A Kernel Defined over Qualitative Spaces of Orders of Magnitude

Núria Agell* **and Xari Rovira***
ESADE, Universitat Ramon Llull
Av. Pedralbes, 62
08034 Barcelona (Spain)
{agell, rovira}@esade.edu

Mónica Sánchez* **and Francesc Prats***
MA2, Universitat Politècnica de Catalunya
Pau Gargallo, 5
08028 Barcelona (Spain)
{monica.sanchez, francesc.prats}@upc.es

Abstract

This paper lies within the domain of learning algorithms based on kernels of Support Vector Machines. A kernel is constructed over the discrete structure of absolute orders of magnitude spaces. This kernel is based on an explicit function, defined from the space of k-tuples of qualitative labels to a feature space, which captures the remoteness between the components of the patterns by using certain weights exponentially. A simple example that allows interpreting the kernel in terms of proximity of the patterns is presented.

Keywords: Learning Algorithms, Support Vector Machines, Orders of Magnitude Reasoning.

1. Introduction

The construction of machines cable to learn from data is one of the main goals of Artificial Intelligence. Lately, different learning machines based on kernels, such as Support Vector Machines (SVM), have been developed and studied in depth because of their numerous applications and their efficiency in the learning process.

One of the most important steps in the construction of Support Vector Machines is the development of kernels adapted to the different structures of the data in real world problems [2], [3], [5].

Within the frame of Artificial Intelligence, a key factor in situations in which one has to obtain some conclusions from imprecise data, is to be able to use variables described via orders of magnitude. One of the goals of Qualitative Reasoning is just to tackle problems in such a way that the principle of relevance is preserved [7]; that is to say, each variable involved in a real problem is valued with the required level of precision.

In classification processes the situation in which the numerical values of some of the data are unknown, and only their qualitative descriptions are available - given by their absolute or relative orders of magnitude - is not unusual. In other situations, the numerical values, even though they might be available, are not relevant for solving the proposed problem. This paper starts from *absolute orders of magnitude models* [8], [9], which work with a finite set of symbols or qualitative labels obtained via a partition of the real line, where any element of the partition is a basic label. These models provide a mathematical structure which unifies sign algebra and interval algebra trough a continuum of qualitative structures built from the rougher to the finest partition of the real line. This mathematical structure, the Qualitative Algebras or Q-Algebras, have been studied in depth [1], [9].

In recent studies, some kernels have been constructed over certain discrete structures, for example for linguistic text classification [5] and [6]; nevertheless, there is no kernel available to work with data described in a space of orders of magnitude.

This work presents a kernel over a qualitative space of absolute orders of magnitude, based on an explicit function

* GREC- Knowledge Engineering Research Group

defined over labels. This kernel will be used for classification in learning algorithms based on kernels, in particular in Support Vector Machines, as a part of the development of the MERITO (Analysis and Development of Innovative Soft-Computing Techniques with Expert Knowledge Integration. An Application to Financial Credit Risk Measurement) project, in which different tools for the measurement of the financial credit risk are analysed.
Often, the classification function cannot be expressed as a simple linear combination of the attributes or input variables. Support Vector Machines are learning systems, which use linear functions in a feature space of higher dimension as classification functions by using several kernels [4], [10] and [11].
The mapping between the initial space and the feature space can be defined explicitly in advance, in order to construct an inner product that will give raise to the kernel. However, it is also possible, on the contrary, to construct a kernel directly, which allows for the implicit definition of the function from the data space into the feature space, in which linear learning machines operate. In this work the kernel is constructed following the first option mentioned above.

In Section 2 the absolute orders of magnitude model with granularity n, OM(n), constructed via a symmetric partition of the real line, is presented. Section 3 gives the basic concepts of Support Vector Machines and highlights the importance of kernels for these kinds of learning algorithms. In Section 4 an explicit function from the quantity space into the feature space is defined; in Section 5, this function allows the construction of a kernel to be able to work in spaces OM(n). The paper ends with several conclusions and outlines some proposals for future research.

2 The absolute orders of magnitude model

In this section the absolute orders of magnitude model is described [1]. The model we use is a generalisation of the model introduced in [9]. The number of labels chosen for describing a given real problem depends on its characteristics.
The absolute orders of magnitude model of granularity n, OM(n), is defined from a symmetric partition of the real line in 2n+1 classes:

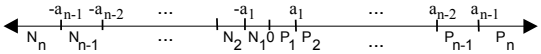

Fig.1. Partition of the real line

where $N_i=[-a_i,-a_{i-1})$, $0=\{0\}$ and $P_i=(a_{i-1},a_i]$.
Each class is named *basic description or basic element,* and, using the notation intruduced in [1], is represented by a label of the set S_1:

$$S_1=\{N_n, N_{n-1}, ..., N_1, 0, P_1, ..., P_{n-1}, P_n\}.$$

Finally, once the partition that defines S_1 is fixed, *the quantity space S* is the set of labels in the form [X,Y] for all $X,Y\in S_1$, with X<Y (i.e., x<y for all $x\in X$ and $y\in Y$):

$$[X,Y]=\begin{cases} X, & \text{if } Y=0; \\ Y, & \text{if } X=0; \\ \text{the smallest interval with respect to inclusion containing X and Y,} & \text{if } X\neq 0 \text{ and } Y\neq 0. \end{cases}$$

An order relation $\leq_P$ is defined in S, *to be more precise than*: given $X,Y\in S$, X is more precise than Y ($X\leq_P Y$) if $X\subseteq Y$. In Figure 2 this order relation is represented graphically:

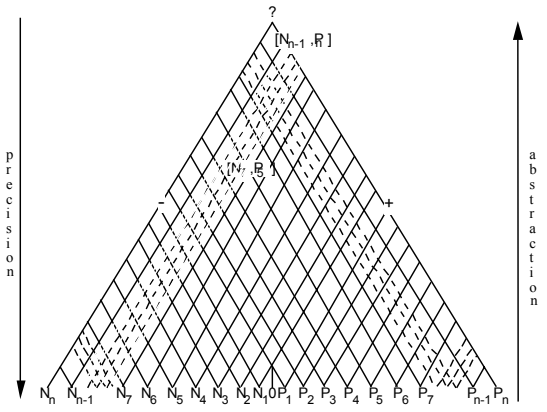

Fig. 2. The order relation $\leq_P$

For all $X\in S-\{0\}$, the *basis of X* is the set $B_X=\{B\in S_1-\{0\}: B\leq_p X\}$; and for all $X\in S$, the *extended basis of X* is the set $B_X^*=\{B\in S_1 : B\leq_p X\}$.
The *qualitative equality* relation is defined as follows: given $X,Y\in S$, they are q-equals, $X\approx Y$, if there exists $Z\in S$ such that $Z\leq_P X$ and $Z\leq_P Y$. This means that they have a common basic element, i.e., $B_X^*\cap B_Y^*\neq\varnothing$. The pair (S, $\approx$) is called a *qualitative space of orders of magnitude*; and, taking into account that it has 2n+1 basic elements, it is said that (S, $\approx$) has granularity n.
In (S, $\approx$) it is considered the mapping $\|\ \|$ from S to $\mathbb{N}^+\cup\{0\}$ defined by:

$$||X|| = Card\, B_{X_0} - Card\, B_X.$$

Where for each $X\in S$ it is considered X_0 as the 0-expansion defined as $\psi_0(X)= X_0=\text{Min}\{Y\in S: X\leq Y \text{ and } 0\leq Y\}$, and Card means the number of elements in a set. It has been shown that under certain conditions on the given partition, the mapping $\|\ \|$ satisfies the following properties:

1) $\|X\|\geq 0$, and $\|X\|=0$ if, and only if $X\approx 0$.

2) $\|X\|=\|\ominus X\|$.
3) $\|X\oplus Y\|\leq\|X\|+\|Y\|$.

Note that $\|\ \|$ satisfies the classical properties of norms except the first one where equality is changed by qualitative equality. The pair $(S, \|\ \|)$ is called qualitative normed space.

Finally, in order to work with qualitative and quantitative data simultaneously, it is useful to consider the qualitative expression of a set A, denoted by [A] and that it is defined by the smallest element of S with respect to inclusion that contains A.

3. Kernels in Support Vector Machines

In this work we preopose a methodology, which will allow SVM to be used when the input data are described by their orders of magnitude.
Before building an appropriate kernel for this kind of discrete spaces, let us remind ourselves of the basic concepts of Support Vector Machines and kernel functions, introduced by Vapnik in 1979 [10].
The SVM are used in learning problems, where the input data are not linearly separable. From a non-linear mapping the input data are imbedded into a space named feature space, potentially of higher dimension, in which the separability of the data can be obtained in a linear manner. In Figure 3 a scheme of this process can be observed.

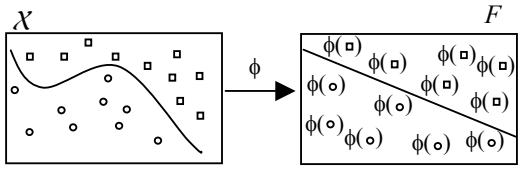

Fig.3. An application from the input space to the feature space

That is to say, noting the input space by $\mathcal{X}$ and the feature space by F, a machine of non-linear classification is built in two steps.
First, a non-linear mapping $\phi:\mathcal{X}\rightarrow F$ transforms the input data to the feature space, and afterwards an algorithm of linear separation is used in this new space.
An important characteristic of the learning process of a SVM is the fact that only a few elements of the training set are meaningful for the classification. These elements, named *support vectors (SV)*, are ones closest to the separator hyperplane. In Figure 4 the support vectors are doubly marked.

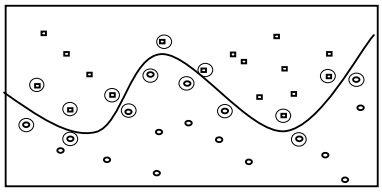

Fig.4. Binary classification of non linearly separable patterns by means of SVM

Let T be the set of input-output training data pairs:

$$T = \{ (\mathbf{x}_i, y_i), \mathbf{x}_i \in \mathcal{X}, y_i=\{-1,+1\ \} \},$$

where labels +1 and -1 represent the two different classes of objects in $\mathcal{X}$.
Let's assume that the training set is separable by a hyperplane; then the linear decision function can be written as:

$$f(\mathbf{x}) = \text{sign}\left(\sum_{\mathbf{x_i}\in SV} \alpha_i y_i (\mathbf{x_i}\cdot\mathbf{x}) + b \right)$$

for all $\mathbf{x}$ in $\mathcal{X}$, where α_i and b are the hyperplane coefficients and $\mathbf{x}_i$ are the elements in T closest to the hyperplane, which have been chosen as support vectors in the learning process. But the choice of a linear function seems to be very restrictive. In the general case of non-linear separability, the decision function turns out to be a non-linear function which appears by substituting the inner product in $\mathcal{X}$ by an inner product in the feature space F, given by the function K such that:

$$K(\mathbf{x}_i,\mathbf{x}_j) = \phi(\mathbf{x}_i)\cdot\phi(\mathbf{x}_j).$$

Such a function K is called a *kernel*. The name kernel is derived from Integral Operation Theory, and a characterization of this kind of functions is given by Mercer's theorem [11].
This leads to the non-linear decision function:

$$f(\mathbf{x}) = \text{sign}\left(\sum_{\mathbf{x_i}\in SV} \alpha_i y_i K(\mathbf{x_i},\mathbf{x}) + b \right)$$

A very important advantage of this methodology is that it is not necessary to have an input space with an inner product, i.e. it works for non-Euclidean spaces. Special kernels of that type have been used with many different kinds of data in the input space: to categorize text documents, for protein classification, to classify images, etc., mapping data into a feature space F, which is a Euclidean space. Different applications can be found in [5] and [6].

4. An explicit feature mapping ϕ from a space $[OM(n)]^k$

Following the method used in [5] to obtain a kernel over a discrete space, and in particular to define a kernel over the space S^k, in this section we define explicitly a feature function ϕ from the quantity space S^k to a feature space F. Further on, in section 5, the kernel will be obtained by the composition of ϕ with the inner product in F.
This process begins with the definition of the functions' *basic expansions*, introduced in section 2 in the particular case of the expansion of zero. Such a function, associated to a given basic element U, maps each element X in S to the minimum label that is *less precise* than X and, at the same time, than U.

Given $U \in S_1$ we call *U-expansion* the map $\Psi_U: S \longrightarrow S$, such that:

$$\Psi_U(X) = \text{Min}\{Y \in S : X \leq_P Y \text{ and } U \leq_P Y\} = [X \cup U].$$

It is the smallest interval with respect to the inclusion containing X and U.

From now on X_U will mean the image of X by Ψ_U. It's easy to see that this map is well defined in the sense that the minimum that is used in the definition exists and it is unique for all $X \in S$. The map satisfies:

a) $X = X_U$, if, and only if, $X \approx U$ (i.e. $U \subset X$)

b) $X \approx X_U$ and $U \approx X_U$

It is necessary to note that X_U does not depend on the values of the landmarks used to determine the real line partition.

As an example to illustrate this map, considering the absolute orders of magnitude model with granularity 2, OM(2), then:

$$S_1 = \{N_2, N_1, 0, P_1, P_2\}$$

and $S = S_1 \cup \{[N_2,N_1],[N_1,P_1],[P_1,P_2],[N_2,P_1],[N_1,P_2],?\}$.

Choosing $U = N_1$, therefore, it is obtained:

$$\Psi_{N_1}(N_2) = \Psi_{N_1}([N_2, N_1]) = [N_2, N_1]$$

$$\Psi_{N_1}(N_1) = N_1$$

$$\Psi_{N_1}(0) = \Psi_{N_1}(P_1) = \Psi_{N_1}([N_1, P_1]) = [N_1, P_1]$$

$$\Psi_{N_1}(P_2) = \Psi_{N_1}([P_1, P_2]) = \Psi_{N_1}([N_1, P_2]) = [N_1, P_2]$$

$$\Psi_{N_1}([N_2, P_1]) = [N_2, P_1]$$

$$\Psi_{N_1}(?) = ?$$

Related to this map, and inspired in the definition of qualitative norm [1] explained in section 2, it is defined in S, for a fixed $U \in S_1$, the map "*remoteness with respect to U*", $a_U : S \longrightarrow N$, such that:

$$a_U(X) = \text{Card}(B_{X_U}) - \text{Card}(B_X).$$

For all $U \in S_1$, the map a_U satisfies:

a) $a_U(X) = 0$, if, and only if $X \approx U$

b) $a_U(X) = \underset{B \in B_X}{\text{Min}}\, a_U(B)$

For any $X \in S$, the "further" the basics in B_X are with respect to the basic element U in the ordered set S_1, the greater is the value of $a_U(X)$, with the exception of zero.

Considering again the space OM(2) and the basic element $U = N_1$ to show how a_U works:

$$a_{N_1}(N_2) = a_{N_1}(P_1) = a_{N_1}([P_1, P_2]) = 1$$

$$a_{N_1}(N_1) = a_{N_1}([N_2, N_1]) = a_{N_1}([N_1, P_1]) = a_{N_1}([N_2, P_1])$$

$$= a_{N_1}([N_1, P_2]) = a_{N_1}(?) = 0$$

$$a_{N_1}(P_2) = a_{N_1}(0) = 2$$

Looking to the basic elements in S_1-{0}, the basic P_2 is the "furthest" with respect to N_1, and it has, by the map a_U defined over the space OM(2), the greatest value $a_{N_1}(P_2) = 2$

Finally, and as a prior step for the definition of ϕ, the map ϕ_U associated to any basic element $U \in S_1$, over the space S^k is defined by:

$$\phi_U(\mathbf{X}) = \phi_U(X_1,...,X_k) = \left(\lambda^{a_U(X_1)},\ldots,\lambda^{a_U(X_k)}\right),$$

for some $\lambda \in]0,1[$. The decay factor λ between 0 and 1 is used in each component to weight the remoteness between two elements in S.

The map ϕ_U transforms each element in S^k into an element in $[0,1]^k$, which reflects the remoteness of **X**'s components with respect to the basic element U. In this way, the components in **X** that are qualitatively equal to U take the value 1 in the corresponding component in $\phi_U(\mathbf{X})$, and less than 1 if they are not. In general, values near 1 in the components of $\phi_U(\mathbf{X})$ mean that their respective components of **X** are "close" to U.

Now the explicit feature mapping can be defined, $\phi : S^k \longrightarrow F$, which will allow moving data from the quantity space S^k to the feature space *F*.

For all $\mathbf{X} \in S^k$, the vector $\phi(\mathbf{X})$ is:

$$\phi(\mathbf{X}) = (\phi_U(\mathbf{X}))_{U \in S_1} = (\phi_{N_n}(\mathbf{X}),\ldots,\phi_{P_n}(\mathbf{X}))$$

where the feature space *F* of vectors $\phi(\mathbf{X})$ is a subset of $[0,1]^{k(2n+1)}$.

5. Construction of a kernel in an orders of magnitude space

Once the explicit function ϕ has been defined on the space S^k, the kernel is defined via the Euclidean product existing in the space *F*; for all **X**,**Y** belonging to S^k, it is considered:

$$K(\mathbf{X},\mathbf{Y}) = \langle \phi(\mathbf{X}), \phi(\mathbf{Y}) \rangle$$

The explicit kernel expression is as follows: given two k-tuples of qualitative labels, $\mathbf{X} = (X_1,...,X_k)$ and $\mathbf{Y} = (Y_1,...,Y_k)$:

$$K(\mathbf{X},\mathbf{Y}) = \sum_{U \in S_1} \langle \phi_U(\mathbf{X}), \phi_U(\mathbf{Y}) \rangle =$$

$$\sum_{U \in S_1} \left\langle \left(\lambda^{a_U(X_1)},\ldots,\lambda^{a_U(X_k)}\right), \left(\lambda^{a_U(Y_1)},\ldots,\lambda^{a_U(Y_k)}\right) \right\rangle =$$

$$\sum_{U \in S_1} \sum_{i=1}^{k} \lambda^{a_U(X_i)} \lambda^{a_U(Y_i)} = \sum_{U \in S_1} \sum_{i=1}^{k} \lambda^{a_U(X_i) + a_U(Y_i)}$$

From its own definition the function K defined over $S^k \times S^k$ is a kernel and it is not necessary to verify that Mercer conditions are fulfilled [11].

Next, an example with an effective calculus with the kernel considered is given. This example will allow interpreting the results obtained in terms of "remoteness".

Example. Consider an OM(2) space with basic labels $\{N_2, N_1, 0, P_1, P_2\}$ and three patterns **X**, **Y**, **Z**, given by terns of S^3.
Let be, $\mathbf{X}=(P_1,[N_2,N_1],[N_1,P_2])$, $\mathbf{Y}=([P_1, P_2], N_1,0)$ and $\mathbf{Z}=(N_2, P_1,[N_2, N_1])$, then:

$\phi_{NG}(\mathbf{X})=(\lambda^2,\lambda^0,\lambda^1)$, $\phi_{NP}(\mathbf{X})=(\lambda^1,\lambda^0,\lambda^0)$, $\phi_0(\mathbf{X})=(\lambda^1,\lambda^1,\lambda^0)$, $\phi_{PP}(\mathbf{X})=(\lambda^0,\lambda^1,\lambda^0)$, $\phi_{PG}(\mathbf{X})=(\lambda^1,\lambda^2,\lambda^0)$.

$\phi_{NG}(\mathbf{Y})=(\lambda^2,\lambda^1,\lambda^3)$, $\phi_{NP}(\mathbf{Y})=(\lambda^1,\lambda^0,\lambda^2)$, $\phi_0(\mathbf{Y})=(\lambda^1,\lambda^1,\lambda^0)$, $\phi_{PP}(\mathbf{Y})=(\lambda^0,\lambda^1,\lambda^2)$, $\phi_{PG}(\mathbf{Y})=(\lambda^0,\lambda^2,\lambda^3)$.

$\phi_{NG}(\mathbf{Z})=(\lambda^0,\lambda^2,\lambda^0)$, $\phi_{NP}(\mathbf{Z})=(\lambda^1,\lambda^1,\lambda^0)$, $\phi_0(\mathbf{Z})=(\lambda^2,\lambda^1,\lambda^1)$, $\phi_{PP}(\mathbf{Z})=(\lambda^2,\lambda^0,\lambda^1)$, $\phi_{PG}(\mathbf{Z})=(\lambda^3,\lambda^1,\lambda^2)$.

Therefore, it is:

$$K(\mathbf{X},\mathbf{Y}) = \langle \phi(\mathbf{X}), \phi(\mathbf{Y}) \rangle = (\lambda^4+\lambda^1+\lambda^4) + (\lambda^2+\lambda^0+\lambda^2) + (\lambda^2+\lambda^2+\lambda^0) + (\lambda^0+\lambda^2+\lambda^2) + (\lambda^1+\lambda^4+\lambda^3) = 3\lambda^4+\lambda^3+6\lambda^2+2\lambda+3,$$

$$K(\mathbf{Y},\mathbf{Z}) = \langle \phi(\mathbf{Y}), \phi(\mathbf{Z}) \rangle = (\lambda^2+\lambda^3+\lambda^3) + (\lambda^2+\lambda^1+\lambda^2) + (\lambda^3+\lambda^2+\lambda^1) + (\lambda^2+\lambda^1+\lambda^3) + (\lambda^3+\lambda^3+\lambda^5) = \lambda^5+6\lambda^3+5\lambda^2+3\lambda$$

As can be seen the kernel has been constructed from a function ϕ, which has been defined by means of a set of weights used exponentially. Those exponents capture the concept of "remoteness" of each pattern component with respect to each one of the basic labels. Therefore, the more qualitatively near components two patterns are, the more similar they will be considered to be, because their Euclidian product will be higher.
In the given example, X and Y are two patterns very similar (qualitatively equal component by component); on the contrary, the components of Y and Z very distant. In Figure 5, the values taken by K(X,Y) and K(Y,Z) can be seen.

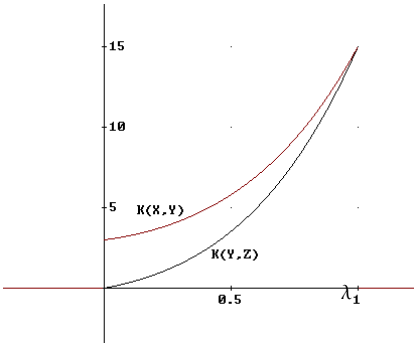

Fig.5. Comparison of the values given by the kennel with respect to parameter λ.

If the functions that represent the values given by the kernel are observed in both cases, it can be seen that, for any value between 0 and 1 given to λ, a greater value is always obtained in the case of the more similar patterns.

6. Conclusions and future research

The present work belongs to a wider project, which aims at motivating, defining, and analysing the viability of the use of learning machines in structures defined in orders of magnitude spaces.
The focus of this paper is the construction of a kernel to be used in problems for which the input variables are described in terms of qualitative values of orders of magnitude. For this reason the kernel has been built from a set of weights considered exponentially by an evaluation of its qualitative information. This set of weights captures the known information about de "remoteness" between qualitative values.
Although this paper has focused on a classification problem by using Support Vector Machines, the methodological aspects considered and given can be used in any learning system based on kernels.
As a future work, the implementation of the given method to be applied in problems of classification and multi-classification might be considered.
In particular, and within the MERITO project, supported by the Spanish Ministry of Science and Technology, the methodology given in this paper is going to be used. The project addresses the prediction and measurement of financial credit risk. The results obtained by using input variables defined over orders of magnitude spaces will be compared with the ones obtained by using numerical values.
Considering an OM(n), several concepts can be analysed to measure the degree of "remoteness" or "closeness" between labels, it seems to be reasonable to look for other suitable kernels in these kinds of sets. With regard to open problems and future work, the following comments can be made:

- To define new concepts to measure the degree of "remoteness" or "closeness" between qualitative labels.
- To choose different parameters of decay λ in the Euclidean product expression depending on the length of the intervals defining the basic labels.
- To define new kernels combining numeric and qualitative data.

Acknowledgements

This work was partially supported by the MCyT (Spanish Ministry of Science and Technology) MERITO project (TIC2002-04371-C02).
Cecilio Angulo's valuable remarks and suggestions are gratefully acknowledged.

References

[1] Agell, N. *Estructures matemàtiques per al model qualitatiu d'ordres de magnitud absoluts*. Ph. D. Thesis Universitat Politècnica de Catalunya, 1998.
[2] Angulo, C. *Aprendizaje con màquina núcleo en entornos de multiclasificación.* Ph. D. Thesis Universitat Politècnica de Catalunya, 2001.

[3] Burges, C. A tutorial on support vector machines for pattern recognition, *Data Mining and Knowledge Discovery*, 2 (1998), 1-47.
[4] Cortes, C; Vapnik,V. Support vector networks, *Machine Learning*, 20 (1995), 273–297.
[5] Cristianini, N.; Shawe-Taylor, J. *An Introduction to Support Vector Machines and other Kernel-based learning methods*. Cambridge University Press. 2000.
[6] Lodhi, H; Saunders, C; Sahwe-Taylor, J;Cristianini, N.; Watkins, C. Text Classification Using String Kernels. *Journal of MachineLearning Research,* (2): 419-444, 2002.
[7] Forbus, K. D. Commonsense physics. A: *Annals Revue of Computer Science*. 1988, pàg. 197-232.
[8] Piera, N. *Current Trends in Qualitative Reasoning and Applications*. Monograph CIMNE, núm. 33. International Center for Numerical Methods in Engineering, 1995.
[9] Travé-Massuyès, L.; Dague, P.; Guerrin, F. *Le Raisonnement Qualitatif pour les Sciences de l'Ingénieur*. Hermès, 1997.
[10] Vapnik, V. *Estimation of Dependences Based on Empitical Data* [in russian]. Nauka, 1979. (English translation: Springer Verlag , 1982).
[11] Vapnik, V. *The nature of statistical learning theory*, Springer Verlag New York, 1995.