Explaining the Genetic Basis of Complex Quantitative Traits through Prediction Models

Oscar Luaces[†] José R. Quevedo[†] Miguel Pérez-Enciso[‡,*]

Jorge Díez[†] Juan José del Coz[†]

Antonio Bahamonde[†]

[†]Artificial Intelligence Center. University of Oviedo at Gijón, Asturias, Spain

[‡]Departament of Food and Animal Science,

[*]Veterinary School, Universitat Autònoma de Barcelona, 08193 Bellaterra, Spain

Institut Català de Recerca i Estudis Avançats, 08010 Barcelona, Spain

July 15, 2009

Abstract

The functional characterization of genes involved in many complex traits (phenotypes) of plants, animals or humans can be studied from a computational point of view using different tools. We propose prediction, from the Machine Learning point of view, to search for the genetic basis of these traits. However, trying to predict an exact value of a phenotype can be too difficult to obtain a confident model; but predicting an approximation, in the form of an interval of values, can be easier. We shall see that trustable and useful models can be obtained from this relaxed formulation. These predictors may be build as extensions of conventional classifiers or regressors. Although the prediction performance in both cases are similar, we show that from the classification field it is straightforward to obtain a principled and scalable method to select a reduced set of features in these genetic learning tasks. We conclude comparing the results so achieved in a real world data set of barley plants with those obtained with state-of-the-art methods used in the biological literature.

1 Introduction

There are a number of complex traits or phenotypes of plants, animals or humans that can be explained by their genetic descriptions or genotypes. One of the main purposes of modern genomics is to identify which are the genetic features (genes) responsible for the observed variability in these traits.

From a computational point of view, when we do not consider any pedigree information, genetic descriptions may be represented by vectors of feature values or genetic markers that are discrete items codifying the allelic composition, usually via SNP (Single Nucleotide Polymorphism). On the other hand, a phenotype may be either a continuous or a discrete variable.

The type of relationship between genotypes and phenotypes, at its most basic level, is simply an association between a genetic marker and a phenotype that one can measure (Lynch and Walsh, 1998). However, most traits (phenotypes) are genetically complex; they are influenced by environmental causes and involve the joint action of many markers (Mauricio, 2001). This relationship has been recently formalized in terms of *prediction* power of phenotypes from genotypes (Bedo *et al.*, 2008; Lee *et al.*, 2008; Wray *et al.*, 2007; Meuwissen *et al.*, 2001).

Besides setting a formal framework, prediction has some interesting benefits (Lee *et al.*, 2008). Thus, prediction is a natural way to assess the genetic risk of a disease in healthy individuals (Wray *et al.*, 2007). Moreover, it has been reported (Meuwissen *et al.*, 2001) that artificial selection on genetic values predicted from markers could substantially increase the rate of genetic gain in animals and plants.

In the references quoted above, the tools used for prediction are based on regression, even when the phenotype is represented by a binary variable. Regression is the obvious approach when the phenotype, the target value to be predicted, is continuous. However, frequently the genotype description is only partially available, and the environmental influence in the phenotype acts as a noise source, hindering the induction of a predictor. The consequence is that regressors may exhibit only modest performance scores, even when a strong relationship between genotype and phenotype exists.

To overcome the weaknesses of regression in this kind of learning tasks, we shall try to learn models to predict approximations to the phenotype values instead of exact values. To implement this idea, we relax the specifications of metric regression changing target points by intervals. Obviously, interval predictors are more reliable than metric regressors: the targets are broader.

There are two basic routes to make effective this approach. From the point of view of regression, to learn hypotheses whose predictions are intervals of a radius $\epsilon > 0$, we may use the so-called ϵ-insensitive loss function, the loss used by Regression Support Vector Machines.

Additionally, we shall explore a second option, based on classification. We make a simple discretization of phenotype values into intervals of equal frequency; thus, predicting one of such intervals is a classification task. Moreover, we can take advantage of the ordering of those intervals; in fact, in this framework, learning the genetic basis of a phenotype is an ordinal regression task.

Moreover, the explanatory power of these classifiers can be additionally improved if we use set-valued classifiers. These classifiers return only one class (interval) when it is sufficiently sure, but in doubtful situations they may opt for predicting two or more classes (a wider interval). Set-valued classifiers have received different names in the literature. In (Corani and Zaffalon, 2008) the authors present an algorithm to learn set-valued classifiers called Naïve Credal; it is an extension of the Naïve Bayes classifier to imprecise probabilities.

In (Alonso et al., 2008; del Coz et al., To appear in 2009) these classifiers are called nondeterministic. We used them both for multiclass classification and for ordinal regression. The aim of the so-called nondeterministic hypothesis is to predict a set of classes (consecutive in ordinal regression) as small as possible, while still containing the true class. We proved that an optimal trade-off between size of predictions and successful classifications can be derived from the set of posterior probabilities for each entry of the input space. In other words, nondeterministic classifiers are built over a deterministic classifier (which always predicts one class) that provides posterior probabilities.

In this paper we present a slight modification of the algorithms of (Alonso et al., 2008; del Coz et al., To appear in 2009) to face the genetic learning task introduced above. But the main novelty is that we establish that the 0/1 loss of a deterministic classifier is an upper bound of the nondeterministic loss of its nondeterministic counterpart. The consequence is that those tools specially devised to improve deterministic classifiers' performance can be used in nondeterministic learning tasks. More specifically, feature selection algorithms designed for deterministic classifiers can be used with nondeterministic learners to obtain a fully scalable method, suitable for dealing with large genetic data sets. Notice that there are no methods for selecting features specially devised for regressors producing intervals.

Feature selection is very important in genetic tasks. When the genetic descriptions have physical locations in the genome sequence, the set of relevant attributes (i.e. positions in the genome sequence) to predict a quantitative (continuous valued) phenotype are called QTL (quantitative trait loci).

We shall see that the search for QTLs and for prediction models can be successfully tackled with the approach presented here. For this purpose, in the last section of this paper we report a set of experiments done with a well-known data set of barley plants (Wenzl et al., 2006; Bedo et al., 2008). We studied several traits, with different strength of genetic basis. We compared the prediction scores obtained by metric and interval regression, multiclass classification, ordinal regression, and set-valued or nondeterministic classification. Additionally, the selection of QTLs was compared with state-of-the-art methods used in the biological literature.

2 Regressions and Classifications

From a formal point of view, learning tasks can be presented in the following general framework. Let $\mathcal{X}$ be an input space, and let $\mathcal{Y}$ be an output space. A learning task is given by a training set $S = \{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)\}$ drawn from an unknown distribution $\Pr(X, Y)$ from the product $\mathcal{X} \times \mathcal{Y}$. The aim of such a task is to find a hypothesis h (of a space $\mathcal{H}$ of functions from $\mathcal{X}$ to $\mathcal{Y}$) that optimizes the expected prediction performance (or risk) on samples S' independently and identically distributed (i.i.d.) according to the distribution $\Pr(X, Y)$:

$$R^{\Delta}(h) = \int \Delta(h(\boldsymbol{x}), y)\; d(\Pr(\boldsymbol{x}, y)), \tag{1}$$

where $\Delta(h(\boldsymbol{x}), y)$ is a loss function that measures the penalty due to the prediction $h(\boldsymbol{x})$ when the true value is y.

Depending on the type of the output space $\mathcal{Y}$, learning tasks have different names, and they require quite different tools to be accomplished. Thus, when $\mathcal{Y}$ is a finite set, we have classification tasks; the loss function is usually the so-called 0/1 loss. When the hypothesis is a deterministic function h, the loss is 0 when $h(\boldsymbol{x}) = y$, and it is 1 otherwise:

$$\Delta_D\,(h(\boldsymbol{x}), y) = 1 - (h(\boldsymbol{x}) = y). \tag{2}$$

In this kind of tasks the aim is to maximize the number of successful classifications.

If $\mathcal{Y}$ is a metric space (usually the set of real numbers), the learning job is usually accomplished as a regression task. The typical loss function is a similarity measure applied to the set of predictions and true values. The linear loss or absolute deviation, for instance, computes

$$\Delta_{ad}\,(h(\boldsymbol{x}), y) = |h(\boldsymbol{x}) - y|. \tag{3}$$

Correlation between predictions and true values is another typical way to measure the goodness of a hypothesis.

On the other hand, when the focus is the relative ordering of predictions and its coherence with the ordering of the true classes, the goodness of a hypothesis is established with the area under a receiver operating characteristic (ROC) curve (AUC for short). If h is a hypothesis, its loss evaluated on a test set S' is $1 - AUC$, where

$$\mathrm{AUC} = \frac{\sum_{\{i,j: y'_i > y'_j\}} \left(1_{h(\boldsymbol{x}'_i) > h(\boldsymbol{x}'_j)} + (\frac{1}{2})_{h(\boldsymbol{x}'_i) = h(\boldsymbol{x}'_j)} \right)}{\sum_{i,j} 1_{y'_i > y'_j}}. \tag{4}$$

Given a data set $S = \{(\boldsymbol{x}_i, y_i) : i = 1, \ldots, n\}$, where $\boldsymbol{x}_i$ represents the genotype and y_i is a quantitative (i.e. a real number) phenotype of an individual i, we can rewrite S discretizing the range of phenotype values in a set of k bins of equal frequency. Therefore, $\mathcal{Y} = \{1, \ldots, k\}$. The aim is to redefine phenotypes in a scale of k ordered qualitative values. In fact, we shall use $k = 5$; thus, we can see this process as a translation into a ranking scale of 5 qualitative ranks: very low, low, medium, high, very high. Rewritten in this way, S can be used as a training set of a classification learning task. But in fact, it is an ordinal regression task since the classes are not only discrete, but also ordered.

3 Nondeterministic Predictions

When the output space $\mathcal{Y}$ has a linear ordering, its elements $y \in \mathcal{Y}$ may be interpreted as ranks, and it is possible to consider intervals as possible sets of predictions. Thus, we define

Definition 1. *A nondeterministic (or set-valued) hypothesis is a function h from the input space to the set of non-empty intervals (subsets of consecutive ranks) of $\mathcal{Y}$*

$$h : \mathcal{X} \longrightarrow Intervals(\mathcal{Y}). \tag{5}$$

In the next subsections we are going to introduce two approaches to learn these kind of hypothesis, one based on classification, and other based on regression.

In nondeterministic classifications, we would like to favor those decisions of h that contain the true ranks, and a smaller rather than a larger number of ranks. In other words, we interpret the output $h(\boldsymbol{x})$ as an imprecise answer to a query about the right rank of an entry $\boldsymbol{x} \in \mathcal{X}$. Thus, the nondeterministic classification can be seen as a kind of Information Retrieval task for each entry.

On the other hand, in regression we shall make use of the loss used by Regression Support Vector Machines. The aim is to optimize the distances of true values to intervals of a given width centered in the values returned by regression hypotheses.

3.1 Measuring the Goodness of Nondeterministic Predictions

In Information Retrieval, performance is compared using different measures in order to consider different perspectives. The most frequently used are Recall (proportion of all relevant documents that are found by a search) and Precision (proportion of retrieved documents that are relevant). In symbols, in a query (i.e. an entry $\boldsymbol{x}$), if y is the true (relevant) class, Recall is defined by

$$R(h(\boldsymbol{x}), y) = 1_{y \in h(\boldsymbol{x})}. \tag{6}$$

Precision is given by

$$P(h(\boldsymbol{x}), y) = \frac{1_{y \in h(\boldsymbol{x})}}{|h(\boldsymbol{x})|}. \tag{7}$$

However, it is more informative to measure a tradeoff between Recall and Precision. The harmonic average of the two amounts is used to capture the goodness of a hypothesis in a single measure. In the weighted case, the measure is called F_β. Moreover, since F_β is bounded in $[0,1]$ and it measures the goodness, the loss (Δ_{ND}) is given by the complementary: $1 - F_\beta$. Thus, for a nondeterministic classifier h and a pair $(\boldsymbol{x}, y)$,

$$F_\beta(h(\boldsymbol{x}), y) = \frac{(1+\beta^2) \cdot P \cdot R}{\beta^2 P + R} = \frac{1+\beta^2}{\beta^2 + |h(\boldsymbol{x})|} \cdot 1_{y \in h(\boldsymbol{x})}. \tag{8}$$

Once we have the definition of F_β for individual entries, it is straightforward to extend it to a test set. So, when S' is a test set of size m, the average nondeterministic loss on it will be computed by

$$\begin{aligned} R^{\Delta_{ND}}(h, S') &= \frac{1}{m} \sum_{j=1}^{m} \Delta_{ND}(h(\boldsymbol{x'_j}), y'_j) = \frac{1}{m} \sum_{j=1}^{m} \Big(1 - F_\beta(h(\boldsymbol{x'_j}), y'_j)\Big) \\ &= \frac{1}{m} \sum_{j=1}^{m} \left(1 - \frac{1+\beta^2}{\beta^2 + |h(\boldsymbol{x'_j})|} 1_{y'_j \in h(\boldsymbol{x}'_j)}\right). \end{aligned} \tag{9}$$

It is important to realize that for a deterministic hypothesis h this amount is the average 0/1 loss, since all predictions are singletons, $|h(\boldsymbol{x})| = 1$. Thus, the nondeterministic loss used here is a generalization of the error rate of deterministic classifiers.

Furthermore, the average Recall and Precision on test sets can be similarly defined. In this case, the Recall on a test set is the proportion of times that $h(\boldsymbol{x'})$ includes y', and is thus another generalization of the deterministic accuracy. In symbols,

$$Recall(h, S') = \frac{1}{m} \sum_{j=1}^{m} 1_{y' \in h(\boldsymbol{x'})}. \tag{10}$$

Finally, from the point of view of regression, the effect of the discretization of continuous classes can be simulated somehow if we use the so-called $\underline{\epsilon\text{-insensitive}}$ loss function, the loss used by Regression Support Vector Machines. If ϵ is a positive value, this loss does not penalize predictions whose distance to true values is below ϵ; in symbols,

$$\Delta_\epsilon(h(\boldsymbol{x}), y) = \max\{0, |h(\boldsymbol{x}) - y| - \epsilon\}. \tag{11}$$

In this context, the $\underline{\text{Recall}}$ (Eq. 10) can be generalized to

$$Recall(h, \epsilon, S') = \frac{1}{m}\sum_{j=1}^{m} 1_{y' \in [h(\boldsymbol{x'})-\epsilon, h(\boldsymbol{x'})+\epsilon]}. \tag{12}$$

3.2 Deterministic and Nondeterministic Classifiers

In this section we present a couple of results that allow us to build nondeterministic from deterministic classifiers. In the general ordinal regression setting presented in Section 2, let $\boldsymbol{x}$ be an entry of the input space $\mathcal{X}$, and let us now assume that we know the posterior probabilities of ranks, $\Pr(rank = j|\boldsymbol{x})$ for $j \in \{1, \ldots, k\}$. In this context, we wish to define a nondeterministic hypothesis whose predictions are intervals of one or two consecutive ranks:

$$h_{ND}(\boldsymbol{x}) = Z \in Int_2(k) = \{I \in Intervals\{1, \ldots, k\} : |I| \leq 2\}, \tag{13}$$

where $|I|$ stands for the number of ranks included in the interval I. We shall prove that an optimal hypothesis $h_{ND}(\boldsymbol{x})$ can be computed by the Algorithm 1, a version of the algorithms presented in (Alonso $\underline{\text{et al.}}$, 2008; del Coz $\underline{\text{et al.}}$, To appear in 2009).

Proposition 1 (Correctness)**.** *If the conditional probabilities* $\Pr(j|\boldsymbol{x})$ *are known, Algorithm 1 returns the prediction* $h(\boldsymbol{x})$ *of one or two consecutive ranks that minimizes the risk given by the loss* $1 - F_\beta$ *(Eq. 9).*

Algorithm 1 Algorithm for computing the optimal prediction with one or two consecutive ranks for an entry $\boldsymbol{x}$ provided that the posterior probabilities of ranks are given

```
**Input:** individual description $\boldsymbol{x}$
**Input:** $\{\Pr(j|\boldsymbol{x}) : j = 1, \ldots, k\}$
**Input:** $\beta$: trade-off between Recall and Precision
$t = \arg\max \{\Pr(j|\boldsymbol{x}) : j = 1, \ldots, k\}$
$s = \arg\max \{\Pr(j|\boldsymbol{x}) + \Pr(j+1|\boldsymbol{x}) : j = 1, \ldots, k-1\}$
**if** $\Pr(t|\boldsymbol{x}) \geq \frac{1+\beta^2}{\beta^2+2}\big(\Pr(s|\boldsymbol{x}) + \Pr(s+1|\boldsymbol{x})\big)$ **then**
    **return** $[t, t]$
**else**
    **return** $[s, s+1]$
**end if**
```

Proof. To minimize the risk (Eq. 1), it suffices to compute

$$\Delta^{\boldsymbol{x}}_{ND}(Z) = \sum_{y \in \mathcal{Y}} \Delta_{ND}(Z, y) \Pr(y|x) \tag{14}$$

with $Z \in Int_2(k)$. Then, we only have to define

$$h(\boldsymbol{x}) = \arg\min\{\Delta^{\boldsymbol{x}}_{ND}(Z) : Z \in Int_2(k)\}. \tag{15}$$

First we shall prove that when Z is an interval, say $Z = [s_0, s_1]$, given $\boldsymbol{x}$, the value of Equation (14) can be expressed in function of the length of the interval $l = s_1 - s_0 + 1$ and the probability of the interval. In fact, with a probability of $1 - \Pr(Z|\boldsymbol{x})$, we expect a loss of 1: the true rank will not be one of the interval Z. On the other hand, with the probability of Z, the true rank will be in $h(\boldsymbol{x})$, and therefore the loss will be 1 minus the F_β of the prediction $h(\boldsymbol{x}) = Z = [s_0, s_1]$. In symbols,

$$\begin{aligned}
\Delta^{\boldsymbol{x}}_{ND}(Z) &= \left(1 - \sum_{j=s_0}^{s_1} \Pr(j|\boldsymbol{x})\right) 1 + \left(\sum_{j=s_0}^{s_1} \Pr(j|\boldsymbol{x})\right) \left(1 - \frac{1+\beta^2}{\beta^2 + l}\right) \\
&= 1 - \frac{1+\beta^2}{\beta^2+l} \sum_{j=s_0}^{s_1} \Pr(j|\boldsymbol{x}). \qquad (16)
\end{aligned}$$

Therefore, the intervals with lowest loss of length 1 (respectively, 2) can be determined only by their posterior probabilities. According to the Algorithm 1, let $[t, t]$ and $[s, s+1]$ be the best intervals of length 1 and 2 respectively. Thus, the interval with the lowest risk, given $\boldsymbol{x}$, will be $[t, t]$ if the following holds

$$1 - \frac{1+\beta^2}{\beta^2+1} \Pr(t|\boldsymbol{x}) \leq 1 - \frac{1+\beta^2}{\beta^2+2}\bigg(\Pr(s|\boldsymbol{x}) + \Pr(s+1|\boldsymbol{x})\bigg), \qquad (17)$$

and will be $[s, s+1]$ otherwise, which is equivalent to the predicate of the conditional of the Algorithm (line 6) as we wanted to prove. □

In practice, the actual role of β in Algorithm 1 is to fix the thresholds to decide the number of ranks to predict. In the experiments reported below the aim is to optimize F_1. With higher values of β in the Algorithm, the size of predictions would increase; we would obtain a higher $\underline{\text{Recall}}$ with a lower $\underline{\text{Precision}}$.

Using the nomenclature of this section, the deterministic counterpart of h_{ND} is given by

$$h_D(\boldsymbol{x}) = \arg\max\{\Pr(j|\boldsymbol{x}) : j = 1, \ldots, k\}. \qquad (18)$$

The next proposition establishes the relationship between these hypothesis in terms of the corresponding loss functions.

Proposition 2 (Bound)**.** *The risk given by the* $0/1$ *loss of the deterministic* h_D *is an upper bound of the risk given by the nondeterministic loss of the hypothesis* h_{ND}

Proof. It suffices to see that for each $\boldsymbol{x}$ (see Eq. 14),

$$\Delta_D^{\boldsymbol{x}}(h_D(\boldsymbol{x})) \geq \Delta_{ND}^{\boldsymbol{x}}(h_{ND}(\boldsymbol{x})).$$

If the size of $h_{ND}(\boldsymbol{x})$ is 1, then both loss functions return the same value. When the length of the nondeterministic prediction is 2, according to the Algorithm, there are some $s, t \in \{1, \ldots, k\}$ such that

$$h_{ND}(\boldsymbol{x}) = [s, s+1], \qquad h_D(\boldsymbol{x}) = t.$$

Again according to the Algorithm, (see Eq. 17), we have that

$$\Pr(t|\boldsymbol{x}) < \frac{1+\beta^2}{\beta^2+2}\Big(\Pr(s|\boldsymbol{x}) + \Pr(s+1|\boldsymbol{x}) \Big).$$

Hence, using (Eq. 16),

$$\Delta_D^{\boldsymbol{x}}(h_D(\boldsymbol{x})) = 1 - \Pr(h_D(\boldsymbol{x})|\boldsymbol{x}) \geq 1 - \frac{1+\beta^2}{\beta^2+2} \Pr(h_{ND}(\boldsymbol{x})|\boldsymbol{x}) = \Delta_{ND}^{\boldsymbol{x}}(h_{ND}(\boldsymbol{x})).$$

□

The consequence of this proposition is that reducing the loss of h_D is a reasonable strategy to reduce the loss of its nondeterministic counterpart, h_{ND}. Therefore, all the efforts to improve somehow nondeterministic classifiers will be supported by the deterministic part. This includes the grid search to adjust parameters, and the use of filters designed to select features in deterministic classifiers.

4 Experimental Results

To test the proposals of this paper we performed a number of experiments. The first aim of the experiments reported in this section was to compare the performance of point-valued versus interval-valued predictions. To predict intervals we used ϵ-regression, multiclass and nondeterministic classifiers. The second objective of the experiments was to test nondeterministic selection of features.

We used a publicly available data set about barley plants whose source is (Wenzl et al., 2006). However, in order to compare the results obtained, we used the version described in (Bedo et al., 2008). The genotypes considered are a subset of a collection of RFLP, DArT and SSR markers. The genotypes of individuals of barley plants in (Bedo et al., 2008) version are codified by 367 binary (0/1) features to express the presence/absence of parental allele calls.

The phenotypic data for 9 traits, measured in up to 16 different environments, were downloaded from the GrainGenes website[1], and preprocessed by the authors of (Bedo et al., 2008). These traits are: α-amylase, diastatic power, heading date, plant height, lodging, malt extract, pubescent leaves, grain protein content, and yield. The number of individuals described in this data set is 94.

In 8 times out of 9 the traits were represented by continuous values. The exception is the presence of pubescent leaves: a binary predicate. Since this trait clearly induces a classification task, we did not consider it in the experiments reported here given that the focus of this paper is to handle continuous traits. Therefore, we used the 8 continuous traits. To standardize the scores achieved by the learners used in these experiments, we used the percentiles of traits instead of their actual values. Therefore,

[1] http://wheat.pw.usda.gov/ggpages/SxM

all trait values range in the interval $[0, 100]$.

Additionally, as was explained in Section 2, we discretized the trait values into 5 bins of equal frequency. The discretized version of traits were used both to employ the filter FCBF (Yu and Liu, 2004), and to handle the data sets as classification learning tasks. Let us recall that FCBF orders the attributes of a learning task according to their relevance to induce class values. For this purpose FCBF uses the so called symmetrical uncertainty (a normalized version of the mutual information) between attribute and class values, which must be discrete, not necessarily ordered. Then the filter rejects those attributes that somehow are redundant with other attributes higher in the order of relevance. The filter is very fast, and performed very well in these experiments. In fact, FCBF has been frequently used for dealing with genetic data, see (Saeys et al., 2007).

In all cases, the performance of predictions reported in this section were estimated using a 10-fold cross validation. In all the scores we used the filter FCBF in an internal loop.

Metric Regression. First, we discuss the results achieved by metric regression. We used a popular SVM implementation for dealing with regressions tasks, LibSVM (Chang and Lin, 2001) with a linear kernel. To adjust the C parameter we performed an internal grid search (a 10-fold cross validation) with $C = 10^e$, $e \in \{-2, -1, 0, 1, 2\}$. The parameter ϵ was searched in $\{0.1, 0.75, 1.75\}$. Table 1

The first column of Table 1 shows that all traits have a modest performance. In fact, the average of all correlations between traits and predictions is only 0.72. In (Bedo et al., 2008), the authors used the fraction of variance explained to measure the goodness of the prediction models; in any case, the scores of the predictive power of phenotypes from genotypes reported in (Bedo et al., 2008) are equivalent to those reported in Table 1. Supported on these scores, the authors stated that their regression method (a variant of SVM) identified the relevant genetic features (QTL) for the barley data set that broadly coincided with known QTL locations.

Considering the hypotheses learned as a tool to order barleys according to a given phenotype, using the AUC, we measured the coherence of the orderings induced by the hypotheses and the true orderings; see Section 2. The second column of Table 1 reports these scores. Roughly speaking, AUC scores repeat the information given by correlations; in fact, the correlation between the first two columns is 0.97.

However, the most disappointing results are those concerning the prediction power of regressors. To explain the scores of the remaining columns of Table 1, let $S' = \{(\boldsymbol{x'_i}, y'_i) : i = 1, \ldots, m\}$ be a test set, and let h be a regressor. The set of residuals is then the collection of differences of the true and the predicted classes. In symbols,

$$residuals = \big\{y'_i - h(\boldsymbol{x'_i}) : i = 1, \ldots, m\big\}. \tag{19}$$

We report the average of the absolute values of residuals in the column labeled by mad (mean absolute deviation), see (Eq. 3). The residuals have normal distributions with a mean and a standard deviation shown in the last two columns of Table 1. Table 2

Using Approximated Values for Phenotypes with Deterministic Classifiers. As was mentioned above, the way used in this paper to represent phenotypes values as approximations is to discretize them into a reduced number of ordered bins. We used 5 bins of equal frequency. With the learning tasks so transformed, we conducted a number of experiments trying to explain the relationship between genotypes and phenotypes by means of multiclass classifiers and ordinal regressors given that classes (or ranks) are ordered. First we discuss the scores obtained with deterministic classifiers. In all cases we used a linear kernel.

The first two columns of Table 2 report the scores of SVOR (Chu and Keerthi, 2005), a state-of-the-art learner for ordinal regression tasks. The classifiers learned by SVOR return one rank in the 5-scale range for each individual.

The achievements of SVOR in AUC are similar to those obtained by metric regression. The proportion of successful classifications vary from 0.33 to 0.47; the average is 0.41. Since 0.20 would be the average score of a random classifier, the performance of SVOR is not very good for some traits.

The scores of multiclass classifiers are showed in the remainder columns of Table 2; they are clearly worse than those of ordinal regressors. The reason is that the classes are ordered, and ordinal regressors can take advantage of this fact. We used the Logistic Regression (LR) implementation of LibLinear, a large-scale logistic regression method (Lin et al., 2008) specially devised for handling data sets with thousands or millions of features and thousands of entries; in other words, LibLinear is an adequate learner for dealing with genetic data. To complete the comparisons, we used the multiclass classification version of LibSVM (Chang and Lin, 2001). In both cases we used a linear kernel, and we implemented an internal grid search (a 10-fold cross validation) for the parameter $C = 10^e$, $e \in \{-2, -1, 0, 1, 2\}$.

Approximated Values for Phenotypes with Nondeterministic Classifiers and Regressors. In order to improve the proportion of times that predictions include the true rank (Recall, Eq. 10), we used nondeterministic predictors. Let us first deal with nondeterministic classifiers. Thus, using the Algorithm 1, and aiming to optimize the F_1 score, we built two nondeterministic classifiers using the multiclass classifiers employed for the experiments reported in Table 2. The first one, based on Logistic Regression is called nd(LR), the second, based on SVM, is called nd(SVM).

To compute posterior probabilities of ranks we took advantage of their ordering.

Thus, if k is the number of ranks, we trained $k-1$ classifiers to learn posterior probabilities $\Pr(y \leq i|\boldsymbol{x})$ for $i = 1, \ldots, k-1$ and $\boldsymbol{x} \in \mathcal{X}$. Then, we estimate posterior probabilities of ranks using

$$\Pr(y = i|\boldsymbol{x}) = \Pr(y \leq i|\boldsymbol{x}) - \Pr(y \leq i-1|\boldsymbol{x}), \tag{20}$$

assuming that $\Pr(y \leq k|\boldsymbol{x}) = 1$ and $\Pr(y \leq 0|\boldsymbol{x}) = 0$. Some problematic cases could return negative probabilities for some ranks; in these cases, we normalize the values.

The scores in Recall (successful classifications) achieved by nondeterministic classifiers outperform the accuracy of SVOR. In 7 out of 8 datasets, nd-classifiers performed better than SVOR. Of course, this is due to the fact that nd-classifiers are allowed to predict more than one rank. But if we want to grasp the relationship between genotypes and phenotypes, the increase in successful classifications compensate for the slight increase in the size of predictions. The average radius (referred to as ϵ) of such predictions is just 12.3 in the case of nd(LR), and 14.0 for nd(SVM). Notice that SVOR used an $\epsilon = 10$: all intervals predicted by SVOR include 20 percentiles.

As was mentioned in Section 3.1, we can simulate the set-valued (or nondeterministic) approach extending metric regression to become ϵ-regression (see Eq. 11). Thus, we have tested the proportion of successful predictions (Recall, Eq. 12) with two different radiuses. With $\epsilon = 10$ we were trying to compare ϵ-regression with ordinal regression; the scores of SVOR outperform those of ϵ-regression in 5 out of 8 datasets. The differences with nd-classifiers are almost always against ϵ-regression with $\epsilon = 10$. On the other hand, the scores of nd-classifiers are lower in general than those of ϵ-regression when we used $\epsilon = 15$.

Table 3

Table 4

4.1 Feature Selection for Nondeterministic Classifiers

Using the Proposition 2 of Section 3.2, we made a first selection of the features of barley datasets using the filter FCBF. The number of features so selected varied from 12 to 17 with an average of 14.3, see Table 4. The second step performed was designed in order to obtain a scalable method able to be used in other situations where the number of features selected by the filter could still be too high. Thus, we used a simple linear search, using a 10-fold cross validation, in the ordered list of features returned by the filter. The subset selected was the one with the best deterministic accuracy, using the deterministic counterpart of the nd-classifiers (Eq. 18). The results were the same with SVM and LR, and the number of features so selected were reduced to a range of 3 to 14 with an average of 8.9.

The scores of Recall and average radius ($\bar{\epsilon}$) are quite similar to those achieved without selection, see Table 4. The features obtained for each trait are listed in Table 5. This table includes (Bedo et al., 2008) the markers selected by some state-of-the-art methods frequently used in the biological literature: Composite Interval Mapping (CIM), single Marker Regression (MR), and Statistical Machine Learning (SML). The features (markers in the biological sense) are represented by centimorgans (cM): distance along a chromosome. The features selected by our method are similar to those selected by the others methods. However, the main difference can be found in the computational principles used.

Plant height. To illustrate the differences between regression-based approaches and the nondeterministic selection method, we are going to discuss in detail the selection about the trait plant height. The prediction power of all alternative methods discussed here have poor scores in Recall. Figure 1 depicts the true plant heights percentiles (represented by bullets), and the intervals predicted (using a 10-fold cross validation) by the nondeterministic classifier (nd(LR) with selection of features) and by metric regression with $\epsilon = 15$. Notice that the size of the tubes are quite different; in the nd(LR) classifier, the average size of the radius (ϵ) is 11.8, while the regression predictor uses a constant radius of 15 percentiles. However, in both cases, around 50% of cases are not in the predicted interval.

For plant height, the first attribute selected by the nondeterministic method is 2H(118.9); it is also reported as a relevant (QTL) for this trait by the methods SML, CIM and MR (Bedo et al., 2008). The merits of this attribute can be straightforwardly explained in a discretized phenotype setting. It provides a description of individuals from which it is possible to predict the right class 39% of times. Since the attribute is binary, it is only possible to reach a 40% of successful classifications; notice that it is only possible to produce 2 different predictions while the set of ranks has 5 elements. Therefore, 2H(118.9) is an almost perfect predictor. This classifier is telling us that all (with the exception of 1 sample) barleys with percentiles in $(80, 100]$ have value 0 in 2H(118.9); and all barleys with percentiles in barleys $[0, 20]$ have value 1 in 2H(118.9). This explanation can not be drawn from the report that the correlation of 2H(118.9) and the percentile of plant height is 0.65. In this context correlation is quite opaque.

The top 5 features selected by our method are the following markers: 2H(118.9), 2H(0.0), 3H(117.4), 3H(74.5), 5H(65.9). The first, third, and fifth attributes of this list are also reported as relevant (QTL) for this trait by the methods SML, CIM and MR (Bedo et al., 2008). In general, it seems that a bit more sophisticated selection procedure (instead of a simple linear search) could remove some features from our list of QTLs. But we did not implement such a procedure in order to ensure scalability.

Table 5

Figure 1

5 Conclusions

Once it is established that there is a genetic basis for a complex behavior expressed by a measurable phenotype, the aim is to explain this relationship in a useful way. In many important cases phenotypes are represented by continuous numbers and then the genotype/phenotype relations can be typically explained using regression tools.

However, phenotypes are frequently influenced by environmental causes, and the joint action of many genetic features. In these cases, simple regression models provide predictors with poor performance. The approach presented here proposes to handle continuous phenotypes as approximations represented by intervals of percentiles. We have proved, in a real case, that the reliability of interval-valued predictors are considerably higher than that of point-valued regressors. We discussed two implementations of this idea. The first is based on regression with a loss function that uses a tolerance parameter ϵ. The second one discretizes the target values into a set of k bins of equal frequency, and then employs classifiers. However, the sizes of predicted intervals do not need to be the same for all genotypes, and this option can be incorporated if we use set-valued or nondeterministic classifiers: they may predict wider intervals for doubtful genotypes.

Additionally, handling phenotypes as approximations opens the possibility of using feature selectors of high performance, and whose complexity in terms of computation time is very low. In this sense, we have proved that deterministic classifiers can be delegated to select a set of features for nondeterministic tasks. Therefore, without heuristic jumps, we can use filters designed for deterministic classifiers to obtain a fully scalable method suitable for dealing with large genetic data sets.

Acknowledgements

The authors acknowledge the people who shared the barley data used in this paper (Wenzl et al., 2006; Bedo et al., 2008). The research reported in this paper was supported in part by the Spanish Ministerio de Ciencia e Innovación [TIN2008-06247] to Oscar Luaces, José R. Quevedo, Jorge Díez, Juan José del Coz, and Antonio Bahamonde, and [AGL2007-65563-C02-01/GAN] to Miguel Pérez-Enciso.

Author Disclosure Statement

No competing financial interests exist.

References

Alonso, J., del Coz, J. J., Díez, J., Luaces, O., and Bahamonde, A., 2008. Learning to predict one or more ranks in ordinal regression tasks. In Daelemans, W., Goethals, B., and Morik, K., eds., Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD), LNAI 5211, 39–54. Springer.

Bedo, J., Wenzl, P., Kowalczyk, A., and Kilian, A., 2008. Precision-mapping and statistical validation of quantitative trait loci by machine learning. BMC GENET 9.

Chang, C.-C. and Lin, C.-J., 2001. LIBSVM : A library for support vector machines. http://www.csie.ntu.edu.tw/∼cjlin/libsvm.

Chu, W. and Keerthi, S. S., 2005. New approaches to support vector ordinal regression. In Proceedings of the ICML'05, 145–152. Bonn, Germany.

Corani, G. and Zaffalon, M., 2008. Learning Reliable Classifiers From Small or Incomplete Data Sets: The Naive Credal Classifier 2. J MACH LEARN RES 9, 581–621.

del Coz, J. J., Díez, J., and Bahamonde, A., To appear in 2009. Learning nondeterministic classifiers. J MACH LEARN RES .

Lee, S., van der Werf, J., Hayes, B., Goddard, M., and Visscher, P., 2008. Predicting Unobserved Phenotypes for Complex Traits from Whole-Genome SNP Data. PLOS GENET 4.

Lin, C.-J., Weng, R. C., and Keerthi, S. S., 2008. Trust region newton method for logistic regression. J MACH LEARN RES 9, 627–650.

Lynch, M. and Walsh, B., 1998. Genetics and analysis of quantitative traits. Sinauer Sunderland, Ma.

Mauricio, R., 2001. Mapping quantitative trait loci in plants: uses and caveats for evolutionary biology. NAT REV GENET 2, 370–381.

Meuwissen, T., Hayes, B., and Goddard, M., 2001. Prediction of Total Genetic Value Using Genome-Wide Dense Marker Maps. GENETICS 157, 1819–1829.

Saeys, Y., Inza, I., and Larrañaga, P., 2007. A review of feature selection techniques in bioinformatics. BIOINFORMATICS 23, 2507—2517.

Wenzl, P., Li, H., Carling, J., Zhou, M., Raman, H., Paul, E., Hearnden, P., Maier, C., Xia, L., Caig, V., Ovesná, J., Cakir, M., Poulsen, D., Wang, J., Raman, R., Smith, K., Muehlbauer, G., Chalmers, K., Kleinhofs, A., Huttner, E., and Kilian, A., 2006. A high-density consensus map of barley linking DArT markers to SSR, RFLP and STS loci and agricultural traits. BMC GENOMICS 7.

Wray, N., Goddard, M., and Visscher, P., 2007. Prediction of individual genetic risk to disease from genome-wide association studies. GENOME RES 17, 1520–1528.

Yu, L. and Liu, H., 2004. Efficient Feature Selection via Analysis of Relevance and Redundancy. J MACH LEARN RES 5, 1205–1224.

List of Figures

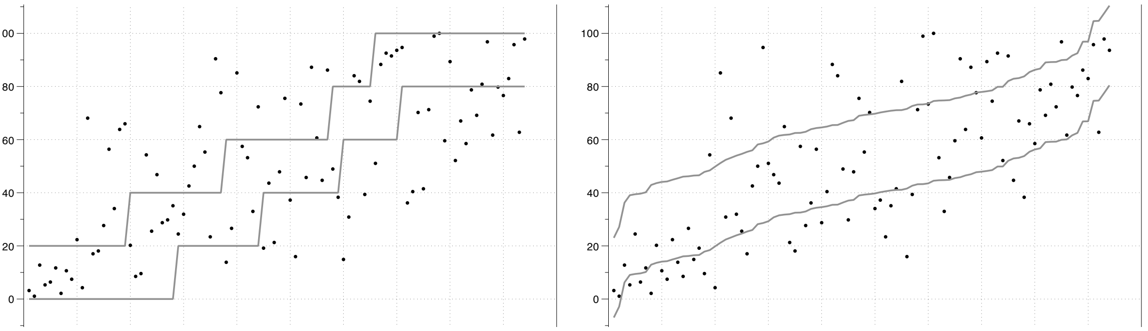

Figure 1: True plant height percentiles (●) and the intervals predicted (using a 10-fold cross validation) by <u>nd(LR)</u> with selection of features (left) and by metric regression (right) with $\epsilon = 15$. To ease the readability, the horizontal axis represents the indexes of barley samples ordered according to their predictions. Notice that therefore, the orderings of predictions in the horizontal axes are different in both pictures

$\Uparrow$

Oscar Luaces[†], José R. Quevedo[†], Miguel Pérez-Enciso[‡,*], Jorge Díez[†], Juan José del Coz[†], Antonio Bahamonde[†]

[†]Artificial Intelligence Center. University of Oviedo at Gijón, Asturias, Spain, [‡]Departament of Food and Animal Science,

[*]Veterinary School, Universitat Autònoma de Barcelona, 08193 Bellaterra, Spain

Institut Català de Recerca i Estudis Avançats, 08010 Barcelona, Spain

Figure 1 (of 1)

List of Tables

Table 1: Metric regression scores: correlations (corre.), Area Under the ROC Curve (AUC), mean absolute deviation (mad), and mean and standard deviation (std) of the residuals

				residuals	
Traits	corre.	AUC	mad	mean	std
α-amylase	0.76	0.80	14.56	0.24	17.99
diastatic	0.86	0.83	12.30	0.10	15.33
malt	0.64	0.75	17.85	0.53	22.19
heading	0.82	0.80	12.32	-0.66	16.25
height	0.71	0.77	16.35	1.20	20.07
lodging	0.74	0.79	14,00	0.23	17.90
protein	0.61	0.72	20.83	-0.43	25.76
yield	0.59	0.70	20.15	0.72	24.98
average	0.72	0.77	16.05	0.24	20.06

⇑

Oscar Luaces[†], José R. Quevedo[†], Miguel Pérez-Enciso[‡,*], Jorge Díez[†],
Juan José del Coz[†], Antonio Bahamonde[†]

[†]Artificial Intelligence Center. University of Oviedo at Gijón, Asturias,
Spain, [‡]Departament of Food and Animal Science,
[*]Veterinary School, Universitat Autònoma de Barcelona, 08193
Bellaterra, Spain
Institut Català de Recerca i Estudis Avançats, 08010 Barcelona, Spain

Table 1 (of 5)

Table 2: Classification scores for multiclass deterministic learners: ordinal regression, logistic regression (LR) and SVM. The label suc. stands for the proportion of successful classifications (Recall)

	Ordinal		Multiclass Classification			
	Regression		LR		SVM	
Traits	suc.	AUC	suc.	AUC	suc.	AUC
α-amylase	0.33	0.71	0.30	0.67	0.24	0.61
diastatic	0.50	0.83	0.34	0.73	0.42	0.75
malt	0.37	0.77	0.35	0.71	0.43	0.75
heading	0.45	0.80	0.34	0.79	0.33	0.79
height	0.47	0.79	0.39	0.73	0.43	0.80
lodging	0.37	0.77	0.35	0.71	0.35	0.73
protein	0.36	0.74	0.36	0.67	0.35	0.70
yield	0.40	0.75	0.35	0.70	0.36	0.69
average	0.41	0.77	0.35	0.71	0.36	0.73

⇑

Oscar Luaces[†], José R. Quevedo[†], Miguel Pérez-Enciso[‡,*], Jorge Díez[†], Juan José del Coz[†], Antonio Bahamonde[†]

[†]Artificial Intelligence Center. University of Oviedo at Gijón, Asturias, Spain, [‡]Departament of Food and Animal Science,

[*]Veterinary School, Universitat Autònoma de Barcelona, 08193 Bellaterra, Spain

Institut Català de Recerca i Estudis Avançats, 08010 Barcelona, Spain

Table 2 (of 5)

Table 3: Classification scores for nondeterministic classifiers and $\epsilon-$regressors with tubes (ϵ) bigger than usual. The label suc. stands for the proportion of successful classifications (Recall). The average size of predictions is represented as the average tube $\bar{\epsilon}$

	nd(SVM)		nd(LR)		Metric regression Recall with ϵ-tube (Eq. 12)	
Traits	suc.	$\bar{\epsilon}$	suc.	$\bar{\epsilon}$	$\epsilon = 10$	$\epsilon = 15$
α-amylase	0.42	14.3	0.38	12.2	0.41	0.56
diastatic	0.46	14.1	0.48	12.1	0.44	0.58
malt	0.50	13.9	0.52	12.1	0.30	0.44
heading	0.50	13.0	0.47	12.4	0.48	0.58
height	0.56	14.7	0.48	11.8	0.38	0.52
lodging	0.53	14.0	0.49	12.3	0.41	0.61
protein	0.45	14.3	0.46	13.2	0.31	0.48
yield	0.53	14.0	0.46	12.0	0.32	0.44
average	0.49	14.0	0.47	12.3	0.38	0.53

⇑

Oscar Luaces[†], José R. Quevedo[†], Miguel Pérez-Enciso[‡,*], Jorge Díez[†],
Juan José del Coz[†], Antonio Bahamonde[†]
[†]Artificial Intelligence Center. University of Oviedo at Gijón, Asturias,
Spain, [‡]Departament of Food and Animal Science,
[*]Veterinary School, Universitat Autònoma de Barcelona, 08193
Bellaterra, Spain
Institut Català de Recerca i Estudis Avançats, 08010 Barcelona, Spain

Table 3 (of 5)

Table 4: Nondeterministic classification scores with feature selection performed as described in Section 4.1

	nd(SVM)			nd(LR)			FCBF
Traits	suc	$\bar{\epsilon}$	# Attrib.	suc	$\bar{\epsilon}$	# Attrib.	# Attrib.
α-amylase	0.41	13.5	9	0.35	11.9	9	15
diastatic	0.52	14.3	11	0.48	11.8	11	15
malt	0.47	14.5	14	0.46	12.3	14	17
heading	0.57	14.2	3	0.54	12.8	3	12
height	0.55	14.5	9	0.48	12.4	9	13
lodging	0.55	14.3	9	0.54	12.3	9	15
protein	0.49	15.3	12	0.39	12.2	12	14
yield	0.53	14.4	4	0.46	12.6	4	13
average	0.51	14.4	8.9	0.46	12.3	8.9	14.3

⇑

Oscar Luaces[†], José R. Quevedo[†], Miguel Pérez-Enciso[‡,*], Jorge Díez[†], Juan José del Coz[†], Antonio Bahamonde[†]

[†]Artificial Intelligence Center. University of Oviedo at Gijón, Asturias, Spain, [‡]Departament of Food and Animal Science,

[*]Veterinary School, Universitat Autònoma de Barcelona, 08193 Bellaterra, Spain

Institut Català de Recerca i Estudis Avançats, 08010 Barcelona, Spain

Table 4 (of 5)

Table 5: List of QTLs identified using different approaches. The column labeled by nd selec. reports the ordered list of markers returned by the filter FCBF followed by the selection process described in the text. The next columns report (Bedo et al., 2008) the markers selected by Composite Interval Mapping (CIM), single Marker Regression (MR), and Statistical Machine Learning (SML). The second part of the Table lists markers selected by CIM and MR ordered by chromosome. In all cases, markers are represented by centimorgans (cM): distance along a chromosome

Trait	nd selec.	CIM	MR	SML
α-amylase	1H(119.3),	1H(126.4), 2H(84.9),	1H(96.3, 113.8)	1H(113.4), 2H(93.1),
	2H(86.0, 154.7),	3H(8.9, 25.0, 115.3),	2H(84.9), 3H(36.2)	3H(20.6),
	3H(16.7),	4H(127.1),	4H(127.1),	5H(68.2, 94.4, 183.9)
	4H(114.0, 57.0)	5H(65.7, 94.4,182.8),	5H(68.1, 135.4, 178.0)	7H(68.6, 96.4)
	5H(57.8), 6H(89.8)	7H(68.6, 96.5)	6H(23.7)	
	7H(68.6)		7H(68.6, 137.3)	
diastatic	1H(1.1, 108.1)	1H(1.8,71.7)	1H(1.5,57.6,108.1)	1H(1.5)
	2H(121.3, 35.2)	2H(10.4,35.2,135.4)	2H(83.7,131.0)	2H(134.4)
	3H(14.0)	3H(36.2,112.1)	3H(36.2,78.9)	3H(38.5)
	4H(0.0,131.4, 86.6)	4H(1.1,127.1)	4H(1.1,127.1)	4H(1.1)
	5H(89.6)	5H(79.8,140.4)	5H(68.5)	5H(80.8)
	6H(5.9),7H(68.6)	6H(80.4),7H(61.5)	6H(36.5,77.2)	7H(67.0)
			7H(67.0)	
malt	1H(3.6, 48.9, 96.3)	1H(1.7, 50.1, 132.0)	1H(1.5, 48.9)	
	2H(10.4,82.7,133.2)	2H(4.2,128.9)	2H(78.6, 93.1, 134.4)	2H(133.2)
	3H(72.3,156.7)	3H(22.8)	3H(168.9)	
	4H(85.4,131.4)	4H(90.6,145.7)	4H(90.1)	4H(86.6)
	5H(1.1,157.9)	5H(99.3, 181.6)	5H(127.2,177.2)	5H(179.1)
	7H(14.9,67.0)	7H(23.8,70.1)	6H(23.7), 7H(68.6)	7H(68.6)
heading		1H(135.9)	1H(1.1)	
	2H(91.2, 121.3)	2H(91.2, 121.3)	2H(71.9, 121.3)	2H(91.2, 117.8)
		3H(86.6, 173.3)		
		4H(78.6, 92.6)		
		5H(68.1), 6H(144.1)		
	7H(106.5)	7H(52.8)	7H(120.0)	
height	1H(31.6, 91.1)		1H(35.5, 91.1)	
	2H(0.0, 118.9)	2H822.5, 91.2, 120.7)	2H(14.8, 46.7, 118.9)	2H(117.8)
	3H(74.5, 117.4)	3H(117.4)	3H(117.4)	3H(117.4)
	4H(129.1)	4H(78.6, 92.6)	4H(142.2)	
	5H(65.9)	5H(68.1), 6H(144.1)	5H(71.5, 177.2)	5H(68.1)
	7H(55.4)	7H(52.8)	6H(149.4), 7H(53.4)	

⇑

Oscar Luaces[†], José R. Quevedo[†], Miguel Pérez-Enciso[‡,*], Jorge Díez[†],

Juan José del Coz[†], Antonio Bahamonde[†]

[†]Artificial Intelligence Center. University of Oviedo at Gijón, Asturias, Spain, [‡]Departament of Food and Animal Science,

[*]Veterinary School, Universitat Autònoma de Barcelona, 08193 Bellaterra, Spain

Institut Català de Recerca i Estudis Avançats, 08010 Barcelona, Spain

Table 5 (of 5)