AgriEngineering

Article

Classifier's Performance for Detecting the Pecking Pattern of Broilers during Feeding

Rogério Torres Seber [1], Irenilza de Alencar Nääs [1], Daniella Jorge de Moura [1] and Nilsa Duarte da Silva Lima [2,*]

[1] School of Agricultural Engineering, University of Campinas, Av. Cândido Rondon, 501 Barão Geraldo, Campinas 13083-875, SP, Brazil
[2] Department of Animal Science, Federal University of Roraima, Boa Vista 69300-000, RR, Brazil
* Correspondence: nilsa.silva.lima@gmail.com

Abstract: Broiler feeding is an efficient way of evaluating growth performance, health, and welfare status. This assessment might include the number of meals, meal period, ingestion rate, meal intervals, and the proportion of time spent eating. These parameters can be predicted by studying the birds' pecking activity. The present study aims to design, examine, and validate classifying algorithms to determine individual bird pecking patterns at the feeder. Broilers were reared from 1 to 42 days, with feed and water provided ad libitum. A feeder equipped with a force sensor was installed and used by the birds starting at 35 days of age, to acquire the pecking force data during feeding until 42 days. The obtained data were organized into two datasets. The first comprises 17 attributes, with the supervised attribute 'pecking detection' with two classes, and with 'non-pecking' and 'pecking' used to analyze the classifiers. In the second dataset, the attribute 'maximum value' was discretized in three classes to compose a new supervised attribute of the second dataset comprising the classes' non-pecking, light pecking, medium, and strong. We developed and validated the classifying models to determine individual broiler pecking patterns at the feeder. The classifiers (KNN, SVM, and ANN) achieved high accuracy, greater than 97%, and similar results in all investigated scenarios, proving capable of performing the task of detecting pecking.

Keywords: broiler feeding; data mining; precision livestock farming

check for updates

Citation: Seber, R.T.; de Alencar Nääs, I.; de Moura, D.J.; da Silva Lima, N.D. Classifier's Performance for Detecting the Pecking Pattern of Broilers during Feeding. *AgriEngineering* **2022**, *4*, 789–800. https://doi.org/10.3390/agriengineering4030051

Academic Editor: Andrea Pezzuolo

Received: 31 July 2022
Accepted: 5 September 2022
Published: 8 September 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Chicken pecking studies are crucial, since this attribute is related to feeding, growth, and consequent performance [1–3]. This assessment might include the number of meals, meal period, ingestion rate, meal intervals, and the proportion of time spent eating [4]. Birds learn that pecking is the action that leads to ingestion [1] and spend much time pecking the litter as a natural behavior. Assessing the bird's behavior at the feeder provides an opportunity for observation of growth measures and allows alternative management and housing strategies [5].

The pecking activities of broilers have been previously studied to predict weight gain [2]; however, it was reported that pecking follows a discontinuous-event pattern, and the actual time of contact of the beak with the feed is short and difficult to record. Automated ways of predicting those activities have been studied, such as using time-series recordings of feed levels [6], using computer vision [7,8], and using radio frequency identification (RFID) devices [9]. The pecking sound in the feeder has also been studied by [10], and it is suggested that sound analysis can be used to supervise the broiler feeding behaviors at the flock level. Xuyong et al. [9] developed a structured query language (SQL) database management system that recorded real-time broiler feeding behavior and weight gain. Faysal et al. [11] recommended internet of things (IoT) and computer vision technology for monitoring poultry farms. Such developments associated with initiatives of precision livestock farming will help transform the poultry industry.

Machine learning algorithms refer to a predictive modeling problem where a class label is predicted for a given example of input data. Such a concept has been used for several purposes. Analyzing data from wearable accelerometers using two machine learning models, K-Nearest Neighbor (KNN) and Support Vector Machine (SVM) [12], classified specific broiler behaviors. You et al. [13] described a supervised machine learning method to detect anomalies in real-time broiler body weight recorded by the system. The tested machine learning algorithms were KNN, random forest classifier (RF), SVM, and artificial neural network (ANN). The authors discovered that RF was a more effective anomaly detection algorithm for this data type. Yang et al. [14] developed a CNN-based posture change detection in untrimmed depth videos to identify dangerous sow movements inside a farrowing pen. Therefore, machine learning algorithms have been proven to be valuable tools to classify and predict animal behavior within precision livestock farming.

Improving the recognition performance of the feeding activity in the feeder using machine learning technology enables the detection of the pecking of broilers. The present study aims to devise, test, and validate classifying algorithms for the determination of individual bird pecking patterns at the feeder. This paper uses both pressure sensing on the feeding dish and a vision system data to classify the pecking actions of the broilers.

2. Materials and Methods

This experiment was carried out according to the guidelines of the Declaration of Helsinki and approved by the Animal Ethics Committee, protocol number 5278-1/2019 (CEUA-Unicamp).

2.1. Experimental Setup and Data Collection

In an experimental chamber, seven male Cobb®-500 broilers were reared from 1 to 42 days, with feed and water ad libitum. We adopted similar conditions as those recommended by the breeders when reared on-farm. The experimental chamber was equipped with a feeder, pendant drinker, temperature sensors, air humidity control, electric heater (used in the initial growth phase), air renewal, mechanical cooling, and dimmable LED lighting. A feeder equipped with a force sensor (Figure 1) was installed and used by the birds starting at 35 days of age to acquire the pecking force data during feeding until 42 days. Details about the experimental procedure are provided in the study of Seber et al. [15].

(**a**) (**b**)

Figure 1. Schematic view of the sensor (**a**) and a photograph of the broilers pecking the feed (**b**).

The data acquisition and signal processing module (QuantumX—MX840A amplifier, manufacturer Hottinger Baldwin Messtechnik—HBM) was integrated in real time using software (CatmanEasy version 4.2, manufacturer Hottinger Baldwin Messtechnik—HBM, Darmstadt, Germany). The whole system was connected to a computer for storing and processing the signals. The signal from the sensor was converted into an electrical value, and further, into a digital value. A video camera (Sharp Corporation, 470 lines with a 3.6 mm converging lens) was used to acquire and synchronize the images and signals. The video images showed when the birds pecked, and we used the signals from the sensor to check when the birds pecked and calculate the average feed intake per pecking.

2.2. Data Mining Approach

The study aimed to compare different classifiers to predict the broilers' pecking at the feeder, as shown in Figure 2.

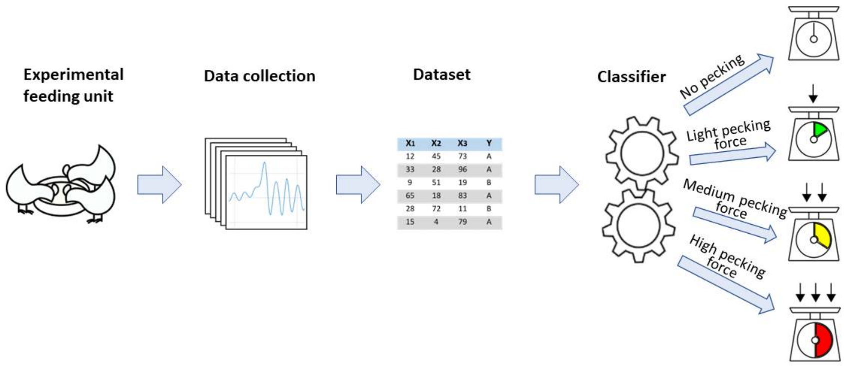

Figure 2. Schematic view of the process used in the current study.

The study included two datasets presented below in Tables 1 and 2. The first dataset comprises 17 attributes, including the supervised attribute 'pecking detection' with two classes, 'non-pecking' and 'pecking,' used for the initial study's exploratory response of the classifiers. For the construction of the second dataset, the attribute 'maximum value' (Table 1) was discretized into three classes to compose a new supervised attribute of the second dataset comprising the classes' non-pecking, light pecking, medium, and strong (Table 2), as illustrated in Figure 3.

Table 1. Attributes included in the first dataset in the detection of pecking of broilers with two classes.

Feature Number	Feature Name	Unit
1	Minimum value	-
2	Maximum value	-
3	Average value	-
4	Standard error	-
5	Variance	-
6	Standard deviation	-
7	Median	-
8	25° percentile	-
9	75° percentile	-
10	Skewness	-
11	Kurtosis	-
12	Coefficient of variation	-
13	Signal entropy	-
14	First frequency of the signal spectrum	Hertz
15	The amplitude of the first frequency of the signal spectrum	dB
16	Second frequency of the signal spectrum	Hertz
17	The amplitude of the second frequency of the signal spectrum	dB
Peck detection		Classes
Non-pecking		B_0
Pecking		B_1

The supervised attribute of the first dataset is segmented into "pecking" (majority class), with 547 observations, and "non-pecking" (minority class), with 193 observations (Table 1). The two datasets with 740 observations showed no missing data; the attributes were numerical and normalized with the Z-score criterion filter.

Table 2. Attributes included in the second dataset in the detection of pecking of broilers with four classes.

Feature Number	Attribute	Unit
1	Minimum value	-
2	Maximum value	-
3	Average value	-
4	Standard error	-
5	Variance	-
6	Standard deviation	-
7	Median	-
8	25th percentile	-
9	75th percentile	-
10	Skewness	-
11	Kurtosis	-
12	Coefficient of variation	-
13	Signal entropy	Hertz
14	First frequency of the signal spectrum	dB
15	The amplitude of the first frequency of the signal spectrum	Hertz
16	Second frequency of the signal spectrum	dB
Peck detection		Classes *
Non-pecking		B_0
Light peck		B_1
Medium peck		B_2
Strong peck		B_3

* Broiler pecking the feed plate.

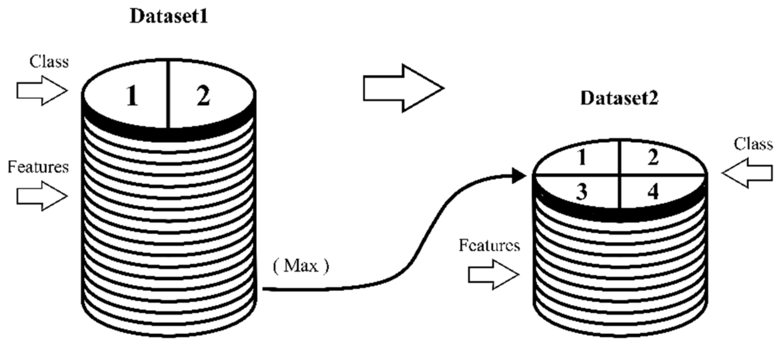

Figure 3. Schematic view of the process used to transform Dataset 1 into Dataset 2.

The supervised pecking detection attribute with four classes was constructed with the 'maximum value' attribute discretized in light, medium, and strong pecking (B_1, B_2, B_3), forming four classes with the 'non-pecking' attribute. This data set was discretized in Weka® 3.8.5 software (Waikato Environment for Knowledge Analysis, University of Waikato, Hamilton, New Zealand) by the filter ('discretize') parameterized in three Bins, giving rise to the supervised attribute with three classes of pecking intensity.

- B_1 (light pecking) from 540 samples in the force range [$-\infty$, ... , 1.89 (gf)];
- B_2 (medium pecking) from 36 samples in the force range [1.89, ... , 3.70 (gf)];
- B_3 (strong peck) from 7 samples in the force range [3.70, ... , ∞].

2.2.1. Classifiers

The classification task was performed in the Weka 3.8 software by the KNN, SMO (SVM), and ANN classifiers. The classifier choice was based on criteria established for the diversification of classifier categories. The KNN algorithm supports numerical and categorical attributes, classifies multiclass, and is used as a classifier or regressor. However, it is a slow algorithm because it does not generate a model and needs to process the entire set of observations to perform the classification, hence the designation 'lazy' [16]. On the other hand, the SVM and ANN algorithms can perform the classification faster by generating models. The SVM algorithm first classifies two classes. However, two methods

(One-vs.-One and One-vs.-All) allow, by the decomposition and training of simpler subsets, the application in multiclass tasks by supporting numerical and categorical attributes, and it can act as a classifier or regressor. In addition, the methods generate good generalizations even under a high number of attributes [17–24].

2.2.2. Application of Classifiers

Dataset 1 was the database for the KNN, SVM, and ANN algorithms applied without adjusting the classifiers' hyperparameters and reducing their dimensionality. In Dataset 2, the MLP algorithm had the hyperparameters adjusted to (_Search method (weka 'LinearNNsearch-Brute force'), _Number of neighbors (3), _Distance Metric (Manhattan), and_Distance Weight (squaredInverse)), and the classifiers were used first in the complete dataset and later in the dataset reduced by attribute selection. All models obtained were processed with '10fold cross-validation'.

2.2.3. Attribute Selection

The selection of attributes was used as a strategy to reduce the size of Dataset 2 to evaluate a possible increase in the performance of the classifiers. The attribute selection methods used were PCA, Chi-square (χ^2), Wrapper, CFS, infoGain, and GainRatio.

2.3. Classifier Performance Evaluation Metrics

In the present study, the evaluation of the classification performed by the algorithms KNN, SVM, and ANN was expressed by the metrics listed in Table 3, adapted from Hay et al. [25]. The confusion matrix is the basis for elaborating the related metrics.

Table 3. Confusion matrices for two tested classes.

	Predicted		**Total**
True	TP FN	FP TN	P N
Total	Total	Total	P + N

TP refers to the classifier's truly positive observations correctly predicted positive, while TN is the truly negative observations correctly predicted negative by the classifier. FP is the negative observations incorrectly classified as positive, and FN is the positive observations incorrectly classified as negative. Table 4 shows the evaluation metrics and the corresponding equations (Equations (1)–(9)).

Table 4. The evaluation metrics and corresponding equations.

Evaluation Metrics	**Equation**	
Accuracy, % (match rate)	$\frac{TP+TN}{P+N}$	(1)
Errors in classification, % (1—Accuracy)	$\frac{FP+FN}{P+N}$	(2)
Kappa statistics	$\frac{TP}{P}$	(3)
Sensitivity, rate of true positives (TP Rate ⇔ Sensitivity ⇔ Recall)	$\frac{TN}{N}$	(4)
Specificity, rate of true negatives	$\frac{FP}{N}$ ou $1 - \frac{TN}{N}$	(5)
False positive rate (FP Rate ⇔ 1—Specificity)	$\frac{TP}{TP+FP}$	(6)
Precision	$\frac{2 \times Precision \times Sensitivity}{Precision+Sensitivity}$	(7)
F-Measure	$\frac{TP}{P}$	(8)
MCC	$\frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP) \times (TP+FN) \times (TN+FP) \times (TN+FN)}}$	(9)

3. Results and Discussion

3.1. Dataset 1

The first step of our study was processing Dataset 1 using only two classes (non-pecking and pecking). When applying the classifiers KNN, SVM, and ANN, we obtained the results presented in Table 5.

Table 5. Overall classifier performance metrics for two classes (NB and B) using Dataset 1.

Algorithm	Accuracy (%)	Classification Error (%)	Kappa	Mean Absolute Error	Root Mean Square Error	Relative Absolute Error (%)	Root Relative Square Error (%)
KNN	99.59	0.40	0.99	0.006	0.06	1.43	14.48
SVM	99.46	0.54	0.98	0.005	0.07	1.40	16.74
ANN	99.73	0.27	0.99	0.005	0.05	1.35	12.60

Table 6 presents the performance metrics of the used algorithms applied to Dataset 1.

Table 6. Performance metrics by class for pecking.

Algorithm	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	Class
KNN	1.00	0.02	0.99	1.00	0.99	0.99	0.99	B
	0.98	0.00	1.00	0.98	0.99	0.99	0.99	NB
SVM	1.00	0.02	0.99	1.00	0.99	0.98	0.99	B
	0.98	0.00	1.00	0.98	0.99	0.98	0.99	NB
ANN	1.00	0.01	0.99	1.00	0.99	0.99	0.99	B
	0.99	0.00	1.00	0.99	0.99	0.99	0.99	NB

The KNN, SVM, and ANN models present general results and, therefore, are semi-large among the algorithms used; they provide metrics with very low false positive rates that can correspond. Considering that all were similar, it would be necessary to choose the best classification algorithm based on other criteria essential to the algorithm or following application purpose criteria. In this sense, any algorithm could be used to classify bird pecking. However, other criteria could help in choosing one of the algorithms that would fit and integrate a product as an automatic feeder, as all models presented high MCC results (0.98–0.99), indicating a measure of model quality by class [26].

The second stage of the present study processed Dataset 2 with four classes (non-pecking, light peck, medium peck, and strong peck) using the same classifiers, KNN, SVM, and ANN, and later Dataset 2 was reduced due to attribute selection.

3.2. Dataset 2

Table 7 presents the classifier performance metrics with and without feature selection for four classes (non-pecking, light peck, medium peck, and strong peck) using Dataset 2.

Table 8 presents the accuracy and kappa of the tested algorithms using four classes and applying the Dataset 2.

Table 7. Classifier performance metrics for four classes using Dataset 2.

Algorithm	Method	Accuracy (%)	Classification Error (%)	Kappa	Mean Absolute Error	Root Mean Square Error	Relative Absolute Error (%)	Root Relative Square Error (%)
KNN	No selection *	98.65	1.35	0.97	0.01	0.09	5.49	25.78
	PCA	93.38	6.62	0.86	0.03	0.18	15.04	53.17
	χ^2	98.65	1.35	0.97	0.01	0.09	5.49	25.78
	Wrapper	99.46	0.54	0.99	0.05	0.05	2.10	5.21
	CFS	98.51	1.49	0.97	0.001	0.09	4.12	25.20
	InfoGain	98.38	1.62	0.96	0.01	0.09	4.41	26.32
	GainRatio	98.38	1.62	0.96	0.01	0.09	4.41	26.32
SVM	No selection *	97.84	2.16	0.95	0.25	0.31	107.80	92.27
	PCA	90.81	9.19	0.79	0.26	0.32	110.50	95.16
	χ^2	97.84	2.16	0.95	0.25	0.31	107.80	92.27
	Wrapper	97.97	2.03	0.96	0.25	0.31	107.75	92.21
	CFS	95.68	4.32	0.90	0.25	0.32	108.67	93.22
	InfoGain	97.57	2.43	0.95	0.25	0.31	107.95	92.42
	GainRatio	97.30	2.70	0.94	0.25	0.32	108.04	92.51
ANN	No selection *	98.38	1.62	0.96	0.01	0.08	4.56	24.84
	PCA	92.84	7.16	0.84	0.05	0.17	19.55	48.86
	χ^2	97.84	2.16	0.95	0.25	0.31	107.80	92.27
	Wrapper	98.92	1.08	0.98	0.01	0.07	4.52	20.04
	CFS	99.05	0.95	0.98	0.01	0.07	5.73	20.78
	InfoGain	98.78	1.22	0.97	0.01	0.07	3.99	20.57
	GainRatio	98.51	1.49	0.97	0.01	0.08	4.74	24.52

* No selection indicates that all attributes were used in the model.

Table 8. Performance of the accuracy and kappa with Dataset 2 using four classes.

	KNN		SVM		RNN	
Method	**Accuracy (%)**	**Kappa**	**Accuracy (%)**	**Kappa**	**Accuracy (%)**	**Kappa**
No selection *	97.84	0.95	97.84	0.95	98.38	0.96
PCA	93.38	0.86	90.81	0.79	92.84	0.84
χ^2	97.84	0.95	97.84	0.95	98.38	0.96
Wrapper/KNN	99.46	0.99	-	-	-	-
Wrapper/SVM	-	-	97.97	0.96	-	-
Wrapper/ANN	-	-	-	-	98.92	0.98
CFS	98.51	0.97	95.68	0.90	99.05	0.98
InfoGain	98.38	0.96	97.57	0.95	98.78	0.97
GainRatio	98.11	0.96	97.30	0.94	98.51	0.97

* No selection indicates that all attributes were used in the model.

A previous study identified the fertility of eggs from hens using an SVM classifier, classifying the fertility of the egg into two classes (infertile egg and fertile egg); from five parameters inserted in the classifier, it obtained an average accuracy of 84.57% [27]. Another predictive model for early detection of chicken egg fertility used neural networks [28]. The results showed that the predictive model had a lower error rate than the prediction made through the manual candling process; the overall accuracy was 97%, and the validation accuracy was 93.3%.

3.3. Comparison of Attribute Selection Methods

We compared the attribute selection methods, and the results are presented in Table 9.

Table 9. Comparison attributes selected by different methods.

Methods	Selected Attributes *
No selection	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16
PCA	1, 3, 9, 10, 11, 13, 15
χ^2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16
Wrapper/KNN	2, 3, 8, 9, 12
Wrapper/SVM	2, 3, 4, 5, 8, 9, 12, 16
Wrapper/ANN	1, 3, 4, 9, 12
CFS	4, 9, 12
InfoGain	1, 3, 4, 5, 8, 9, 10, 11, 12, 16
GainRatio	1, 3, 4, 5, 8, 9, 10, 12, 15, 16

* Attributes are defined in Table 1.

3.4. Selection of Attributes

PCA (Principal Component Analysis)

Retained attributes: 1_Min, 2_Mean, 3_Stderror, 4_Variance, 5_Stddev, 6_Median, 7_25prcntil, 8_75prcntil, 9_Skewness, 10_Kurtosis, 11_Coeffvar, 14_Ampl1, 15_Freq2, 16_Ampl2.

The principle adopted for discarding variables using PCA focused on the statement that a component with a low eigenvalue (λ) is less important. Consequently, the variable that dominates this component must be less important or redundant. According to Jolliffe [29], any $\lambda \leq 0.70$ contributes very little to the explanation of the data, and it can explain 90% of the data variability with the retained attributes.

Qui-square (χ^2)

Retained attributes: 1_Min, 2_Mean, 3_Stderror, 4_Variance, 5_Stddev, 6_Median, 7_25prcntil, 8_75prcntil, 9_Skewness, 10_Kurtosis, 11_Coeffvar, 12_SigEntropy, 13_Freq1, 14_Ampl1, 15_Freq2, 16_Ampl2.

The chi-square (χ^2) method for discarding variables is described as follows. The χ^2 method evaluates the attributes individually, using this measure, concerning the meta-attribute. The higher the χ^2 value, the more likely it is that the variables (attribute and class) are correlated. There are two hypotheses:

H0: *there is no association between attributes (independence)*;

H1: *there is an association between the attributes*.

The null hypothesis H0 is rejected if χ^2 is greater than the critical value provided by a statistical table. For one degree of freedom, the critical value is 3.841.

Wrapper/KNN

Retained attributes: 2_Mean, 3_Stderror, 8_75prcntil, 9_Skewness, 12_SigEntropy.

Wrapper/SVM

Retained attributes: 2_Mean, 3_Stderror, 4_Variance, 5_Stddev, 8_75prcntil, 9_Skewness, 12_SigEntropy, 16_Ampl2.

Wrapper/ANN

Retained attributes: 1_Min, 3_Stderror, 4_Variance, 9_Skewness, 12_SigEntropy.

The Wrapper method evaluates sets of attributes using machine learning algorithms. The algorithm works as a black box to find the best subsets of attributes, being an approach dependent on the machine learning algorithm. The compatibility of the attribute selection algorithm with the classification algorithm is a requirement for the Wrapper method.

CFS (Correlation Feature Selection)

The CFS presented the attributes' 4_Variance, 9_Skewness, and 12_SigEntropy', indicating that these attributes correlate highly with the response attribute.

According to the CFS method, a set of attributes is considered good if it has two characteristics; the first is to contain attributes that are highly correlated with the meta-attribute, and the second is to contain attributes that are not correlated with each other. The attribute selection methods CFS and Wrapper formed sets of attributes with very similar characteristics.

InfoGain

Ranked attributes: 12_SigEntropy, 4_Variance, 5_Stddev, 10_Kurtosis, 3_Stderror, 1_Min, 9_Skewness, 8_75prcntil, 11_Coeffvar, 16_Ampl2, 15_Freq2, 2_Mean, 13_Freq1, 14_Ampl1, 6_Median, and 7_25prcntil.

GainRatio

Ranked attributes: 12_SigEntropy, 5_Stddev, 4_Variance, 10_Kurtosis, 3_Stderror, 1_Min, 8_75prcntil, 9_Skewness, 15_Freq2, 16_Ampl2, 11_Coeffvar, 2_Mean, 14_Ampl1, 6_Median, 13_Freq1, and 7_25prcntil.

The InfoGain and GainRatio attribute selection methods are methods of calculating and descending the ordering of attributes by gaining information. However, the InfoGain method is sensitive to attributes with many samples, which can cause a bias in selecting attributes. The GainRatio method attempts to minimize the sensitivity to attributes with many samples. The InfoGain and GainRatio methods only calculate and rank attributes from highest to lowest information gain. The criterion for selecting the attributes depends entirely on the analyst respecting the hierarchy of values; it chooses the attribute's cut-off point, starting from the lowest to the highest. The criterion adopted for selection was to exclude the six attributes with the lowest information gain.

The difference in accuracy and precision by class presented slight variation. Observing the MCC and then arbitrating the computational cost as a criterion to select the best classifier, we can infer that the SVM is the best classifier. The current study presents the application of three classifiers with very different algorithms (KNN, SVM, and ANN). The KNN classifier uses the entire dataset to perform the classification; for each new classification, it must calculate the distance from the new sample to the entire existing dataset. This implies the maximum computational effort for each new sampling. The ANN classifier demands a great computational effort to train the model and can classify the same sample differently. The SVM classifier prepares a deterministic model for a database, and each new sample uses the same model for the sample classification, which is more straightforward. The computational cost also depends on the number of attributes needed to characterize a sample. When the classifier does not significantly degrade the performance, when using the smallest possible number of attributes, the best classifier for this study is obtained [22,26,27,30–32].

Venkatesan et al. [31] used the SVM algorithm to perform signal processing in different application areas. When classifying arrhythmic beats, results indicated that the performance of the SVM classifier was better than other classifiers based on machine learning. Another approach of digital image processing with a minimum number of resources compared to existing systems considers computer vision based on a microcontroller for classifying tomatoes, detecting levels of ripeness and defects due to diseases using the SVM classifier-obtained experimental results and comparative analyses with similar methods; the effectiveness of the proposed system was proven over existing systems in the sorting and grading of tomatoes [32]. In another study that used SVM, the classification of broad and narrow leaf plants was conducted by the SVM algorithm for weed discrimination. In the results, the accuracies were compared with a conventional method of data aggregation based on the evaluation of Vegetation Indices by Normalized Difference (NDVIs) considering two different wavelengths; the results showed that using the Gaussian kernel SVM provided better discrimination accuracy than that obtained using the discrete NDVI-based aggregation algorithm [30].

4. Conclusions

We developed and validated the classifying models to determine individual broiler pecking patterns at the feeder. In all tested scenarios, the classifiers performed similarly.

Due to its use of computational time, we suppose that the best classifier was the SVM, as this classifier is swift and overcomes the other tested classifiers in terms of time taken to conduct the observations.

Observing the results obtained for Dataset 1, whose performance evaluation metrics (accuracy and Kappa) of the KNN, SVM, and ANN classifiers presented very close and high values (99% and 0.9, respectively), we concluded that there was no significant difference between the algorithms in the classification task. In addition, with the algorithms able to be classified with very high accuracy, it was not necessary to perform the selection of attributes.

The accuracy of the same classifiers (KNN, SVM, and ANN) trained by Dataset 2, derived from Dataset 1, was slightly lower (97%), which motivated us to apply attribute selection techniques to explore possible improvements in overall performance by observing the metrics (accuracy and kappa). The strategy of a benchmark for selecting attributes was successful, as the accuracy and kappa values rose for the three classifiers: the KNN (from 97.84% and 0.95 to 99.46% and 0.99), the SVM (from 97.84% and 0.95 to 97.97% and 0.99) and the ANN (from 98.38% and 0.96 to 98.92% and 0.98). Although the KNN classifier has obtained the highest accuracy value, kappa is the classifier with the disadvantage of having the highest computational cost.

Author Contributions: Conceptualization, R.T.S., N.D.d.S.L. and I.d.A.N.; methodology, R.T.S., N.D.d.S.L. and I.d.A.N.; software, R.T.S.; validation, R.T.S. and N.D.d.S.L.; formal analysis, R.T.S. and N.D.d.S.L.; investigation, R.T.S., N.D.d.S.L., D.J.d.M. and I.d.A.N.; data curation, R.T.S.; writing—original draft preparation, R.T.S.; writing—review and editing, R.T.S., N.D.d.S.L., D.J.d.M. and I.d.A.N.; visualization, R.T.S., N.D.d.S.L. and I.d.A.N.; supervision, D.J.d.M. and I.d.A.N.; project administration, R.T.S. All authors have read and agreed to the published version of the manuscript.

Funding: Coordination of Improvement of Higher Education Personnel (CAPES): funding code 001.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data will be available upon request.

Acknowledgments: The authors are grateful to the Coordination of Improvement of Higher Education Personnel (CAPES) for the scholarship.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

ANN	Artificial Neural Network
CEUA	Ethics Committee on the Use of Animals
CFS	Correlation Feature Selection
CNN	Convolutional Neural Network
FN	False Negative
FP	False Positive
IoT	Internet of things
KNN	K-Nearest Neighbor
LED	Ligth Emiting Diode
N	Total negative classification
NDVI	Normalized Difference Vegetation Index
P	Total positive classification
PCA	Principal Component Analysis
RF	Random Forest
RFID	Radio Frequency Identification
SQL	Structured Query Language
SVM	Support Vector Machine
TN	Truly Negative
TP	Truly Positive

References

1. Hogan, J.A. Pecking and feeding in chicks. *Learn. Motiv.* **1984**, *15*, 360–376. [CrossRef]
2. Yo, T.; Vilarino, M.; Faure, J.M.; Picard, M. Feed pecking in young chickens: New techniques of evaluation. *Physiol. Behav.* **1997**, *61*, 803–810. [CrossRef]
3. Neves, D.P.; Mehdizadeh, S.A.; Tscharke, M.; de Alencar Nääs, I.; Banhazi, T.M. Detection of flock movement and behaviour of broiler chickens at different feeders using image analysis. *Inf. Process. Agric.* **2015**, *2*, 177–182. [CrossRef]
4. Cook, R.N.; Xin, H.; Nettleton, D. Effects of cage stocking density on feeding behaviors of group-housed laying hens. *Trans. ASABE* **2006**, *49*, 187–192. [CrossRef]
5. Gates, R.S.; Xin, H. Comparative analysis of measurement techniques of feeding and drinking behaviour of individual poultry subjected to warm environmental condition. In Proceedings of the ASABE International Meeting, Sacramento, CA, USA, 29 July–1 August 2001; ASAE Paper no. 014033.
6. Gates, R.S.; Xin, H. Extracting poultry behavior from time-series weigh scale records. *Comput. Electron. Agric.* **2008**, *62*, 8–14. [CrossRef]
7. Youssef, A.; Exadaktylos, V.; Berckmans, D.A. Towards real-time control of chicken activity in a ventilated chamber. *Biosyst. Eng.* **2015**, *135*, 31–43. [CrossRef]
8. Li, G.; Zhao, Y.; Purswell, J.L.; Du, Q.; Chesser, G.D., Jr.; Lowe, J.W. Analysis of feeding and drinking behaviors of group-reared broilers via image processing. *Comput. Electron. Agric.* **2020**, *175*, 105596. [CrossRef]
9. Tu, X.; Du, S.; Tang, L.; Xin, H.; Wood, B. A real-time automated system for monitoring individual feed intake and body weight of group housed turkeys. *Comput. Electron. Agric.* **2011**, *75*, 313–320. [CrossRef]
10. Aydin, A.; Berckmans, D. Using sound technology to automatically detect the short-term feeding behaviours of broiler chickens. *Comput. Electron. Agric.* **2016**, *121*, 25–31. [CrossRef]
11. Faysal, M.A.H.; Ahmed, M.R.; Rahaman, M.M.; Ahmed, F. A Review of groundbreaking changes in the poultry industry in Bangladesh using the internet of things (IoT) and computer vision technology. In Proceedings of the International Conference on Automation, Control and Mechatronics for Industry 4.0, Rajshahi, Bangladesh, 8–9 July 2021; pp. 1–6.
12. Yang, X.; Zhao, Y.; Street, G.M.; Huang, Y.; To, S.F.; Purswell, J.L. Classification of broiler behaviours using triaxial accelerometer and machine learning. *Animals* **2021**, *15*, 100269. [CrossRef]
13. You, J.; Lou, E.; Afrouziyeh, M.; Zukiwsky, N.M.; Zuidhof, M.J. A supervised machine learning method to detect anomalous real-time broiler breeder body weight data recorded by a precision feeding system. *Comput. Electron. Agric.* **2021**, *185*, 106171. [CrossRef]
14. Yang, X.; Zheng, C.; Zou, C.; Gan, H.; Li, S.; Huang, S.; Xue, Y. A CNN-based posture change detection for lactating sow in untrimmed depth videos. *Comput. Electron. Agric.* **2021**, *185*, 106139. [CrossRef]
15. Seber, R.T.; Moura, D.J.D.; Lima, N.D.D.S.; Nääs, I.D.A. Smart Feeding Unit for Measuring the Pecking Force in Farmed Broilers. *Animals* **2021**, *11*, 864. [CrossRef] [PubMed]
16. Aha, D.W.; Kibler, D.; Albert, M.K. Instance-based learning algorithms. *Mach. Learn.* **1991**, *6*, 37–66. [CrossRef]
17. Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* **1995**, *20*, 273–297. [CrossRef]
18. Hearst, M.A.; Dumais, S.T.; Osuna, E.; Platt, J.; Scholkopf, B. Support vector machines. *IEEE Intell. Syst. Appl.* **1998**, *13*, 18–28. [CrossRef]
19. Platt, J.C. Sequential Minimal Optimization: A Fast Algorithm for Training Suppor Vector Machines. In *Advances in Kernel Methods-Support Vector Learning*; Scholkopf, B., Burges, C.J.C., Smola, A.J., Eds.; M.I.T. Press: Cambridge, MA, USA, 1999; pp. 185–208.
20. Burges, C.J.C. A Tutorial on Support Vector Machines for Pattern Recognition. *Data Min. Knowl. Discov.* **1998**, *2*, 121–167. [CrossRef]
21. Vapnik, V. *Statistical Learning Theory*; Wiley: New York, NY, USA, 1998.
22. Kaul, A.; Raina, S. Support vector machine versus convolutional neural network for hyperspectral image classification: A systematic review. *Concurr. Comput.* **2022**, *34*, e6945. [CrossRef]
23. Haykin, S.; Lippmann, R. Neural networks, a comprehensive foundation. *Int. J. Neural Syst.* **1994**, *5*, 363–364.
24. Abiodun, O.I.; Jantan, A.; Omolara, A.E.; Dada, K.V.; Umar, A.M.; Linus, O.U.; Arshad, H.; Kazaure, A.A.; Gana, U.; Kiru, M.U. Comprehensive Review of Artificial Neural Network Applications to Pattern Recognition. *IEEE Access* **2019**, *7*, 158820–158846. [CrossRef]
25. Han, J.; Kamber, M.; Pei, J. *Data Mining: Concepts and Techniques*, 3rd ed.; Elsevier: Waltham, MA, USA, 2012; pp. 364–368.
26. Chicco, D.; Jurman, G. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genom.* **2020**, *21*, 1–13. [CrossRef] [PubMed]
27. Saifullah, S.; Suryotomo, A.P. Identification of chicken egg fertility using SVM classifier based on first-order statistical feature extraction. *arXiv* **2022**, arXiv:2201.04063.
28. Fadchar, N.A.; Dela Cruz, J.C. Prediction Model for Chicken Egg Fertility Using Artificial Neural Network. In Proceedings of the IEEE 7th International Conference on Industrial Engineering and Applications (ICIEA), Bankok, Thailand, 16–21 April 2020; pp. 916–920. [CrossRef]
29. Jolliffe, I.T. Discarding variables in a principal component analysis. I: Artificial data. *J. R. Stat. Soc. Ser. C Appl. Stat.* **1972**, *21*, 160–173. [CrossRef]

30. Akbarzadeh, S.; Paap, A.; Ahderom, S.; Apopei, B.; Alameh, K. Plant discrimination by Support Vector Machine classifier based on spectral reflectance. *Comput. Electron. Agric.* **2018**, *148*, 250–258. [CrossRef]
31. Venkatesan, C.; Karthigaikumar, P.; Paul, A.; Satheeskumaran, S.; Kumar, R. ECG signal preprocessing and SVM classifier-based abnormality detection in remote healthcare applications. *IEEE Access* **2018**, *6*, 9767–9773. [CrossRef]
32. Kumar, S.D.; Esakkirajan, S.; Bama, S.; Keerthiveena, B. A microcontroller based machine vision approach for tomato grading and sorting using SVM classifier. *Microprocess. Microsyst.* **2020**, *76*, 103090. [CrossRef]