SFS and K-Means

Amrith Deepak

Abstract:

The glass data showing type of glass, Iris data showing species of Iris plants, and spambase data examining spam were analyzed using stepwise forward selection with Naïve Bayes and with k-means. The glass and spambase data performed slightly better with the Naïve Bayes algorithm and the iris data performed slightly better with the k-means algorithm, but the difference wasn't significant. Both algorithms performed best on the spambase data.

Problem Statement:

This project is examining how well stepwise forward selection works and examining how K-Means performs compared to Naïve Bayes. These algorithms are being evaluated on how they perform on different datasets to classify data. 9 attributes are being used to predict type of glass, length and width statistics to predict species of iris plants, and 57 email attributes are being used to predict whether email is spam.

Hypothesis:

Stepwise forward selection should give a small number of features to focus on. It is predicted that Naïve Bayes will perform better on the spambase and iris data, where there are fewer number of classes, but worse in the glass data where there are many classes. Naïve Bayes also has a conditional independence assumption, which doesn't hold, but K-means generally works well with variables having similar variance, and prior probabilities, which may not hold.

Algorithms:

SFS works by adding features one at a time as long as adding the feature improves performance. Each iteration, one feature is added and the feature added is the feature that gives maximum performance at that step. Naïve Bayes is being used to test the performance.

The Naïve Bayes algorithm calculates all the conditional probabilities of each variable given each value of the variable we are trying to predict. It's an application of Bayes theorem. To test the algorithm, all the conditional probabilities are multiplied for all values of the variable being predicted and the greatest conditional probability is used.

K-means is a clustering algorithm that partitions variables into k clusters. K-means uses expectation maximization and tries to make points within each cluster as similar as possible and points outside the cluster as different as possible. At the end, the clusters are being assigned to classes.

Experimental Approach:

For each of the three datasets, the data was split into training and testing data. The algorithms described above were followed. SFS was used to choose features and Naïve Bayes was used at each step to test performance. For k-means, after creating clusters, each cluster was associated with a class and the accuracy of classification was compared to Naïve Bayes. In k-means, the features given by SFS were used.

Results:

Dataset	**Naïve Bayes Performance**	**K-Means Performance**
Glass	0.2	0.16
Iris	0.2933	0.3158
Spambase	0.6439	0.6368

The Naïve Bayes performed well on the spambase data where it classified with accuracy 0.6439. In the glass and iris data, it performed with 0.2 and 0.2933 accuracy respectively.

The K-Means Clustering performed well on the spambase data where it classified 447 correctly and 255 incorrectly. In the iris data, it classified 12 correctly and 26 incorrectly. In the glass data, it classified 8 correctly and 42 incorrectly.

Both the algorithms performed well in the spambase data and poorly on the iris and glass data. In the iris data, k-means performed slightly better and in the glass and spambase data, k-means performed slightly worse, but the difference isn't significant.

Discussion:

They both performed very similarly, but the k-means performed slightly better in the iris data and slightly worse in the spambase and glass data. It seems like the training data wasn't very well representative of the testing data in both the iris and glass data, but was better in the spambase data. One other reason for weaker performance can be the limitation of SFS in that it chooses the best feature for each step, but a combination of other features can sometimes give better overall performance. One example of this is also shown in the output. The accuracy also seems to be higher in the spambase data where there are only two classes compared to iris where there are 3 classes and glass where there are 7 classes.

The average silhouette coefficient for the glass, iris, and spambase datasets were 0.834, 0.669, and 0.877 respectively. All of these are positive values close to 1, which means that the clusters are well separated. Although there were distinct clusters, the performance didn't seem to be very high. This could partially be due to the mapping between clusters and classes not being so clear.

Summary:

The glass and spambase data performed slightly better with the Naïve Bayes algorithm and the iris data performed slightly better with the k-means algorithm, but the difference wasn't significant. Both algorithms performed best on the spambase data. Naïve Bayes is a classification algorithm, and K-Means was a clustering algorithms, but in the code each cluster was assigned to a class. The code trained with approximately 2/3 of the data and tested with approximately 1/3 of the data.

References:

Guyon, Isabelle, and Elisseeff Andre´. "An Introduction to Variable and Feature Selection." *Journal of Machine Learning Research 3*, Mar. 2003, www.jmlr.org/papers/volume3/guyon03a/guyon03a.pdf.

Kanungo, Tapas, et al. "An Efficient k-Means Clustering Algorithm: Analysis and Implementation." *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, July 2002, www.cs.umd.edu/~mount/Projects/KMeans/pami02.pdf.

Datasets:

https://archive.ics.uci.edu/ml/datasets/Glass+Identification

https://archive.ics.uci.edu/ml/datasets/Iris

https://archive.ics.uci.edu/ml/datasets/Spambase