How to learn consumer preferences from the analysis of sensory data by means of support vector machines (SVM)

Antonio Bahamonde, Jorge Díez, José Ramón Quevedo,
Oscar Luaces, and Juan José del Coz
Centro de Inteligencia Artificial. Universidad de Oviedo at Gijón,
Campus de Viesques, E-33271 Gijón (Asturias), Spain
{antonio, jdiez, quevedo, oluaces, juanjo}@aic.uniovi.es

Abstract. *In this paper we discuss how to model preferences from a collection of ratings provided by a panel of consumers of some kind of food product. We emphasize the role of tasting sessions, since the ratings tend to be relative to each session and hence regression methods are unable to capture consumer preferences. The method proposed is based on the use of Support Vector Machines (SVM) and provides both linear and nonlinear models. To illustrate the performance of the approach, we report the experimental results obtained with a couple of real world datasets.*

1. Introduction

The quality or acceptability of market products can be measured in a number of different dimensions. Sensory analysis is concerned with those aspects that are principally appreciated through sensory impressions and is typically used by food industries and breeders to improve certain production decisions (Murray et al., 2001; Muñoz, 2002). In this paper, we discuss the analyses of so-called sensory data that include the assessment of products provided by a panel of consumers, i.e. groups of consumers who are asked to rate their degree of acceptance of a sample of food products on a scale.

The type of analysis that we shall discuss in this paper consists in the search for functions able to relate product descriptions with consumer preferences, what it is usually called external preference mapping (McEwan, 1996). On the other hand, the so called internal preference mapping only uses consumers' ratings, and it is applied to discover market segments formed by consumers with similar tastes that can be differentiated from other segments (Van Kleef, Van Trijp, and Luning, 2006). The available methods for tackling these problems employ a combination of analysis of variance (ANOVA, see for instance (Lea, Næs, Rodbotten, 1997)), principal component analysis (PCA, see (Ellekjær, Ilseng, and Næs, 1996)), and regression (McEwan, 1996; Tenenhaus, Pagès, Ambroisine, and Guinot, 2005).

In all cases, these methods require that all the panel members have to rate all food samples. However, this is a difficult request when dealing with food products; we

must realize that frequently the size of the sample prevents panel members from tasting all products. Then, we cannot ask our panel members to spend long periods rating the whole set of food samples. Typically, each consumer only participates in one or a small number of tasting sessions, usually in the same day. Note that tasting a large sample of foods may be physically impossible in a short period of time or that the number of tests performed may damage the sensory capacity of consumers.

Independently of the numbers of tastes made by each consumer, a first attempt to learn preference functions may consist of a regression method. We shall show that regression methods (linear or nonlinear, such as Partial Least Squares (PLS)) are not adequate in general for finding preference functions. The main reason is that consumer ratings cannot be interpreted as absolute assessments. Consumers tend to rate their preferences in a relative way, comparing objects with the other samples in the same batch or tasting session. There is a kind of *batch effect* that often biases the ratings. Thus, an object presented in a batch surrounded by worse objects will probably obtain a higher rating than if it were presented together with better objects.

To avoid the relative meaning of consumer ratings, we will use an Artificial Intelligence method based on a family of Machine Learning algorithms called Support Vector Machines (SVM), see (Vapnik, 1998; Schölkopf, Smola, 2002; Shawe-Taylor, Cristianini, 2004). Artificial Intelligence techniques have proven their usefulness in the food industry, as they incorporate new methods to construct functions induced from by a set of data; see, for instance (Goyache et al., 2001).

The method of learning preference functions using SVM will be explained for linear and nonlinear cases, and it does not require all the products to have been tasted by all the panel members. To illustrate the performance of the method we shall report the achievements obtained with real world datasets compiled from panels rating beef and cider.

2. Conceptual framework

When consumers are involved, sensory data may include the assessment of products provided by two different kinds of panels. The first is made up of a small group of expert, trained judges; these will describe each product by attribute-value pairs. Expert panelists are thus required to possess enough sensory accuracy so as to discriminate between different and similar products; note that experts are not necessarily asked to rate the overall quality of products. This kind of panel will play the role of a bundle of sophisticated sensors, usually acting in combination with certain chemical, biological or physical devices. Expert descriptions, on the other hand, are ratings on an ordinal scale of different aspects of products related to the taste, odor, color, etc. of such products. Here we must assume that a rating of “7” (in say, texture) means the same for a given expert in every product, though not necessarily for every expert.

The second kind of panel is made up of untrained consumers who are asked to rate their degree of acceptance of the tested products on a scale. Therefore, we use matrices to relate product descriptions with consumer ratings. Here, each row represents a product rated by a consumer in a given tasting *session*. Each product is described by a vector whose components are expert assessments and other variables obtained from chemical, biological or physical analysis.

In order to acquire some knowledge from this kind of data, Artificial Intelligence offers different possibilities from the field of Machine Learning. For instance, supervised inductive learning deals with sets of *training examples* that contain pairs of inputs and the attached outputs of a function that has to be found. The inputs are described by a set of attribute values, while the outputs are in fact another attribute of the examples called *class*. Its type determines the approach and even the name of the learning task. Regression is used when the class is a continuous number and categorical classification is employed when the class or output of the training examples is one of a finite set of symbolic categories. Notice that in this paper we use training in this Machine Learning sense; that is, training sets are the inputs of Machine Learning algorithms.

Our aim in this paper is to learn consumer preferences. Here the training material can be expressed as in regression problems: the description of each object is then followed by a number that assesses the degree of consumer satisfaction. In symbols, if E is the set of food samples, the training set will be

$$T_r = \{(\mathbf{x}, r(\mathbf{x})) : \mathbf{x} \in E\}, \quad (\mathbf{1})$$

where $r(\mathbf{x})$ stands for the rating given to sample $\mathbf{x}$. Thus, if we have a vectorial way to describe the objects in E, we can try to use regression from T_r to induce a function that maps object descriptions into ratings. However, this is not a reliable way of capturing people's preferences. The reason is that tasting sessions are not included in these models. In the next section, we shall present an in-depth discussion of the importance of considering sessions of each rating and consumer (Joachims, 2002; Bahamonde et al. 2004; Díez et al. 2004).

2.1. Regression and preference judgments

The consequence of the batch effect, as defined above, is that ratings included in the training set T_r (Equation 1) are not reliable, unless we consider explicitly the sessions. However, we could think that this effect is solely another source of noise, like many others that appear when dealing with real world data. In this case, the repercussions on model constructions are so profound that they invalidate the use of any regression method on T_r. We shall present experimental evidence of this fact at the end of the paper. For now, however, let us illustrate how regression can mislead the learning process with the graphs in Figure 1 (Díez et al. 2004).

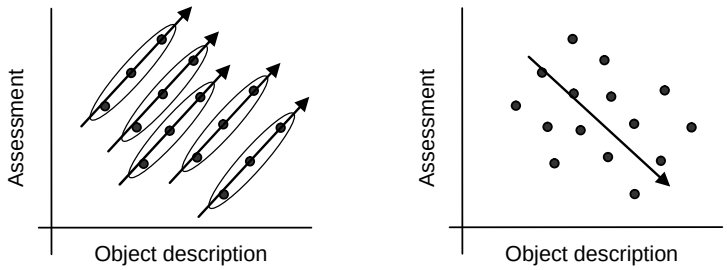

Figure 1.- The batch effect prevents the use of regression

Let us assume that consumers have to rate objects whose description may be given by a single number. In this case, each rating may be represented as a two dimensional point on a coordinate plane, as in Figure 1. If three samples are rated in each tasting sessions and if consumers are affected by the batch effect, then the ratings of 5 tasting sessions might be those represented on the left-hand side of Figure 1, where an ellipse surrounds the data of each session. The tasting session may be attended by the same consumer or by different consumers with different scales in mind.

In any case, notice that the message transmitted by consumers in each session is that the product is more preferable as the description is given by a higher number: *the more, the better*. However, if we do not take into account the sessions and join all ratings, as on the right-hand side of Figure 1, then the resulting cloud of points suggests that the evaluators are expressing that *the more, the worse*. The trend function found by linear regression now has a negative slope.

The conclusion that may be drawn from this example is that it is necessary to consider the rating sessions explicitly. Thus, the training set T_r (Equation 1) has to be upgraded to a more general setting which will be used in the rest of this paper. Hence, following Herbrich, Graepel, and Obermayer (2000) and Joachims (2002), we shall separate the ratings given at each tasting session. We shall then create a *preference judgment* set

$$PJ = \{(\mathbf{u}, \mathbf{v}) : \mathbf{u}, \mathbf{v} \in E, r(\mathbf{u}) > r(\mathbf{v}), \text{session}(\mathbf{u}) = \text{session}(\mathbf{v})\}. \quad (\mathbf{2})$$

In other words, the training material for learning algorithms will be the set of all pairs (**u**, **v**) of objects in E as if they were presented in the same session and the rating of the first object were higher than that of the second.

Given that the set of Equation 2 will be the input for inducing a useful model, we can try to acquire preference judgments directly from consumers instead of ratings. As pointed out in (Cohen, Shapire, and Singer, 1999; Buck, Wakeling, Greenhoff, and Hasted, 2001), obtaining preference information may be easier and more natural than obtaining ratings. Moreover, if we represent consumer ratings by preference judgment pairs, we no longer need to assume that a rating of "7" means the same thing to every consumer in every session (Cohen et al., 1999).

Once we have collected a dataset PJ of preference judgments, we can try to learn a membership function to predict whether a given ordered pair (**u**, **v**) of objects is suitable or not for inclusion in PJ; in other words, a two-argument function pref(**u**, **v**) that returns a numerical measure of how certain it is that **u** should be ranked before **v**.

The idea is that preference judgments provide positive and *negative* (reversing the order) examples of consumer preferences. It is then possible to learn a binary classification from PJ, which is a way of learning to order food products according to consumer preferences.

However, there is another approach to capturing the preference criteria of consumers. This consists in learning a real *preference* or *ranking function* **f** defined from the space of food products in such a way that **f(u) > f(v)** whenever **u** is preferable to **v**; i.e. when (**u, v**) $\in$ PJ. The advantage of this approach is that the functions so learned can be used to assess products coherently with consumer preferences expressed in PJ. Although a preference function **f** can be used to order a set of products, the knowledge of **f** itself is usually the most valuable reward. This functional approach to learning preferences was followed in several other studies (Tesauro, 1989; Herbrich et al., 2000; Joachims, 2002; Bahamonde et al. 2004).

2.2. Class separation and learning preferences

2.2.1 Linear approach

Before presenting the general learning method, let us first consider a simplified case. Let us thus assume that food products can be faithfully represented by real vectors; then the set E of food samples may be considered as a subset of $\mathbf{R}^d$. To learn from a dataset PJ of preference judgments, we shall try to find a real *ranking* (*or preference*) *function* $\mathbf{f}: \mathbf{R}^d \rightarrow \mathbf{R}$ that maximizes the probability of having $\mathbf{f}(\mathbf{u}) > \mathbf{f}(\mathbf{v})$ whenever **u** is preferable to **v**. If we *now* restrict the hypothesis space to linear functions, PJ gives rise to a set of constraints

$$(\mathbf{u},\mathbf{v}) \in PJ \quad \Leftrightarrow \quad \mathbf{f}(\mathbf{u}) > \mathbf{f}(\mathbf{v}) \quad \Leftrightarrow \quad \mathbf{f}(\mathbf{u}-\mathbf{v}) > 0 \quad \Leftrightarrow \quad \mathbf{f}(\mathbf{v}-\mathbf{u}) < 0 \qquad (\mathbf{3})$$

Therefore, ranking functions can be learned using a binary classification algorithm able to discriminate the class according to the sign returned, as happens with Support Vector Machines (SVM) (Vapnik, 1998; Schölkopf, Smola, 2002; Shawe-Taylor, Cristianini, 2004). Notice that the training set is

$$T = \{(\mathbf{u} - \mathbf{v}; +1), (\mathbf{v} - \mathbf{u}; -1): (\mathbf{u},\mathbf{v}) \in PJ\}. \qquad (\mathbf{4})$$

The learned function passes through the origin of coordinates and is thus defined by

$$\mathbf{f(z)} = \langle \mathbf{w}, \mathbf{z} \rangle = \sum_{j=1}^{d} w_j z_j \qquad (\mathbf{5})$$

where **w** is a weight vector, and $\langle \mathbf{w},\mathbf{z} \rangle$ stands for the scalar product of **w** and **z**. The return of **f** on an object representation **z** can be thought of as the *assessment* of **z** in the sense that **f**(**z**) will be used to predict preferences between **z** and other products. On the other hand, the weight vector **w** is the director vector of a hyperplane ($\langle \mathbf{w},\mathbf{x} \rangle = 0$) called the *assessment hyperplane*. From a geometrical point of view, the distance (in

the direction of **w**) from the hyperplane to each object is proportional to the value returned by **f**. In fact (see Figure 2),

$$\text{distance}((\langle \mathbf{w}, \mathbf{x} \rangle = 0; \mathbf{z}) = \frac{\langle \mathbf{w}, \mathbf{z} \rangle}{\|\mathbf{w}\|} = \frac{\mathbf{f}(\mathbf{z})}{\|\mathbf{w}\|} \quad (\mathbf{6})$$

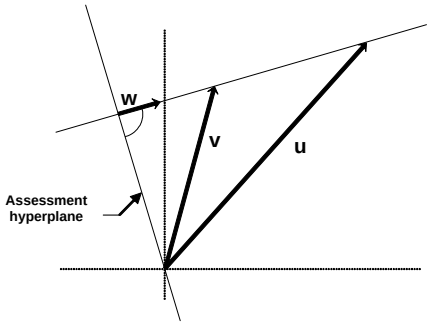

Figure 2.- The assessment of each object is proportional to the distance of the vector that represents it to the (assessment) hyperplane perpendicular to **w**. In the picture, **u** is preferable to **v**, since it is farther from this hyperplane.

2.2.2 An illustrative example

To illustrate the last subsection, let us consider a simple case where food products are described by two dimensional vectors. The first dimension represents a percentage (p), in which higher values are more appreciated by consumers. The second dimension represents the presence of a typical defect (d), the value of which consumers would therefore like to reduce as much as possible. However, what is the relative importance of these two factors in consumer preferences? According to the method described above, we have to collect a set of preference judgments; for instance,

PJ = {[(100%, 0); (100%, 4)], [(75%, 1); (85%, 10)], …}.

In other words, PJ reports, among other things, that 100% with no defects is better than 100% with 4 defects, and that consumers can accept 75% instead of 85% if the number of defects is only 1 instead of 10. The training set of Equation 4 is then built with the differences of product descriptions; in this case, we obtain

T = {((0, -4); +1), ((0, 4); -1), ((-10, -9); +1), ((10, 9); -1), …}.

Trained with T, a linear SVM finds the following linear function

assessment(p, d) = 0.41 * p – 0.94 * d.

Notice that p values are weighted by a positive value (0.41), while d is weighted negatively (-0.94). The whole values of T and the geometric representation of the assessment function are depicted in Figure 3. Let us stress that the geometric role of the hyperplane

$$assessment(p, d) = 0.41 * p – 0.94 * d = 0$$

is to separate the positive and negative differences of T. In other words, the assessment function is in fact a classification device.

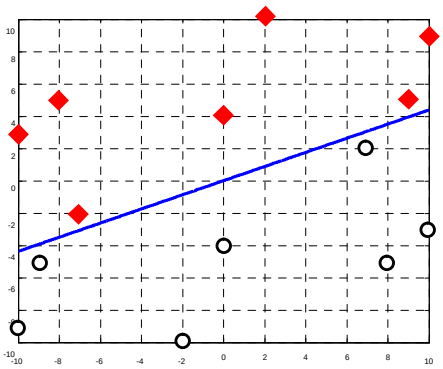

Figure 3.- The training set and the assessment hyperplane that separates the positive differences (represented by circles) from the negative ones (represented by diamonds)

2.2.3 Nonlinear preference or ranking functions

Of course, it will not always be possible to find a linear function able to separate the positive from the negative differences of the training set T (Equation 4). In terms of preferences, the assessment of an object will not always be proportional to the components of a vectorial description. It is not true that the more (or the less), the better; for instance, the amount of sugar or salt in our favorite foods usually has an optimal point and any increase or decrease from that point of equilibrium is frequently rejected. To consider nonlinear separation and preference functions, it is possible to use a *kernel trick* (Herbrich et al., 2000).

To introduce the concept of the kernel function, let us consider a mapping ϕ from the set of food products E into a subset of $\mathbf{R}^h$. In symbols, ϕ: E → $\mathbf{R}^h$. This aim of this mapping is to provide a new vectorial representation in (usually) a higher dimensional space called the *feature space*. Typically, ϕ may be a polynomial representation. In the example in the previous subsection, for instance, if we wish to represent food samples by 2nd degree polynomials in variables p and d, the mapping may be given by:

$$\phi(p,d) = \left(p^2, \sqrt{2}p, \sqrt{2}pd, d^2, \sqrt{2}d, 1\right) \in \mathbf{R}^6 . \qquad (7)$$

If we now repeat the method described in Section 4.1 using $\phi(\mathbf{x})$ instead of $\mathbf{x}$, we can obtain a linear function in the products of at the most two variables as the assessment function. In other words, the assessment function will be a 2nd degree polynomial in the variables that describe food products.

In practice, however, it is not possible to write full polynomial expansions explicitly. The reason is that the dimensions of the feature space are too great. For instance, if E has 150 components, the feature space of 3rd degree polynomials has 585,276 dimensions.

Fortunately, SVM do not use the components of the input vectors; they only use their scalar products. Therefore, the components of expansions of $\phi(\mathbf{x})$ are not necessary if we are able to compute scalar products of the form

$$K(\mathbf{x}, \mathbf{y}) = \langle \phi(\mathbf{x}), \phi(\mathbf{y}) \rangle. \quad (\mathbf{8})$$

In the case of g-degree polynomials, it is not difficult to see that

$$K(\mathbf{x}, \mathbf{y}) = \langle \phi(\mathbf{x}), \phi(\mathbf{y}) \rangle = (\langle \mathbf{x}, \mathbf{y} \rangle + 1)^g. \quad (\mathbf{9})$$

Functions defined like K in Equations 8 and 9 are called *kernel functions*. The ability of SVM to handle feature space representations through kernel functions is called the *kernel trick*. For more information on kernels, see (Schölkopf and Smola, 2002; Shawe-Taylor, and Cristianini, 2004).

Returning to learning consumer preferences, let us assume that the space E of food products can be mapped onto a feature space $\mathcal{F}$ using ϕ: $E \rightarrow \mathcal{F}$. Then, using the procedure in 2.2.1, in order to learn a ranking or preference function, we only have to find a *linear* separation in $\mathcal{F}$ for the training set

$$T_{\mathcal{F}} = \{(\phi(\mathbf{x}^{(1)}) - \phi(\mathbf{x}^{(2)}); +1), (\phi(\mathbf{x}^{(2)}) - \phi(\mathbf{x}^{(1)}); -1): (\mathbf{x}^{(1)}, \mathbf{x}^{(2)}) \in PJ \}. \quad (\mathbf{10})$$

Therefore, the input space that we need is in fact the product of E by itself,

$$T_{E \times E} = \{(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}; +1), (\mathbf{x}^{(2)}, \mathbf{x}^{(1)}; -1): (\mathbf{x}^{(1)}, \mathbf{x}^{(2)}) \in PJ \}. \quad (\mathbf{11})$$

and is mapped onto the feature space $\mathcal{F}$ using

$$\Psi: E \times E \rightarrow \mathcal{F}, \ \Psi(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}) = \phi(\mathbf{x}^{(1)}) - \phi(\mathbf{x}^{(2)}). \quad (\mathbf{12})$$

Hence, the associated kernel to this transformation is

$$\begin{aligned} \mathbf{K}(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}; \mathbf{x}^{(3)}, \mathbf{x}^{(4)}) &= \langle \Psi(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}), \Psi(\mathbf{x}^{(3)}, \mathbf{x}^{(4)}) \rangle = \langle (\phi(\mathbf{x}^{(1)}) - \phi(\mathbf{x}^{(2)})), (\phi(\mathbf{x}^{(3)}) - \phi(\mathbf{x}^{(4)})) \rangle \\ &= \langle \phi(\mathbf{x}^{(1)}), \phi(\mathbf{x}^{(3)}) \rangle - \langle \phi(\mathbf{x}^{(1)}), \phi(\mathbf{x}^{(4)}) \rangle - \langle \phi(\mathbf{x}^{(2)}), \phi(\mathbf{x}^{(3)}) \rangle + \langle \phi(\mathbf{x}^{(2)}), \phi(\mathbf{x}^{(4)}) \rangle \\ &= K(\mathbf{x}^{(1)}, \mathbf{x}^{(3)}) - K(\mathbf{x}^{(1)}, \mathbf{x}^{(4)}) - K(\mathbf{x}^{(2)}, \mathbf{x}^{(3)}) + K(\mathbf{x}^{(2)}, \mathbf{x}^{(4)}) \end{aligned} \quad (\mathbf{13})$$

where $K(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$ is the kernel associated with the transformation ϕ of the *individual* objects. The kernel $\mathbf{K}$ thus built is called Herbrich's kernel attached to K.

The separation function induced by a classification SVM from $T_{E \times E}$ with kernel $\mathbf{K}$ will be a function $\mathbf{F}$: $\mathbf{E} \times \mathbf{E} \rightarrow \mathbf{R}$ of the form

$$\mathbf{F}(\mathbf{x}, \mathbf{y}) = \sum_{s \in S} \alpha_s z_s \langle \phi(\mathbf{x}_s^{(1)}) - \phi(\mathbf{x}_s^{(2)}), \phi(\mathbf{x}) - \phi(\mathbf{y}) \rangle = \sum_{s \in S} \alpha_s z_s \mathbf{K}(\mathbf{x}_s^{(1)}, \mathbf{x}_s^{(2)}, \mathbf{x}, \mathbf{y}) \quad (\mathbf{14})$$

where $\mathbf{S}$ is a set of indexes for the so called support vectors, and $\mathbf{z}_s$ are the classes. This function has an important property that is an immediate consequence of the kernel definition:

$$\mathbf{F}(\mathbf{x},0) > \mathbf{F}(\mathbf{y},0) \quad \Leftrightarrow \quad \mathbf{F}(\mathbf{x},\mathbf{y}) > 0 \tag{15}$$

Thus, we define the following assessment function:

$$\mathbf{f}: \mathbf{E} \rightarrow \mathbf{R}, \qquad \mathbf{f}(\mathbf{x}) = \mathbf{F}(\mathbf{x},0). \tag{16}$$

It is trivial to see that $\mathbf{f}$ is as coherent with the preference judgments of PJ as $\mathbf{F}$ is a separating function for $T_{\mathbf{E}\times\mathbf{E}}$. Moreover, in the case of using the linear kernel as K, $\mathbf{f}$ coincides with the function defined in Equation 5.
The expression of $\mathbf{f}$ may be simplified, given that it is possible to skip a constant term from $\mathbf{F}(\mathbf{x},0)$; this constant is 0 in the linear case. Thus, in practice, the general assessment function $\mathbf{f}$ is given by

$$\mathbf{f}(x) = \sum_{s\in S} \alpha_s z_s \left\langle \phi(\mathbf{x}_s^{(1)}) - \phi(\mathbf{x}_s^{(2)}), \phi(\mathbf{x}) \right\rangle = \sum_{s\in S} \alpha_s z_s \left(K(\mathbf{x}_s^{(1)}, \mathbf{x}) - K(\mathbf{x}_s^{(2)}, \mathbf{x}) \right) \tag{17}$$

Notice that when K is the polynomial kernel (Equation 9), the assessment function of Equation 17 is also a polynomial. With the use of polynomials and other kernels, the assessment functions that can be obtained using this method cover all useful nonlinear possibilities.

3. Empirical study

In this section we report the scores achieved with two real world datasets from sensory panels. The first one covers ratings of beef, while the second dataset deals with a slightly alcoholic drink, traditional cider from Asturias (Spain). The aim is to show the consequences of the batch effect: how it negatively affects regression methods and how preference or ranking functions can overcome this effect.

3.1 Methodology

There are a number of possible approaches to combining the preferences of different consumers. A first attempt may consist in joining together the preference judgments of all consumers to thus form one training set (recall Equations 2 and 4). However, disagreements between individual consumers may be too numerous, which would damage the overall learning process. Notice that the aim is to model the general opinion of consumers of a given food product.

We should therefore somehow aggregate consumer preferences. If we have some knowledge about the possible existence of differentiated market segments, the first step should be to ascertain the clusters of consumers that represent these segments

(Westad, Hersleth, and Lea, 2004; Díez et al. 2005; Díez et al. 2006) to then proceed to learn from the union of their preference judgments. However, if there is no evidence in favor of significant divergences, a wise way of modeling the general opinion may be to first summarize the opinions of consumers.

In the empirical study reported here, we computed the mean of the ratings obtained by each sample of meat or cider for each session. This was the class used for regression algorithms. On the other hand, taking into account the sessions, a joint PJ dataset was computed using Equations (2, 4, 11). See (Luaces et al. 2004; Del Coz et al. 2004; Díez et al. 2004; Díez et al. 2005; Díez et al. 2006) for a detailed discussion of these options.

3.1.1 Consumer ratings of beef

The first dataset comes from a study carried out to determine the features that entail consumer acceptance of beef from seven Spanish breeds (Gil et al., 2001; Sañudo et al. 2004). Each piece of meat was described by: the weight of the animal, aging time, breed, 6 physical features describing its texture and 12 sensory characteristics rated by 11 different experts (132 ratings). Given that breed was represented by 7 Boolean features, the whole description of each piece of meat uses 147 features.

In each testing session, 4 or 5 pieces of meat were tested and a group of consumers were asked to rate (on a scale of 1 to 10 points) only three different qualities: tenderness, flavor and acceptance. These three data sets have over 2420 preference judgments. The number of consumers involved in this panel was 393 (Sañudo et al. 2004).

3.1.2 Panelists' ratings of cider

In the dataset on traditional cider from Asturias (Picinelli et al., 2000), the description of each cider was given solely by 64 chemical and physical features, without any sensory description given by trained experts. In fact, the consumers here were a set of 14 candidates to become experts, and the rating sessions (of 3, 4 or 5 ciders) were carried out during the training and selection stage. These potential experts were asked to rate a high number of qualities of ciders: bouquet, color, acidity, bitterness, 4 additional visual aspects and 3 more flavor-related aspects. The scales used for these purposes ranged from 1-3 to 1-9. We thus have 11 qualities of cider, i.e. 11 different datasets of over 225 preference judgments computed using again Equations (2, 4, 11).

Notice that these datasets do not come strictly from consumers' preferences, since the potential panelists were doing a descriptive test. However, the aim of the empirical study described in this section is to show the influence of the batch effect, and these datasets, as we will see below, are contaminated by this effect too. So, we included these datasets in the comparative reported in this paper.

3.1.3 Comparison of models obtained by regression and preference learning

We performed a comparison between the scores achieved by preference approaches and those obtained by regression methods. Note that the argument used in subsection 2.1 to discard regression in favor of preference learning was mainly graphical. What we shall show in this section is that the batch effect has disastrous consequences in regression in real world datasets.

Learning from the datasets described above, regression methods induce functions aimed at predicting numerical ratings of consumers from the description of food products. To consider different options, we experimented with a simple linear regression and with a well-reputed nonlinear regression algorithm: Cubist, a commercial product from *RuleQuest Research*.

To interpret regression results, we used the *relative mean absolute deviation* (rmad), which is computed from the *mean absolute distance* or deviation, mad, of the function **f** learned by the regression method. In symbols, if T_t is a test set,

$$\mathrm{mad}(\mathbf{f}) = \frac{1}{|T_t|}\sum_{\mathbf{x}\in T_t}|\mathbf{f}(\mathbf{x}) - \mathrm{class}(\mathbf{x})| \qquad \mathrm{rmad}(\mathbf{f}) = 100\frac{\mathrm{mad}(\mathbf{f})}{\mathrm{mad}(\mathbf{mean})} \qquad (\mathbf{18})$$

where **mean** is the constant predictor that returns the mean value in all cases, and class(**x**) is the class of a case **x**. The idea of rmad is to measure the relative improvement of a regressor with respect to a trivial baseline. Hence, if the rmad of a function **f** is 100, the average absolute difference between predicted and real values is the same as that given by the constant predictor, which always predicts the mean value of the class. Conversely, an rmad(**f**) of 0 means that the prediction of **f** is always perfect; we assume that **mean** is never perfect, i.e. we assume that the class is not constant.

On the other hand, to enable a fair comparison between regression and preference learning approaches, we also tested regression models on preference judgment test sets, calculating their misclassifications: the percentage of preference judgments (**u**, **v**) where **f**(**u**) > **f**(**v**) for a model **f**. In symbols, if PJ_t is a test set,

$$\mathrm{mis}(\mathbf{f}) = \frac{100}{|PJ_t|}\sum_{(u,v)\in PJ_t}\mathbf{1}(\mathbf{f}(u) > \mathbf{f}(v)) \qquad (\mathbf{19})$$

where **1**(predicate) is the function that returns 1 if and only if the predicate is true.

To estimate rmad and mis of all models on unseen cases, we used 10 fold cross-validation. This is a statistical method frequently used in Machine Learning. The method proceeds as follows. First, it randomly breaks a dataset T into 10 partitions; then, for each partition p, it learns a model from all the data not in that part (T-p), and evaluates the model so found on p. Finally, cross-validation returns the mean evaluation through all 10 parts.

3.2 Results

The second and third columns of Tables 1 and 2 report the rmad achieved by least squares linear regression and by a nonlinear regression algorithm such as Cubist. The scores show that regression methods are unable to learn any useful knowledge: their rmad is above 100 in almost all cases, i.e. the **mean** predictor usually performs better.

Additionally, in order to test the performance of the preference learning approach, we computed the sets of preference judgments using Equations (2, 4, 11). To learn from each PJ dataset, we used SVM^{light} (Joachims, 1998) with linear and polynomial kernels as described in subsections 2.2.1 and 2.2.3. In this case, the percentages of misclassifications (Equation 19) are reported in Tables 1 and 2 in columns labelled by 'SVM lin.' and 'SVM Poly'.

The performance of regression models, whether linear or nonlinear, is very poor. Even when these regression models are tested on preference judgment sets, the percentage of misclassifications is over 40%, clearly higher than the percentage obtained when using the preference learning approach. Note that when we are dealing with binary classification, the worst score has a mis estimation of 50%. SVM-based methods, on the other hand, can reduce the mis of regression to reach an average of nearly 30% with a linear kernel (Equations 4 and 5) and by nearly 20% if the kernel is a 2nd degree polynomial (Equations 9-14 and 16-17). Notice that these results demonstrate on real cases the benefits of using preference learning instead of simply regression the data. The predictive quality of regression is very poor, and then it is not possible to obtain any useful knowledge from it.

The rationale underlying the improvement when using nonlinear kernels may be explained by bearing in mind that the positive appreciation of food products usually requires equilibrium of their components, and the increase or decrease of any value from that point is frequently rejected.

Table 1.- Beef. 10-fold cross-validation rmad and mis achieved using regression (linear and nonlinear) and preference learning with SVM with linear and polynomial kernels. The 393 consumers rated their preferences on a scale of 1-10

	Regression		Preferences			
Dataset	Linear	Nonlin.	SVM lin.	SVM Poly.	Linear	Nonlin.
Tenderness	96.3%	97.8%	29.6%	19.4%	41.5%	43.1%
Flavor	99.3%	103.4%	32.7%	23.8%	43.8%	46.5%
Acceptance	94.0%	97.2%	31.9%	22.1%	38.4%	40.2%
Average	**96.51%**	**99.49%**	**31.39%**	**21.79%**	**41.24%**	**43.27%**

Table 2.- Cider from Asturias. 10-fold cross-validation rmad and mis achieved using regression (linear and nonlinear) and preference learning with SVM with linear and polynomial kernels. The 14 assessors rated cider features on scales ranged from 1-3 to 1-9

	Regression	Preferences

Dataset	Linear	Nonlin.	SVM lin.	SVM Poly.	Linear	Nonlin.
Acidity	103.0%	109.4%	29.9%	18.0%	40.0%	42.4%
Bitterness	105.8%	111.9%	30.5%	23.1%	56.0%	47.4%
Flavor-1	105.3%	111.7%	27.2%	17.1%	42.4%	44.3%
Flavor-2	107.2%	116.0%	28.6%	17.9%	45.6%	45.0%
Flavor-3	110.3%	107.7%	33.6%	17.7%	43.8%	41.8%
Bouquet	104.0%	110.2%	26.4%	21.0%	43.5%	42.7%
Color	98.4%	109.9%	26.1%	17.8%	41.3%	43.4%
Visual-1	103.2%	113.0%	25.9%	13.4%	41.7%	43.1%
Visual-2	102.3%	112.0%	34.0%	20.0%	43.8%	45.7%
Visual-3	107.2%	120.5%	25.3%	20.6%	45.6%	49.3%
Visual-4	98.7%	97.2%	23.0%	14.0%	36.5%	38.2%
Average	**104.12%**	**110.87%**	**28.24%**	**18.23%**	**43.65%**	**43.92%**

4. Conclusions and implications

We have reported a method for modeling consumer preferences when we have a dataset gathering consumer ratings on some kind of food product. The method is especially devised for this context and has been tested in a couple of real world datasets reported in the previous section. We stress the relevance of taking into account the tasting sessions in which the ratings were obtained. The reason is that consumer ratings tend to have only a relative meaning with respect to each session; they cannot be taken as absolute values. This implies that regression methods fail when they try to capture consumer preferences.

The method proposed in this paper uses an Artificial Intelligence tool, Support Vector Machines (SVM). These *machines*, employed in classification tasks, are algorithms based on the optimization of the margin between classes by means of well-known and well-founded techniques of Quadratic Programming, see (Vapnik, 1998; Schölkopf, Smola, 2002; Shawe-Taylor, Cristianini, 2004). Using the so-called *kernel trick*, these SVM can build linear and nonlinear functions able to separate classes in a binary classification dataset. To handle consumer preferences, the core idea is to consider the **u**-**v** differences of vectorial representations of food products. These differences may be classified as *positive*, when **u** represents a product that should be ranked before **v**, and *negative* otherwise.

Roughly speaking, the method assigns a (ranking or preference) function to a group of consumers. What is noteworthy is that those functions can be mathematically manipulated for several purposes; for instance, to discover guides to improve consumer acceptance of products. Furthermore, preference functions can be compared and then used to discover groups or clusters of people with closely related tastes; in other words, market segments with differentiated requirements from a kind of food product. This possibility was explored in (Díez et al. 2005; Díez et al. 2006).

The use of SVM in Food Science opens new possibilities since they can handle products described by thousands of attributes; for instances, food products can be defined by gene expression data from DNA microarrays. Moreover, the data can be represented in high dimensional spaces with clear sensorial interpretations, as happens

with polynomial kernels. But SVM are only one part of the so-called kernel methods (Schölkopf, Smola, 2002; Shawe-Taylor, Cristianini, 2004), where there are *kenelized* (that is nonlinear) versions of Principal Component Analysis (KPCA), with an evident applications in Food Science that will appear in the next few years.

The software used in the experiments reported in this paper is publicly available for almost any platform including desktop PCs, and it is implemented in C[1] with (or without) an interface to MatLab[2]. Therefore, SVM software can be used without any special requirements, and it can be extended and adapted to cope successfully with new an improved sensory applications. On the other hand, the sensory data used in the experiments, as described in section 2, has a format familiar for Food Scientists; in the Annex we reproduce a part of the database used for modeling beef consumer preferences.

Acknowledgements

The authors wish to thank: Carlos Sañudo and the Animal Production team at the Universidad de Zaragoza for providing us with the sensory quality dataset of beef (Gil et al., 2001; Sañudo et al. 2004); Anna Picinelli and her colleagues at the SERIDA Cider group (Asturias) for allowing us to use their cider datasets (Picinelli et al., 2000); Thorsten Joachims for his SVM^{light} (Joachims, 1998) and the authors of Spider (Weston et al.), a MatLab toolbox that includes kernel based algorithms. Both SVM^{light} and Spider were used in the experiments reported in this paper; and one anonymous reviewer whose comments improved the manuscript. The authors were supported in part under grant TIN2005-08288 by MEC (Ministerio de Educación y Ciencia of Spain).

Referentes

Bahamonde, A.; Bayón, G. F.; Díez, J.; Quevedo, J. R.; Luaces, O.; del Coz, J. J.; Alonso, J.; Goyache, F. (2004). Feature subset selection for learning preferences: a case study. Proceedings of the 21st International Conference on Machine Learning, ICML 2004, Banff, Canada, pp. 49-56.

Buck, D., Wakeling, I., Greenhoff, K., and Hasted, A. (2001). Predicting paired preferences from sensory data. Food quality and preference 12:481-487.

[1] http://svmlight.joachims.org/

[2] http://www.kyb.tuebingen.mpg.de/bs/people/spider/

Cohen, W.W., Shapire, R.E., and Singer, Y. (1999). Learning to order things. Journal of Artificial Intelligence Research 10:243-270.

Del Coz, J. J.; Bayón, G. F.; Díez, J.; Luaces, O.; Bahamonde, A.; Sañudo, C. (2004). Trait selection for assessing beef meat quality using nonlinear SVM. Proceedings of the 18th Annual Conference on Neural Information Processing Systems (NIPS 2004). Vancouver, British Columbia, Canada, pp. 321-328.

Díez, J.; Bayón, G.F.; Quevedo, J. R.; del Coz, J. J.; Luaces, O.; Alonso, J.; Bahamonde, A. (2004). Discovering relevancies in very difficult regression problems: applications to sensory data analysis. In Proceedings of the European Conference on Artificial Intelligence (ECAI '04), Valencia, Spain, pp. 993-994.

Díez, J.; del Coz, J. J.; Sañudo, C.; Albertí, P.; Bahamonde, A. (2005). A Kernel Based Method for Discovering Market Segments in Beef Meat. Proceedings of the 9th European Conference on Principles and Practice of Knowledge Discovery in Databases, ECML/PKDD'2005, Porto, Portugal, pp. 462-469.

Díez, J.; del Coz, J.J.; Bahamonde, A.; Sañudo, C.; Olleta, J.L.; Macie, S.; Campo, M.M.; Panea, B.; and Albertí, P. (2006). Identifying market segments in beef: Breed, slaughter weight and ageing time implications. *Meat Science*, in Press, available online.

Ellekjær, M. R.; Ilseng, M. A.; and Næs, T. (1996). A case study of the use of experimental design and multivariate analysis in product improvement. *Food Quality and Preference*, 7(1), 29-36.

Gil, M., Serra, X., Gispert, M., Oliver, M., Sañudo, C., Panea, B., Olleta, J., Campo, M., Oliván, M., Osoro, K., García-Cachan, M., Cruz-Sagredo, R., Izquierdo, M., Espejo, M., Martín, M., Piedrafita, J.: The effect of breed-production systems on the myosin heavy chain 1, the biochemical characteristics and the colour variables of longissimus thoracis from seven Spanish beef cattle breeds. Meat Science 58 (2001) 181–188

Goyache, F., Bahamonde, A., Alonso, J., López, S., del Coz J.J., Quevedo, J.R., Ranilla, J., Luaces, O., Alvarez, I., Royo, L., and Díez, J. (2001). The usefulness of artificial intelligence techniques to assess subjective quality of products in the food industry. Trends in Food Science and Technology 12 (10):370-381.

Herbrich, R.; Graepel, T.; and Obermayer, K.: Large margin rank boundaries for ordinal regression. In A. Smola, P. Bartlett, B. Scholkopf, and D. Schuurmans, editors, Advances in Large Margin Classifiers, 115–132. MIT Press, Cambridge, MA, (2000)

Joachims, T. (1998). Making large-scale SVM learning practical. Advances in Kernel Methods - Support Vector Learning, MIT Press, Cambridge MA. Chapter 11.

Joachims, T. (2002). Optimizing search engines using clickthrough data. In Proc. ACM Conference on Knowledge Discovery and Data Mining.

Lea, P.; Naes, T.; Rodbotten, M. (1997). Analysis of Variance for Sensory Data. John Wiley & Son, New York.

Luaces, O.; Bayón, G.F.; Quevedo, J.R.; Díez, J.; del Coz, J.J.; Bahamonde, A. (2004). Analyzing sensory data using nonlinear preference learning with feature subset selection. Proceedings of the 15th European Conference of Machine Learning, (ECML 04). Pisa, Italia, pp. 286-297.

McEwan, J.A.. Preference mapping for product optimization. In: Naes, T.; Risvik, E. (ed): "Multivariate analysis of data in sensory science", Elsevier Science B.V., 1996. Pp. 71-102

Muñoz, A. M. (2002). Sensory evaluation in quality control: an overview, new developments and future opportunities. Food Quality and Preference 13(6): 329-339

Murray, J.M., Delahunty, C.M., and Baxter, I.A. (2001). Descriptive sensory analysis: Past, present and future. Food Research International 34:461-471.

Picinelli, A., Suárez, B., Moreno, J., Rodríguez, R., Caso-García, L., Mangas, J.: Chemical characterization of Asturian cider. Journal of Agricultural and Food Chemistry 48 (2000) 3997–4002

Sañudo, C.; Macie, E.S.; Olleta, J.L.; Villarroel, M.; Panea, B.; Albertí, P.. The effects of slaughter weight, breed type and ageing time on beef meat quality using two different texture devices. Meat Science, 66 (2004), 925–932

Schölkopf, B.; Smola, A. J. (2002). Learning with kernels. MIT Press.

Shawe-Taylor, J.; Cristianini, N. (2004). Kernel methods for pattern analysis. Cambridge University Press.

Tenenhaus, M.; Pagès, J.; Ambroisine, L.; and Guinot, C. (2005). PLS methodology to study relationships between hedonic judgements and product characteristics. *Food Quality and Preference*, 16(4), 315-325.

Tesauro, G. (1989). Connectionist learning of expert preferences by comparison training. In Proc. Neural Information and Processing Systems. pp. 99-106.

Van Kleef, E.; Van Trijp, H.C.M.; and Luning, P. (2006). Internal versus external preference analysis: An exploratory study on end-user evaluation. *Food Quality and Preference*, 17(5), 387-399.

Vapnik, V. (1998). Statistical learning theory. John Wiley, New York.

Westad, F.; Hersleth, M.; and Lea, P. (2004). Strategies for consumer segmentation with applications on preference data. *Food Quality and Preference*, 15(7-8), 681-687

Weston, J.; Elisseeff, A.; BakIr, G.; Sinz, F.: SPIDER: object-orientated machine learning library. http://www.kyb.tuebingen.mpg.de/bs/people/spider/

Annex

Table 3.- Sensory data collected from panels of experts and consumers. Each product is described by expert assessments in addition to other chemical, biological or physical analysis outputs

Animal Info.

Animal	Aging	Weight	…	k80
812	7	Heavy	…	21.73
816	21	Light	…	34.58
833	1	Heavy	…	59.68
845	21	Light	…	24.27
…				

Consumers' Panel

Animal	Cons.	Session	Tender.	Flavor	Accep.
812	24	1	4	6.5	5
845	24	1	5.5	7.5	7
812	15	2	6	7	6.5
845	15	2	4.5	6	5.5
…					

Experts' Panel

Animal	Expert	Fibrosis	…	Odor
812	1	5	…	4
845	1	3.5	…	6.5
812	11	4.5	…	7
845	11	5.5	…	3
…				

Overall Acceptance Dataset

Expert sensory appreciations							Physical & Biological attributes			Consumer preferences		
Expert-1				Expert-11								
Fibro.	…	Odor	…	Fibro.	…	Odor	Aging	…	k80	Session	Consumer	Rating
5	…	4	…	4.5	…	7	7	…	21.73	1	24	5
3.5	…	6.5	…	5.5	…	3	21	…	24.27	1	24	7
5	…	4	…	4.5	…	7	7	…	21.73	2	15	6.5
3.5	…	6.5	…	5.5	…	3	21	…	24.27	2	15	5.5
	…		…		…					…	…	…