GMR
The Original by FUNPEC-RP

Genome-wide classification of dairy cows using decision trees and artificial neural network algorithms

E. Rodríguez[1], J. Waissman[2], P. Mahadevan[3], C. Villa[1], B.L. Flores[1] and R. Villa[1]

[1] Institute of Engineering, Autonomous University of Baja California, Mexicali, BC, México
[2] Department of Mathematics, University of Sonora, Hermosillo, Sonora, México
[3] Biology Department, University of Tampa, Tampa, FL, United States of America

Corresponding author: E. Rodríguez
E-mail: edelmira.rodriguez.alcantar@uabc.edu.mx

Genet. Mol. Res. 18 (4): gmr18407
Received June 24, 2019
Accepted December 07, 2019
Published December 31, 2019
DOI http://dx.doi.org/10.4238/gmr18407

ABSTRACT. We compared two techniques of machine learning for the identification of cows that will be good producers of milk based on their genome-wide information. Data from a genome-wide genotyping panel, consisting of 164312 single nucleotide polymorphism markers (SNPs), within the 29 autosomal chromosomes, from 1092 Holstein cow samples were used for this study. Sample cows were divided as high-milk producers and low-milk producers based on their estimated breeding value of the 305 day average milk yield. Seven data sets were generated that grouped chromosomes with the highest number of SNPs related to milk production for prediction. Decision trees and artificial neural network algorithms were trained and tested, and the performance of prediction was computed. The mean prediction accuracy obtained with the decision tree algorithm was 92.44%, with a maximum of 94.5%, while the mean prediction accuracy obtained with the artificial neural network algorithm was 82.19%, with a maximum of 87.3%. Also, the decision tree algorithm permitted the identification

of the most dominant single nucleotide polymorphism for prediction, which is situated within a milk-related quantitative trait locus in chromosome 14. Finally, our results add new evidence to support that machine learning algorithms may be used for managing genome-wide SNP markers, for implementing classification and prediction tools in the cattle industry.

Key words: Genome-wide analysis; SNP; Decision tree; Artificial neural network; Classification; Milk yield

INTRODUCTION

After the sequencing of the bovine genome (Consortium, 2009), a revolution in High Throughput Genotyping Technologies (HTGT) was developed enabling the inspection of thousands of Single Nucleotide Polymorphisms (SNPs) in the complete genome (Matukumalli et al., 2006, 2009). HTGT is capable of capturing genetic markers with high density and highly correlated structure. They have been used to elucidate genetic structure for differentiation of cattle breeds (Villa-Angulo et al., 2009), to perform genome-wide association studies (Jiang et al., 2010; Salomón-Torres et al., 2015; González et al., 2017), and to predict genomic values for genetic improvement programs (VanRaden, 2008; Hayes et al., 2010; Pryce et al., 2012; Meuwissen et al., 2013; Yudin et al., 2016).

Traditionally, issues such as high dimensionality or highly correlated data structures, with a small number of observations and a large number of predictive variables, have hampered the analysis of large SNP datasets. Most solutions have focused on classical statistical models (i.e., Wright-Fisher model), or regression analysis for estimating relationships among variables. Recently, machine-learning methods have started to be seriously considered for dealing with large genomic datasets (Ehret et al., 2015; Li et al., 2018; Schrider et al., 2018).

To date, machine learning methods have been applied in genome-wide association studies, gene network pathway analyses and genomic prediction of phenotypic values. However, the applicability of machine learning methods, such as decision trees, and artificial neural networks, for the selection or classification of cows based on genome-wide genotypes have not been explored before.

In this study, we applied two machine-learning algorithms for classifying Holstein cows as “high-milk producer” and “low-milk producer”, based on hidden genetic patterns captured by a genome-wide genotyping panel. Machine learning (ML) is an area of artificial intelligence based on the idea that computer systems can learn by analyzing data in searching for patterns to generate a model capable of making predictions. A learning problem can be defined as the problem of improving some performance measure, through some training, when performing a task (Jordan et al., 2015). ML has two main categories: supervised learning methods (Kotsiantis et al., 2007) and unsupervised learning methods (Ghahramani, 2004). Supervised methods make use of samples with known labels for training, and the model that is generated is used to make predictions about new examples with unknown labels, whereas

unsupervised methods search for structures in datasets without using labels. Learning can be used to predict categorical data (classification) or to predict real value data, which is called regression (Libbrecht et al., 2015).

We show that ML can be used in the selection of cattle (Schrider et al., 2018), in the application of predicting the class to which a cow belongs ("high-milk producer" or "low-milk producer"). This classification is obtained by means of ML algorithms training with a dataset with known genotypes and phenotypes. In particular, we used decision trees (Rokach et al., 2014) and artificial neural network (Goodfellow et al., 2016) techniques.

MATERIAL AND METHODS

Samples dataset

The dataset used for this work was obtained from Chen's work (Chen et al., 2018). Data is publicly available from https://doi.org/10.5061/dryad.cs133. Data consist of genotype samples from 1092 Holstein cows, from a panel of 164312 SNP markers within 29 autosomal chromosomes. Genotypes are coded as 0, 1 and 2 for minor allele homozygous, heterozygous, and major allele homozygous, respectively. This is categorical information, rather than numerical values. Phenotype measures for different traits are provided.

We selected the estimated breeding value (EBV) of the 305-day average milk yield as a phenotype. In addition, we searched in the Bovine Quantitative Trait Loci (QTL) database (https://www.animalgenome.org, accessed May-25, 2019), and we selected those chromosomes containing the most significant number of QTL related to milk production, to perform the analysis.

Table 1 presents for each chromosome, the number of QTLs related to milk production. We selected chromosomes 14, 6, 5, 20 and 1 with 51, 36, 31, 25 and 23 QTL associated, respectively. Chromosome 14 contains the highest number of QTLs associated with milk production.

Table 2 presents the number of SNPs assayed in each chromosome.

Table 1. Number of QTLs related to milk production by chromosome in the cattle genome.

Chromosome	Number of QTLs related to milk production	Chromosome	Number of QTLs related to milk production	Chromosome	Number of QTLs related to milk production
1	23	11	7	21	20
2	16	12	8	22	5
3	21	13	13	23	18
4	12	14	51	24	1
5	31	15	4	25	6
6	36	16	10	26	22
7	18	17	22	27	8
8	8	18	10	28	7
9	10	19	16	29	9
10	14	20	25		

Table 2. Number (#) of SNPs assayed in each chromosome.

Chromosome	#SNPs	Chromosome	#SNPs	Chromosome	#SNPs
1	7338	11	7120	21	5871
2	7049	12	6640	22	3765
3	8064	13	6736	23	4548
4	7572	14	4004	24	3622
5	7733	15	5574	25	5773
6	5312	16	5269	26	3536
7	7465	17	4750	27	3492
8	6088	18	7579	28	3680
9	5273	19	6108	29	5004
10	5952	20	3395		

Quality Control filters

Quality Control (QC) filters were applied to initial data to ensure the overall quality of samples and a consistent set of genotypes. The filters included removal of all animals that had >20% of missing genotypes, all SNPs that violated Hardy-Weinberg frequency distribution, as applied in (Cleveland et al., 2012), and all SNPs that had a Minor Allele Frequency (MAF) <5%. After this QC procedure, the dataset consisted of 52475 SNP markers from 1092 animals, considering the 29 autosomal chromosomes.

Principal component analysis

Principal Component Analysis (PCA) is a valuable contribution of applied linear algebra (Shlens, 2003). Formally, PCA is an orthogonal linear transformation that transforms the data to a new coordinate system such that the highest variance by any projection of the data comes to lie on the first coordinate (called the first principal component), the second-highest variance on the second coordinate, and so on. PCA is theoretically the optimum transform for a given data in the least square terms. For dimensionality reduction in a dataset, we can use PCA by retaining those characteristics of the dataset that contribute most to its variance, by keeping lower-order principal components. Such low-order components often contain the “most important” aspects of the data (Alwakeel et al., 2010).

The procedure for obtaining PCAs can be described as follows: Given a matrix X^T of m samples with n features (dimensions),

$$X = \begin{pmatrix} x_1^{(1)} & \cdots & x_n^{(1)} \\ \vdots & \ddots & \vdots \\ x_1^{(m)} & \cdots & x_n^{(m)} \end{pmatrix} \quad \text{(Eq. 1)}$$

whose mean vector M and covariance C are described by $M = E(X) = [m_1, m_2, \dots, m_n]^T$ and $C = E((X - M)(X - M)^T)$, respectively. Calculate the eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_n$ and the eigenvectors $P_1, P_2, \dots, P_d$ of the covariance matrix C; arrange them according to the eigenvalue magnitude and select the d first eigenvectors to represent the n variables, $d < n$. The principal components are the vectors $P_1, P_2, \dots, P_d$ (Villa-Angulo et al., 2009).

The complexity of data was visually inspected by applying multiple correspondence analysis (MCA), which is a generalization of PCA analysis for categorical data, to each dataset. Figure 1 shows the plot of the two principal components (PC0 and PC1) for the dataset containing all sampled SNPs from chromosome 14. Blue marks (+) correspond to "high-milk producer" cows, while the red marks (*) correspond to "low-milk producer" cows. We can notice that data appears mixed, which means that there is no visual distinction of cow groups. Then, a classification method should be selected to be good enough to process data and discern cow groups.

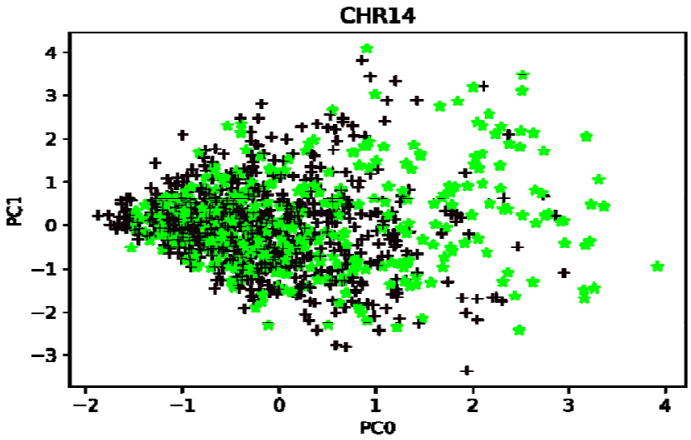

Figure 1. Plot of PC0 vs PC1 for the dataset containing all sampled SNPs from chromosome 14. Blue marks (+) correspond to "high-milk producer" cows, while the red marks (*) correspond to "low-milk producer" cows. The plot shows the high complexity of data.

Data processing using Python

We used a cluster computer running on Linux CentOS 6.7 Operative System, with a Core i5-2500 Quad-Core with 3.30 GHz processor, 4 GB RAM DDR3 1333 MHz, and Python programming language (version 3.6.8), with scikit-learn (Pedregosa et al., 2011) a machine learning library. Before starting training, each filtered dataset was coded using a one-hot encoding procedure, recommended when managing categorical information. Genotype values 0, 1 and 2 were converted to (1,0,0), (0,1,0) and (0,0,1), respectively. The Python sklearn.preprocessing package has a series of functions that facilitated the transformation.

In order to prepare the data for classification, the filtered and coded dataset was divided into two sets: One for training, which was used for adjusting the parameters of the methods, and one for the test, whose data was never used for training, and for which the real accuracy of the classification was measured. The procedure applied is as follows: the initial dataset (filtered dataset), was randomly divided into 10% for the test set and 90% for the training set. Then, the training set was divided again, but through a 5-fold cross-validation strategy, and then used for training. We applied an exhaustive search over specified parameter values of an estimator, with two ML algorithms, a decision tree classifier, and an artificial neural network.

Decision tree classifier

A Decision Tree (DT) is an efficient tool for the solution of classification problems (Xu et al., 2005). The decision tree consists of nodes and edges. There is a root node, internal nodes, and leaf nodes. The sides can be input or output to a node. The root node does not have incoming edges. All other nodes have exactly one incoming edge. An internal node is a node with outgoing edges, and the other nodes are called leaves (Rokach et al., 2014). During the learning process of the classification decision tree, the samples in each interior node are divided into subsets according to the value of an attribute. Recursively, the process is repeated in each derived node. The process is called recursive partition. The recursion is finished when a sample's subset at one node has the same target value, when splitting does not improve prediction, or when splitting is impossible because of user-defined constraints (Kim, 2016).

In the case of the DT, we used two different stop criterion values (gini (Rokach et al., 2014) and entropy (Kim, 2016)), as well as different splitter methods. And, the random state (seed) used in the splitting method was assigned values from 1 to 1000. A total number of 4000 different combinations of parameters was performed, looking for the one that yields the best results.

Artificial neural network

A feedforward backpropagation Artificial Neural Network (ANN), also called feedforward networks or multilayer perceptron, has the goal of approximating some function $f^*(X, \theta)$ and learn the value of the parameters θ that result in the best function approximation to solve the problem raised (Goodfellow et al., 2016). An ANN consists of three types of layers in general, i.e., input layer, hidden layers, and output layer. The total number of layers can be different for different applications. There are nodes (neurons) on each layer which connect to the input from the input interface or from the nodes of the previous layer, and to output layer or the next layer. Associated with the arcs which connect two nodes are weights. The weights of the ANN can be changed during the learning process. (Jenq et al., 1998) The flow begins when a sample, $X = (x_1, x_2, \dots, x_{n0})$, enters the input layer, passes through each of the hidden layers and ends when it leaves the output layer. Figure 2 shows the typical topology of a three layers ANN; where the input layer has n_0 input values, there are n_1 neurons in one hidden layer, and one neuron in the output layer. [SNP_1,…,SNP_i,…,SNP_{n0}] are input values, w_{ji} is the weight in the edge from neuron i in some layer to neuron j in the next layer. For the analysis in this study, the input to the neural network corresponds to the set of genotypes, for all SNPs, from each sample, and the output of the network corresponds to the phenotype (in a binary categorical form).

The output of neuron i (in a layer different from the initial layer), is the value that throws the activation function for the sum of all input values to the neuron i, plus the bias, $f_{act}(\sum_i x_i \cdot w_{ji} + bias)$. This value will be sent to each of the neurons in the next layer. The activation function in the neurons of the hidden layers may be different from that used in the neurons of the output layer since care must be taken that the range of output values is required. The error, which is the discrepancy between the actual output and expected output can then be used as a guide to modify the weights. The weight modifications can be done through backward propagation of errors from the last layer back to the input.

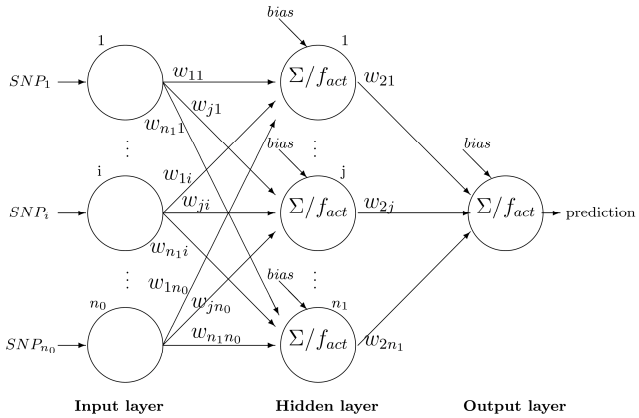

Figure 2. The typical topology of a feedforward three layers artificial neural network.

The output of neuron j (in a layer different from the initial layer), is the value that throws the activation function for the sum of all input values to the neuron j, plus the bias, $f_{act}(\sum_i \ x_i \cdot w_{ji} + bias)$. This value will be sent to each of the neurons in the next layer. The activation function in the neurons of the hidden layers may be different from that used in the neurons of the output layer since care must be taken that the range of output values is required. The error, which is the discrepancy between the actual output and expected output can then be used as a guide to modify the weights. The weight modifications can be done through backward propagation of errors from the last layer back to the input.

The learning process is an iterative operation. In the forward phase, the input patterns will be fed into the input layer of the system. Each hidden layer does the computation and forwards the activation values to the next layer in the chain of the network and eventually, the results reach the output layer. The output layer computes the errors based on the observed output and desired output. These errors will then be backpropagated, by using the backpropagation formula mentioned earlier, from the output layer through the hidden layers and finally reach the input layer. The modification of the weights can be done, during the backward phase, on the arcs connecting the nodes of the layers (Jenq et al., 1998). An outline of the aforementioned training procedure is shown in Figure 3.

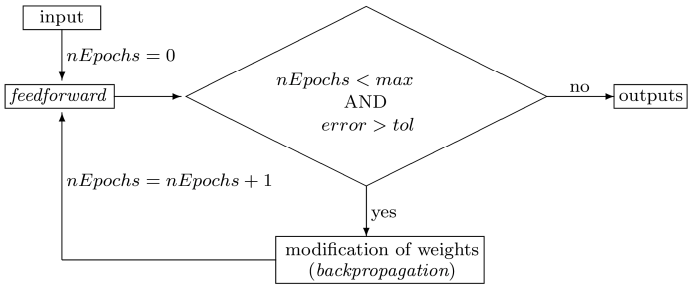

Figure 3. The training procedure for an artificial neural network algorithm. Input patterns are fed into the input layer; the hidden layer does the computation and forwards the activation values to the output layer. The output layer computes the errors based on the observed output and desired output. Errors are backpropagated by using the backpropagation formula, from the output layer through the hidden layer and finally reaching the input layer. Modification of the weights is done during the backward phase. Running the algorithm for the complete set of training patterns is called an Epoch. The complete training consists of a certain number of epochs until reaching a minimal error or a specified number of epochs.

For the implementation, we use different options in the activation function f_{act} (logistic (Rumelhart et al., 1985) and ReLu (Goodfellow et al., 2016) among others. Also, the number of neurons in the hidden layer was tested from 1 to 25 neurons. Other studied parameters were the optimization methods applied (stochastic gradient descent, adam, …), the L2 regularization term α (from 0.1 to 1), and other parameters related to the optimization algorithm (like the number of epochs and the random state). We studied 54000 combinations of hyperparameters to select the best for using with our dataset.

Samples categorization by milk production

In order to implement a classification of cows using genome-wide genotypic information, we selected the EBV for the 305-days' average milk yield as the phenotype of interest. This value was already included in the data from Chen's work (Chen et al., 2018). Its original values ranged from -8.576 to 16.838. For classification purposes, we transformed phenotype values to categorical values, labeling as "low-milk producer" when phenotype was ≤0, and "high-milk producer" when phenotype was >0. As a result, we had 670 low-milk producer and 422 high-milk producer cows.

RESULTS

To verify the applicability of DT and ANN techniques for the identification of cows that will be good producers of milk from their genomic-wide information, we generated different subsets containing SNPs from groups of chromosomes with the high number of QTLs related to milk production. We generated seven subsets, called dataset_1, dataset_2, …, dataset_7; Table 3 shows the chromosomes and the number of SNPs before and after QC filters in each dataset.

Table 3. Number of SNPs contained in each subset, before and after QC filters.

Dataset	Chromosomes	#SNPs before QC filters	#SNPs after QC filters
1	1,..., 29	164312	52475
2	14	4004	1230
3	6, 14	9316	2736
4	5, 6, 14	17049	4866
5	5, 6, 14, 20	20444	5804
6	1, 5, 6, 14, 20	27782	7901
7	1, 14	11342	3327

We applied DT and ANN algorithms and documented the accuracy of classification. Table 4 presents the results obtained with both algorithms. Next sections explain the procedure we followed for obtaining the results.

Table 4. Accuracy of classification obtained with decision tree and artificial neural network algorithms.

Dataset	Classification using decision trees		Classification using neural networks	
	Accuracy test (%)	Number of nodes in the Tree	Accuracy test (%)	Number of neurons in the hidden layer
1	93.6	67	79	8
2	91.8	117	87.3	1
3	91.8	107	83.6	7
4	90.9	93	80.9	23
5	90.9	95	81.8	18
6	93.6	93	82.7	22
7	94.5	97	80	1

Classification with decision trees

We selected each of the seven subsets of SNPs (after QC filters) described in Table 3. We applied the DT algorithm with different hyperparameters and chose the combination that produced the best accuracy for the training set. After the training phase, we used the model to predict the outcome of the test set. From Table 4, column 1 shows the dataset number, column 2 shows the classification accuracy, and column 3 shows the number of nodes in the tree. As we can see, the best classification result was obtained for dataset_7, resulting in an accuracy of 94.5%. The worst classification accuracy was obtained for dataset_4 and dataset_5, resulting in an accuracy of 90.9%. The average classification accuracy was 92.44%. The execution time varied according to the dataset used. In the case of dataset_1, which is the largest; the processing time was 32 hours.

With decision trees, when making decisions by the hierarchical analysis of the variables (taken individually), it is relatively easy to calculate the importance of each of the variables (SNPs) in the decision-making process. The *DecisionTreeClassifier* Python class performs the identification of the most important SNP. In all the datasets, the algorithm selected the SNP in position 1455997 from chromosome 14 as the most influential SNP. Its resulting influential effect was of ~0.46.

To investigate if the most influential SNP was associated with Quantitative Trait Loci (QTL), a genomic QTL search was performed using data from the Cattle QTL database (https://www.animalgenome.org/cgi-bin/QTLdb/BT/index). The SNP resulted associated with QTL #121637 related to 305-days' milk yield, spanning from 1.4 to 5.3 Mbp in chromosome 14 in the Holstein breed.

The SNP annotation was verified in NCBI resources (https://www.ncbi.nlm.nih.gov/nuccore/NC_037341.1?report=graph), looking for genes that are within the range of 110,000 bp on both sides of the SNP. Nine genes were found, from which one gene is not annotated. The unannotated gene is LOC104973964, while the rest are protein-coding genes: LOC112441461, LOC787628, LY6H, GPIHBP1 LY6E, LY6L, GML, and CYP11B1. In previous research, Yang et al (Yang et al., 2017) found the gene GPIHBP1 related to the fat milk trait in dairy cattle. Further investigation of the relationship of the listed genes with the phenotype of interest (beyond the scope of this paper) would be needed.

In order to determine if chromosome 14 is the one that has the most influence on the obtained result, we removed chromosome 14 from all datasets and repeated the training and test procedure. The results are shown in Table 5.

Table 5. Results of classification by decision trees algorithm when removing SNPs from chromosome 14 of datasets.

Chromosomes Dataset	Accuracy test (%)	Number of nodes in the tree
All except 14	73.6	149
1	70	221
6	70.9	231
5, 6	72.7	229
5, 6, 20	71.8	223
1, 5, 6, 20	72.7	193

In all cases, the results are much lower than those shown in Table 4, which indicates that the information on chromosome 14 is really important for the phenotype that we analyzed.

Classification with artificial neural networks

We applied the neural networks algorithm to the same groups of SNPs listed in Table 3, and used the same procedure looking for the combination of training-test sets that produced the best results. The topology of the neural network implemented as a multilayer perceptron with one input layer, one hidden layer, and one neuron output layer. The number of neurons in the hidden layer was gradually incremented from 1 to 25 looking for the hidden layer size that produced the best classification accuracy for the testing set. Table 4 presents the results for the seven SNP subsets. The best classification accuracy was 87.3%, obtained from dataset_2 and a hidden layer with 1 neuron. The worst classification accuracy was 79%, obtained from dataset_1 and a hidden layer of 8 neurons. The average classification accuracy was 82.19%. The approximate total execution time using the largest dataset (including the 29 chromosomes) was ~10000 hours (we simultaneously executed a run for each seed, requiring ~28 days, each).

In order to determine if DT performed statistically better than ANN, we generated two sets containing the accuracies of both algorithms and applied a Wilcoxon test, resulting in a p-value = 0.015.

DISCUSSION

The implementation of ML algorithms for categorical classification of cows from genome-wide SNP genotypes has not been reported before. Ehret et al., (2015) implemented ANN algorithms for predicting complex traits in Holstein-Friesian and German Fleckvieh cattle. They predicted milk yield, protein yield and fat yield values using a genome-wide panel of 50K SNP markers. Their aim was to use ANN to capture hidden patterns associated with milk traits in the genetic structure of cattle and predict continuous values (non-categorical). They evaluated the correlation of predicted values with real values and achieved a maximum correlation of 0.67. (Li et al., 2018) used three different ML

algorithms for identifying a subset of SNP markers for predicting genomic breeding values (GEBV). Their phenotype of interest was the live body weight from 2093 Brahman cattle. They used a traditional statistical method to estimate the GEBV of body weight using a set of 38082 SNP markers. Then, they used Random Forest, Gradient Boosting, and Extreme Gradient Boosting algorithms to identify subsets of SNPs enough information to accurately predict the GEVBs. They evaluated the correlation of predicted values with ML against statistical predicted values and achieved a maximum correlation of 0.46. By comparing these previously reported results with the results of our study (maximum accuracy of 94.5%), we can see that categorical prediction clearly outperforms the prediction of continuous values (non-categorical), even when comparing correlations is different to comparing percentages of successes and errors. It clearly demonstrates that generating a categorization induces relaxation to the problem, making categorical classification more accurate.

Here we show that machine learning successfully enables the categorical classification of high and low milk producer cows by inspecting genome-wide SNP genotypic information. The fact that the DT algorithm achieves better predictions than the ANN algorithm in much shorter processing time, makes it more suitable to implement a specific prediction tool for this kind of data. Another advantage of the DT algorithm is that it allows the identification of the most influential SNPs for classification, which can be used as a possible indication of the association of the SNP with the economic trait. Finally, our results add new evidence to support that machine learning algorithms can be used for managing genome-wide SNP markers, for implementing classification and prediction tools in the cattle industry.

ACKNOWLEDGMENTS

The Program for Professional Development Teacher, for the superior type (PRODEP), is kindly acknowledged for providing a Ph.D. scholarship to Edelmira Rodríguez Alcántar. Also, the High-Performance Computing Area of the University of Sonora (ACARUS), is thanked for the support provided in hardware and software.

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

REFERENCES

Alwakeel M and Shaahban Z (2010). Face Recognition Based on Haar Wavelet Transform and Principal Component Analysis via Levenberg-Marquardt Backpropagation Neural Network. *Eur. J. Sci. Res.* 42: 25-31

Chen Z, Yao Y, Ma P, Wang Q, et al. (2018). Haplotype-based genome-wide association study identifies loci and candidate genes for milk yield in Holsteins. *PLoS One.* 13: e0192695. https://doi.org/10.1371/journal.pone.0192695

Cleveland MA, Hickey JM and Forni S (2012). A Common Dataset for Genomic Analysis of Livestock Populations. *G3* 2: 429-435. https://doi.org/10.1534/g3.111.001453

Consortium TBH (2009). Genome-wide survey of SNP variation uncovers the genetic structure of cattle breeds. *Science.* 324: 528-532. https://doi.org/10.1126/science.1167936

Ehret A, Hochstuhl D, Gianola D and Thaller G (2015). Application of neural networks with back-propagation to genome-enabled prediction of complex traits in Holstein-Friesian and German Fleckvieh cattle. *Genet. Sel. Evol.* 47: 22. https://doi.org/10.1186/s12711-015-0097-5

Ghahramani Z (2004). Unsupervised Learning. In: Advanced lectures on machine learning. Springer, Berlin, Heidelberg, Berlin, Heidelberg, pp 72–112

González ME, González VM, Montaño MF, Medina GE, et al. (2017). Genome-wide association analysis of body conformation traits in Mexican Holstein cattle using a mix of sampled and imputed SNP genotypes. *Genet. Mol. Res.* 16:gmr16029597. https://doi.org/10.4238/gmr16029597

Goodfellow I, Bengio Y and Courville A (2016). Deep Feedforward Networks. In: Deep Learning. MIT Press

Hayes B and Goddard M (2010). Genome-wide association and genomic selection in animal breeding. *Genome.* 53: 876-883. https://doi.org/10.1139/G10-076

Jenq JJ and Li W (1998). Feedforward backpropagation artificial neural networks on reconfigurable meshes. *Futur Gener Comput Syst* 14: 313-319. https://doi.org/10.1016/s0167-739x(98)00036-3

Jiang L, Liu J, Sun D, Ma P, et al. (2010). Genome wide association studies for milk production traits in Chinese Holstein population. *PLoS One*, 5: e13661.https://doi.org/10.1371/journal.pone.0013661

Jordan MI and Mitchell TM (2015). Machine learning: Trends, perspectives, and prospects. *Science.* 349: 255-260. https://doi.org/10.1126/science.aaa8415

Kim K (2016). A hybrid classification algorithm by subspace partitioning through semi-supervised decision tree. *Pattern Recognit* 60: 157-163. https://doi.org/10.1016/j.patcog.2016.04.016

Kotsiantis SB, Zaharakis I and Pintelas P (2007). Supervised Machine Learning : A Review of Classification Techniques. In: Emerging artificial intelligence applications in computer engineering. IOS Press, pp 3-24

Li B, Zhang N, Wang YG, George AW, et al. (2018). Genomic prediction of breeding values using a subset of SNPs identified by three machine learning methods. *Front. Genet.* 9: 237. https://doi.org/10.3389/fgene.2018.00237

Libbrecht MW and Noble WS (2015). Machine learning applications in genetics and genomics. *Nat. Rev. Genet.* 16: 321. https://doi.org/10.1038/nrg3920

Matukumalli LK, Grefenstette JJ, Hyten DL, Choi IY, et al. (2006). SNP-PHAGE - High throughput SNP discovery pipeline. *BMC Bioinformatics.* 7: 468. https://doi.org/10.1186/1471-2105-7-468

Matukumalli LK, Lawley CT, Schnabel RD, Taylor JF, et al. (2009). Development and Characterization of a High Density SNP Genotyping Assay for Cattle. *PLoS One.* 4: e5350. https://doi.org/10.1371/journal.pone.0005350

Meuwissen T, Hayes B and Goddard M (2013). Accelerating Improvement of Livestock with Genomic Selection. *Annu. Rev. Anim. Biosci.* 1: 221-237. https://doi.org/https://doi.org/10.1146/annurev-animal-031412-103705

Pedregosa F, Varoquaux G, Gramfort A, Michel V, et al. (2011). Scikit-learn: Machine Learning in Python. *J. Mach Learn Res.* 12: 2825-2830

Pryce JE and Daetwyler HD (2012). Designing dairy cattle breeding schemes under genomic selection: A review of international research. *Anim. Prod. Sci.* 52:107-114. https://doi.org/10.1071/AN11098

Rokach L and Maimon OZ (2014). Data mining with decision trees, theory and applications. World Scientific Publishing Company

Rumelhart DE, Hinton GE and Williams RJ (1985). Learning internal representations by error propagation. Calif Univ San Diego La Jolla Inst Cogn Sci

Salomón-Torres R, González-Vizcarra VM, Medina-Basulto GE, Montaño-Gómez MF, et al. (2015). Genome-wide identification of copy number variations in Holstein cattle from Baja California, Mexico, using high-density SNP genotyping arrays. *Genet. Mol. Res.* 14:11848-11859. https://doi.org/10.4238/2015.October.2.18

Schrider DR and Kern AD (2018). Supervised Machine Learning for Population Genetics: A New Paradigm. *Trends Genet* .34:301-312. https://doi.org/10.1016/j.tig.2017.12.005

Shlens J (2003). A tutorial on principal component analysis. *arXiv Prepr* arXiv:1404.

VanRaden PM (2008). Efficient Methods to Compute Genomic Predictions. *J. Dairy Sci.* 91: 4414-4423. https://doi.org/10.3168/jds.2007-0980

Villa-Angulo R, Matukumalli LK, Gill CA, Choi J, et al. (2009). High-resolution haplotype block structure in the cattle genome. *BMC Genet.* 10: 19. https://doi.org/10.1186/1471-2156-10-19

Xu M, Watanachaturaporn P, Varshney PK and Arora MK (2005). Decision tree regression for soft classification of remote sensing data. *Remote Sens. Environ.* 97: 322-336. https://doi.org/10.1016/j.rse.2005.05.008

Yang J, Liu X, Wang D, Ning C, et al. (2017). Functional validation of GPIHBP1 and identification of a functional mutation in GPIHBP1 for milk fat traits in dairy cattle. *Sci. Rep.* 7: 1-10. https://doi.org/10.1038/s41598-017-08668-6

Yudin NS, Lukyanov KI, Voevoda MI and Kolchanov NA (2016). Application of reproductive technologies to improve dairy cattle genomic selection. *Russ. J. Genet. Appl. Res.* 6: 321-329. https://doi.org/10.1134/S207905971603014X

