University of Castilla-La Mancha

A publication of the

Department of Computer Science

Mining the ESROM: A study of breeding value prediction in Manchego sheep by means of classification techniques plus attribute selection and construction

by

Jose A. Gámez

Technical Report **#DI-05-01-3** January 2005

DEPARTAMENTO DE INFORMÁTICA
ESCUELA POLITÉCNICA SUPERIOR
UNIVERSIDAD DE CASTILLA-LA MANCHA
Campus Universitario s/n
Albacete - 02071 - Spain
Phone +34.967.599200, Fax +34.967.599224

Mining the ESROM: A study of breeding value prediction in Manchego sheep by means of classification techniques plus attribute selection and construction

Jose A. Gámez

January 24, 2005

Abstract

Manchego sheep is the native breed in Castilla-La Mancha (a region of Spain). Its two main products are Manchego cheese and Manchego lamb, representing more than 50% of the final animal production in the region. Because of these economical implication and with the aim of improving Manchego sheep production, a selection scheme (called ESROM) based on the animal genetic merit was started fifteen years ago. One of the major points in the selection scheme is the estimation of the breeding value, and its use in flock replacements. In the ESROM scheme, the breeding value is estimated by using BLUP animal model, which is a complex method based on relating different traits by linear equations, and solving the system by simultaneously taking into account all the available information.

In this paper we study the use of data mining techniques to deal with breeding value classification. The goal of the paper is far enough of replacing BLUP in breeding value estimation, on the contrary, our goal is to learn in a supervised way from the results produced by BLUP, and to use the learned models to provide preliminary information about the breeding value of an animal. The advantages of using those models is that few information is required and the estimation can be done as soon as the data (about a few variables) is ready for a given animal, allowing to take early decisions or to delay them until a deeper study is carried out.

We start the data mining process identifying a proper data set from the whole available data. Then we use standard classification techniques combined with feature subset selection to identify good attribute subsets to be used as predictors. Attribute selection is done on the basis of filter and wrapper algorithms, and we also proposed a filter+wrapper algorithms which provide close to wrapper results with a remarkable smaller computational cost. We also show that the classifiers accuracy can be considerably improved (around a 4% on the average) by using attribute construction. Finally we discuss about some tasks performed in the ESROM scheme in relation with the obtained classification models.

Keyworkds: Manchego sheep, selection scheme, breeding value, classification algorithms, data mining, attribute selection, attribute construction.

1 Introduction

In Castilla-La Mancha (a region of Spain with more than a million and a half citizens who live in the 79.000 km^2 territory) the sheep cattle represents one of the key components in the regional economy. For an idea, according to a report of 2001 [ITAP, 2001], in Castilla-La Mancha the ovine production represents 15% of the agricultural production and more than 50% of the final animal production. Although in the region coexist several ovine breeds, is the native Manchego sheep [Gallego et al., 1994] which best fits to the natural habitat and to the extreme continental climatology of the region (cold winters, long dry summers, scarce rainfall and large daily temperature changes), and it is this natural adaptation which allow them: (1) to exploit by pasture all the resources at their disposal; and (2) to be fertile during all the year. There are two main final products from Manchego sheep: (1) Manchego cheese[1] and Manchego lamb[2]. The excellent quality of these products becomes plain by their consumption figures: Manchego lamb has increases its sales a 34% from 2002 to 2003 and Manchego cheese represents the 44.4% of the cheese commercialized in Spain, and its exports outside Spain have increased from 0.3% in 1987 to 34% in 2003 (being USA, France and Germany the main consumers).

However, not all are congratulations for Manchego sheep. Thus, due to the crisis suffered during the last years by the market of sheep meat, milk production has attained a leading role in the sheep cattle, and foreign breeds represent a menace to Manchego breed because in some cases those foreign breeds have a greater milk production (though this figure does not always means greater net profit). Being aware of this risk, several public organizations and authorities of the region have opted to the improvement of production data in Manchego sheep, specially when its potential is tremendous. To achieve this goal the Selection Scheme for Manchego Sheep (ESROM) was created in 1987.

The ESROM Selection Scheme (SS), which is similar to other selection schemes developed for other breeds, includes a series of activities whose joint purpose is the *genetic improvement of the breed with respect to the production of milk*, and is run by several organizations: AGRAMA (National Association of Manchego sheep breeders), the Regional Government of Castilla-La Mancha, CERSYRA (Regional Center for Animal Selection and Reproduction),

[1] Controlled by a Guarantee of Origin [CRDOQM, 2004] since 1984 which requires that it is make only from milk obtained of Manchego ewes raised in the region of Castilla-La Mancha (which in 2001 represents 40% of the total ewe milk in Castilla-La Mancha).

[2] Controlled by a Specific Guarantee [CRDECM, 2004] since 1995

and the Spanish Government (Ministry of Agriculture). An evidence of the SS success is the 25 extra liters produced at each lactation by the ewes obtained by artificial insemination inside of the SS.

The SS has four main tools:

1. *Genealogical ranking*. It is a register of all the ewes in a stock-farm submitted to the milk controls performed by the SS. It contains (among others) data about the genetic merit of the animal (shown by the percentile - 10%, 20%, etc, ...- in which the ewe is ranked with respect to its herd and with respect to the full census of controlled Manchego ewes).

2. *Stud catalog*. Males included in the SS having a very high genetic merit (computed from its daughters genetic merit).

3. *Milk production control*. A report of the lactation of each ewe containing all the data referred to the quantity and quality of milk produced by the ewe during the lactation, normalized to 120 days and 6% of fat (to allow comparisons).

4. *Stud market*. Market of males obtained by artificial insemination, which follows the racial standard[3] of Manchego sheep and whose mother is above the percentile 70% in the genealogical ranking. The adquisition of males in the Stud market constitutes an easy way to improve the genetic merit of a herd for those stock farmers that cannot enter in the technical part of the ESROM program (artificial insemination and lactational controls).

As we can realize, the key parameter in the SS is the estimation of animals genetic merit or *Breeding Value* (BV), because it is this value, computed by using the data about the controlled lactations, which allow us to place them in the genealogical ranking and to be entered (or not) in the stud catalog or market. Besides, the SS encourages stock breeders to select their flock replacements on the basis of the animal genetic merit. In our case, the breeding value of an animal is a numeric (real) value which represents the deviation of the animal with respect to the averaged breeding value of the Manchego ewes born in 1990 (referred to as the base year).

[3]Described in `http://www.agrama.org/prototipo.html` (in Spanish)

The estimation of the breeding value[4] is done by using the BLUP (Best Linear Unbiased Predictor) methodology, concretely the *animal model* [poner referencia] is used which is the most sophisticated method of BLUP analysis available . BLUP is a contrasted methodology that evaluates the BV of an animal by attempting to separate out the genetic factors influencing the animal merit from the non-genetic factors like feeding or management. During the computation, BLUP uses all the available information to carry out the BV estimation: lactational data about any ewe (dead or alive) which in some moment was controlled, genealogical information about the animal, its relatives and the rest of animals in its herd, and the information about all the herds which are under control by the SS. Finally, all this information is linked by means of equations and is *simultaneously* analyzed by taking into account any correlation between the different traits.

Therefore, the estimation of the breeding value by using BLUP is a complex process, that in the case of our SS is carried out each six months in a specialized center. Furthermore, the BV of an animal is a dynamic value, because it can change from a measurement to the next one due to changes in the own animal production data, due to changes in its relatives data, due to changes in its herd, etc, ...

The goal of this paper is to work on the prediction/classification of the breeding value inside the ESROM Scheme by using techniques from machine learning [Mitchell, 1997] and data mining [Fayyad et al., 1996] fields. These techniques are embraced by a broader field, called artificial intelligence or intelligent systems, whose application to agriculture has gained interest during the last years [Murase, 2000, Farkas, 2003]. Obviously, our goal it is not to replace the use of BLUP methodology, but to study the possibilities of predicting the BV of an animal by using a data-driven approach, which will use (by far) less information than BLUP and that is simpler and can be used as soon the information for a given animal is ready, without having to wait for the full cattle six-month evaluation. Furthermore, we focus our analysis in primipara ewes, because it is after the first birth and its corresponding lactation when ewes are evaluated for first time, and so, it is of interest to have, as soon as possible, an approximation of its genetic merit in order to take early decision about its inclusion or not in the production-line.

To achieve this goal we have structured the paper in eight sections apart from this introduction. In Section 2 we describe the data sources used in this work as well as the data

[4]In fact we should use *Estimated Breeding Value* (EBV), but for the sake of simplicity we maintain the notation of *Breeding Value* (BV), although it is clear that we are dealing with estimations all the time.

selection carried out according to our task. Section 3 is devoted to data transformation, especially to the discretization of the breeding value variable in order to transform the task from numerical prediction to classification. Classification algorithms used are described in Section 4, while Section 5 describes the initial classification process carried out. In Section 6 we apply variable selection while in Section 7 attribute construction is used. Finally, Section 8 is devoted to briefly discuss the obtained results and in Section 9 we conclude and outline future work.

2 Data preparation and selection

After understanding the application domain and identifying the goal of the process, the next step is to prepare the data. Data preparation [Fayyad et al., 1996] is an important process (usually one of the most time consuming) which comprises a series of stages as creating a target data set and cleaning and preprocessing it. This section and the next one deal with this important topic of a data mining project.

2.1 Data preparation

In a data mining project the source of data is usually a data-warehouse, however AGRAMA does not have a data-warehouse in its organization. In AGRAMA, the data[5] is stored in a relational data base system, as a set of tables which are linked among them by means of a set of attributes used as keys (the structure is shown in figure 1). There are six main tables in the system:

- Animals. This table contains historical data about the animals recorded by the organization: sex, type of birth, date of birth, stock farm, etc, ... It contains approximately 248000 records.

- Qualifications. This table contains data about the morphological qualification given to the animals. The more higher the qualification is, the closest is the animal to the racial standard of Manchego sheep breed.

- Lactations. This table contains a record for each controlled lactation: animal, date, amount of milk, percentage of fat, etc, ... It contains approximately 390000 records.

[5]In this work we have use as source data from years 1989 to 2003

- Mammals observations. This table contains data about ewes udders.
- Observations. This table contains general observations about the animal, its stock-farm, etc, ...
- BV. This table contains data about the breeding value given to an animal: BV, confidence/reliability about the BV estimation, etc, ...

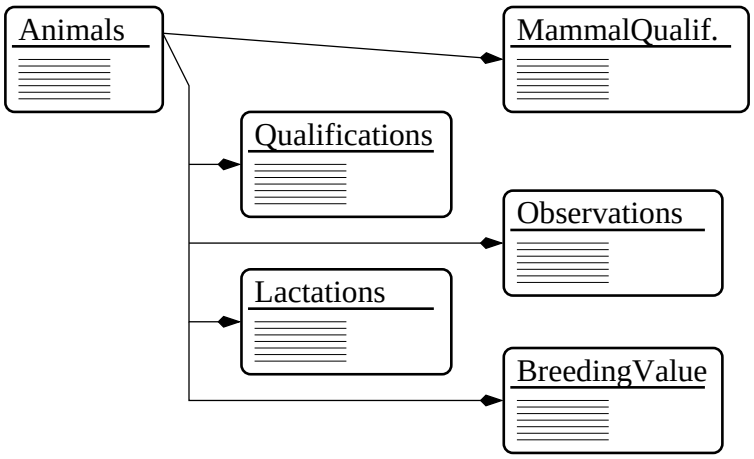

Figure 1: Structure of data tables in AGRAMA

From these tables our first goal is to identify/specify the part of the database to be mined. That is, we need to obtain a single (flat) table containing those variables (and records) which could have influence in the determination of the BV. This table can be obtained[6] by means of SQL queries performed against our data table structure by using the animal identification as primary key, and will constitute our initial dataset or *minable view* (MV). To obtain the MV we have to take into account the following questions:

- Variables. Which variables should be included in the MV?. To solve this problem we have required the help of the technical staff of AGRAMA. The set of selected variables is listed in table 1.
- Data integration. As we are working with different data sources, some integration problems arise. As an example, most tables store a record for each animal, but Lactations table stores a record for each lactation, and given that an animal can have 0, 1 or more controlled lactations, we have a 1-to-n relation. Therefore, making a join in which Lactations table is involved, will yield a table in which the same animal can be represented

[6]Since virtual relations are called *views* in the field of databases, the set of task-relevant data for data mining is called a *minable view* [Han and Kamber, 2001]

by several records. To fix this problem, we have summarized the information about the controlled lactations of an animal by using the following fields:

- NLact. Number of controlled lactations performed to the animal.
- AvLactNorm. Amount of milk produced during a controlled lactation, averaged over NLact.
- MaxLactNorm. The amount of milk produced in the best controlled lactation to the animal (the maximum value).
- AvLact120. The same as AvLactNorm but considering only the first 120 days of the controlled lactation.
- MaxLact120. The same as MaxLactNorm but considering only the first 120 days of the controlled lactations.

After this process, the new Lactations table only have a record per animal, and thus, we can obtain our MV containing the variables listed in table 1 by using SQL queries.

Another problem to be considered in data preparation is *data cleaning*. By data cleaning we understand the detection (and its treatment) of possible errors in data, outliers and missing values. In our case we have deal to with the two following situations:

- By inspection we have detected some errors due to data acquisition. Concretely, there are 169 records with sex=*male* but which have data about lactations. Obviously, the value of sex for these records is wrong, being the correct one *female*.
- As the MV is obtained by SQL queries, a *join* is carried out between Lactations and Animals tables. As a consequence, all the animals which do not have any lactation (i.e., they do not appear in the lactations table), will have missing values for the lactation variables (Nlact, AvLactNorm, etc, ...) in the resulting MV. However, we know that the correct value for these variables is not missing, but 0. By making this substitution, the number of missing values for these variables decreases from 6% to 0% in Nlact and from 12% to 6% in the remaining lactaction variables.

2.2 Data Selection

Although we have already performed some kind of data selection by deciding which variables/attributes should be included in our MV, this task has not still finished.

Table 1: Variable description

Variable	Type	Missing	Description
1. Sex	nominal(2)	0%	Sex (male,female) of the animal
Data about the BV			
2. BVFather	numeric	0%	BV of animal father
3. ReBVF	numeric	0%	Confidence on the value assigned to BVFather
4. BVMother	numeric	0%	BV of animal mother
5. ReBVM	numeric	0%	Confidence on BVMother value
6. BVMaternalGM	numeric	39%	BV of animal maternal grand mother
7. ReBVMGM	numeric	39%	Confidence on BVMaternalGM value
8. BVParentalGM	numeric	3%	BV of animal parental grand mother
9. ReBVPGM	numeric	3%	Confidence on BVParentalGM value
10. BVMaternalGF	numeric	67%	BV of animal maternal grand father
11. ReBVMGF	numeric	67%	Confidence on BVMaternalGF value
12. BVParentalGF	numeric	15%	BV of animal parental grand father
13. ReBVPGF	numeric	15%	Confidence on BVParentalGF value
14. **BV**	numeric	0%	BV of the animal. This is the GOAL variable
15. ReBV	numeric	0%	Confidence on the value assigned to BV
Environmental data			
16. TypeOfBirth	nominal(6)	0%	Number of children in the animal childbirth
17. StockFarm	nominal(117)	0%	Stock farm to which the animal belong
18. FatherStockFarm	nominal(64)	0%	Stock farm to which the animal father belong
19. MotherStockFarm	nominal(127)	0%	Stock farm to which the animal mother belong
Data about mother lactations			
20. NLactM	numeric	12%	Number of controlled lactations to the animal mother
21. AvLactNormM	numeric	13%	Amount of milk produced during a controlled lactation. Averaged over the number of controlled lactations.
22. MaxLactNormM	numeric	13%	Maximum amount of milk produced in the controlled lactations
23. AvLact120M	numeric	13%	Amount of milk produced in the first 120 days of a controlled lactation. Averaged over the number of controlled lactations
24. MaxLact120M	numeric	13%	Maximum amount of milk produced in the 120 days controlled lactations
Data about animal lactations			
25. NLact	numeric	0%	Number of controlled lactations to the animal
26. AvLactNorm	numeric	6%	Same meaning as AvLactNormM
27. MaxLactNorm	numeric	6%	Same meaning as MaxLactNormM
28. AvLact120	numeric	6%	Same meaning as AvLact120M
29. MaxLact120	numeric	6%	Same meaning as MaxLact120M

In this section we will try to select the correct set of records in order to accomplish our data mining task. As our goal is related to the prediction/approximation of the animal BV, we are interested in the selection of those records which best fit to that goal. Following the advice of AGRAMA's experts we have take into account the conditions listed below in order to select those records:

- As the main goal of the ESROM program is to improve the amount of milk produced by the animals, we focus our problem in the prediction of the BV only for female animals.
- In order to have an homogeneous sample, AGRAMA's experts think that the task should focus in animals with exactly one lactation.
- For all the BV variables (BV, BVFather, BVMother, etc, ...) there is an associated variable which measures the confidence on such estimation (ReBV, ReBVF, ReBVM, etc, ...). In order to avoid having a noisy data set, only those records in which we have enough confidence about the computed BV will be considered. Concretely, the experts require a confidence greatest or equal than 0.4 in the case of female animals and greatest or equal than 0.6 in the case of male animals. Thus, we have considered only those records in which the following expression holds: $(\mathsf{ReBV} \geq 0.4)$ and $(\mathsf{ReBVF} \geq 0.6)$ and $(\mathsf{ReBVM} \geq 0.4))$.

After this process our MV has 3087 records. Because of this record selection, there is also changes with respect to the variables included in our MV:

- Variable Sex has been removed because it has the same value (*female*) in all the cases.
- Variable NLact has been removed because it has the same value (*1*) in all the cases.
- Variables MaxLactNorm and MaxLact120 have been removed because they have the same value that AvLactNorm and AvLact120 in all the cases.
- As the confidence in the BV of an animal can be known only after computing such value using BLUP methodology, and our goal is to predict BV, then we have removed variable ReBV.

Therefore, our new MV has 24 variables. To finish this section, we have to note that because of the reduction (at the record level) in the MV, there has been also changes with respect to the number of missing values in some variables. Thus,

- BVMaternalGM and ReBVMGM from 39% to 23%.

- BVParentalGM and ReBVPGM from 3% to 0%.

- BVMaternalGF and ReBVMGF from 67% to 47%.

- BVParentalGF and ReBVPGF from 15% to 3%.

- NLActM from 12% to 1%.

- AvLactNormM, MaxLactNormM, AvLact120M and MaxLact120M from 13% to 2%.

- AvLactNorm and AvLact120 from 14% to 6%.

3 Data Transformation

As we can see in table 1 the BV of an animal is a numerical value, so the task of predicting it constitutes a regression (or numerical prediction) problem. However, in this paper we are interested in dealing with this problem as a classification one.

As was mentioned in the introduction, the information provided to the *stock breeder* about the BV of an animal is the percentile (10%, 20%, 30%,etc, ...) in which the animal is classified. In fact, in the ESROM scheme the animal is retained if it is classified in the percentile 51-60% or higher. Therefore, even a two-class ($BV_{\leq 50\%}$,$BV_{>50\%}$) classification task will be of interest in this problem. However, following the instructions of AGRAMA's experts, we have proposed two classification tasks:

- First: the BV variable has been discretized in its respective quartiles. That is, we have discretized BV in four bins of equal frequency (see fig. 2). The class variable is BV4 = $\{f1, f2, f3, f4\}$. From now on we will refer to this problem as the 4-labels one.

- Second: the BV variable has been discretized in five bins of equal frequency (see fig. 2). The class variable is BV5 = $\{v1, v2, v3, v4, v5\}$. From now on we will refer to this problem as the 5-labels one.

Figure 2 shows an histogram of the BV variable and the two discretizations used in this work. From the point of view of AGRAMA's experts, classifying an animal by using the discretizations provided by BV4 and BV5 is enough informative.

Therefore, from now on we have two different data sets:

- mv4.- This data set consists in the MV described in the previous section, but replacing BV by BV4.

- mv5.- This data set consists in the MV described in the previous section, but replacing BV by BV5.

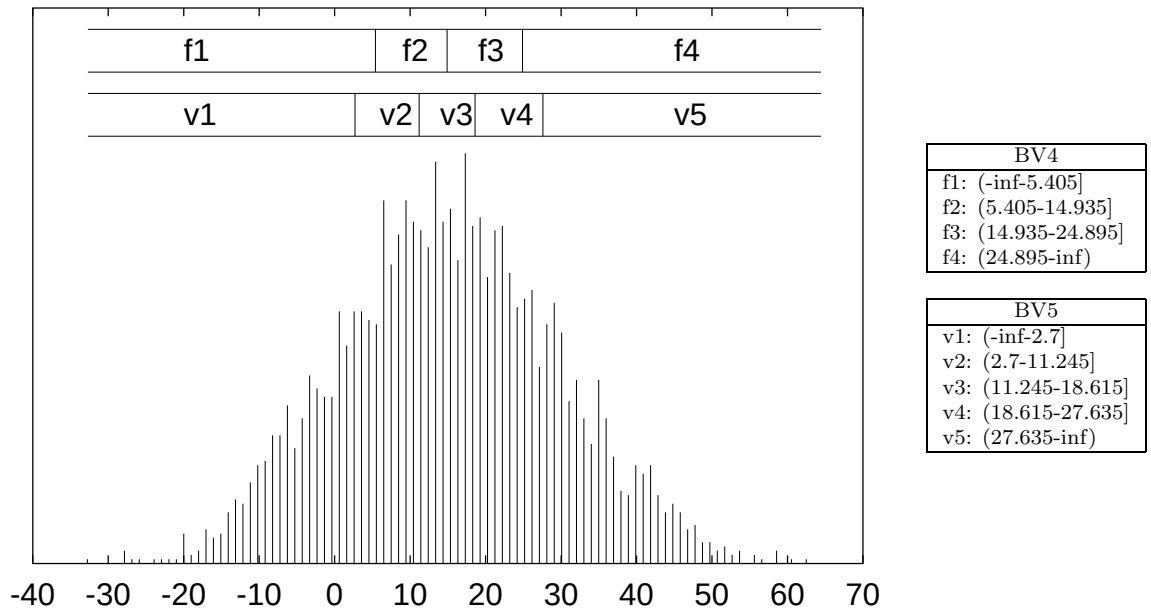

Figure 2: Histogram of BV variable and the two discretizations carried out: BV4={f1, f2, f3, f4} and BV5={v1, v2, v3, v4}

Apart from the transformation of the BV variable, which means to change from a regression problem to a classification one, we have considered also the discretization of the remaining numerical variables. The idea is to work with the two versions, that is, the two datasets described above (mv4 and mv5) and two new datasets (mv4d and mv5d) which corresponds to the discretization of the numerical variables in mv4 and mv5. As we are in a classification problem, we have used a supervised technique to carry out the discretization. Concretely, we have used a standard method of the literature, as it is the Fayyad and Irani algorithm [Fayyad and Irani, 1993] which uses MDL and entropy to find the best cut-off points. A very appreciated feature of this method is that the number of bins has not to be fixed a priori. After the discretization process the number of bins in the discretized variables of mv4d goes from 2 to 24 with a mean of 6.1, and in mv5d goes from 2 to 18 with a mean of 5.9.

4 Classification algorithms

In this section we briefly describe the two (classical) classification algorithms used in this work: decision trees (C4.5) and Naive Bayes.

4.1 Decision trees (C4.5)

Graphically, in a decision or classification tree the inner nodes represent tests about the predictive attributes, the leaves are labels of the class variable and each branch descending from an inner node, asking about attribute X_i corresponds to one of the (possibly discretized) values for this attribute (see figure 3.b). In order to classify a new instance, the algorithm starts at the root node, tests the attribute specified by this node and descends by the appropriate branch to a new node. If this new node is a leaf then its class label is returned as output, otherwise the test process is repeated.

C4.5 [Quinlan, 1986] is a greedy, recursive, top-down algorithm for the induction of decision trees from data. The algorithm starts at the root node with all the available data, and selects the *best* test among the available attributes. This test is placed as root node and the data set is partitioned following the possible outcomes of the test. Then, the process is recursively repeated for each partition until a stopping criterion is met. In C4.5 the *best* attribute is selected by using information gain. Other advantages of C4.5 are the capabilities of dealing with missing values and discretizing on-line the numerical attributes (see [Quinlan, 1986] for details).

4.2 Naive Bayes

The Naive Bayes (NB) classifier [Duda and Hart, 1973] is a probabilistic classifier based on the assumption of conditional independence among the predictive attributes given the class. Because of this independence assumption, the joint probability $P(C, X_1, \ldots, X_n)$ factorizes as:

$$P(C, X_1, \ldots, X_n) = P(C) \prod_{i=1}^{n} P(X_i|C)$$

Therefore, the probabilities to be learnt are:

- A marginal probability distribution for the class variable $P(C)$, which stand for the *a priori* probability of C.

- A conditional probability distribution for each predictive attribute given the class $P(X_i|C)$. If X_i is a nominal variable, then a multinomial distribution is used. If X_i is a numerical variable, then $P(X_i|c_j) = N(\mu, \sigma)$, that is, a Normal distribution learnt for each label c_j of the class variable C.

Figure 3.b shows an example of NB classifier induced for our problem, but considering only three (numerical) predictive variables and a two-labels discretization of the class (BV).

After the NB classifier is induced, the MAP principle is used to classify new instances, that is, given an instance $< x_1, \ldots, x_n >$ we choose the class label c^* such that

$$\begin{aligned} c^* &= \arg\max_{c_j} P(C = c_j | X_1 = x_1, \ldots, X_n = x_n) \\ &= \arg\max_{c_j} \frac{P(C=c_j) \cdot P(X_1=x_1,\ldots,X_n=x_n|c_j)}{P(X_1=x_1,\ldots,X_n=x_n)} \\ &= \arg\max_{c_j} P(C = c_j) \cdot P(X_1 = x_1, \ldots, X_n = x_n | C = c_j) \\ &= \arg\max_{c_j} P(C = c_j) \prod_{i=1}^{n} P(X_i = x_i | C = c_j) \end{aligned}$$

Despite its simplicity and unrealistic independence assumption, the performance of the NB classifier is remarkably successful in practice [Langley et al., 1992, Domingos and Pazzani, 1997, Rish, 2001].

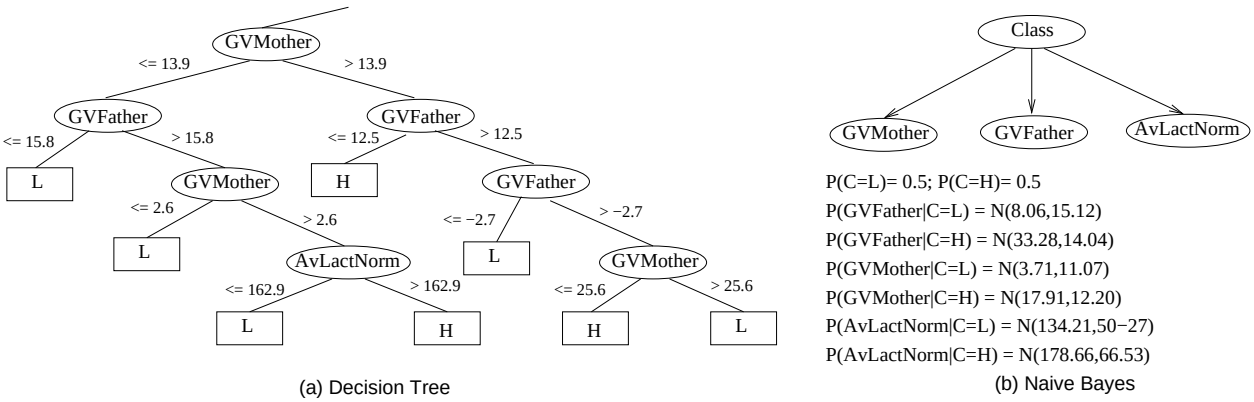

Figure 3: (Fraction of a) Decision tree and Naive Bayes learned for a two labels (*Low* and *High*) classification task of a data set containing only {BVFather, BVMother, AvLactNorm } as (numerical) predictive attributes.

5 Initial classification process

In this section we describe the initial experiments carried out over the MVs described in previous sections, and using the two classifiers introduced in Section 4, or more concretely the implementation of C4.5 (J48) and Naive Bayes provided by WEKA [Witten and Frank, 2000]

(in both cases the default options setting has been used). In order to measure the accuracy of the classifiers over a given data set we use the well known (stratified) K-fold cross validation technique (K=10 [Kohavi, 1995]).

Instead of considering all the variables in the MV, we have run the algorithms starting with a small set of variables, and progressively adding different groups of variables. Concretely, we have carried out the following process (see Table 2):

1. First, we have considered only the BV of both parents as predictive attributes. That is, BVp = {BVFather, BVMother}.

2. Our second approach has been to consider all the BV variables (BVall) as predictive attributes, that is, variables 2 to 13 in Table 1. Surprisingly, there is only a slight improvement in one out the eight cases. Because of this, we maintain this two sets of variables as different starting points for the remaining process.

3. Our third approach consists into adding environmental variables (see Table 1) to BVp and BVall. As we can see in Table 2, the results are considerably worse than in the previous case, specially for the NB algorithm (notice that C4.5 has its own variable selection procedure).

   If we pay attention to the environmental variables (except TypeOfBirth), we can see that they are nominal variables with a large number of possible outcomes. This type of variables can introduce a considerably amount of noise in the learning/classification process because estimating conditional probability tables for them is quite difficult and require a really (and usually unavailable) large sample. Because of this, we have considered the possibility of preprocessing (grouping) these variables. To do this, we have implemented in the Elvira system [Elvira-Consortium, 2002] the KEX method proposed by Berka and Bruha [Berka and Bruha, 1998]. This method reduces the number of values for a nominal variable to $|C| + 1$ labels (or groups), being $|C|$ the number of values (classes) of the class variable (C). The idea of the method described in [Berka and Bruha, 1998] is to study the distribution (P_D) of the class variable for each value x_i of a given variable X. If the distribution significantly differs from the uniform (by using a χ^2 test) then the value x_i is assigned to the group identified by the class label c_i having the largest probability in P_D, otherwise, the value x_i is assigned to a group denoted as *unknown*.

We denote by env^g the group of environmental variables after the grouping process. Replacing env by env^g and repeating the classification process, we can see how the results improve considerably, specially for the NB algorithm. Therefore, from now on, we will use env^g instead of env.

4. Our next step is to add the variables related with lactation data. In this way, we try by adding only the data about mother lactations, lactM (variables 20 to 24 in Table 1); by adding only the data about animal lactations, lact (variables 25 to 29 in Table 1); or by adding all of them, lactM+lact. As we can see in Table 2, the best results are obtained when only animal lactations data is used.

Table 2: Initial classification process

	4 classes				5 classes			
Variables	C4.5	C4.5(d)	NB	NB(d)	C4.5	C4.5(d)	NB	NB(d)
BVp(arents)	72.34	71.79	72.27	70.46	66.18	66.73	**66.47**	**63.59**
BVall	70.46	71.36	65.50	62.71	64.17	66.89	58.11	55.17
BVp+env	69.61	71.79	62.26	62.00	63.20	66.73	54.62	53.84
BVall+env	71.49	71.49	61.80	61.19	63.88	66.47	54.42	53.35
BVp+env^g	72.47	71.92	69.39	67.15	66.12	66.21	62.78	60.61
BVall+env^g	70.52	71.43	64.08	63.04	64.17	66.38	58.08	54.81
BVp+env^g+lactM	71.07	71.36	61.81	58.83	63.59	64.63	53.55	51.15
BVall+env^g+lactM	68.74	71.82	62.52	62.65	61.91	64.95	54.00	54.52
BVp+env^g+lact	**75.38**	**76.39**	**73.05**	**70.52**	**69.42**	**69.58**	64.82	63.46
BVall+env^g+lact	74.44	75.10	70.20	66.08	68.74	68.97	62.13	57.60
BVp+env^g+lactM+lact	72.01	75.64	61.23	60.48	68.51	69.42	54.55	52.96
BVall+env^g+lactM+lact	73.05	75.22	63.04	64.56	66.99	69.00	56.79	57.15
BVall+env+lactM+lact	73.05	75.06	63.04	64.89	66.05	67.74	54.39	55.82
Variables	zR	zR(d)	oR	oR(d)	zR	zR(d)	oR	oR(d)
BVall+env^g+lactM+lact	24.94	24.94	54.49	53.45	19.89	19.89	48.66	47.04
BVall+env+lactM+lact	24.94	24.94	54.49	53.45	19.89	19.89	48.66	47.04

Before to analyze the results obtained, we should realized that our classification problem has been artificially constructed from a regression one, and that all the classes are equally distributed. Because of these reasons, we think that the problem can be considered as difficult and high classification rates should not be expected. In fact, baseline algorithms as ZeroR (zR), which returns the majority class, or OneR (oR) [Holte, 1993], which uses only one variable to do the prediction, obtain the accuracy shown in the last two rows of Table 2.

After this reflexion we can proceed to analyze the results. The last row of Table 2 referred to C4.5 and NB gives us the results obtained when all the variables in the MV are included as predictive attributes. Given the complexity of the problem, these results are not bad,

specially for the decision tree case, however, they are improved when not all the variables are included as predictive attributes. Concretely, in most of the cases the best results are obtained when only BVp+env^g+lact are used as predictive attributes, obtaining an accuracy of 76% in the four labels problem and almost a 70% in the five labels problem. Besides, it is interesting to point out that decision trees obtain their best result for the discretized MV, while NB obtains its best results for the non-discretized MV.

Though these results can (in our opinion) be considered as good results, it is clear that using all the variables as predictive attributes it is not a good idea here. In fact, the previous process (shown in table 2) can be viewed as a *manual* variable selection process. Thus, when only an appropriate subset of variables is used instead of the full set, the algorithms improve their accuracy, specially NB which does not carry out an implicit variable selection process as C4.5 does. From this analysis, we can conclude that it is worth to study in a deeper way the problem of attribute selection, which is the goal of the next section.

6 Feature Subset Selection

Feature (or variable, or attribute) Subset Selection (FSS) is the process of identifying the input variables which are relevant to a particular learning (or data mining) problem [Guyon and Elisseeff, 2003 Liu and Motoda, 1998]. In our case we are interested in identifying those variables with major influence in the prediction of the BV. The goal of FSS can be twice: (1) select the subset of variables yielding the best classification performance, and (2) identify those variables which are relevant for a given task.

As has been mentioned, the process in the previous section can be interpreted as a *manual* FSS process. In this section we study the application of automatic FSS selection techniques to our problem, concretely we have considered the following approaches:

- *Embedded* methods. In this case is the own learning algorithm who carries out the variable selection during the learning process. An example of this approach are decision trees.

- *Filter* methods. These methods use statistical or distance-based measures to evaluate the merit of a variable or subset of variables. They are fast and independent of the machine learning algorithm to be used.

- *Wrapper* methods. They use a machine learning algorithm as part of the selection

process, that is, the merit of a given subset is measured by learning (and evaluating) a model using only that subset of variables. Of course, these methods are computationally expensive and the obtained subset lack of generality because it is tied to the bias of the classifier used during the FSS process.

- *Filter+Wrapper* methods. In this case both approaches are combined in some way.

Embedded FSS Although in section 5 we have used different subsets of variables to build the different classifiers, when decision trees are induced not all the given variables are used. Thus, the subsets S_n (original/numerical MV) and S_d (discretized MV) shown in Table 5 contain the variables actually used by C4.5 when the subset BVp+env^g+lact is used as input variables (notice that this is the case in which C4.5 gets the best accuracy). It is worth noting that C4.5 selects the same subset independently of the number of labels in the class variable, and that the discretized version (which gets the best results in both cases) needs only 5 out the 7 variables used for the original (non-discretized) MV.

If we use decision trees as feature subset selectors and take its output (S_n and S_d) as the input for NB, then we obtain the results shown in the FSS(C4.5) row of Table 4. Thus, NB improves the best result obtained in the previous section for the discretized MV, and also improves (by far) the results obtained when the full set of variables in the MV is used. However, as S_n and S_d are biased by C4.5, we can expect to improve this results by using different FSS methods.

Filtered FSS Our second approach to FSS it is based on the use of filter criteria in order to measure the relevancy between each predictive attribute and the class variable. Concretely, we have used the two following filter measures:

- *Mutual Information (MI).* The mutual information between two given variables X and Y can be interpreted as: "the information that Y tells us about X is the reduction in uncertainty about X due to the knowledge of Y". The mutual information between the class variable (C) and a given attribute X can be computed as:

$$MI(C, X) = H(C) - H(C|X),$$

where H denotes the Shannon entropy.

- *Symmetrical Uncertainty (SU).* This measure evaluates the worth of an attribute (X) by measuring the symmetrical uncertainty with respect to the class variable (C):

$$SU(C, X) = \frac{2 \cdot ((H(C) - H(C|X))}{H(C) + H(X)}.$$

  Basically it uses mutual information, but it is projected onto the $[0, 1]$ interval by applying a sort of normalization. In our opinion SU is quite interesting when the number of states in the involved variables is different.

Table 3 shows the ranking produced by these measures in the four and five labels problems. In the opinion of AGRAMA experts the ranking provided by SU seems to be more accurate than the one provided by MI, as an example, it is well known that BVFather and BVMother are the two variables of major influence in the prediction of BV. Because of this, in the rest of this work we will use the ranking produced by SU.

In filter FSS, after obtaining the ranking, the first k variables are used as the selected subset. Of course, the main problem here is how to select k. In our case and due to the knowledge gained in Section 5, we have decide to use two different values for k:

- $k = 6$. As lactation data seems to play an important role in breeding value prediction/classification, we have choose a value for k which forces to include a lactation variable (concretely AvLac120). We will refer to this subset of variables as S_6 (Table 3).

- $k = 9$. Given that both subsets, S_n and S_d, contain variables from env^g, we have extended our variables selected in order to add the best ranked variables belonging to env^g. We will refer to this subset of variables as S_9.

From the results (Table 4) we can observe that using S_9 instead of S_6 only gets (slightly) better results in one out of the eight cases. However, this situation happens with C4.5 (mv5d) which makes its own FSS process. With respect to the use of S_6, only in one out of the eight cases we get a better result than when S_n or S_d are used as input, and again this happens with C4.5 (mv5). On the other hand, NB algorithm degrades its performance specially in the discretized case, which can be due to the inclusion of redundant variables in the selected subset. In fact, this is one of the main problems of these FSS methods, two variables can be separately relevant for the class, but redundant among them.

Table 3: Variable ranking obtained by filter measures

SU (4 classes)		MI (4 classes)		SU (5 classes)		MI (5 classes)	
0.24851	BVFather	0.58338	BVFather	0.2503	BVFather	0.63229	BVFather
0.16591	BVMother	0.41607	BVParentalGM	0.15456	BVMother	0.40773	BVParentalGM
0.14649	BVParentalGM	0.3715	BVMother	0.1468	BVParentalGM	0.39301	BVMother
0.11889	BVParentalGF	0.28623	BVParentalGF	0.12136	BVParentalGF	0.31166	BVParentalGF
0.08522	ReBVF	0.19802	ReBVF	0.08474	ReBVF	0.21055	ReBVF
0.08272	AvLac120	0.17608	AvLac120	0.07872	AvLac120	0.18079	AvLac120
0.06235	ReBVPGM	0.15862	ReBVPGM	0.06233	ReBVPGM	0.17372	ReBVPGM
0.05637	ReBVPGF	0.1416	MotherStockFarm	0.05625	ReBVPGF	0.17128	MotherStockFarm
0.05575	AvLAc120M	0.13243	StockFarm	0.05414	AvLAc120M	0.15952	StockFarm
0.05567	AvLacNorm	0.12164	ReBVPGF	0.05058	AvLacNorm	0.13044	ReBVPGF
0.0514	MaxLac120M	0.10773	AvLacNorm	0.04962	MaxLac120M	0.11296	AvLAc120M
0.04425	AvLacNormM	0.10719	AvLAc120M	0.04309	MotherStockFarm	0.10694	AvLacNorm
0.03836	BVMaternalGM	0.10551	MaxLac120M	0.04151	AvLacNormM	0.106	MaxLac120M
0.03778	MaxLacNormM	0.08069	AvLacNormM	0.04108	FatherStockFarm	0.08243	AvLacNormM
0.03713	MotherStockFarm	0.07573	BVMaternalGM	0.04026	StockFarm	0.07897	BVMaternalGM
0.03595	FatherStockFarm	0.07443	MaxLacNormM	0.038	BVMaternalGM	0.07873	MaxLacNormM
0.03484	StockFarm	0.0516	FatherStockFarm	0.03685	MaxLacNormM	0.06557	FatherStockFarm
0.01994	BVMaternalGF	0.03526	BVMaternalGF	0.01941	BVMaternalGF	0.03703	BVMaternalGF
0.00932	ReBVMGF	0.01335	ReBVMGF	0.00885	NLacM	0.01341	ReBVMGF
0.00913	NLacM	0.01213	NLacM	0.00827	ReBVMGF	0.01319	NLacM
0.00472	ReBVM	0.00654	ReBVM	0.00395	TypeOfBirth	0.00728	TypeOfBirth
0.00466	ReBVMGM	0.00545	TypeOfBirth	0	ReBVMGM	0	ReBVMGM
0.00323	TypeOfBirth	0.00503	ReBVMGM	0	ReBVM	0	ReBVM

Filter+Wrapper FSS Several combinations of filter plus wrapper FSS can be found in the literature. The most common is applied when there is a large number of predictive attributes, and consists in the selection of a subset of variables by applying a (fast) filter method, and then a wrapper method is applied over that (smaller) subset. In this work, the scenario is different because we only have around 30 predictive variables, and so we propose to apply a different combination of filter and wrapper approaches.

Our idea is to use the filter stage to produce a ranking, but without removing any variable. Then, that ranking will be used to guide the operation of the wrapper algorithm. Concretely, the wrapper stage consists in running over the ordering of variables produced by the filter process and to add a variable to the subset of selected variables only if such inclusion improves the classifier accuracy. As we are in a wrapper phase, the accuracy is obtained by launching the learning algorithm (C4.5 or NB) having as input only the current subset of selected variables. The classical (greedy) stopping criterion for this process is to finish when adding a new variable does not improve the accuracy. However, proceeding in this way the process can be stopped because two correlated variables appear together in the ranking even if any relevant variable appears later in the ranking. Trying to alleviate this problem, the novelty of our proposal consists in the consideration of a lookahead parameter k, which allow us to continue the process (discarding the useless variable) if less than k useless variables have been consecutively discarded. Figure 4 shows the pseudo-code of this FSS algorithm. Of course, if $k = 0$ we get the greedy behavior described above, and if $k = \infty$ we consider all the predictive variables as candidate to be included in the selected subset.

We have experimented with the ranking produced by using SU as filter measure, and

Function FW(data, class, classifier, lookahead)

1. $Ranking \leftarrow$ compute a ranking of predictive attributes with respect to class by using a filter measure (i.e. symmetrical uncertainty, mutual information, ...)
2. $i \leftarrow 0$; $selected \leftarrow \emptyset$; $bestAcc \leftarrow 0.0$; $fails \leftarrow 0$
3. While (i $\leq Ranking.size$) do
   (a) $X \leftarrow Ranking[i]$
   (b) $accuracy \leftarrow$ getAccuracy(classifier, class, data$^{\downarrow selected \cup \{X\}}$)
   (c) If ($accuracy > bestAcc$)
   $selected \leftarrow selected \cup \{X\}; bestAcc \leftarrow accuracy; fails \leftarrow 0$
   (d) Else If ($fails <$ lookahead) $fails \leftarrow fails + 1$
   (e) Else break
   (f) $i \leftarrow i + 1$
4. Return *selected*

Figure 4: Pseudo-code of function FW

with lookahead = 0, 1, 5 and ∞.

From the results (Table 4 and Table 5) we can conclude that with lookahead equals 0 or 1, the algorithm has a similar behavior, just stopping when only the two first variables (BVFather and BVMother) have been selected. There are only two exceptions (out of 16) in which more variables are selected, but in both cases is C4.5 the algorithm used and so, it is probable that it discards some of them. Even with this small subset of variables, NB improves its results in all the cases with respect to the filtering FSS process and in half of the cases with respect to the subset obtained by using C4.5 as feature selector. This fact remarks the idea of having redundant variables among the subset of (*relevant*) variables ranked in the first positions by SU. On the other hand, this is not the case for C4.5, which obtains better results with the subsets provided by the previous FSS approaches, probably because it filters those subsets when inducing the trees.

Things are quite different when lookahead is set to 5. In this case a new subset, S_t = {BVFather, BVMother, AvLact120}, arises as a very good predictor. That is, by using a lookahead of 5, the process is able to discards some (possibly relevant) attributes which are redundant with respect to those already included, but at the same time it continues the process looking for new relevant (but not redundant) attributes. This subset (S_t) is selected in the four cases in which NB is applied and it is complemented with some other variables

when applying C4.5. In all the cases (NB and C4.5) the results obtained when using FW(5) are better than those obtained so far.

To end with the application of FW as feature selector, we have try with lookahead=∞. In this way, we are sure to give an opportunity to all[7] the variables in the data set, without a high increase of the complexity (in fact, the number of subset evaluated is linear in the number of variables). In this case, the results (slightly) improve with respect to lookahead=5, in four of the eight cases. With respect to the subsets of variables obtained, we can see that they consist of S_t complemented with one or two more variables in most of the cases.

To conclude, the FW method presented here has obtained (by far) better results than the filtering approach or the decision trees based FSS process. On the other hand, the number of subset evaluated is at most (lookahead=∞) linear in the number of variables.

Wrapper FSS As described before, the wrapper approach takes advantage of using the learning algorithm during the FSS process. In this way, this approach (usually) obtains better subsets than other methods (like filter), but also has a considerably higher cost. In this work we have use *forward* FSS based on best first search. Forward subset selection works by starting with an empty set of variables and adding a variable at each step of the search. That is, it starts by trying all the subsets containing only a variable and select that with highest accuracy. Then, it tries all the subsets of cardinality two (which contains the previously selected variable) and select the one with higher accuracy. This process go on until a stopping criteria is met (usually, the algorithm stops when the accuracy does not improve). Best first search increases the described forward subset selection by allowing the search method to do backtracking. The main disadvantage of this approach is that the cost grows exponentially with the number of (irrelevant) variables.

In our experiments we have set the number of allowed backtracking levels to five. The results (Table 4 and Table 5) show that in five out of the eight cases we have (slightly) improved the results obtained so far. However, when comparing with FW(∞) we observe that the gain in accuracy obtained by Wrapper is (on the average) less than 0.1%, which show us that FW(∞) is in fact a quite competitive FSS method. With respect to the subsets selected by this approach, again they are extensions of S_t by adding few new variables, that contain in most cases data about mother lactations (which seems to be quite reasonable)

[7] If too many variables are included in the MV then some of them can be removed before to apply FW(∞) by using statistical hypothesis testing based on the well known relation between χ^2 and mutual information.

and confidence measures. With respect to FW(∞) again we obtain that both methods have a similar behavior. On the other hand, the complexity of the wrapper approach is considerably higher with respect to FW(∞) because it evaluates (on the average) about 200 subsets, while the number of subsets evaluated by FW(∞) is linear in the number of variables, so it evaluates exactly 23 subsets over each MV.

Table 4: Accuracy obtained when applying FSS (the superscript makes reference to the subset of variables listed in Table 5)

	4 classes				5 classes			
Selection	C4.5	C4.5(d)	NB	NB(d)	C4.5	C4.5(d)	NB	NB(d)
Best until now	75.38	76.39	73.05	70.52	69.42	69.58	66.47	63.59
FSS (C4.5)	75.38^{n}	76.39^{d}	71.66^{n}	71.68^{d}	69.42^{n}	69.58^{d}	65.24^{n}	65.66^{d}
FSS (Filter-1)	75.32^{6}	75.57^{6}	71.75^{6}	65.21^{6}	69.91^{6}	68.61^{6}	66.15^{6}	59.54^{6}
FSS (Filter-2)	75.25^{9}	75.51^{9}	70.72^{9}	63.82^{9}	68.74^{9}	68.97^{9}	63.46^{9}	56.88^{9}
FSS (FW(0))	72.34^{p}	71.79^{p}	72.27^{p}	70.46^{p}	66.31^{b}	66.73^{p}	66.47^{p}	63.59^{p}
FSS (FW(1))	72.34^{p}	76.48^{a}	72.27^{p}	70.46^{p}	66.31^{b}	66.73^{p}	66.47^{p}	63.59^{p}
FSS (FW(5))	76.13^{t}	76.48^{a}	$\mathbf{78.23}^{t}$	75.22^{t}	70.07^{e}	70.39^{f}	71.46^{t}	67.22^{t}
FSS (FW(∞))	76.55^{c}	76.48^{a}	$\mathbf{78.23}^{t}$	75.25^{2}	70.07^{d}	70.39^{f}	$\mathbf{71.60}^{g}$	$\mathbf{67.38}^{g}$
FSS (Wrapper)	$\mathbf{76.62}^{h}$	$\mathbf{76.77}^{i}$	$\mathbf{78.23}^{t}$	$\mathbf{75.32}^{j}$	$\mathbf{70.52}^{k}$	$\mathbf{71.69}^{3}$	70.20^{l}	$\mathbf{67.38}^{g}$

Table 5: Variables selected by the different algorithms

Id	Set
S_n	$S_t \cup$ {AvLacNorm, StockFarm, TypeOfBirth, MotherStockFarm}
S_d	$S_t \cup$ {AvLacNorm, StockFarm}
S_6	$S_t \cup$ {BVParentalGM, BVParentalGF, ReBVF}
S_9	$S_t \cup$ {BVParentalGM, BVParentalGF, ReBVF, ReBVPGM, MotherStockFarm, StockFarm}
S_p	{BVFather, BVMother}
S_a	$S_t \cup$ {AvLAcNorm,ReBVF,ReBVMGM,BVParentalGF,ReBVPGF,AvLac120M}
S_b	{BVFather, BVMother,BVParentalGM}
$\mathbf{S}_t$	{BVFather, BVMother,AvLact120}
S_e	$S_t \cup$ {BVParentalGM,ReBVPGM,AvLact120M}
S_f	$S_t \cup$ {ReBVF,BVMaternalGF,NLactM,AvLactNormM,AvLact120M}
S_c	$S_t \cup$ {FatherStockFarm, BVMaternalGM, BVMaternalGF,MaxLactNormM}
S_2	$S_t \cup$ {ReBVMGM}
S_d	$S_t \cup$ {FatherStockFarm}
$\mathbf{S}_g$	$S_t \cup$ {ReBVM, FatherStockFarm}
$\mathbf{S}_h$	$S_t \cup$ {BVMaternalGM,BVMaternalGF,AvLact120M}
$\mathbf{S}_i$	$S_t \cup$ {BVMaternalGF,AvLact120M}
$\mathbf{S}_j$	$S_t \cup$ {ReBVMGM,ReBVM}
$\mathbf{S}_3$	$S_t \cup$ {ReBVM}
$\mathbf{S}_k$	$S_t \cup$ {ReBVMGF, MaxLactNormM}
S_l	$S_t \cup$ {AvLactNorm,MotherStockFarm,BVMaternalGF,ReBVMGF,MaxLactNormM,AvLact120M ReBVF}

7 Attribute Construction

Attribute (or variable or feature) construction is the process of deriving new attributes from the original ones. The idea [Matheus and Rendell, 1989] is to apply a set of constructive

operators to the existing attributes resulting in the construction of one or more new attributes more appropriate for the description of the target concept.

Attribute construction can have two different goals [Guyon and Elisseeff, 2003]: achieving best reconstruction of the data or being more efficient for making predictions. In this work we are interested in the (supervised) second goal. Although attribute construction pretends to discover dependences between some attributes and so it is a very domain-specific task, we focus in a data-driven approach which uses generic attribute construction methods. In this work we create new features by applying simple arithmetic functions to subsets of variables. Here, we consider only the numerical attributes included in our MV, and restrict the selected subsets to pairs of variables, so the number of constructed attributes using a given function is bounded by $O(n^2)$, n being the number of numerical attributes. Concretely, we have used the operators sum $(+)$ and product $(*)$ as functions, and given that both operations are commutative, $\frac{n^2-n}{2}$ features are generated for each one. The choice of these two operators obey to the fact that they are simple, and summation can be appropriate to combine attributes measured in *similar* scales, while product can be appropriate to combine attributes measured in different scales.

After the attribute construction, $n^2 - n$ new features have been added to the MV, n being 19. Thus, our new MV has 366 variables (the class plus 23 original attributes plus 342 constructed attributes). We have ranked the 365 predictive attributes by using SU as filter measure and we get that the two first features are: BVFather+BVMother and BVFather*BVMother. This fact is not surprising at all because two reasons:

- These two attributes were ranked as the two of greater importance with respect to the class variable (see Table 3), and
- The experts frequently use the pedigree index as predictor, which is computed as $\frac{\text{BVFather}+\text{BVMother}}{2}$.

Therefore, we have identified a good predictor by using the data driven constructive approach which agrees with the domain experts knowledge. The question now is if more interesting combinations have been identified. Table 6 shows the ranking of the first 28[8] variables by using SU as filter measure. As we can see in all of them, one of the two primary attributes BVFather or BVMother appears in the constructed variable, being in the top position those yielded by the combination of BVFather with lactation mother data, so it

[8]We have listed the first variables until the primary variables BVFather and BVMother are included

seems that more interesting combinations have been identified.

Table 6: First ranked variables using SU in the data set including generated features

SU	Variable	SU	Variable
0.43157	BVFather+BVMother	0.21643	BVFather+ReBVPGF
0.33019	BVFather*BVMother	0.21228	BVFather*AvLact120
0.25478	BVFather*AvLactNormM	0.2108	BVFather+BVParentalGM
0.25095	BVFather*MaxLact120M	0.20161	BVFather*BVParentalGM
0.25038	BVFather*AvLact120M	0.20064	BVMother+BVParentalGM
0.25037	BVFather+ReBVM	0.19916	BVFather*AvLactNorm
0.24851	BVFather	0.19514	BVFather*NLactM
0.24758	BVFather+ReBVF	0.19336	BVFather*ReBVPGF
0.24449	BVFather+ReBVPGM	0.18111	BVMother*BVParentalGM
0.24291	BVFather*ReBVF	0.17228	BVFather+BVParentalGF
0.2416	BVFather*MaxLactNormM	0.16983	BVMother*ReBVF
0.24115	BVFather*ReBVM	0.1671	BVMother+ReBVF
0.24055	BVFather+NLactM	0.16617	BVMother+ReBVM
0.23919	BVFather*ReBVPGM	0.16591	BVMother

As the current MV has 365 predictive attributes, and taking into account that 342 come from an attribute construction process we can be almost sure that many of them will be irrelevant or redundant with respect to our classification process. Therefore, the attribute selection process is even more necessary than in our previous experiments, so we have carried out the same feature subset selection process described in the previous section over our new (larger) MV. From the results (Table 7) we can draw the following conclusions:

- The accuracy of the classification has been improved considerably with respect to the results obtained over the MV without constructed attributes. Concretely, by using FSS-Wrapper, the accuracy has augmented a 3.6% (on the average) for the 4 labels problem and a 5% (on the average) for the 5 labels problem.

- The number of attributes selected is quite far of the 365 available features, being less than ten in the best cases. In general, NB needs less attributes than C4.5 and FSS-Wrapper selects less attributes than FW(∞).

- FSS-Wrapper gets better results than FW(∞) in both criteria: accuracy and number of required attributes. While the gain in accuracy is (on the average) less than one point, the number of selected attributes is drastically reduced in some cases. On the other hand, FSS-Wrapper is by far more complex because it needs to evaluate (on the average) about 4800 subsets while FSS-FW(∞) is linear in the number of attributes, that is, it evaluates exactly 365 subsets.

- With respect to the selected attributes, BVFather+BVMother is always selected, in fact, it is the only feature selected in 13 cases (see Table 7), and is complemented only

by BVFather*BVMother in 8 cases (which improves only slightly the accuracy of the classification). When more attributes are selected, as is the case of FSS-FW(∞) and FSS-Wrapper, it seems that the insertion of the constructed attributes allow the classifiers to adapt better to some areas of the solution space. However, it seems that the attributes are selected to improve the accuracy, but no semantic interpretation can be (at least easily) obtained from that selection. As an example, Table 8 shows the attributed selected in four out of the eight best cases, concretely those containing less attributes. From it, we can see that there is not a pattern in the selected attributes (apart of BVFather+BVMother), e.g., AvLacNorm is always used (which agree with our previous experiments) but combined with different attributes.

Table 7: Results obtained applying FSS over the MV enlarged by attribute construction. The subscript represents the number of attributes included in the selected subset. For some subsets we have used a superscript to identify the subset.

	4 classes				5 classes			
Selection	C4.5	C4.5(d)	NB	NB(d)	C4.5	C4.5(d)	NB	NB(d)
Best until now	76.62	76.77	78.23	75.32	70.52	71.69	71.60	67.38
FSS (FW(0))	73.67_1	73.99_2	73.96_1	74.21_2	68.32_1	69.19_2	69.68_1	69.16_1
FSS (FW(1))	73.67_1	73.99_2	73.96_1	74.21_2	68.32_1	69.19_2	69.68_1	69.16_1
FSS (FW(5))	76.80_{14}	73.99_2	73.96_1	74.21_2	68.80_3	72.56_{10}	69.68_1	69.16_1
FSS (FW(∞))	79.20_{30}	80.50_{18}	78.55_{10}	$\mathbf{80.21}_{14}$	73.76_{26}	76.26_{42}	73.83_{12}	$\mathbf{74.99}_{12}$
FSS (Wrapper)	$\mathbf{80.00}^m_8$	$\mathbf{81.90}_{18}$	$\mathbf{79.30}^y_4$	$\mathbf{80.21}^o_6$	$\mathbf{75.30}_9$	$\mathbf{76.50}_8$	$\mathbf{74.80}^r_5$	74.80_8

Table 8: Variables included in some selected subsets.

identifier	selected variables
m	BVFather+BVMother, BVFather+ReBVMGF, ReBVMGF+BVPArentalGF, AvLacNorm*AvLac120, BVMaternalGM*BVMaternalGF, ReBVPGM*AvLac120M, ReBVMGF*ReBVPGF, NLacM*MaxLacNormM
y	BVFather+AvLac120, BVFather+BVMother, BVMother+BVMaternalGF, ReBVM+REBVMGM
o	ReBVMGM, AvLac120+ReBVM, BVFather+BVMother, AvLacNorm*AvLac120, ReBVF*ReBVM, ReBVMGF*NLacM
r	AvLac120+BVMother, BVFather+BVMother, ReBVF+REBVMGM, BVParentalGM+ReBVMGF, AvLac120*ReBVF

8 Discussion

In this section we discuss the results obtained through the process described in the previous sections. First of all, during the analysis we should take into account the nature of our problem and the fact that we are dealing with a classification problem which arises by a

transformation process from a prediction task. Because of this, *artificially* created classes (frontiers) are likely ill-defined and so very good results are not expected.

From our initial classification process (Section 5) there are some points worthing to be discussed:

- When all the variables are considered, C4.5 improves (by far) the results obtained by NB. As C4.5 only uses a subset of the available attributes, this is a clear clue that variable selection is appropriate in this task.

- During the *manual* FSS process we realize that most environmental variables represents a problem because of its cardinality. A cardinality reduction process has been carried out for these variables by using KEX algorithm, which yields to better classification results specially for NB algorithm. Apart from this benefit, the output of KEX provided us with a like-clustering[9] of the herds as a function of the breeding value.

- As expected, the best results are obtained using only a subset of the available variables. The most informative predictors are parents BV but in six of the eight cases the best results are obtained when using parents breeding value, environmental variables and animal lactational data.

- The accuracy of the classification is relatively good (76% in the four labels problem and almost a 70% in the five labels one), especially if we consider the starting point (OneR and the use of all the variables). The improvement with respect to the starting point is specially remarkable in the case of NB. On the other hand, in both problems (four and five labels), the best results are achieved in the discretized case.

Despite the initial results are not bad and we have identified an interesting subset of relevant variables, we think that a finer FSS can help to improve the results in both directions. From the process described in Section 6 we remark the following points:

- The filtering process confirm us which variables are more relevant with respect to BV variable: breeding value variables and lactational data. However, many of them are redundant when considered together, and so using the filter approach alone is not a good idea.

[9]Results are not fully interpretable because many herds fall in the *unknown* category.

- The filter+wrapper approach proposed in Section 6 identifies a strong predictor subset: $S_t = \{$BVFather, BVMother, AvLact120$\}$. In fact, it gives the best results in the four labels problem and its performance is quite close to the best one in the five labels problem. Besides, it complexity is considerably smaller than wrapper FSS.

- In the remaining cases the best performance is achieved by using wrapper FSS. In all the cases the subsets selected are formed by adding to S_t two or three variables, related to grand mother breeding value, confidence on breeding values or lactational data about the animal mother. Only in one out of the eight cases environmental data is used (FatherStockFarm).

- When using FSS, C4.5 obtains similar results in both cases, discretized and non-discretized, while NB gets significantly best results in the non-discretized case. We think that a possible explanation to this fact is the like-normal distribution shape exhibited by most of the variables selected in the winner cases. On the other hand, and because of the FSS process, now NB and C4.5 have similar performance.

- Finally, one of our goals have been fulfilled, as is the identification of *relevant subsets* of variables which yields (in all the cases) better accuracy results than the cases studied in section 5.

With respect to our data-driven attribute construction process, we have induced new good predictors from the available attributes. However, it is necessary to carry out a FSS over the enlarged MV in order to identify interesting subsets. The following remarks worth in our opinion:

- The key constructed attribute is BVFather+BVMother which has been considered more relevant than the addition of BVFather and BVMother (see Table 6). This variable has shown a great classification power (see Table 7), being even slightly better than the *pedigree index* used by experts.

- There are other good constructed predictors, many of them combining BVFather with genetic or lactational data, and it is clear that they play an important role in the classification process, because the increase achieved in the accuracy (4.3% on the average in the winner cases).

- In this case wrapper FSS has shown (in general) a superior performance than the FW approach, with respect to the number of selected variables. With respect to this fact we should notice that the wrapper method used in this work allow backtracking, while FW has a greedy behavior and once a variable is included in the selected subset it cannot be removed.

Below we include the confusion matrices (Tables 9 and 10) of the best cases with respect to accuracy for the four and five labels problems. As expected, most errors are in the classes frontiers. We will use this confusion matrices to discuss about using these classifications with respect to three selection processes carried out in the ESROM scheme:

Table 9: Confusion matrix for the best result obtained (81.9% of accuracy) in the four labels problem.

	classified as			
actual	f1	f2	f3	f4
f1 ($\leq 25\%$)	667	100	4	1
f2 ($25\% - 50\%$)	63	595	107	6
f3 ($50\% - 75\%$)	0	114	570	88
f4 ($> 75\%$)	0	4	73	695

Table 10: Confusion matrix for the best result obtained (76.5% of accuracy) in the five labels problem.

	classified as				
actual	v1	v2	v3	v4	v5
v1 ($\leq 20\%$)	512	98	5	0	1
v2 ($20\% - 40\%$)	61	443	104	8	2
v3 ($40\% - 60\%$)	1	85	422	105	5
v4 ($60\% - 80\%$)	0	3	85	441	88
v5 ($> 80\%$)	0	1	4	70	543

- *Inclusion of ewes in the preliminary catalog.* When a new herd is considered for its inclusion in the SS, only those ewes which pass a threshold about morphological qualification, milk production and genetic merit, will be included in a preliminary catalog[10]. With respect to breeding value the ewe has to be in the top 50% of the population.

[10]Only second generation descendants of these preselected ewes will have the opportunity of being included in the final ESROM catalog.

This task can be directly attacked by using the four labels problem. In fact, we can collapse the confusion matrix (Table 9) into a two labels problem ($GV > 50\%$), obtaning the confusion matrix shown in the left part of Table 11. From this confusion matrix we can see that the accuracy in the classification is 92.35%, which is a high ratio, even taking into account that the target concept using during the learning stage was not this binary classification but a four labels one. On the other hand, if we want to be more prudent (with respect to the animals selected by the classifier) in this task, we can use the five labels classifier by setting our target to be $GV > 60\%$. In this case we obtain the (collapsed) confusion matrix shown in the right part of Table 11. The degree of accuracy (93%) is similar to the previous one (notice that again this not was the concept target during the learning phase), but now there are 111 of the 121 false positives which actually belong to v3 class, and so we can expect that many of them are in the percentile $50\% - 60\%$.

- *Selection of ewes as candidate mothers for the stud market.* Only ewes with BV greater or equal to the 70% will be used (by means of artificial insemination) to produce males for the stud market.

  This task can be approached in a *prudent* way by using the four labels classification, and setting $GV > 75\%$ as our target concept. Left part of Table 12 shows the collapsed confusion matrix. As we can see the accuracy is really high, 94.43%, and 88 of the 95 false positives belong actually to the f3 class, and so we can expect that many of them actually are in the $70\% - 75\%$ band. Of course, we can be even more prudent, and use five labels classification with concept target $GV > 80\%$. In this case we get a similar accuracy (94.46), but the collapsed confusion matrix (right part of Table 12) show us that 88 of the 96 false positives belong to class v4 and so it is quite likely that they are in the $70\% - 80\%$ decile.

- *Selection of ewes as mothers for ewes replacement.* When artificial insemination is used for ewes replacement, the mothers are selected from those being above the 80% percentile.

  Five labels classification can be used to approach this task, it is enough to use $GV > 80$ as target concept. The confusion matrix is the one shown in Table 12 (right), having an accuracy of 94.46%, however in this case the figure 96 represents tha actual true

positives.

Table 11: Collapsed confusion matrices for concept targets $BV > 50\%$ and $BV > 60\%$.

	classified as	
actual	$\leq 50\%$	$> 50\%$
$\leq 50\%$	1425	118
$> 50\%$	118	1426

	classified as	
actual	$\leq 60\%$	$> 60\%$
$\leq 60\%$	1731	121
$> 60\%$	93	1142

Table 12: Collapsed confusion matrices for concept targets $BV > 75\%$ and $BV > 80\%$.

	classified as	
actual	$\leq 75\%$	$> 75\%$
$\leq 75\%$	2220	95
$> 75\%$	77	695

	classified as	
actual	$\leq 80\%$	$> 80\%$
$\leq 80\%$	2373	96
$> 80\%$	75	543

To finish with this discussion, we would like to remark the fact that using the obtained classifiers should be considered as a *preliminary* and fast decision criterion, but finer tools can be use to tackle with dubious cases. Furthermore, if we want to be more conservative in our decisions, we can use C4.5 and NB as rankers, and a minimal (probability) threshold can be set in order to classify an animal with respect to a target concept. Finally, a two-level classification [Ferri et al., 2004] can be use by training a second classifier using only the dubious cases for the first one, in this way we think that those cases in the classes frontiers will be more correctly managed.

9 Concluding remarks

An study of the breeding value classification in Manchego sheep breed has been carried out in this paper. This task is one of the key points in the Selection Scheme (ESROM) used to improve the quality and production figures of Manchego sheep. Starting from the data provided by AGRAMA and following a careful data preparation process we have obtained a set of minable views (four or five classes and discretized or non-discretized attributes) which have been used to classify the breeding value by means of two classical standard algorithms: NB and C4.5. We have shown that feature selection is a key process with a twofold benefit: (1) identification of small subsets of variables to be used as predictors, and (2) an improvement in the accuracy of the classification. Furthermore, a data-driven

attribute construction process has been carried out over the numerical attributes included in the minable view. From this process some interesting attributes have been identified and also the classification accuracy has been considerably improved. Finally, we have analyzed the obtained results linking them with possible selection tasks performed inside the ESROM scheme.

For future work we plan to use more sophisticated bayesian classifiers, because even the results obtained by NB are good and competitive with those obtained by C4.5, it is clear that the class-conditional independence assumption is not true in this domain, so we plan to use Bayesian networks classifiers [Friedman et al., 1997] in order to allow (possible limited) dependences among the predictive attributes. On the other hand we plan to deal with the problem as it is in nature, that is, as a numerical prediction one.

Acknowledgments

The author is grateful to Cesar Domínguez and AGRAMA for providing the data used in the work as well as many advice. This work has been partially supported by Spanish Ministerio de Ciencia y Tecnología, Junta de Comunidades de Castilla La Mancha and FEDER under projects TIC2001-2973-CO5-05 and PBC-02-002.

References

[Berka and Bruha, 1998] Berka, P. and Bruha, I. (1998). Discretization and grouping: Preprocessing steps for data mining. In *Proceedings of Principles of Data Mining and Knowledge Discovery PKDD'98*, LNAI 1510, pages 239–245. Springer Verlag.

[CRDECM, 2004] CRDECM (2004). Consejo regulador de la denominacin especfica cordero manchego. `http://www.corderomanchego.org` (in Spanish).

[CRDOQM, 2004] CRDOQM (2004). Consejo regulador de la denominacin de origen queso manchego. `http://www.mapya.es/es/alimentacion/pags/Denominacion/htm/queso_manchego.htm` (in Spanish).

[Domingos and Pazzani, 1997] Domingos, P. and Pazzani, M. (1997). On the optimality of the simple bayesian classifier under zero-one loss. *Machine Learning*, 29(2-3):103–130.

[Duda and Hart, 1973] Duda, R. and Hart, P. (1973). *Pattern classification and scene analysis*. John Wiley and Sons.

[Elvira-Consortium, 2002] Elvira-Consortium (2002). Elvira: An environment for creating and using probabilistic graphical models. In *Probabilistic Graphical Models*.

[Farkas, 2003] Farkas, I. (2003). Special issue: Artificial intelligence in agriculture. *Computers and Electronics in Agriculture*, 39(1-3).

[Fayyad and Irani, 1993] Fayyad, U. and Irani, K. (1993). Multi-interval discretization of continuous-valued attributes for classification learning. In *Proceedings of the International Joint conference on Artificial Intelligence (IJCAI)*, pages 1022–1029. Morgan And Kaufmann.

[Fayyad et al., 1996] Fayyad, U., Piatetsky-Shapiro, G., and Smyth, P. (1996). From data mining to knowledge discovery in databases. *AI Magazine*, 17:37–54.

[Ferri et al., 2004] Ferri, C., Flach, P., and Hernández-Orallo, J. (2004). Delegating classifiers. In *Twenty-first International Conference (ICML'04)*, pages 297–304.

[Friedman et al., 1997] Friedman, N., Geiger, D., and Goldszmidt, M. (1997). Bayesian network classifiers. *Machine Learning*, 29:131–163.

[Gallego et al., 1994] Gallego, L., Torres, A., and Caja, G., editors (1994). *Cattle sheep: Manghega breed (in Spanish)*. Ediciones Mundi-Prensa.

[Guyon and Elisseeff, 2003] Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182.

[Han and Kamber, 2001] Han, J. and Kamber, M. (2001). *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers.

[Holte, 1993] Holte, R. (1993). Very simple classification rules perform well on most commonly used datasets. *Machine Learning*, 11:63–91.

[ITAP, 2001] ITAP (2001). Cattle sheep for milk production in Castilla-La Mancha (in Spanish). Technical Report Num. 52, Instituto Tcnico Agronmico Provincial.

[Kohavi, 1995] Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the International Joint conference on Artificial Intelligence (IJCAI)*, pages 1137–1145. Morgan And Kaufmann.

[Langley et al., 1992] Langley, P., Iba, W., and Thompson, K. (1992). An analysis of bayesian classifiers. In *Proceedings of the 10th National Conference on Artificial Intelligence (AAAI)*, pages 223–228. MIT Press.

[Liu and Motoda, 1998] Liu, H. and Motoda, H. (1998). *Feature Extraction Construction and Selection: a data mining perspective*. Kluwer Academic Publishers.

[Matheus and Rendell, 1989] Matheus, C. and Rendell, L. (1989). Constructive induction on decision trees. In *Proceedings of the International Joint conference on Artificial Intelligence (IJCAI)*, pages 645–650. Morgan And Kaufmann.

[Mitchell, 1997] Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill.

[Murase, 2000] Murase, H. (2000). Special issue: Artificial intelligence in agriculture. *Computers and Electronics in Agriculture*, 29(1-2).

[Quinlan, 1986] Quinlan, J. (1986). Induction of decision trees. *Machine Learning*, 1:81–106.

[Rish, 2001] Rish, I. (2001). An empirical study of the naive bayes classifier. In *Proceedings of IJCAI-01 workshop on Empirical Methods in AI*, pages 41–46.

[Witten and Frank, 2000] Witten, I. and Frank, E. (2000). *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann.