclassification rate of only 5.88% was achieved. None of the class one or class three wines were misclassified, and only nine instances of class two wine were misclassified.

When the k-Nearest Neighbor classification algorithm was applied to the wine dataset with the non-normalized attributes of Total Phenols, Flavanoids, and OD280/OD315 of Diluted Wines, a misclassification rate of 20.92% was achieved. One instance of class one wine was misclassified, thirty-one instances of class two wine were misclassified, and none of the class three wine was misclassified. When these attributes were normalized, a misclassification rate of only 20.92% was achieved. Two instances of the class one wine were misclassified, thirty instances of class two wine were misclassified, and none of the class three wine was misclassified.

*For a tabular view of the results of the algorithm on the datasets, please consult the appendix.

Issues

One issue was encountered when the algorithm was run. A few instances of the wine dataset were split between two classes. That is to say, the nearest neighbors included an equal amount of neighbors from two classes. This issue was resolved by classifying the instance into the dataset that included the nearest neighbor to the instance. This was done to maximize the correct classification percentage of the algorithm and ensure that the instance was classified to the dataset from which its attributes most closely resembled.

Appendix

MISCLASSIFICATION PERCENTAGES

IRIS DATASET

	All Attributes		Petal Length and Petal Width	
	Non-Normalized	**Normalized**	**Non-Normalized**	**Normalized**
Setosa	0/150 = 0%	0/150 = 0%	0/150 = 0%	0/150 = 0%
Versicolor	0/150 = 0%	1/150 = 0.67%	0/150 = 0%	0/150 = 0%
Virginica	9/150 = 6%	10/150 = 6.67%	7/150 = 4.67%	11/150 = 7.33%
Total	6%	7.33%	4.67%	7.33%

WINE DATASET

	All Attributes		Phenols, Flavanoids, OD280/OD315	
	Non-Normalized	**Normalized**	**Non-Normalized**	**Normalized**
Class 1	7/153 = 4.58%	0/153 = 0%	1/153 = 0.65%	2/153 = 1.31%
Class 2	23/153 = 15.08%	9/153 = 5.88%	31/153 = 20.26%	30/153 = 19.61%
Class 3	12/153 = 7.84%	0/153 = 0%	0/153 = 0%	0/153 = 0%
Total	27.45%	5.88%	20.92%	20.92%

References

Aleshunas, J. (2004). ID3 – Iris Data. Retrieved from: Supplemental Excel Data Sets: http://mercury.webster.edu/aleshunas/MATH%204500/Supplemental%20Excel%20Data%20Sets.htm

Aleshunas, J. (2004). Wine. Retrieved from: Supplemental Excel Data Sets: http://mercury.webster.edu/aleshunas/MATH%204500/Supplemental%20Excel%20Data%20Sets.htm

Dunham, Margaret H. (2003). Data Mining: Introductory and Advanced Topics. New Jersey: Pearson Education Inc.

Giudici, Paolo. (2003). Applied Data Mining: Statistical Methods for Business and Industry. England: John Wiley & Sons Ltd.

Mitchell, Tom M. (1997). Machine Learning. Oregon: McGraw-Hill.

Witten, I., & Frank, E. (2000). Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations. California: Morgan Kaufmann Publishers.