# Supervised evaluation of Voronoi partitions

Sylvain Ferrandiz [(1)(2)] * Marc Boullé [(1)]

May 17, 2006

[(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

**Abstract**

Since its introduction, the nearest neighbor rule has been widely refined and there exists many techniques for prototypes selection or construction. The underlying structure of such rules is the Voronoi partition induced by the prototypes. Construction of the best Voronoi partition often relies on the generalisation performance and thus faces the risk of overfitting the data.

In this paper, we adopt a descriptive approach for the supervised evaluation of medoid-based Voronoi partitions. The resulting criterion measures the discrimination of the classes, is parameter free and prevents from overfitting. Experiments on real and synthetic datasets illustrate these properties. Although this criterion is not related to the classifying task, the accuracy and robustness of the induced classifier are also compared with standard methods, such as the nearest neighbor rule and the linear vector quantization method.

**Keywords** : Supervised classification – nearest neighbor rule – Voronoi tesselations – partitioning – data-dependent evaluation

## 1 Classification and Voronoi partitions

Data mining aims at extracting information from data sources. With the increasing number of collected data, preprocessing techniques which summarize and clean the databases before the modelling step become more and more appealing. As soon as a dissimilarity notion is available, Voronoi partitions [11] become a key tool for many of these techniques. Examples of Voronoi partitions are given in Figure 1.

Such partitions have many useful characteristics. The partitioning paradigm is intrinsically well-suited for discrimination of the behaviors. Voronoi partitions

*2, avenue Pierre Marzin, 22307 Lannion Cedex, France. Tel.: +33 296 051 430; E-mail: sylvain.ferrandiz@francetelecom.com.

assign to each detected behavior a typical example. By grouping similar objects, reliable information can be extracted by considering the objects in each cell jointly : effects of the outliers decrease. Finally, the definition of a Voronoi partition merely relies on a set of objects, which makes the set of partitions easy to handle. For these attractive properties, Voronoi partitions have been widely used for clustering databases.

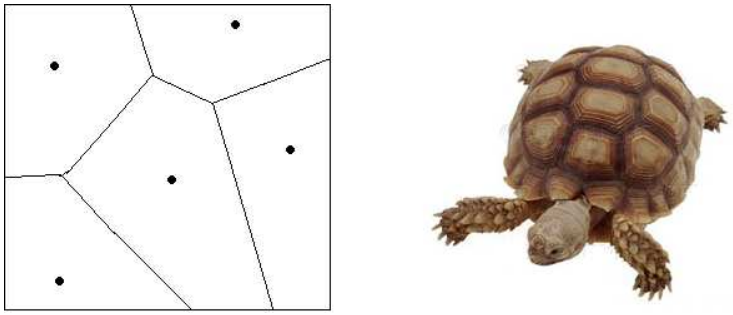

Figure 1: Examples of synthetic and real Voronoi partitions for the euclidean metric.

In the unsupervised context, the *dissimilarity measure* of a partition just sums the dissimilarities between each instance and its nearest representative. The best partition is the one with minimum dissimilarity. If the euclidean metric is applied, the K-means algorithm [9] is a stochastic method which optimizes this criterion. Firstly, a set of $K$ prototypes is picked out from the set of instances at random. Then, in each cell, the mean of the instances is computed and the set of the means replaces the initial set of prototypes. A more accurate partition results from this and the process is iterated up to stability. The method is known to converge quickly. However, it must theoretically deal with metric spaces and is efficient for euclidean spaces only.

K-medoids methods [6] overcome this constraint by considering sets of prototypes included in the set of instances. The only required input is the matrix of dissimilarities between any two instances. For example, instances can be described by categorical values and dissimilarities can be measured by the Hamming metric. Such methods are naturally combinatorial which makes them less computationally tractable.

In [5], two optimisation heuristics are proposed. PAM (for Partitioning Around Medoids) is a two phases algorithm, which incrementally builds a set of prototypes and then performs swaps between prototypes and instances. CLARA (for Clustering LARge Applications) relies on sampling and applies PAM on successive samples in order to save memory space and computing time. CLARANS (for Clustering Large Applications based on RANdomized Search) is a hill-climbing search with multiple randomized starts [10].

Working with a dissimilarity based criterion in an unsupervised context

forces the user to specify the number $K$ of prototypes. Indeed, the whole set of instances is the one minimizing the dissimilarity (which is null in this case) and the dissimilarity always decreases with the increase of $K$. Then, there is no hope for automatically selecting $K$ by optimizing such a criterion. In some cases, like assigning customers to salesmen, the number of prototypes is obviously given by the problem setting. But one often has to apply a heuristic.

In the context of density estimation, Normalised Maximum Likelihood (or Stochastic Complexity) principle [12] is a theoretical framework which allows to compare mixture models with different number of components. The idea consists in penalizing the likelihood of a model for the given sample by the sum of the likelihoods over every samples. For multinomial data, the criterion is made computationally tractable in [8].

In the supervised context, a label lying in a finite alphabet is assigned to each instance and the data thus pertain to predetermined classes. A natural way for partitioning such data consists in applying unsupervised methods to each class and putting together the resulting prototypes [4]. In case of well-separated classes, this heuristic behaves satisfactorily but fails if the classes are mixed. Indeed, the positioning of the prototypes for each class does not take into account the instances in the other classes. Furthermore, the specification of the number of representatives in each class remains an open issue.

LVQ (for Linear Vector Quantization [7]) aims at improving the position of $K$ preselected prototypes. The initial prototypes may results from applying the K-means algorithm in each class separately or sampling from each class. The on-line version processes one instance at a time, adapting the position of the nearest prototype according to the class labels : if the instance and the prototype share the same label, the prototype moves toward the instance, else it moves away. LVQ uses all the instances for positioning the representatives and thus handles the case of mixed classes. But the choice of the number $K$ is still left to the user.

In this paper, we present a new supervised finite-data evaluation criterion for medoid-based Voronoi partitions of different sizes with frequential distributions. As the instances are labelled, we do not focus on the global dissimilarity of the instances but on the discrimination of the labels : a good partition must result from a compromise between the number of cells and the discrimination of the labels between cells. The adopted descriptive approach allows to make this compromise formally explicit.

The remainder of the paper is organized as follows. We first present the related works (section 2) and describe the new criterion (section 3). Then, we evaluate the descriptive approach on synthetic and real data (section 4) and compare with the predictive classical one, represented by the LVQ method (section 5).

## 2 Related works

Supervised medoid-based partitioning is an extension of the instance selection paradigm. Instance selection consists in selecting prototypes among the instances of the sample and assigning their initial label to these instances, while supervised partitioning considers all labels in each cell. Many instance selection techniques have been proposed. A comparative study is the aim of [16].

Information Bottleneck principle has been applied to supervised data clustering, by extracting a compact representation of the instances under a constraint on the mutual information between the clustering and the labels [15]. The problem is turned into a variational problem, which can be explicitly solved provided that the joint density of the instances and the labels is known. A bottom-up greedy agglomerative heuristic is applied which produces a hierarchy of partitions. The method works with full knowledge about the densities. In practice, the densities are estimated from their corresponding empirical densities. Furthermore, the "right" number of groups is fixed heuristically, as the evaluation criterion cannot be used to compare partitions with different sizes.

The Discriminative Clustering approach [14] consists in obtaining as discriminating clusters as possible according to the labels. A finite-data entropy-shaped evaluation criterion is proposed, derived from a MAP (for Maximum A Posteriori) estimation. In order to allow a gradient optimization, the criterion is extended by introducing smooth parameterized membership functions. The method falls into the category of Vector Quantization methods. The euclidean metric is used as the measure of dissimilarity. Though entropy measures are known to provide a good evaluation of the discrimination of the labels, it cannot be used alone for selecting the number of prototypes.

The idea of balancing the number of cells and the discrimination of the labels has led to propose a heuristic criterion which evaluates any general partition $\pi$ [2] :

$$c(\pi) = \text{Impurity}(\pi) + \beta \text{Penalty}(\pi).$$

The Impurity term evaluates the dispersion of the labels by taking the ratio of the overall number of minority examples to the number of instances. The Penalty term is essentially the square root of the ratio of the number of groups to the number of instances. $\beta$ $(0 \leq \beta \leq 3)$ is a parameter to tune.

Albeit possessing the capacity to compare partitions with different sizes, this criterion results from heuristic choices, is thus parametric and measures a non explicit quantity. Estimation of $\beta$ requires a cross-validation step, which is time demanding.

## 3 Supervised evaluation of data-dependent partitions

We adopt a descriptive approach and make the compromise between the number of prototypes and the discrimination of the distributions formally explicit. From

this results a supervised finite-data evaluation criterion for medoid-based Voronoi partitions with frequential distributions. This meaningful criterion allows to compare partitions with different sizes.

## 3.1 Notations

Let us fix the notation used throughout the following analysis. 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 $\mathbb{L} = \{l_j\}$ of size $J$ and the instances in a space $\mathbb{X}$. A dissimilarity measure $\delta : \mathbb{X} \times \mathbb{X} \to \mathbb{R}_+$ is defined.

Given a subset $P \subset \mathbb{X}$, the *Voronoi partition* $V(P) = (V(p))_{p \in P}$ relying on $P$ is defined by :

$$\forall p \in P,\ V(p) = \left\{ x \in \mathbb{X};\ p = \arg\min_{p' \in P} \delta(x, p') \right\}.$$

Thus, for $p \in P$, the *Voronoi cell* $V(p)$ contains the points $x$ for which $p$ is the most similar element in $P$, with respect to the dissimilarity measure $\delta$. The element $p$ is called the *representative* or the *prototype* of its cell $V(p)$.

In the supervised context, we seek a Voronoi partition of the space with a multinomial distribution defined in each cell. Thus, we define a *(descriptive) model* as a couple $(v, \psi)$ with $v$ a Voronoi partition with $K$ cells and $\psi$ a matrix giving the probability of label $j$ $(1 \leq j \leq J)$ in cell $k$ $(1 \leq k \leq K)$ at the position $(k, j)$. The number of cells of such a model will be denoted $k(v, \psi)$.

Given a Voronoi partition $v$ with $K$ cells $v_1, \ldots, v_K$, the size of $D^{(x)} \bigcap v_k$ is denoted $N_k$ $(1 \leq k \leq K)$ and the size of $\{X_n \in D^{(x)} \bigcap v_k;\ Y_n = l_j\}$ is denoted $N_{kj}$ $(1 \leq j \leq J)$. Thus, $N = N_1 + \cdots + N_K$ and $N_k = N_{k1} + \cdots + N_{kj}$.

## 3.2 Formalization

A model $M = (V, \Psi)$ is evaluated according to the probability $p(M, D^{(y)}/D^{(x)})$. This probability can be written as the product $p(M/D^{(x)})p(D^{(y)}/M, D^{(x)})$. Informally, instead of describing the relationship between the instances and the labels directly (i.e. the probability $p(D^{(y)}/D^{(x)})$), we firstly describe a model depending on the instances ($p(M/D^{(x)})$) and then we rely on the model in order to describe the relationship ($p(D^{(y)}/M, D^{(x)})$).

The $p(M/D^{(x)})$ part balances the contribution of the model $p(D^{(y)}/M, D^{(x)})$, in order to prevent from overfitting the data. Unlike the classical bayesian approach, which needs to set a prior $p(M)$ on the whole set of models, the descriptive approach allows to exploit a data-dependency. Furthermore, the $p(D^{(y)}/M, D^{(x)})$ part is defined without assuming that $M$ generates the data. The approach then provides a finite-data solution to the problem of overfitting.

Now, we focus on the problem of specifying $p(M, D^{(y)}/D^{(x)})$. Denoting $K = k(V, \Psi)$ :

$$p(V, \Psi, D^{(y)}/D^{(x)}) = p(K, V, \Psi, D^{(y)}/D^{(x)}),$$

which enables to compare different-sized models. The description of the models relies on the description of the number of cells, then the selection of the prototypes and finally the description of the distribution of the labels. In other words, we iterate the dependency using Bayes' formula :

$$p(V, \Psi, D^{(y)}/D^{(x)}) = p(K/D^{(x)})p(V/K, D^{(x)})p(\Psi, D^{(y)}/K, V, D^{(x)}).$$

At this step, we assume that the behavior of the multinomial distributions in the cells are conditionally independent from each other, which means :

$$p(\Psi, D^{(y)}/K, V, D^{(x)}) = \prod_{k=1}^{K} p(\Psi_k, D_k^{(y)}/V_k, D_k^{(x)}),$$

with $V_k$ the $k^{th}$ cell of $V$, $\Psi_k$ the model distribution of the labels in $V_k$ (i.e. the $k^{th}$ row of $\Psi$), $D_k^{(x)}$ the instances in $V_k$ and $D_k^{(y)}$ their labels. Then, in each cell, the model distribution is considered first and the relationship between $D_k^{(x)}$ and $D_k^{(y)}$ is considered next. In other words, we apply one last time the Bayes' rule and we have :

$$\begin{aligned} p(V, \Psi, D^{(y)}/D^{(x)}) &= p(K/D^{(x)})p(V/K, D^{(x)}) \\ &\quad \prod_{k=1}^{K} p(\Psi_k/V_k, D_k^{(x)})p(D_k^{(y)}/V_k, \Psi_k, D_k^{(x)}). \end{aligned}$$

Informally, the overall description is carried out hierarchically, which consists in applying the Bayes' theorem iteratively, and the description of the relationship between the instances and their labels is made independently in each cell.

### 3.3 Specification

We turned the problem of defining the probability $p(M, D^{(y)}/D^{(x)})$ into four similar but simpler sub-problems. Now, we propose analytic formulas for the above probabilities. The idea consists mainly in specifying the support of the involved variable and then adopting a uniform prior.

For the number of groups $K$, i.e the probability $p(K/D^{(x)})$, possible values lie between 1 and $N$. Applying a uniform prior on these values, we have :

$$p(K/D^{(x)}) = \frac{1}{N}.$$

The Voronoi partition is uniquely characterized by its prototypes. According to the data-dependency, we restrict ourselves to prototypes that are medoids and consider sets of prototypes that are subsets of $D^{(x)}$. Adopting a uniform prior would lead to use the classical $\binom{N}{K}$ binomial coefficient, as $K$ representatives among $N$ instances have to be specified. But this coefficient is symetric with respect to the number $K$ of cells for fixed $N$, and we prefer lower $K$. We select

the coefficient $\binom{N+K-1}{K-1}$, which increases with $K$, is close to $\binom{N}{K}$ for low values of $K$, null if $K = 1$, and thus models quite well our preference :

$$p(V/K, D^{(x)}) = \frac{1}{\binom{N+K-1}{K-1}}.$$

In the $k^{th}$ cell, we exploit the data-dependency and consider a restricted support for the possible multinomial distributions by taking into account only rational probabilities with $N_k$ as denominator. Formally, the support is

$$\left\{ \left( \frac{n_{k1}}{N_k}, \ldots, \frac{n_{kJ}}{N_k} \right) ; \sum_{j=1}^{J} n_{kj} = N_k \right\},$$

the cardinality of which is $\binom{N_k+J-1}{J-1}$ $(1 \leq k \leq K)$. Applying a uniform prior gives :

$$p(\Psi_k/V_k, D_k^{(x)}) = \frac{1}{\binom{N_k+J-1}{J-1}}.$$

At this point, the partition and the multinomial distribution in each cell are given and it remains to specify the label of each instance in each cell. In each cell, we suppose that the frequencies of the labels follow a multinomial distribution. The support is restricted according to the data-dependency : for the $k^{th}$ cell $(1 \leq k \leq K)$, the problem is the same as putting the elements of the cell in $J$ boxes, under the condition that the $j^{th}$ box contains $N_{kj}$ elements $(1 \leq j \leq J)$. The multinomial coefficient gives the exact number of such possibilities and we obtain :

$$p(D_k^{(y)}/V_k, D_k^{(x)}) = \frac{1}{\frac{N_k!}{N_{k1}! \ldots N_{kJ}!}}.$$

Finally, taking the negative log of $p(M, D^{(y)}/D^{(x)})$, a model $M$ is evaluated according to the following formulae :

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

Since Shannon's work [13], negative log of probabilities can be interpreted as code lengths, measured in nats if the log is natural. An example of evaluation is given on the figure 3.3.

## 3.4 Summary

According to the Stirling's approximation ($\log x! \approx x \log x - x + O(\log x)$), the last term of the above criterion is related to entropy and mutual information. The sum over the cells behaves asymptotically as $N$ times the conditional entropy of the distribution of the $Y_n$'s given the clusters assignment function :

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

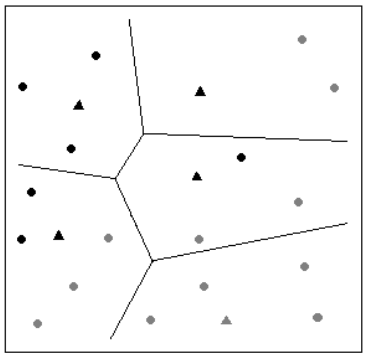

Figure 2: Evaluation of a partition with 5 prototypes (triangle instances) in a two classes problem (grey and black) with 22 instances. Specifying $K = 5$ requires $\log 22$ nats; specifying which instances are prototypes requires $\log \binom{27}{5}$ nats; specifying the frequencies in each cell requires $\log 7 + \log 6 + 2\log 5 + \log 4$ nats; specifying the relationship in each cell requires $\log \frac{6!}{3!3!} + \log \frac{4!}{2!2!} + \log \frac{3!}{2!}$ nats. The overall evaluation is 28.62 nats.

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 cutting.

The new criterion makes the probability $p(D^{(y)}, M/D^{(x)})$ explicit and is thus meaningful, being a probabilistic measure of the ability for a model to discriminate the labels. The consideration of the data-dependency at each step is the mainspring which makes the criterion

- finite-data,
- non-parametric (in the computational meaning),
- able to evaluate partitions with different sizes

without

- making the iid assumption on the sample,
- defining a generative process of the $X_n$'s,
- defining a generative process of the $Y_n$'s given the $X_n$'s,
- adopting a parametric prior.

In a few words, this criterion is well-suited for extracting reliable information from data without further knowledge, which is exactly the aim of the data preparation task.

# 4 Evaluation of the descriptive approach

We perform experiments on real and synthetic datasets in order to illustrate the properties of our criterion. In all the experiments we use the $L_1$ metric as dissimilarity measure. Firstly, we set up a synthetic problem to illustrate the discrimination ability of the criterion; secondly, we study the tolerance to the increasing presence of mislabelled data and finally, we investigate its predictive accuracy and robustness on real data.

## 4.1 The adopted heuristic

The criterion measures the discrimination capacity of medoid-based Voronoi partitions. An exhaustive search through the whole space of models (which cardinality is $2^N$) is unrealistic. We use the CLARANS heuristic proposed in [10] within an unsupervised context. The method swaps a prototype for a non prototype iteratively. Starting from an arbitrary set of $K$ prototypes, swaps are performed at random, evaluated, and triggered if the value of the criterion decreases. The number of swaps is controlled by the parameter $MaxSwapNumber$. If the criterion does not decrease after $MaxSwapNumber$ swaps, the algorithm stops and restarts with a new set of prototypes. The number of start is controlled by the parameter $LocalMinimumNumber$. More precisely, the pseudo code of the CLARANS($K$) algorithm is :

- $i \leftarrow 1$
- $Best \leftarrow NULL$
- **For** $i = 1 \ldots LocalMinimumNumber$ **Do**
  - $j \leftarrow 1$
  - $Current \leftarrow$ pick up an arbitrary set of $K$ prototypes
  - **While** $j \leq MaxSwapNumber$ **Do**
    - $Swapped \leftarrow$ swap an element in $Current$ for an element not in $Current$
    - **If** $Swapped$ has a lower cost than $Current$
      - $Current \leftarrow Swapped$
      - **Break**
    - **Else**
      - $j \leftarrow j + 1$
  - **If** $Current$ has a lower cost than $Best$
    - $Best \leftarrow Current$
- **Return** $Best$

According to the experiments carried out in [10], the authors suggest to set $LocalMinimumNumber = 2$ (i.e to use the version which finds two local optima) and $MaxSwapNumber = p \times K(N-K)$, with $p = 1.25$. We adopt the same tuning. This gives a computational complexity of $O(K^2N^2)$ for CLARANS($K$). As the presented criterion is able to compare sets of prototypes with different sizes, the algorithm CLARANS($K$) is applied iteratively for $1 \leq K \leq K_{max}$. This gives an overall complexity of $O(K_{max}^3 N^2)$

## 4.2 Discrimination

The first experiment aims at illustrating the notion of discrimination that the criterion captures. We generate a two classes dataset with 2000 points lying inside the 2d ball of radius 1 for the $L_\infty$ metric. The distribution of the classes is $(0.9, 0.1)$ in the upper-right and lower-left corners, whereas it is $(0.6, 0.4)$ in the upper-left and lower-right corners. We optimize the criterion with the CLARANS heuristic, with $K_{max} = 8$. The method builds 4 cells (cf. Figure 4.2).

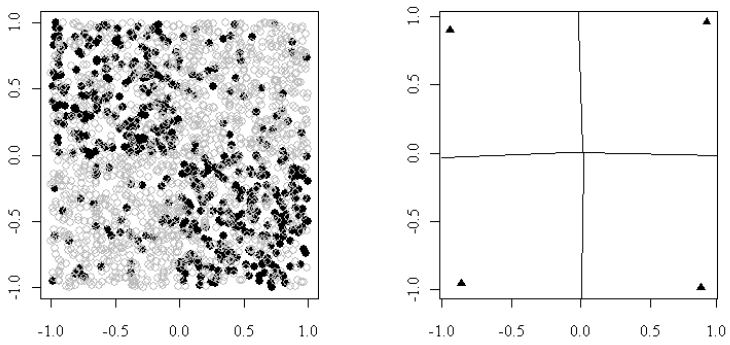


Figure 3: Synthetic dataset for discrimination. The criterion focuses on the discrimination and builds 4 cells.

Many criteria focus on the purity of the resulting clusters and thus take into account the majority class only. In other words, they focus on the prediction accuracy. In case of well separated classes, the approach is safe. But, as soon as the classes are mixed, it fails. In the discussed example, the majority class is the same everywhere : purity and prediction accuracy are irrelevant notions in this context. At the opposite, our criterion detects significant variations of the conditional distributions. Though the example is a toy one, it illustrates a classical situation. In practice, classes are often mixed or the most interesting class might be the minority one. This is the case for real problems such as fraud detection.

## 4.3 Mislabelled data

In a second experiment, we study the tolerance of the criterion to the increasing presence of mislabelled data into the dataset. We generate datasets in the same way as in the first experiment, the distribution of the classes being $(1-\alpha, \alpha)$ in the upper-right and lower-left corners and $(\alpha, 1-\alpha)$ in the upper-left and lower-right corners. The parameter $\alpha$ varies from 0 (XOR problem) to 0.5 (pure noise problem). We optimize the criterion with the CLARANS heuristic again, with $K_{max} = 8$. The optimization is run ten times. The number of prototypes is reported in the Figure 4.3.

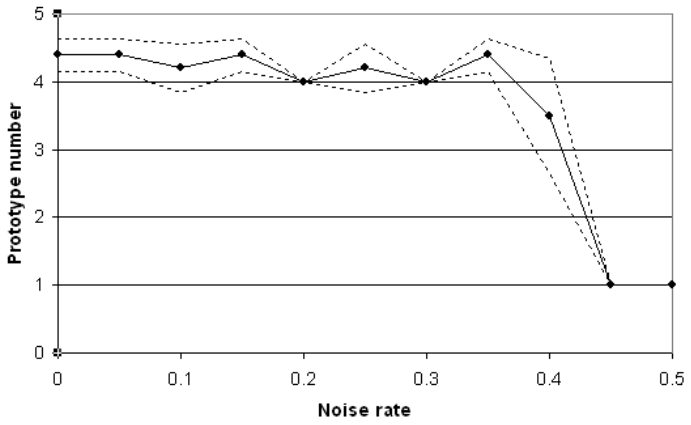


Figure 4: Tolerance to noise. For each value of the noise rate $\alpha$, the method is run ten times. The figure shows the mean and the standard deviation of the resulting number of prototypes.

The criterion is effective as long as the data contains less than 40% of mislabelled data. The method sometimes falls into a local optima and builds more than 4 cells. By compromising between the number of cells and the discrimination, the criterion is able to decide that no significant discrimination can be done : if the best partition has one single cell, the $X_n$'s and the $Y_n$'s are independent. This is the case for the noisy and next to noisy problems ($\alpha = 0.5$ and $\alpha = 0.45$).

## 4.4 Accuracy and robustness

On one hand, the Nearest Neighbor (NN) rule, which assigns the class label of its nearest neighbor to an unseen instance [3], can be thought of as a simple medoid-based partitioning method (creating one group per instance by considering the whole set $D^{(x)}$ as the set of prototypes). On the other hand, a model $(V, \Psi)$ implicitly defines a classifier : if a new instance $X$ belongs to the $k^{th}$ cell $v_k$ of $V$, $X$ is classified as the majority class in $v_k$.

| Dataset | Size | Attributes | Class | Majority Class |
|---|---|---|---|---|
| Iris | 150 | 4 | 3 | 0.33 |
| Wine | 178 | 13 | 3 | 0.40 |
| Heart | 270 | 10 | 2 | 0.56 |
| Bupa | 345 | 6 | 2 | 0.58 |
| Ionosphere | 351 | 34 | 2 | 0.64 |
| Australian | 690 | 6 | 2 | 0.56 |
| Crx | 690 | 6 | 2 | 0.56 |
| Breast | 699 | 9 | 2 | 0.66 |
| Pima | 768 | 8 | 2 | 0.65 |
| Vehicle | 846 | 18 | 4 | 0.26 |
| Waveform | 5000 | 21 | 3 | 0.34 |
| WaveformNoise | 5000 | 40 | 3 | 0.34 |

Table 1: Tested datasets.

The NN rule stores every instances and each classification of a new instance requires to browse the whole set. This limits the deployment of the method. By reducing the number of prototypes, our method solves this problem. But one could wonder whether such a reduction goes with a loss of predictive accuracy, particularly since the criterion does not focus on the accuracy.

The experiment aims at comparing both the accuracy and robustness (measured as the ratio of the test accuracy to the train accuracy) of the classifier induced by our approach and the NN rule. We select datasets (cf Table 1) from the UCI machine learning database repository [1] for which the NN rule performs far better than the majority classifier. As we focus on continuous attributes, we discard the nominal attributes of the Heart, Crx and Australian databases.

Our criterion is still optimized with the CLARANS heuristic, with $K_{max} = 10$, and the overall best partition encountered is returned. The evaluation consists in a stratified five-fold cross-validation. The predictive accuracy of the classifiers are reported in the Table 2, as well as the robustness. We can notice that the robustness of NN equals the test accuracy, since the train accuracy always equals 1. A Student's test at the 5% confidence level is performed to determine whether the differences of performance are significant.

The proposed criterion allows to select a few representative instances (about 1% of the initial database) without any loss of accuracy with respect to the NN rule. Furthermore, the robustness is dramatically increased when compared with the NN rule and nearly equals one. This property is a strong and important one for real-case studies. Indeed, it means that the best set of prototypes according to the criterion will behave quite well on previously unseen data.

The method thus outperforms the Nearest Neighbor rule in terms of predictive accuracy (4 significant wins and 1 significant loss), robustness (10 significant wins and no significant loss) and ease of deployment (1 prototype for 100 in-

| | Test accuracy | | Robustness | |
|---|---|---|---|---|
| Datatsets | SM | NN | SM | NN |
| Iris | 0.97 | 0.95 | 0.99 | 0.95 |
| Wine | 0.84 | 0.82 | 0.93 | 0.82 |
| Heart | 0.69 | 0.59 | 0.91 | 0.59 |
| Bupa | 0.66 | 0.58 | 0.87 | 0.58 |
| Ionosphere | 0.90 | 0.91 | 0.95 | 0.91 |
| Australian | 0.73 | 0.68 | 0.97 | 0.68 |
| Crx | 0.72 | 0.67 | 0.96 | 0.67 |
| Breast | 0.97 | 0.97 | 0.99 | 0.97 |
| Pima | 0.74 | 0.69 | 0.97 | 0.69 |
| Vehicle | 0.63 | 0.67 | 0.96 | 0.67 |
| Waveform | 0.81 | 0.77 | 0.98 | 0.77 |
| WaveformNoise | 0.79 | 0.76 | 0.98 | 0.76 |
| Mean | 0.786 | 0.755 | 0.954 | 0.755 |
| W/D/L | | 4/5/1 | | 10/0/0 |

Table 2: Predictive accuracy and robustness of the supervised medoid-based method (SM) and the nearest neighbor rule (NN) estimated by stratified 5-fold cross-validation. Numbers of Win/Draw/Loss of SM are reported.

stances in average).

### 4.5 Summary

Our approach makes the compromise between the discrimination of the target and the number of cells explicit. The resulting criterion allows to discriminate different behaviors of the target (as shown in the first experiment). By adopting a discriminative point of view, the criterion still behaves satisfactorily in the presence of mislabelled data (as shown in the second experiment). The classifier induced by the best partition enhances the Nearest Neighbor rule in terms of predictive accuracy and robustness (as shown in the third experiment).

## 5 Comparing the descriptive with the predictive approach

The classical approach consists in estimating a classifier with high generalization performance. We use the Linear Vector Quantization method [7] in order to illustrate the main differences with our descriptive point of view.

## 5.1 Description of the LVQ algorithm

LVQ is a state of the art supervised method which deals with Voronoi partitions. Quantization consists in placing the prototypes strategically with respect to the decision boundaries and considers a bigger class of partitions than the class of medoid-based ones, with the hope of a better fit of the data. Let $v_1(t), \ldots, v_K(t)$ ($t \in \mathbb{N}$) be a set of $K$ prototypes and $l_1, \ldots, l_K$ the labels corresponding to the given prototypes. The LVQ algorithm [7] iteratively corrects the position of the initial prototypes according to the following process :

- Sample an instance $X$ (with replacement),
- Consider the nearest prototype $v_k(t)$ of $X$,
- If $X$ and $v_k(t)$ share the same label, move $v_k(t)$ towards $X$ :
  - $v_k(t+1) = v_k(t) + \varepsilon(t)(X - v_k(t))$,
- Else move $v_k(t)$ away from $X$ :
  - $v_k(t+1) = v_k(t) - \varepsilon(t)(X - v_k(t))$.
- For every $k' \neq k$, $v_{k'}(t+1) = v_{k'}(t)$.

LVQ thus optimizes the position of the prototypes. The initial prototypes are either randomly and separately sampled within each class or results from applying $K$-means algorithm separately in each class [4]. The number $K$ of prototypes is a parameter and is fixed heuristically. The learning process is controlled by two parameters : $\varepsilon$, the learning rate, and $\alpha$, the number of iterations. The number of iterations $\alpha$ controls the rate of convergence of the method.

According to Kohonen [7], the same number of initial prototypes is assigned to each class. The learning rate $\varepsilon$ can be set initially to 0.02 and made linearly decreasing to 0. The number of iteration $\alpha$ is fixed to 500 times the number of instances.

## 5.2 Prediction vs description

Our method evaluates the best description of the relationship between the $X_n$'s and the $Y_n$'s in terms of a Voronoi partition with frequential distributions. The usual learning paradigm works with sets of classifiers and focuses on the generalization performance. We illustrate the main difference between the two approaches with the help of a two classes synthetic dataset. 2000 instances are uniformly sampled inside the 2d ball of radius 1 for the $L_\infty$ metric. The upper-left and lower-right corner are pure, while the instances in the two other corners are labelled independently and uniformly. We show in the Figure 5.2 the dataset and the Voronoi partitions resulting from the optimisation of our criterion and the LVQ method respectively, with a number of cell $K = 4$.

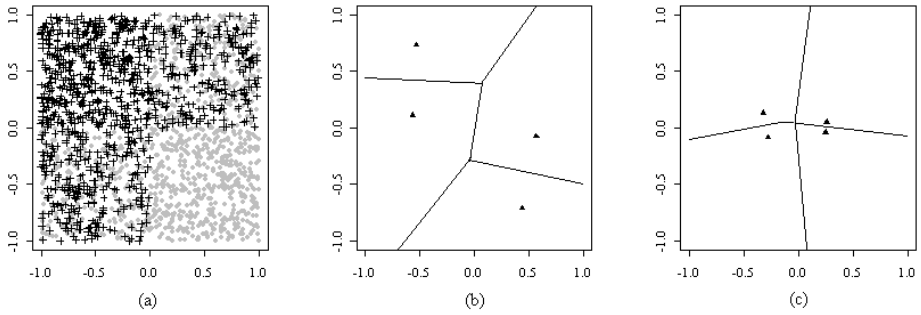


Figure 5: (a) The dataset : lower-left and upper right corners form a no man's land in term of prediction. (b) The predictive approach : quantization places a diagonal border inside the no man's land and prototypes move toward the pure areas. (c) The descriptive approach : our method discriminates behaviors.

From a predictive point of view, the mixed corner are useless. The decision boundary can be moved inside these areas without affecting the accuracy and this is what quantization does. From a descriptive point of view, significant variations of the conditional density constitute very valuable informations. This is exactly what our criterion measures.

The predictive and the descriptive approaches are not in competition with each other but complementary. In the preprocessing steps of the supervised mining task, the user wants to quickly extract relevant and reliable information, without making too much assumptions about the data. The presented criterion allows to perform relevant and reliable discrimination. In the modelling phase, predictive accuracy and generalization performance become the predominating concepts.

### 5.3 Comparison of the predictive accuracy

By taking a majority vote into each cell of the best partition, our method can be turned into a predictive one. We compare its accuracy with the LVQ method on real datasets (cf Tab.1) by a stratified five-fold cross validation. The number $K$ of prototype varies from 1 to 10. The $K$ initial prototypes for the LVQ are either selected randomly in each class or result from applying the K-means algorithm in each class, with the same number of prototypes in each class. A Student's test at the 5% confidence level is performed to determine whether the differences of performance are significant. The results are reported in the Table 3.

The numbers of Win/Draw/Loss of the presented criterion against the best LVQ initialised at random or by the K-means algorithm are 2/8/2 and 3/7/2 respectively. While the experimental protocol favours the LVQ method, the predictive use of our method is shown to be competitive with quantization on

| | Test accuracy | | |
|---|---|---|---|
| Dataset | SM | RLVQ | KMLVQ |
| Iris | 0.97 | 0.97 | 0.98 |
| Wine | 0.84 | 0.78 | 0.75 |
| Heart | 0.69 | 0.68 | 0.70 |
| Bupa | 0.66 | 0.65 | 0.66 |
| Ionosphere | 0.90 | 0.85 | 0.85 |
| Australian | 0.73 | 0.69 | 0.65 |
| Crx | 0.72 | 0.69 | 0.67 |
| Breast | 0.97 | 0.96 | 0.95 |
| Pima | 0.74 | 0.73 | 0.75 |
| Vehicle | 0.63 | 0.54 | 0.55 |
| Waveform | 0.81 | 0.84 | 0.85 |
| WaveformNoise | 0.79 | 0.84 | 0.85 |
| Mean | 0.786 | 0.770 | 0.768 |
| W/D/L | | 2/8/2 | 3/7/2 |

Table 3: Predictive accuracy resulting from the optimisation of the presented supervised criterion (SM) and from LVQ initialised at random (RLVQ) or by the K-means algorithm (KMLVQ). The best predictive accuracy of LVQ (for $1 \leq K \leq 10$) is reported only. Numbers of Win/Draw/Loss of SM are reported.

the tested datasets, as could be expected. Indeed, detection of pure areas is a part of the discrimination task.

The quantization method relies on a bigger set of models than our method. While we evaluate sets of prototypes included in the set of instances, the LVQ algorithm can place the prototypes everywhere in the whole space. This explains the significant differences observed on the Waveform datasets. But the LVQ method is theoretically restricted to deal with euclidean spaces. Furthermore, it is parametric and faces the risk of overfitting. By restricting ourselves to medoid-based Voronoi partitions, we are able to set a non-parametric finite-data criterion which automatically prevents from overfitting the data. The experiment shows that such a structural restriction does not lead to a significant loss in term of predictive accuracy.

# 6 Conclusion

Classification according to the nearest neighbor relies on the construction of a Voronoi partition or, which is the same, a set of prototypes. In this paper, we proposed a supervised evaluation of medoid-based Voronoi partitions. A descriptive approach has been adopted. The resulting criterion, based on a comprehensible compromise between discrimination and number of prototypes, is non-parametric and automatically handles the problem of overfitting. In

practice, this criterion makes validation sets useless and gives reliable results, as shown by a first set of experiments.

While not focusing on the generalization performance, the method builds a medoid-based Voronoi partition which naturally defines a classifier. It is then interesting to evaluate its predictive accuracy. A first experiment shows that the Nearest Neighbor rule classifying on the whole set of instances is outperformed in terms of predictive accuracy and robustness. Furthermore, by selecting few prototypes (about 1% of the initial database), our method makes the Nearest Neighbor rule effective and attractive for real case studies, especially when the deployment of models is considered.

While we evaluate medoid-based Voronoi partitions only, the LVQ algorithm uses quantization in order to finely place the prototypes. The quantization method relies on a bigger set of models with the hope of a better fit of the data. An experiment shows that this is not necessarily the case (2 and 3 wins of our method against 2 losses) and that the predictive accuracy is slightly the same on the average (78.6% against 77%). The main difference is the confidence in the result : quantization algorithm is parametric and requires an heuristic control of overfitting. At the opposite, our criterion is non-parametric and handles the problem of overfitting the data.

In a few words, the optimisation of the presented criterion brings trustworthy descriptive information, which can be used in a prediction purpose as well.

# 7 Acknowledgement

We are grateful to the anonymous referees who provided helpful suggestions, and to Fabrice Clérot, Senior Expert at France Télécom R&D, for his careful proofreading of the manuscript and his comments.

# References

[1] C.L. Blake and C.J. Merz, UCI repository of machine learning databases, http://www.ics.uci.edu/mlearn/MLRepository.html.

[2] C. Eick, N. Zeidat and R. Vilalta, Using representative-based clustering for nearest neighbor dataset editing, Proceedings of the 4th International Conference on Data Mining, 2004, pp. 375–378.

[3] E. Fix and J. Hodges, Discriminatory analysis. Nonparametric discrimination : consistency properties, Technical Report 4, USAF School of Aviation Medicine, Randolf Field, TX, 1951.

[4] T. Hastie, R. Tibshirani and J. Friedman, The elements of statistical learning, Springer, 2001.

[5] L. Kaufman and P.J. Rousseeuw, Clustering by means of medoids, Statistical Data Analysis Based on the L1-Norm, Y. Dodge, Ed. Amsterdam, The Netherlands: North-Holland, 1987, pp. 405–416.

[6] L. Kaufman and P.J. Rousseeuw, Finding groups in data : an introduction to cluster analysis, Wiley & Sons, 1990.

[7] T. Kohonen, Self-organizing maps. Springer, 3rd edition, 2001.

[8] P. Kontkanen, P. Myllymäki, W. Buntime, J. Rissanen and H. Tirri, An MDL framework for data clustering, Advances in Minimum Description Length: Theory and Application, Grunwald, Myung and Pitt, MIT Press, 2005.

[9] J. McQueen, Some methods for classification and analysis of multivariate observations, Fifth Berkeley Symposium on Mathematical Statistics and Probability, Le Cam and Neyman, 1, 1967, pp. 281–297.

[10] R.T. Ng and J. Han, CLARANS: a method for clustering objects for spatial data mining, IEEE Transactions on Knowledge and Data Engineering, 14-5, 2002, pp. 1003–1016.

[11] F.P. Preparata and M.I. Shamos, Computational geometry : an introduction, Springer, 1986.

[12] J. Rissanen, Fisher information and stochastic complexity, IEEE Transactions on Information Theory, 42-1, 1996, pp. 40–47.

[13] C.E. Shannon, A mathematical theory of communication, Bell Systems Technical Journal, 27, 1948, pp. 379–423 and pp. 623–656.

[14] J. Sinkkonen, S. Kaski and J. Nikkilä, Discriminative Clustering: optimal contingency tables by learning metrics, Proceedings of the 13th European Conference on Machine Learning, 2002, pp. 418–430.

[15] N. Slonim and N. Tishby, Agglomerative information bottleneck, Proceedings of Neural Information Processing System, 12, 2000, pp. 617–623.

[16] D.R. Wilson and T.R. Martinez, Reduction techniques for instance-based learning algorithms, Machine Learning, 38-3, 2000, pp. 257–286.