Feature-weighted k-Nearest Neighbor Classifier

Diego P. Vivencio
DC/UFSCar, S. Carlos, SP, Brazil
vivencio@comp.ufscar.br

Estevam R. Hruschka Jr.
DC/UFSCar, S. Carlos, SP, Brazil
estevam@dc.ufscar.br

M. do Carmo Nicoletti
DC/UFSCar, S. Carlos, SP, Brazil
carmo@dc.ufscar.br

Edimilson B. dos Santos
DC/UFSCar, S. Carlos, SP, Brazil
edimilson_santos@dc.ufscar.br

Sebastian D. C. O. Galvão
DC/UFSCar, S. Carlos, SP, Brazil
edimilson_santos@dc.ufscar.br

Abstract

This paper proposes a feature weighting method based on χ2 statistical test, to be used in conjunction with a k-NN classifier. Results of empirical experiments conducted using data from several knowledge domains are presented and discussed. Forty four out of forty five conducted experiments favoured the feature weighted approach and are empirical evidence that the proposed weighting process based on χ^2 is a good weighting strategy.

Keywords: Feature Selection, Instance-Based Learning, Feature Ranking.

1 Introduction

The nearest neighbor (NN) [Cover and Hart, 1967] is a learning process that simply stores the training examples as the representation of the concept. Each time a new instance needs to be classified, its similarity to the stored instances is measured and the new instance inherits the class of its closest instance. As commented in [Mitchell, 1997], NN learning approaches are especially sensitive to the dimensionality curse, thus, the identification of the relevant features in a training set can improve results of a NN learning process.

Feature ranking algorithms are used to identify the relevance of features in a dataset and can be used with many different distance measures. In this work a feature ranking algorithm based on the "Chi-squared" distance measure is applied to rank the features of a dataset, and the ranked list of features is then used to define a feature weighting vector to be embedded in a k-NN classifier. The paper is organized along the following lines. Section 2 describes in a nutshell the class of feature ranking algorithms. In Section 3, an overview of the NN Classifier implemented in this work is given. Section 4 describes the proposed chi-squared (χ^2) feature weighting (χ^2FW) algorithm. A description of the knowledge domains, the experiments, as well as the obtained results are presented in Section 5. Finally, Section 6 highlights the conclusions and points out future works.

2 Feature Ranking

Feature Ranking can be defined as a category of feature selection methods. Feature selection has become the focus of research work in areas where datasets with tens or hundreds of thousands of variables are available [Guyon and Elissef, 2003]. While, in a theoretical sense, having more features should only give us more discriminating power, the real-world provides us with many reasons why this is not generally the case [Koller and Sahami, 1996]. Reunanen [Reunanen, 2003] observes that there can be many reasons for selecting only a subset of features: (i) it is cheaper to measure only a subset of features; (ii) prediction accuracy might be improved through exclusion of irrelevant features; (iii) the predictor to be built is usually simpler and potentially faster when less input features are used; (iv) knowing which features are relevant can give insight into the nature of the prediction problem at hand. Therefore, the problem of focusing on the most relevant information has become increasingly important for machine learning and data mining procedures [Blum and Langley, 1997].

When considering the method's output, feature selection methods can be grouped in two categories: "feature ranking" and "minimum subset selection" algorithms [Liu and Motoda, 1998]. A feature ranking algorithm defines a score to express the relevance of a feature; a subset selection algorithm tries to identify a subset of relevant features. In this work we are more interested in feature ranking algorithms. These methods require the evaluation of each feature using a specific distance metric, for identifying its degree of relevance (DR). The DR is then used to sort the features into a list called "ranked list of features". Figure 1 shows the algorithm in a nutshell.

The distance metric used in this work is chi-squared (χ2) statistical score [Liu and Motoda, 1998]. The motivation for using the χ^2 as a mutual information measure in a feature

1-4244-0703-6/07/$20.00 ©2007 IEEE

weighting task comes from the ability of this measure in ranking features [Witten and Frank, 2000].

Considering a discrete variable *i* which can assume *l* possible values; a discrete variable *j* which can assume *c* possible values; n_{ij} the observed frequency and e_{ij} the expected frequency. Then, the χ^2 test can be used to measure the mutual information between variables *i* and *j* following equation 1.

$$\chi^2 = \sum_{i=1}^{l} \sum_{j=1}^{c} \frac{(n_{ij} - e_{ij})^2}{e_{ij}} \qquad (1)$$

```
Feature Ranking Algorithm
Input: dataset, dist_measure;
Output: ranked list of features
List: ordered features list;
DR:degree of relevance function;
D_R: vector with the degree of relevance
of each feature

begin
  List ← emptylist;
  for each feature f do
    begin
      D_R[f] ←
      DR(f,dist_measure,dataset); /*degree
      of relevance */
      List ← (f,DR[f]);
    endfor;
  order(List,DR);
end.
```

Figure 1. Feature ranking algorithm.

3 Nearest Neighbor Classifier

Basically NN techniques assume the class of the nearest instance from x, as the class of an instance x. In order to determine the nearest instance, NN techniques adopt a distance metric that measures the proximity of instance x to all stored instances. Various distance metrics can be used, including the Euclidean. Figure 2 presents the formal definition of NN technique found in [Cover and Hart, 1967]. The rule described in Figure 2 is more properly called the 1NN rule since it uses only one nearest neighbor. One of the variants of the 1NN rule is the k-NN rule, which considers the k nearest instances $\{i_1, i_2,.., i_k\}$ and decides upon the most frequent class in the set $\{\theta_{i_1}, \theta_{i_2}, \ldots\theta_{i_k}\}$. Provided that the number of training instances is large enough, generally k-NN exhibits a good performance. As mentioned before, a disadvantage of k-NN methods is that all of the training instances must be retained.

It is worth mentioning that when the set of training instances is large, k-NN based methods invest a high computational effort to perform a classification. This happens because for each new query (q) the whole training set T_{NN} needs to be visited.

The idea of weighting features when using a k-NN algorithm is used to give more importance, in the classification process, to relevant features. Consider a dataset containing m features (f1,f2,…fm), where only one (fj) is relevant for classifying instances. Two instances having the same fj value, however, may be distant from each other in the m-dimensional space; this shows that non relevant features can play an important role in the classification process, dominating the distance measure. Trying to minimize this problem, many authors suggest [Wettschereck et al., 1997] the use of weights associated with features in order to guide the computation of the distance between them.

As the main goal of this work is to empirically verify the performance of a k-NN algorithm using a feature weighting method based on χ^2 statistical score, a k-NN algorithm was implemented for classifying instances described by discrete (or ordinal) features and a nominal class. All the implemented functions followed the description given in [Mitchell, 1997].

```
Assume:
n-dimensional feature space.
M classes, numbered 1,2,…,M.
p training instances, each one expressed
as a pair ($x_i$, $\theta_i$), for $1 \le i \le p$ where
a) $x_i$: training instance, expressed by a
vector of pairs attribute-
value $x_i = (x_{i_1}, x_{i_2}, \ldots, x_{i_{1n}})$
b) $\theta_i \in \{1,2,\ldots,M\}$ represents the correct
class of the instance $x_i$


Let $T_{NN}$ = {($x_1$,$\theta_1$), ($x_2$,$\theta_2$), …, ($x_p$,$\theta_p$)} be
the nearest neighbor training set.
Given an unknown instance x, the
decision rule is to decide x is in class
$\theta_j$ if
        $d(x,x_j) \le d(x,x_i)$, for $1 \le i \le p$
where d is some n-dimensional distance
metric.
```

Figure 2. 1-NN formal definition

4 The χ2 Feature Weighting (χ2FW)

The proposed chi-squared (χ^2) feature weighting (χ^2FW) method can be classified as a mutual information approach for assigning features weights [Wettschereck et al., 1997]. In this sense, the mutual information (the Chi-Squared statistical score) between the values of a feature and the class of the training instances are used to assign feature weights.

χ^2FW follows the idea presented in [Daelemans and Bosch, 1992] but instead of using the information gain [Quinlan, 1986] measure, it uses the χ^2 statistical score. As mentioned in Section 2, the motivation for using χ^2 as a mutual information measure in a feature weighting task is due to the ability of this measure in ranking features [Witten and Frank, 2000].

The method can be defined in three steps. Initially the χ^2 score between each feature and the class must be defined using the whole dataset; subsequently, based on a weighting criterion, a vector containing the weights of each feature is created and finally the weights of each feature are used in the k-NN classification task. Figure 3 describes an algorithm for the first and the second steps of this process. The third step is the use of k-NN algorithm based on weights defined in the previous steps.

Many criteria for defining a weight vector may be used. In this work, two criteria were used and the obtained results were compared. The first criterion, called Sequential Weighting (SW), defines a weight vector simply by ranking features ie, the features having the lowest χ^2 score have their weights set to 1, those with the second lowest-scored features have their weight set to 2 and so on. The process goes on until weights are assigned to the highest χ^2 scored features. In datasets where all m features have different χ^2 scores, the highest scored feature will have its weight set to m. The second criterion, called Normalized Weighting (NW), normalizes weights in the interval [0..10]. According to this criterion, features with the highest χ^2 score have their weights set to 10. The other features have their weights linearly established according to their χ^2 score. Therefore, both criteria follow the idea given in [Wettschereck et al., 1997] that a feature weighting algorithm should assign low weights to features that provide little information for classification and higher weights to features that provide more reliable information.

```
$\chi^2$FW algorithm
**Input:** dataset described by features
plus class, criterion;
**Output:** Weight Vector (V)
Chi: vector with the $\chi^2$ score for each
feature;
V: weight vector;
begin
  for each feature f do
     Chi[f] $\leftarrow$ $\chi^2$(class,f,dataset);
  V $\leftarrow$ create_vector(criterion,Chi);
end.
```

Figure 3. χ2 Feature Weighting (χ2FW) algorithm.

In addition to the feature weighting process, our experiments performed k-NN classification using a feature domain normalization step and a distance-weighted learning approach. Both of them as described in [Mitchell, 1997].

5 Experiments

Experiments were conducted using fifteen datasets containing discrete or ordinal features to attempt to identify the consistency of the proposed method. An overview of the datasets is given in Table 1.

Datasets Balance Scale, Congressional Voting Records, Letter, Optic Digits and Wisconsin Breast Cancer were downloaded from the UCI Machine Learning Repository [Merz and Murphy, 1998]. The others are their extended versions. Extended versions were generated inserting randomly distributed features in the corresponding original dataset. For instance, Balance +10 dataset is the original Balance Scale dataset enlarged with the addition of 10 new attributes (integer-valued) whose values were randomly chosen from interval [0..10]. All the extended datasets were obtained using the same process. The only exceptions are the Voting +10 and Voting +20 datasets which were generated with the insertion of binary-valued features, since all the original features are binary-valued.

Table 1. Datasets overview.

Dataset	# instances	**# features**
Balance Scale	625	**4**
Balance +10	625	**14**
Balance +20	625	**24**
Wisconsin Breast Cancer	683	**9**
Breast +10	683	**19**
Breast +20	683	**29**
Congressional Voting Records	232	**16**
Voting +10	232	**26**
Voting +20	232	**36**
Letter	20000	**16**
Letter +10	20000	**26**
Letter +20	20000	**36**
Optic Digits	5620	**64**
Optic +10	5620	**74**
Optic +20	5620	**84**

The average-case analysis of simple k-NN given in [Langley and Iba, 1993] as well as results of works using this learning method, such as those described in [Aha, 1990 ; Langley and Sage, 1997], indicate that the number of training examples needed to reach a given accuracy grows exponentially with the number of irrelevant attributes. The extended datasets were generated to simulate samples containing large amounts of irrelevant information. Thus, it is possible to verify whether χ2FW can contribute to reducing the effect of this undesirable characteristic of k-NN learning or not. All datasets have no missing feature values.

The aim of the experiments was to verify the soundness of the proposed feature weighting method articulated to a k-NN classification task. Thus, we consider that the expected behavior of a consistent feature weighting method is to generate higher Average Correct Classification Rates - ACCRs (obtained in classification using the weighting method) than the ACCRs obtained in classification without the weighting method.

As it is hard to define the best value for k in k-NN classification, for each dataset the experiments used k=1, k=5 and k=10. Trying to minimize the training data bias, the experiments were conducted in a stratified 10-fold cross-validation strategy. Tables 2, 3, 4, 5 and 6 show the ACCRs and the corresponding standard deviations (SDs) obtained. The column named "Original k-NN" shows results obtained using original k-NN algorithm as discussed in section 3. Columns named "Sequential Weighting" (SW) and "Normalized Weighting" (NW) present the ACCRs when using the SW and NW criteria respectively (as described in Section 4).

As mentioned in the Balance Scale dataset documentation [Merz and Murphy, 1998] the four features are equally relevant for describing the concept this dataset represents. Any consistent weighting method consequently should assign the same weight to the four features. First line of Table 2 shows that this is the case for the proposed weighting method, since the obtained values are exactly the same.

Analysis of results in the second and third rows of Table 2 indicate that the weighting method improved the ACCR values; particularly the NW criterion which resulted in better ACCR values for all values of k. In addition, it is interesting to note that original k-NN produced the worst results in the extended datasets. Therefore, the insertion of randomly distributed features contributed to reducing classification accuracy (mainly when using original k-NN). This can be explained by k-NN algorithm sensitivity to irrelevant features [Mitchell, 1997] and has been observed in all experiments conducted in this work.

Table 2. Balance Scale Datasets (ACCR±SD)

Dataset	K	**Original k-NN**	**Sequential Weighting**	**Normalized Weighting**
		ACCR ± SD	ACCR ± SD	ACCR ± SD
Original Balance Scale	**1**	80.47 ± 3.23	80.47 ± 3.23	80.47 ± 3.23
	5	82.39 ± 3.17	82.39 ± 3.17	82.39 ± 3.17
	10	88.00 ± 2.73	88.00 ± 2.73	88.00 ± 2.73
Balance Scale +10	**1**	59.39 ± 7.96	64.97 ± 5.73	**65.92** ± 4.83
	5	69.94 ± 4.97	72.00 ± 2.83	**75.84** ± 4.05
	10	71.37 ± 4.01	75.85 ± 3.18	**83.53** ± 3.42
Balance Scale +20	**1**	52.46 ± 8.40	58.53 ± 6.94	**64.15** ± 5.31
	5	60.16 ± 5.48	69.12 ± 4.46	**76.17** ± 3.68
	10	65.58 ± 5.98	73.92 ± 5.26	**81.45** ± 3.18

Table 3 presents the classification results obtained when using the Wisconsin Breast Cancer dataset and its extensions. In the original dataset, the highest ACCR was achieved using SW criterion (96.63%). Results obtained using the three criteria, however, are very similar and makes it hard to state which method is the best for this dataset. One reason for this behavior may be the fact that most of the original Wisconsin Breast Cancer dataset features are relevant (as showed in [Hruschka Jr. et al., 2004]). On the other hand, the experiments using the extended datasets ie, Breast +10 and Breast +20, show that NW criterion delivered slightly better results than the others. Comparing ACCR values between original k-NN and NW, NW tends to perform better when random features are inserted in this dataset.

Table 3. Wisconsin Breast Cancer Datasets (ACCR±SD)

Dataset	K	**Original k-NN**	**Sequential Weighting**	**Normalized Weighting**
		ACCR ± SD	ACCR ± SD	ACCR ± SD
Original Wisconsin Breast Cancer	**1**	**96.33** ± 3.12	95.02 ± 2.52	95.46 ± 2.64
	5	96.04 ± 2.78	**96.63** ± 2.08	96.18 ± 2.52
	10	95.89 ± 2.76	**96.04** ± 2.78	**96.04** ± 2.95
Breast +10	**1**	91.66 ± 3.58	95.02 ± 2.20	**95.61** ± 2.05
	5	92.54 ± 3.19	95.61 ± 2.66	**96.05** ± 1.39
	10	92.39 ± 3.20	95.61 ± 2.83	**96.19** ± 1.54
Breast +20	**1**	88.72 ± 3.56	94.43 ± 1.94	**95.31** ± 2.35
	5	90.77 ± 2.80	95.60 ± 2.50	**96.04** ± 1.55
	10	91.35 ± 3.00	95.31 ± 2.38	**96.33** ± 1.39

Tables 2 and 3 reveal that the weighting method did not optimize the classification results when using the original Balance Scale and the original Wisconsin Breast Cancer datasets. Table 4, however, shows that for the Congressional Voting Records domain, the weighting method improved results even when working with the original dataset. This suggests that the weighting method found irrelevant features in the original dataset as well as in the extended ones. Table 4 also shows that NW criterion produced the best results in all experiments using this domain.

Table 5 displays the ACCR values achieved using the Optic Digits domain. As with the previous domains, the insertion of irrelevant features has not significantly changed ACCR values when using either weighting criteria (WS and NW). It can be seen, however, that the inserted random features contributed to diminishing ACCR values when using original k-NN.

Table 4. Congressional Voting Records Datasets (ACCR±SD)

Dataset	K	Original k-NN	Sequential Weighting	Normalized Weighting
		ACCR ± SD	ACCR ± SD	ACCR ± SD
Original Congress Voting Records	1	92.72 ± 3.96	95.67 ± 4.10	**96.10** ± 3.81
	5	93.10 ± 4.20	96.52 ± 4.49	**96.96** ± 3.58
	10	93.51 ± 4.25	95.22 ± 4.32	**96.96** ± 3.58
Voting +10	1	90.54 ± 4.40	92.25 ± 3.41	**93.93** ± 6.22
	5	93.08 ± 4.70	93.93 ± 4.68	**95.23** ± 4.33
	10	91.36 ± 5.44	93.08 ± 5.09	**94.36** ± 4.13
Voting +20	1	90.49 ± 5.69	93.95 ± 4.17	**94.38** ± 4.58
	5	91.77 ± 5.96	92.64 ± 6.49	**96.12** ± 4.28
	10	90.91 ± 6.63	92.21 ± 6.41	**95.25** ± 4.75

Table 6. Letter Datasets (ACCR±SD)

Dataset	K	Original k-NN	Sequential Weighting	Normalized Weighting
		ACCR ± SD	ACCR ± SD	ACCR ± SD
Original Letter	1	88.46 ± 0.72	90.46 ± 0.85	**90.90** ± 0.85
	5	90.06 ± 0.82	90.92 ± 0.93	**90.97** ± 0.93
	10	90.38 ± 0.77	90.31 ± 0.76	**90.53** ± 0.76
Letter +10	1	80.85 ± 0.74	90.12 ± 0.56	**90.93** ± 0.65
	5	84.75 ± 0.68	90.98 ± 0.63	**91.24** ± 0.65
	10	85.75 ± 0.55	**90.74** ± 0.42	90.52 ± 0.44
Letter +20	1	73.56 ± 0.98	88.26 ± 0.68	**91.01** ± 0.64
	5	79.83 ± 0.65	89.58 ± 0.74	**91.26** ± 0.73
	10	81.39 ± 0.39	89.57 ± 0.67	**90.59** ± 0.66

Table 5. Optic Digits Datasets (ACCR±SD)

Dataset	K	Original k-NN	Sequential Weighting	Normalized Weighting
		ACCR ± SD	ACCR ± SD	ACCR ± SD
Original Optic Digits	1	88.81 ± 0.64	**89.91** ± 0.92	89.89 ± 0.75
	5	90.84 ± 0.81	91.25 ± 0.61	**91.42** ± 0.77
	10	90.80 ± 0.99	**91.50** ± 1.03	91.49 ± 0.84
Optic +10	1	86.19 ± 0.90	89.86 ± 1.34	**89.96** ± 1.22
	5	89.54 ± 1.20	91.44 ± 0.92	**91.48** ± 1.06
	10	89.73 ± 1.26	91.05 ± 1.02	**91.30** ± 1.17
Optic +20	1	84.80 ± 0.92	89.57 ± 1.10	**89.79** ± 1.12
	5	88.75 ± 0.73	91.41 ± 0.83	**91.58** ± 0.80
	10	89.02 ± 1.14	90.84 ± 1.17	**91.27** ± 0.98

It is worth mentioning that the insertion of 10 and 20 random features in the Optic Digits domain tends to generate less impact in the classification results because of the large number of features used for describing this dataset. Inserting 10 new features in a dataset already containing 64 features (Optical Digits) should produce less impact than inserting 10 new features in a dataset containing only 4 features (Balance Scale).

The last domain used in this work is the Letter dataset [Merz and Murphy, 1998] and the results of the experiments are in Table 6 and are consistent with what has been said previously. The NW criterion delivered the best ACCR values in eight out of nine experiments. Based on results it is possible to state that when using original k-NN the ACCR values become lower as the number of irrelevant features increases. On the other hand, when using χ2FW the ACCR values tend to stabilize even in the presence of many irrelevant features.

6 Conclusions

This paper proposes, describes and evaluates a new weighting method, based on the χ2 statistical test, for a k-NN classifier. Comparative analyses of results were carried out and the proposed method appears to be very promising.

The work shows accuracy values obtained using k-NN (for k=1, k=5 and k=10) on 15 datasets – 5 of them are from the UCI Machine Learning Repository [Merz and Murphy, 1998] and the remaining 10 are their extended versions. Results of 45 experiments using a stratified 10-fold cross-validation strategy are described and discussed. Considering both proposed weighting vector generation criteria (Sequential Weighting and Normalized Weighting), forty-four out of forty five experiments favoured the proposed χ2FW method.

It is interesting to mention that the artificial insertion of irrelevant features in the original datasets, induced greater differences among ACCR values (comparing weighted k-NN versus original k-NN). Therefore, the proposed weighting process tends to have good performance in datasets with a large number of irrelevant features. As commented in [Blum and Langley, 1997], this behavior is an important issue in a feature selection (or feature weighting) algorithm.

This paper also proposes two vector weighting generation criteria namely Sequential Weighting (SW) and Normalized Weighting (NW). Analysis of data in the previous tables indicates that NW criterion does achieve the best results (compared with either SW or original k-NN) in 39 out of 45 experiments, which can be considered a good performance. We intend next to analyze NW performance in datasets with greater value intervals (for example, [0..100] instead of [0..10] used in this experiments).

Another interesting line of research would be to verify if the dependence between features can influence the behaviour of our proposed method. Therefore, instead of only verifying the relation between each feature and the class, it would be interesting to analyze the relation between each pair of features. Presently we are conducting an empirical comparison with other weighting methods, still using k-NN as the classifier algorithm.

References

[1] [Cover and Hart, 1967] Cover, T. and Hart, P., Nearest neighbor pattern classification, IEEE Transactions on Information Theory 13, 1967, pp 21–27.

[2] [Mitchell, 1997] Mitchell, T., Machine Learning, The McGraw-Hill Companies, Inc, 1997.

[3] [Guyon and Elissef, 2003] Guyon, I. and Elisseeff, A., An introduction to variable and feature selection, Journal of Machine Learning Research*,* 3, pp. 1157-1182, 2003.

[4] [Koller and Sahami, 1996] Koller, D. and Sahami, M., Toward optimal feature selection*,* Proceedings of the 13th International Conference on Machine Learning, pp. 284-292, July, 1996.

[5] [Reunanen, 2003] Reunanen, J., Overfitting in making comparisons between variable selection methods, Journal of Machine Learning Research*,* 3, pp. 1371-1382, 2003.

[6] [Blum and Langley, 1997] Blum, A. L. and Langley, P., Selection of relevant features and examples in machine learning, Artificial Intelligence, pp. 245-271, 1997.

[7] [Liu and Motoda, 1998] Liu, H. and Motoda, H., Feature Selection for Knowledge Discovery and Data Mining. Kluwer Academic, 1998.

[8] [Witten and Frank, 2005] Witten, I. H. and Frank, E., Data Mining – Practical Machine Learning Tools and Techniques with Java Implementations, second edition, Morgan Kaufmann Publishers, USA, 2005.

[9] [Wettschereck et al., 1997] Wettschereck, D., Aha, D. & Mohri, T., A review and empirical evaluation of feature weighting methods for a class of lazy learning algorithms, Artificial Intelligence Review, 11:273-314, 1997.

[10] [Daelemans and Bosch, 1992] Daelemans, W. and Bosch, A., Generalization performance of backpropagation learning on a syllabification task, Proc. of TWLT3: Connectionism and Natural Language Processing, pp. 27-37, 1992.

[11] [Quinlan, 1986] Quinlan, J. R., Induction of decision trees, Machine Learning, 1:81-106, 1986.

[12] [Merz and Murphy, 1998] Merz, C. J. and Murphy, P. M., UCI Repository of Machine Learning Databases, [http://www.ics.uci.edu]. Irvine, CA, University of California.

[13] [Langley and Iba, 1993] Langley, P. and Iba, W., Average-case analysis of a nearest neighbor algorithm. In: Proceedings IJCAI-93, Chambery, France, 889-894, 1993.

[14] [Aha, 1990] Aha, D., A study of instance-based algorithms for supervised learning tasks: mathematical, empirical and psychological evaluations. Doctoral Dissertation, Department of Information and Computer Science, University of California, Irvine, CA, 1990.

[15] [Langley and Sage, 1997] Langley, P. and Sage, S., Scaling to domains with many irrelevant features. Computational Learning Theory and Natural Learning Systems. Vol. 4, MIT Press, Cambridge, MA, 1997.

[16] [Hruschka Jr. et al., 2004] Hruschka Jr., E. R., Hruschka, E. R. and Ebecken, N. F. F., Feature Selection by Bayesian Networks, Lecture Notes in Artificial Intelligence. Berlin: Springer-Verlag, v.3060. p.370 – 379, 2004.