

Chapter 12

Feature Selection

Xiaogang Su
Department of Statistics
University of Central Florida

Outline

- Why Feature Selection?
- Categorization of Feature Selection Methods
 - Filter Methods
 - Wrapper Methods
 - Embedded
- RELIEF

Why Feature Selection?

- It is cheaper to measure less variables.
- The resulting classifier is simpler and potentially faster.
- Prediction accuracy may improve by discarding irrelevant variables.
- Identifying relevant variables gives more insight into the nature of the corresponding classification problem.
- Alleviate the “curse of dimensionality”.

Categorization of Feature Selection Methods

- Wrapper:
 - Feature selection takes into account the contribution to the performance of a given type of classifier.

- Filter:
 - Feature selection is based on an evaluation criterion for quantifying how well feature (subsets) discriminate the two classes.

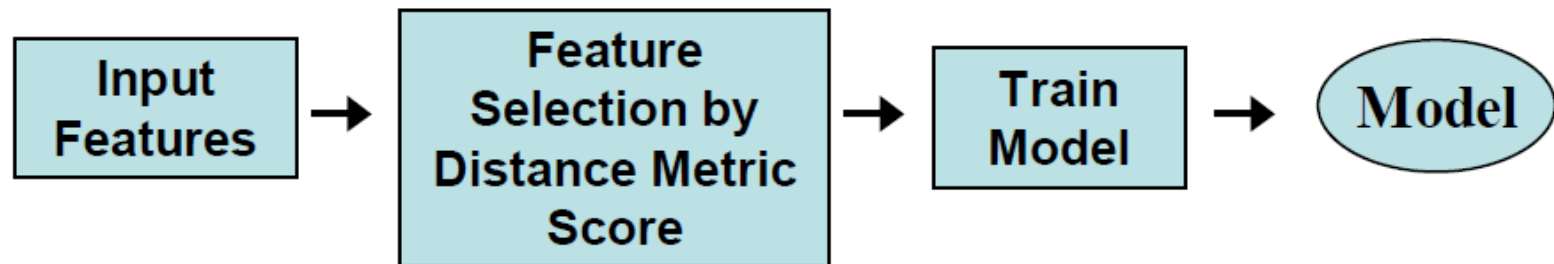
- Embedded:
 - Feature selection is part of the training procedure of a classifier (e.g. decision trees).

Embedded Methods

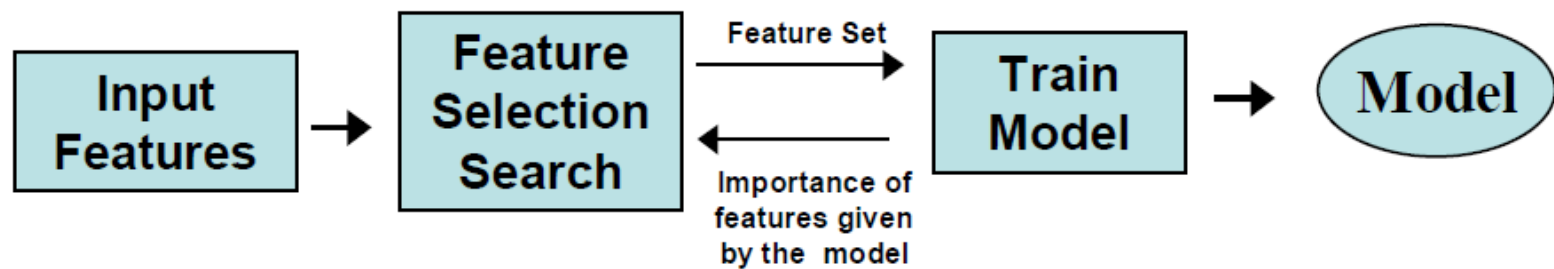
- Attempt to *jointly* or *simultaneously* train both a classifier and a feature subset.
- Often optimize an objective function that jointly rewards accuracy of classification and penalizes use of more features.
- Intuitively appealing.
- **Example**: tree-building algorithms

Filter and Wrapper Methods

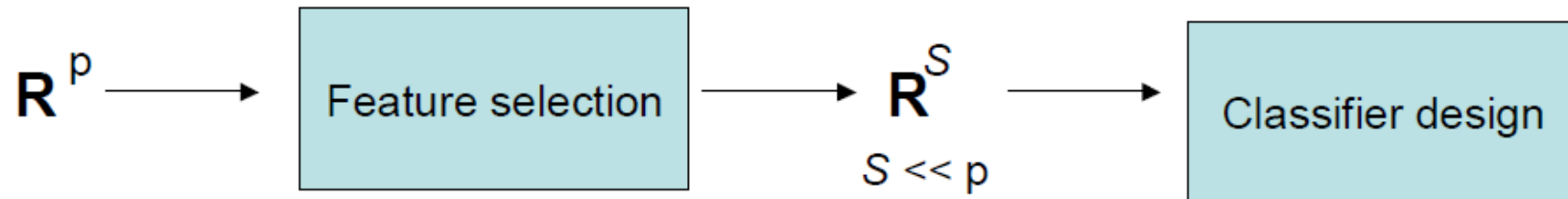
Filter Approach



Wrapper Approach



Filter Methods



- Features are scored independently and the top S of them will be used by the classifier.
- **Score:** correlation, mutual information, t-statistic, F-statistic, p-value, tree variable importance ranking, etc.

Several Filter Methods for Variable Screening

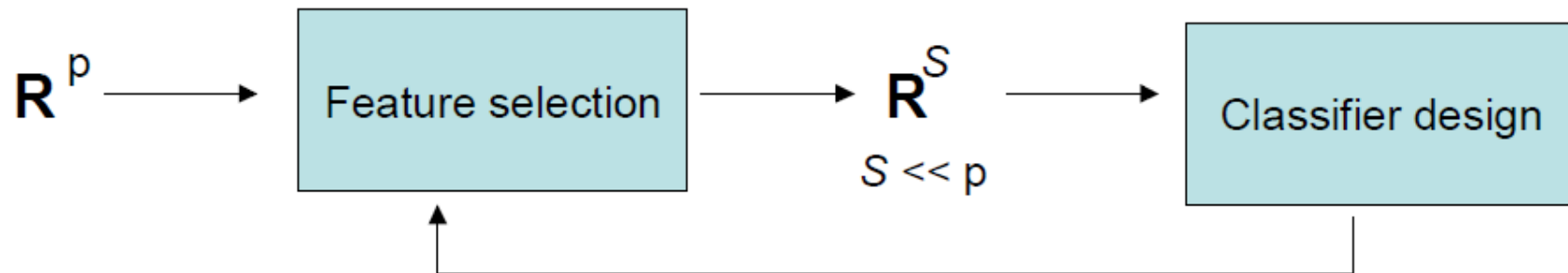
- Correlation matrix; VIF (Variance Inflation Factor)
- **RELIEF** (Kira and Rendell, 1992) ranks variables as per distance.
- **FOCUS** (Almuallim and Dietterich, 1991) searches for smallest subset that completely discriminates between target classes.

In filter methods, variables are evaluated independently and not in context of a learning algorithm.

Problems with Filter Methods

- Redundancy in selected features: features are considered independently and not measured on the basis of whether they contribute new information.
- Interactions among features generally cannot be explicitly incorporated (some filter methods are smarter than others).
- Classifier has no say in what features should be used: some scores may be more appropriate in conjunction with some classifiers than others.

Wrapper Methods



- Done in an iterative manner. Many feature subsets are scored based on classification performance of a classifier and the best is used.
- **Problems:**
 - Computationally expensive: for each feature subset to be considered, a classifier must be built and evaluated.
 - No exhaustive search is possible (2^p subsets to consider) : generally greedy algorithms only.
 - Risk for overfitting.

RELIEF

- Initialized by Kira K, Rendell L, *10th Int. Conf. on AI*, 129-134, 1992. Having been extended, the first version of RELIEF handles only binary responses.
- **Idea**: Relevant features make (1) nearest examples of same class closer and (2) nearest examples of opposite classes more far apart.
- RELIEF assigns weights to variables based on how well they separate samples from their nearest neighbors from the same and from the opposite class.

RELIEF - Steps

- Normalize data first.
- Set the relevance of every feature as zero
- For each example or observation in training set:
 - Find nearest example or observation from same (hit) and opposite class (miss)
 - Update weight of each feature by adding $abs(example - miss) - abs(example - hit)$

Algorithm RELIEF for Data with Binary Responses.

-
- Initialize all importance measures $W_j = W(X_j) = 0$ for $j = 1, 2, \dots, p$;
 - Do $k = 1, 2, \dots, K$
 - Randomly select an observation $\{k : (y_k, \mathbf{x}_k)\}$, call it “Obs- k ”;
 - Find the observation m , called ‘the nearest miss’, which is closest, in terms of the distance in \mathbf{x} , to “Obs- k ” and has response y_m equal to y_k ;
 - Find the observation h , called ‘the nearest hit’, which is closest, in terms of the distance in \mathbf{x} , to “Obs- k ” and has response y_h unequal to y_k ;
 - Do $j = 1, 2, \dots, p$,
 - Update $W_j \leftarrow W_j - \text{diff}(x_{kj}, x_{mj})/m + \text{diff}(x_{kj}, x_{hj})/m$;
 - End do;
 - End do.
-

RELIEF- Algorithm

In the outlined algorithm, function $\text{diff}(x_{kj}, x_{k'j})$ computes the difference between x_{kj} and $x_{k'j}$, the values of predictor X_j for observations k and k' . While various distance definitions can apply to measure the difference, it is defined in the original proposal, for categorical predictors, as

$$\text{diff}(x_{kj}, x_{k'j}) = \begin{cases} 0 & \text{if } x_{kj} = x_{k'j} \\ 1 & \text{otherwise} \end{cases}$$

and for continuous predictors as

$$\text{diff}(X_{kj}, X_{k'j}) = \frac{|x_{kj} - x_{k'j}|}{\max(X_j) - \min(X_j)},$$

where $\max(X_j)$ and $\min(X_j)$ are the maximum and minimum of X_j , respectively. With this formulation, it is no need to normalize or standardize variables that are measured in different scales. An implementation of RELIEF is available in R package: `dprep`.

Function relief {dprep}

```
relief(data, nosample, threshold, vnom)
```

Arguments

data	the dataset for which feature selection will be carried out
nosample	number of instances drawn from the original dataset
threshold	the cutoff point to select the features
vnom	a vector containing the indexes of the nominal features

References

- KIRA, K. and RENDEL, L. (1992). The Feature Selection Problem: Traditional Methods and a new algorithm. *Proc. Tenth National Conference on Artificial Intelligence*, MIT Press, 129-134.
- KONONENKO, I., SIMEC, E., and ROBNIK-SIKONJA, M. (1997). Overcoming the myopia of induction learning algorithms with RELIEFF. *Applied Intelligence* 7: 39-55.