A Survey of Feature Selection Techniques

Barak Chizi
Tel-Aviv University, Israel

Lior Rokach
Ben-Gurion University, Israel

Oded Maimon
Tel-Aviv University, Israel

INTRODUCTION

Dimensionality (i.e., the number of data set attributes or groups of attributes) constitutes a serious obstacle to the efficiency of most data mining algorithms (Maimon and Last, 2000). The main reason for this is that data mining algorithms are computationally intensive. This obstacle is sometimes known as the "curse of dimensionality" (Bellman, 1961).

The objective of Feature Selection is to identify features in the data-set as important, and discard any other feature as irrelevant and redundant information. Since Feature Selection reduces the dimensionality of the data, data mining algorithms can be operated faster and more effectively by using Feature Selection. In some cases, as a result of feature selection, the performance of the data mining method can be improved. The reason for that is mainly a more compact, easily interpreted representation of the target concept.

The filter approach (Kohavi , 1995; Kohavi and John ,1996) operates independently of the data mining method employed subsequently -- undesirable features are filtered out of the data before learning begins. These algorithms use heuristics based on general characteristics of the data to evaluate the merit of feature subsets. A sub-category of filter methods that will be refer to as rankers, are methods that employ some criterion to score each feature and provide a ranking. From this ordering, several feature subsets can be chosen by manually setting

There are three main approaches for feature selection: wrapper, filter and embedded.

The wrapper approach (Kohavi, 1995; Kohavi and John,1996), uses an inducer as a black box along with a statistical re-sampling technique such as cross-validation to select the best feature subset according to some predictive measure.

The embedded approach (see for instance Guyon and Elisseeff, 2003) is similar to the wrapper approach in the sense that the features are specifically selected for a certain inducer, but it selects the features in the process of learning.

BACKGROUND

Feature selection algorithms search through the space of feature subsets in order to find the best subset. This subset search has four major properties (Langley, 1994): starting point, search organization, evaluation strategy, and stopping criterion.

Starting Point: Selecting a point in the feature subset space from which to begin the search can affect the direction of the search.

Search Organization: A comprehensive search of the feature subspace is prohibitive for all but a small initial number of features.

Evaluation Strategy: How feature subsets are evaluated (filter, wrapper and ensemble).

Stopping Criterion: A feature selector must decide when to stop searching through the space of feature subsets.

FEATURE SELECTION TECHNIQUES

This section provides a survey of techniques for each strategy described on previous sections.

Feature Filters

The filter methods were the earliest approaches for feature selection. All filter methods use general properties of the data in order to evaluate the merit of feature subsets. As a result, filter methods are generally much faster and practical than wrapper methods, especially for using it on data of high dimensionality. Detailed experiments for each method presented below can be found in Hall's work (1999) and on Liu and Motoda (1998) book on Feature Selection.

FOCUS

Almuallim and Dietterich (1991) describe an algorithm originally designed for Boolean domains called FOCUS. FOCUS exhaustively searches the space of feature subsets until every combination of feature values is associated with one value of the class. After selecting the subset, it passed to ID3 (Quinlan, 1986), which constructs a decision tree.

LVF

Similar algorithm to FOCUS is LVF(Liu and Setiono, 1996) describe. LVF is consistency driven and can handle noisy domains if the approximate noise level is known a- priori. Every round of execution LVF generates a random subset from the feature subset space. If the chosen subset is smaller than the current best subset, the inconsistency rate of the dimensionally reduced data described by the subset is compared with the inconsistency rate of the best subset. If the subset is at least as consistent as the best subset, the subset replaces the best subset.

An Information Theoretic Feature Filter

Koller and Sahami (1996) described a feature selection algorithm based on information theory and probabilistic reasoning. The rationale behind this technique is that, since the goal of an induction algorithm is to estimate the probability distributions over the class values, given the original feature set, feature subset selection should attempt to remain as close to these original distributions as possible.

An Instance Based Approach to Feature Selection – RELIEF

RELIEF (Kira and Rendell, 1992) uses instance based learning to assign a relevance weight to each feature. The weight for each feature reflects its ability to single out the class values. The Features are ranked by its weights and chosen by using a user- specified threshold. RELIEF randomly choosing instances from the training data. For every instance RELIEF samples the nearest instance of the same class (nearest hit) and finds the opposite class (nearest miss). The weight for each feature is updating according to how well its values differentiate the sampled instance from its nearest hit and nearest miss. Feature will gain a high weight if it differentiates between instances from different classes and has the same value for instances of the same class.

Simba and G-Flip

Gilad-Bachrach et al. (2004), introduced a new approach called SIMBA (Iterative Search Margin Based Algorithm), which outperforms RELIF. This approach introduces the idea of measuring the quality of a set of features by the margin it induces. To overcome the drawback of iterative search, Gilad-Bachrach et al (2004), present A Greedy Feature Flip Algorithm called G-Flip. The G-Flip is a greedy search algorithm for maximizing the margin function