Supervised selection of dynamic features, with an application to telecommunication data preparation

Sylvain Ferrandiz[1,2] and Marc Boullé[1]

[1] France Télécom R&D
2, avenue Pierre Marzin,
22307 LANNION Cedex, France
[2] Université de Caen, GREYC,
Campus Côte de Nacre, boulevard du Maréchal Juin
BP 5186
14032 Caen Cedex, France
sylvain.ferrandiz@francetelecom.com
marc.boulle@francetelecom.com

Abstract. In the field of data mining, data preparation has more and more in common with a bottleneck. Indeed, collecting and storing data becomes cheaper while modelling costs remain unchanged. As a result, feature selection is now usually performed. In the data preparation step, selection often relies on feature ranking. In the supervised classification context, ranking is based on the information that the explanatory feature brings on the target categorical attribute.
With the increasing presence in the database of feature measured over time, *i.e.* dynamic features, new supervised ranking methods have to be designed. In this paper, we propose a new method to evaluate dynamic features, which is derived from a probabilistic criterion. The criterion is non-parametric and handles automatically the problem of overfitting the data. The resulting evaluation produces reliable results. Furthermore, the design of the criterion relies on an understandable and simple approach. This allows to provide meaningful visualization of the evaluation, in addition to the computed score. The advantages of the new method are illustrated on a telecommunication dataset.

1 Data preparation and feature ranking

In a data mining project, the data preparation step is a cornerstone. It aims at providing a dataset for the modelling step, that is a row/column table, from primary collected data [3]. Typically, topics like instance representation, instance selection and/or aggregation, missing values handling, feature selection, are to be dealt with. We focus in this paper on feature selection, in the context of supervised classification.

In [7], a check list of the different problems to tackle when performing feature selection is provided. According to this list, we consider in the present paper that :

- we have domain knowledge : the whole search space may be of very large size and domain knowledge limits the evaluation to meaningful features,
- features are commensurate : no normalization has to be carried out preliminarily,
- we are to select subsets of the input variables : the context is that of large databases,
- we assess features individually : for the sake of simplicity and scalability,
- we do not focus on the prediction performance : the context is that of data preparation,
- we cannot make hypotheses on the interdependence or the "noisiness" of the features : this must be detected not hypothesized,
- we want a stable solution : extracted information must be general, not valid on the data at hand only.

Assessing the features individually, more than being simple and scalable, is well-suited to the data preparation step. Indeed, following the classification of [9], this is a filter method, being independent of the choice of a predictor. We can think of the problem as a variable ranking problem. We assume that a high score describes a valuable variable and that variables are sorted decreasingly. As an application, nested subsets progressively incorporating more and more variables of decreasing relevance can be defined in order to build predictors. Classical scoring paradigms are described in the section 2.

With the increasing collecting and storing capacity of many computerized systems, data-miners have to deal with more and more heterogeneous data and face new challenges. As an example, while features used to be static, they are becoming more and more dynamic, being measured over time. To each instance can be associated static continuous values (like the age of the patient), static categorical values (like the hospital in which the patient is being treated), and dynamic features (like an ECG, an EEG). We discuss the problematic of supervised selection of dynamic features in the section 3.

In the static case, the approach adopted in [2] for discretization of a continuous feature and in [1] for grouping the values of a categorical feature provides the user with an evaluation of the amount of information the feature contains relating to the target attribute. As dynamic features are innerly multivariate, any extended version of these static methods to the multivariate case allows to evaluate dynamic features. In [5], the approach is adapted to the case of multivariate features. It relies on partitioning the set of instances and designing a criterion for the evaluation of different partitions. We derive from it a method to evaluate dynamic features in the section 4. For the convenience of the reader, the technicalities around the criterion are postponed to the section 5.

In the section 6, we illustrate the advantages of the new criterion for supervised selection of dynamic features on a telecommunication dataset. As this dataset contains continuous data only, we restrict ourselves to this kind of data in the following. But, as it will become clearer, the proposed indicator can deal with any kind of data.

2 Classical evaluation paradigms

Let us consider a continuous explanatory random variable X and a target attribute Y. We describe in this section the classical approaches for designing a measure of interest of X relating to Y.

In a two-class classification problem, the values of Y can be mapped to the values ± 1. Then, the squared correlation coefficient between X and Y can be used to score the variable X. It can be shown that this correlation coefficient is closely related to the ratio of the between-class variance to the within-class variance, that is to Fisher's criterion, and to the Student's T-test. Variable ranking can thus be turned into classical statistical testing. This approach is limited to two-class problems, is parametric (in the algorithmic and statistical sense) and relies on an asymptotic approximation.

In the considered supervised context, many scores are based on the individual predictive performance of X. Once X is turned into a classifier, the error rate evaluated on a separated validation sample measures such a performance. For example, in a two-class problem, a classifier is obtained by setting a threshold on the values of X. Varying the threshold allows to perform a ROC analysis, measuring the performance of X with the area under the ROC curve [4]. Cutting a continuous attribute into more than two intervals is a discretization problem. By considering more complex cuttings, one has to prevent from overfitting the data [10]. In case of large number of variables, ranking criteria based on predictive performance cannot separate the top ranking variables.

The maximum margin principle can be applied as well. The margin of an instance is the absolute difference between its distance to the nearest example of the same class and its distance to the nearest example of another class. Considering the sum of the margins on X provides an evaluation of X. This evaluation extends straightforwardly to the multivariate case. The resulting feature subset selection problem is tackled in [6]. The use of margins comes with distribution-free generalization bound. These bounds can be very loose in practice.

Another well-known approach is the information theoretic one. It relies on the maximization of the mutual information, which measures how far the joint distribution of X and Y is from independency. The main difficulty is to empirically estimate the mutual information, as it considers the joint distribution and the marginal distributions simultaneously. Some might say that the problem is easier in the categorical case, as the integral becomes a sum. Discretization is then applied, with an information preserving goal, this time. Once again, one has to prevent from overfitting the data [10].

3 When features are dynamic

In the classical case, a feature is static : the marital status, the gender, the salary, etc. As collecting and storing data becomes cheaper, it is more and more usual to monitor features over time. While gender cannot fluctuate over time, the salary and marital status do. The salary curve then has to be considered as a fully qualified feature. This is what we will refer to as a dynamic feature.

Thus, beside static features, the data-miner now encounters dynamic features. While the overall problematic of the data mining tasks remains unchanged (building a classifier, in the supervised classification context, for example), the introduction of such features raises new questions that require a particular treatment. Especially, the question of the representation is strengthened.

While the representation problem for continuous static features is usually turned into a discretization problem, and is quite less considered for categorical static features, the range of possibility is dramatically enlarged for dynamic features :

- the time scale is fixed according to technical constraints and can be unrelevant for the data-mining task,
- data can be noisy,
- relevant information can be hidden,
- . . .

Segmentation, denoising, Fourier's transform and many other algorithms are applied to the primary data and produce different representations. It means that, when ranking features, different representations of the same dynamic feature have to be evaluated too. Indeed, for a particular supervised study, the Haar's transform might be better than the Fourier's one. In the supervised context, such transformations are applied according to the domain knowledge, disregarding the target attribute.

As an example, let us consider head related transfer functions (HRTFs), which describe the acoustic filtering properties of a listener's external auditory periphery and are used in 3-D audio systems. An HRTF results from the application of the Fourier's transform to a particular head-related impulse response. The domain knowledge leads to work with the log of the Fourier's coefficient (as our auditive scale is closer to a logarithmic scale than a linear one), and to adopt a threshold (which corresponds to a threshold of hearing). The distance between HRTFs is usually measured by the euclidean distance. Studies show that weighting schemes produce similarity measure closer to the properties of our auditive system.

A dynamic feature is represented by a new set of explanatory attributes : the Fourier's coefficients, the segment means, etc. Then, a similarity measure between instances is usually defined. This measure itself is part of the representation. Supervised ranking of dynamic features amounts to the evaluation of a set of attributes equipped with a similarity measure.

4 Evaluation of dynamic features

Let us consider that a representation of a dynamic feature is given by a set of descriptive attributes and a similarity measure. In the static case, it is proposed in [2] to consider the problem of the discretization of a continuous attribute as a modelling problem. A model is a partition of the attribute into a set of intervals. The most probable given the data is selected. The originality of the method relies on the definition of the probability of a model given the data, compromising between the complexity of the model (*i.e.* the number of intervals) and goodness of fit to the data (*i.e.* the purity of the target attribute in each interval). In [5], the approach is extended in order to deal with the multivariate case, provided that a similarity measure is defined.

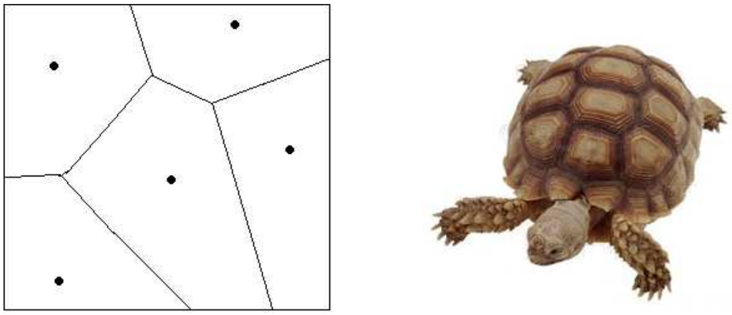

Fig. 1. Examples of synthetic and real Voronoi partitions for the euclidean metric.

While intervals are considered in the static case, partitions are made up with Voronoi cells in the multivariate case. Given a set P of instances, the *Voronoi cell* induced by $p \in P$ contains the points x whose p is the most similar element in P, with respect to a fixed dissimilarity measure. The element p is called the *representative* or the *prototype* of its cell. Examples of Voronoi partitions are given in the Figure 1. The problem of selecting the most probable partition given the data is then turned into an instance selection problem.

In order to perform instance selection, a representation must be fixed, that is a set of attributes and a similarity measure. In this context, a model is no other than a set of prototypes, *i.e.* a subset of the set of instances. For a given representation R and any set of prototypes M, let us evaluate the quality of M with the supervised criterion $c(R, M)$, the definition of which is postponed to the next section. Instance selection is performed by minimizing this criterion and we denote $c^*(R)$ the minimum value of $c(R, M)$:

$$c^*(R) = min_M c(R, M).$$

The resulting evaluation function c^* can be used and helpful for ranking representations : for a given representation, apply a combinatorial optimization algorithm for instance selection and evaluate the representation according to the minimum encountered criterion value. The technicalities of the criterion and the optimization algorithm are discussed in the next section.

The proposed primary criterion $c(M, R)$ sets a compromise between the size of M and the discrimination of the target attributes. Every instance in the database has a nearest prototype, according to the representation R. Each prototype thus supports the distribution of the labels of the instances lying in its Voronoi cell. Increasing the size of M produces distributions that are purer and purer but supported by less and less instances. The criterion $c(M, R)$ quantifies the compromise between the size of M and the reliability of the distributions, in a principled manner.

The criterion $c(M, R)$ is the negative log of the probability of M given the representation R and the data. As the adopted approach provides a regularized criterion, the search for the best set of prototypes is not prone to overfitting. In order to provide a normalized indicator, we consider the following transformation of c^* :

$$g^*(R) = 1 - \frac{c^*(R)}{c_0(R)},$$

where $c_0(R)$ is the criterion value for the empty set of prototypes. This can be interpreted as a compression gain, as negative log of probabilities are no other than coding lengths [11]. The compression gain $g^*(R)$ is greater than 0 (as soon as the empty set is evaluated during the optimization) and less than 1. If $g^*(R) = 0$, the representation R brings no information on the target attribute. The nearer $g^*(R)$ is from 1, the more separable the labels are.

The use of a validation set is very constraining. It limits the size of the training set and introduces useless variance in the result. Cross-validation is often used in order to reduce the variance effects, but is time consuming. Unlike performance based criteria, the compression gain is validation free.

The use of margins is validated by the fact that they provide distribution-free bounds on the generalization performance. In practice, these bounds are often very loose and margins are not innerly meaningful. The compression gain makes sense by quantifying a simple compromise between complexity of the hypotheses and discrimination of the target attribute.

Informational criteria and statistical tests often rely on statistical parametric assumptions (the probability laws are supposed to have a predetermined parametric shape) and possess an asymptotic validity. For a particular finite dataset, the quality of the estimations is not guaranteed or can be very loose as well. Unlike informational criteria or statistical tests, the compression gain is a finite-data criterion.

5 Instance selection : criterion and algorithm

In this section, we describe the criterion and the algorithm from which is derived the new method to evaluate dynamic features.

5.1 Evaluation of sets of prototypes

We first set the notations. We have a finite sample $D = \{X_n, Y_n\}$ of N labelled instances. We denote $D^{(x)} = \{X_n\}$ the set of instances and $D^{(y)} = \{Y_n\}$ the set of labels. The labels lie in an alphabet of size J and the instances in a space $\mathbb{X}$. A dissimilarity measure $\delta : \mathbb{X} \times \mathbb{X} \to \mathbb{R}_+$ is given.

For a set of prototypes $M = \{p_1, \ldots, p_K\} \subset \mathbb{X}$, K is the size of M, N_k $(1 \leq k \leq K)$ is the number of instances whose nearest prototype is p_k and N_{kj} denotes the number of such instances in the j^{th} class. Thus, $N = N_1 + \cdots + N_K$ and $N_k = N_{k1} + \cdots + N_{kJ}$.

The approach adopted in [5] leads to the following supervised evaluation of M :

$$c(M) = \log N + \log \binom{N+K-1}{K} + \sum_{k=1}^{K} \log \binom{N_k + J - 1}{J-1} + \log \frac{N_k!}{N_{k1}! \ldots N_{kJ}!}.$$

This value can be interpreted as the negative log of $p(D^{(y)}, M/D^{(x)})$. The first term of the criterion stands for the choice of the number K of prototypes, the second term for the choice of the K prototypes and the third term for the choice of the output label distributions in each cell. The last sum over the cells, according to the Stirling's approximation $\log x! \approx x \log x - x + O(\log x)$, behaves asymptotically as N times the conditional entropy of the distribution of the Y_n's given the clusters assignment function :

$$\frac{1}{N} \sum_{k=1}^{K} \log \frac{N_k!}{N_{k1}! \ldots N_{kJ}!} \approx -\sum_{k=1}^{K} \sum_{j=1}^{J} \frac{N_{kj}}{N} \log \frac{N_{kj}}{N_k}.$$

The criterion thus evaluates the discrimination of the distributions with a finite-data entropy-related term balanced with a structural weight, which quantifies the complexity of the partitioning. This prevents from overfitting the data.

5.2 The optimization heuristic

In this section, an optimization algorithm is described. It consists in a greedy optimization of a set of prototypes, the complexity of which can be reduced by exploiting the properties of the descriptive criterion. This greedy search is embedded into a meta-heuristic in order to further optimize the criterion.

The greedy heuristic Greedy(M) applies to every set M of p prototypes. Every subset resulting from the removal of an element in M is evaluated. Among those subsets, the winner is the one minimizing the criterion. This process is iterated and applied to every successive winners, until a singleton has been evaluated. The best encountered subset is returned. This method considers $O(p^2)$ subsets and each evaluation requires a search of the nearest prototype for each instance. A straightforward implantation of Greedy(M) has a complexity down to $O(Np^3)$. The properties of the models and the criterion allows to reduce this complexity to $O(Np \log p)$.

The greedy heuristic thus performs many evaluations quickly, as long as the number p of prototypes is not too large. It is then natural to think about applying this algorithm repeatedly. This is done according to the Variable Neighborhood

Search (VNS) meta-heuristic [8], which consists in applying the primary heuristic (*i.e.* the greedy one) to a neighbor of the solution. If the new solution is not better, a bigger neighborhood is considered. Otherwise, the algorithm restarts with the new best solution and a minimal size neighborhood. The process is controlled by specifying the maximum length of the series of growing neighborhoods to explore.

6 Application

We illustrate the advantages of the new evaluation method for dynamic features on a real dataset. The problem is that of dynamic feature selection for data preparation, in the context of supervised classification. The target attribute is a four-class attribute and the distribution of the labels is uniform : 25% of the instances are in class A, 25% in class B, 25% in class C and 25% in class D. We aim at scoring 24 dynamic features. The 24 features are themselves well-ordered and indexed from 1 to 24. Each feature is represented by 7 continuous attributes. Experiments are carried out with the L_1 and L_2 metrics alternatively.

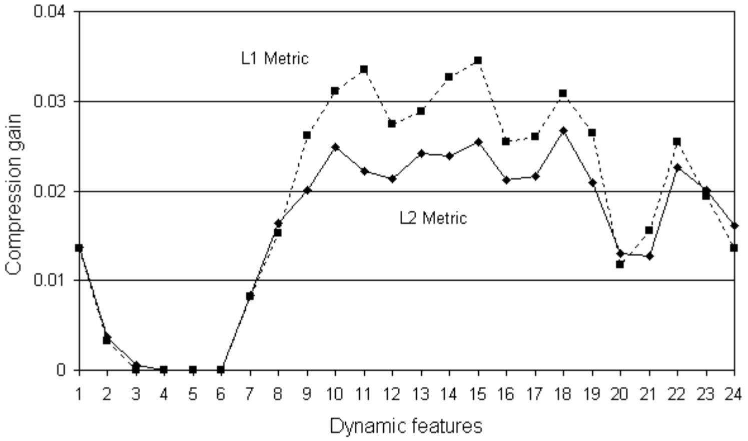

Fig. 2. Evaluation of the 24 dynamic features.

The scoring curve of the dynamic features is plotted in the figure 2. In the case of the L_1 metric, the compression gain of the features from 3 to 6 is null. This means that these dynamic features are uncorrelated with the target attribute. Automatically and with few risks of taking a wrong decision, the data-miner can eliminate those features.

Considering the L_1 and the L_2 metric alternatively, compression gain is higher when using the first one. Under such a hypothesis, the data-miner can perform metric selection according to the compression gain. Here, the L_1 metric should be prefered.

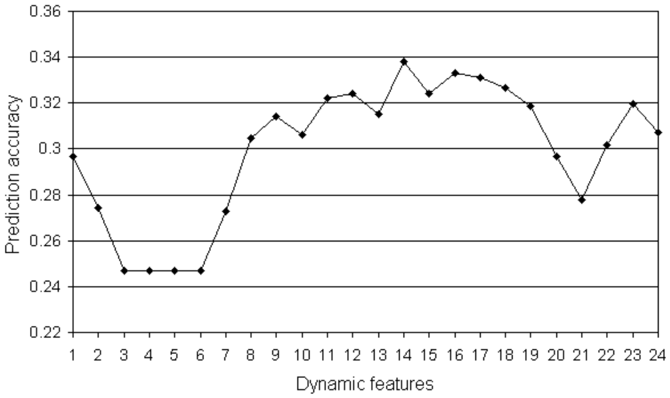

Fig. 3. Prediction accuracy of the 24 dynamic features.

The partition comes with a set of prototypes. The predictive accuracy of the nearest neighbor rule on this set of prototypes is reported on the figure 3 for every dynamic feature. The accuracy is measured by the prediction rate estimated on a separate test set. As can be noticed, the prediction accuracy and the compression gain exhibit the same global behavior.

Furthermore, the number of final prototypes is very low. This allows to provide the user with a visualization of the discrimination. The relating distribution of the target attribute can be plotted. Working with continuous attributes, for a prototype p, the instances whose nearest prototype is p can be averaged. The resulting mean profile assigned to each prototype can be plotted as well. This is done for the 15^{th} feature, which is the most relevant one, in the figure 4. The four classes are very mixed, and none of them clearly dominate the others. This is often the case in practice. In such situations, notions like prediction accuracy and margins are likely to be less meaningful.

Owing to the provided visualization, the approach adopted in this paper allows to make more than dynamic feature ranking. Useful and reliable general knowledge can be extracted. While the B class is not discriminated (for every prototypes, the proportion of instances labelled B is about 25%), the proportion of the D class ranges from 52% to 16%. Coupling this information on the distributions with the mean profiles allows the data-miner to draw the conclusion that instances with profile 4 on the 15^{th} dynamic feature are more likely to be in the class D than others. Furthermore, the peaks on the mean profile can be turned into knowledge as well.

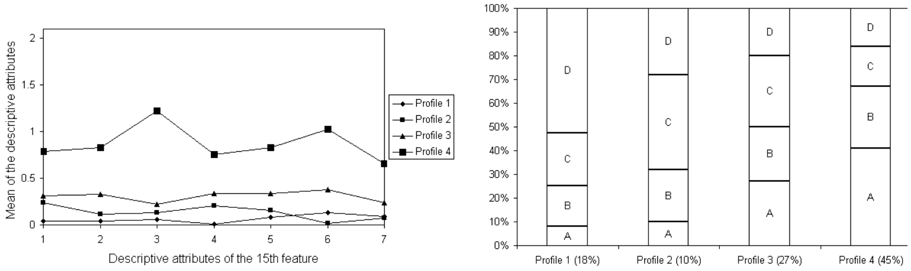

Fig. 4. Visualization of the mean profiles and the related distributions, for the 15^{th} feature. The method select four prototypes. To each prototype p is associated a mean profile : the average of the instances the nearest prototype of which is p. The four extracted profiles are plotted on the left. The labels of the instances can be collected and attributed to the nearest prototype and the four resulting frequential distributions are plotted on the right. The support of each prototype is reported too (Profile 1 (18%) means that the first prototype is the nearest prototype of 18% of the instances).

As the approach is non-parametric and handles automatically the problem of overfitting the data, validation sets are useless. The learning task usually benefits from considering more instances and then produces more accurate classifiers. In the present situation, more instances means more robust estimations of the label distributions and, possibly, a finer detection of behaviors. This is illustrated on the figure 5. Evaluation of the 15^{th} feature is performed with the data from the training AND the validation sets. Using more data allows to distinguish new profiles (6 instead of 4) and the data-miner is able to extract more knowledge, which is still reliable.

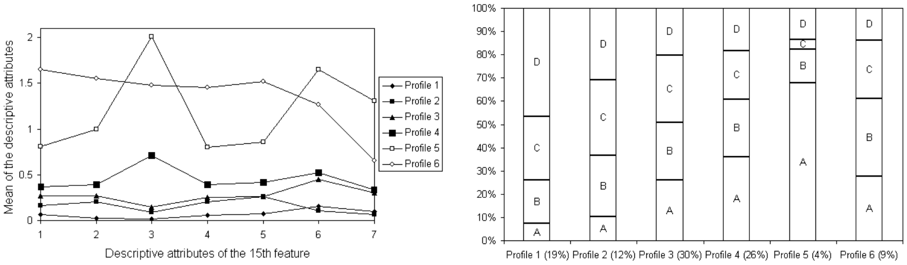

Fig. 5. Visualization of the mean profiles and the related distributions, for the 15^{th} feature, with more data.

7 Conclusion and further work

In this paper, we have discussed the ranking paradigm for feature selection and the particular problem raised by the presence of dynamic features in the database, especially in the context of data preparation. We proposed a new scoring method for such features, based on a criterion applying to instance selection for the nearest neighbor rule. The advantages of the new method fonction are illustrated on a real telecommunication dataset.

Being non-parametric, the method does not require a validation set. The method is able to take advantage of the presence of more instances. As the criterion handles automatically the problem of overfitting the data, the results are reliable. Furthermore, the underlying instance selection, which is performed when evaluating a feature, allows to produce a visualization in addition to the computed score. Finally, although the criterion does not focus on the prediction accuracy, the adopted target discrimination principle exhibits a strong correlation to the accuracy.

References

[1] Boullé, M.: A grouping method for categorical attributes having very large number of values. In: P. Perner and A. Imiya (Eds.), Machine Learning and Data Mining in Pattern Recognition, Springer Verlag, lnai 3587, MLDM 2005, (2005), 228–242

[2] Boullé, M.: A bayesian approach for supervised discretization. Data Mining V, Zanasi and Ebecken and Brebbia, WIT Press (2004) 199–208

[3] Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., Wirth, R.: CRISP-DM 1.0 : step-by-step data mining guide. Applied Statistics Algorithms (2000)

[4] Fawcett T.: ROC Graphs : notes and practical considerations for reseachers. Technical report HPL-2003-4 (2003).

[5] Ferrandiz S., Boullé M.: Supervised evaluation of Voronoi partitions. Journal of intelligent data analysis (2006), to be published

[6] Gilad-Bachrach R., Navot A., Tishby N.: Margin based feature selection - theory and algorithms. Proceedings of the 21'st international conference on machine learning (2004)

[7] Guyon I., Elisseeff A.: An introduction to variable and feature selection. Journal of machine learning research **3** (2003) 1157–1182

[8] Hansen P., Mladenovic N.: Variable neighborhood search: principles and applications. European journal of operational research **130** (2001) 449–467

[9] Kohavi R., John G.H.: Wrappers for Feature Subset Selection. Artificial Intelligence **97** (1997) 273–324

[10] Kohavi R., Sahami M.: Error-based and entropy-based Discretization of continuous features. Proceedings of the 2'nd international conference on knowledge discovery and data mining (1996) 114–119

[11] Shannon C.E.: A mathematical theory of communication. Bell systems technical journal **27** (1948) 379–423 and 623–656