

# Context–Sensitive Attribute Evaluation

**Marko Robnik-Šikonja**

*University of Ljubljana, FRI*

## INTRODUCTION

The research in machine learning, data mining, and statistics has provided a number of methods that estimate the usefulness of an attribute (feature) for prediction of the target variable. The estimates of attributes' utility are subsequently used in various important tasks, e.g., feature subset selection, feature weighting, feature ranking, feature construction, data transformation, decision and regression tree building, data discretization, visualization, and comprehension. These tasks frequently occur in data mining, robotics, and in the construction of intelligent systems in general.

A majority of attribute evaluation measures used are myopic in a sense that they estimate the quality of one feature independently of the context of other features. In problems which possibly involve much feature interactions these measures are not appropriate. The measures which are historically based on the Relief algorithm (Kira & Rendell, 1992) take context into account through distance between the instances and are efficient in problems with strong dependencies between attributes.

## BACKGROUND

The majority of feature evaluation measures are impurity based, meaning that they measure impurity of the class value distribution. These measures evaluate each feature separately by measuring impurity of the splits resulting from partition of the learning instances according to the values of the evaluated feature. Assuming the conditional independence of the features upon the class, these measures are myopic, as they do not take the context of other features into account. If the target concept is a discrete variable (the classification problem) well-known and used measures of these kind are information gain (Hunt et al., 1966), Gini index (Breiman

et al., 1984), j-measure (Smyth & Goodman, 1990), Gain ratio (Quinlan, 1993) and MDL (Kononenko, 1995). Large differences in the impurity of class values before the split, and after the split on a given feature, imply purer splits and therefore more useful features. We cannot directly apply these measures to numerical features, but we can use discretization techniques and then evaluate discretized features. If the target concept is presented as a real valued function (regression problem), the impurity based evaluation heuristics used are e.g., the mean squared and the mean absolute error (Breiman et al., 1984).

The term context here represents related features, which interact and only together contain sufficient information for classification of instances. Note that the relevant context may not be the same for all instances in a given problem. The measures which take the context into account through distance between the instances and are efficient in classification problems with strong dependencies between attributes are Relief (Kira & Rendell, 1992), Contextual Merit (Hong, 1997), and ReliefF (Robnik-Sikonja & Kononenko, 2003). RReliefF is a measure proposed to address regression problems (Robnik-Sikonja & Kononenko, 2003).

For a more thorough overview of feature quality evaluation measures see (Kononenko & Kukar, 2007). Breiman (2001) has proposed random forest learning algorithm which, as a byproduct, can output the utility of the attributes. With large enough data sample which ensures sufficiently large and diverse trees in the forest these estimates are also context-sensitive. For an overview of other recent work, especially in the context of feature subset selection see (Guyon & Elisseeff, 2003). Note that this chapter is solely a machine learning view of feature selection and omits methods for model selection in regression that amount to feature selection. A recent work trying to bridge the two worlds is (Zhou et al., 2006).

## MAIN FOCUS

The main idea of how to take the context of other features into account was first presented in algorithm Relief (Kira & Rendell, 1992), which is designed for two-class problems without missing values. The idea of the algorithm, when analyzing learning instances, is to take into account not only the difference in feature values and the difference in classes, but also the distance between the instances. Distance is calculated in the feature space, therefore similar instances are close to each other and dissimilar are far apart. By taking the similarity of instances into account, the context of all the features is implicitly considered.

The algorithm Relief illustrated in Box 1 randomly selects an instance and finds the nearest instance from the same class (nearest hit) and the nearest instance from the opposite class (nearest miss). Then it updates the quality of each feature with respect to whether the feature differentiates two instances from the same class (undesired property of the feature) and whether it differentiates two instances from opposite classes (desired property). By doing so, the quality estimate takes into account the local ability of the feature to differentiate between the classes. Repeating the whole procedure for large enough sample these local estimates provide a global picture of the feature utility, but the locality implicitly takes into account the context of other features.

Let's say that a given feature explains the change of the class value of the instance, when the change of its values is one of the minimal changes required for

changing the class value. The quality evaluations of Relief algorithms can then be interpreted as the portions of the explained concept i.e., as the ratio between the number of the explained changes in the concept and the number of examined instances.

## ReliefF for Classification and RReliefF for Regression

A more realistic and practically useful variant of Relief is its extensions, called ReliefF for classification and RReliefF for regression problems (Robnik-Sikonja & Kononenko, 2003). Unlike original Relief these two algorithms are able to deal with incomplete and noisy data. The most important difference is in searching for the nearest hit and miss. Noise or mistakes in class and/or feature values significantly affects the selection of nearest hits and misses. In order to make this process more reliable in the presence of noise, ReliefF and RReliefF use several nearest hits and misses and average their contributions to features' quality estimates. ReliefF can be used also for evaluating the feature quality in multi-class problems and to do so it searches for nearest instances from each class. The contributions of different classes are weighted with their prior probabilities. In regression problems the target variable is numerical, therefore nearest hits and misses cannot be used in a strict sense. RReliefF (Regression ReliefF) uses a kind of "probability" that two instances belong to two "different" classes. This "probability" is modeled with the distance between the values of the target variable of two learning instances.

Box 1.

### **Algorithm Relief**

*Input:* set of instances  $\langle x_i, \tau_i \rangle$

*Output:* the vector  $W$  of attributes' evaluations

set all weights  $W[A] := 0.0$ ;

**for**  $i := 1$  **to**  $\#sample\_size$  **do begin**

    randomly select an instance  $R$ ;

    find nearest hit  $H$  and nearest miss  $M$ ;

**for**  $A := 1$  **to**  $\#all\_attributes$  **do**

$W[A] := W[A] - diff(A,R,H)/m + diff(A,R,M)/m$ ;

**end**;

3 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: [www.igi-global.com/chapter/context-sensitive-attribute-evaluation/10840](http://www.igi-global.com/chapter/context-sensitive-attribute-evaluation/10840)

## Related Content

---

### Preference Modeling and Mining for Personalization

Seung-won Hwang (2009). *Encyclopedia of Data Warehousing and Mining, Second Edition* (pp. 1570-1574). [www.irma-international.org/chapter/preference-modeling-mining-personalization/11028](http://www.irma-international.org/chapter/preference-modeling-mining-personalization/11028)

### Enhancing Web Search through Query Expansion

Daniel Crabtree (2009). *Encyclopedia of Data Warehousing and Mining, Second Edition* (pp. 752-757). [www.irma-international.org/chapter/enhancing-web-search-through-query/10904](http://www.irma-international.org/chapter/enhancing-web-search-through-query/10904)

### A Genetic Algorithm for Selecting Horizontal Fragments

Ladjel Bellatreche (2009). *Encyclopedia of Data Warehousing and Mining, Second Edition* (pp. 920-925). [www.irma-international.org/chapter/genetic-algorithm-selecting-horizontal-fragments/10930](http://www.irma-international.org/chapter/genetic-algorithm-selecting-horizontal-fragments/10930)

### Data Warehousing for Association Mining

Yuefeng Li (2009). *Encyclopedia of Data Warehousing and Mining, Second Edition* (pp. 592-597). [www.irma-international.org/chapter/data-warehousing-association-mining/10881](http://www.irma-international.org/chapter/data-warehousing-association-mining/10881)

### Comparing Four-Selected Data Mining Software

Richard S. Segall (2009). *Encyclopedia of Data Warehousing and Mining, Second Edition* (pp. 269-277). [www.irma-international.org/chapter/comparing-four-selected-data-mining/10832](http://www.irma-international.org/chapter/comparing-four-selected-data-mining/10832)