

Acta Technologica Agriculturae 2
Nitra, Slovaca Universitas Agriculturae Nitriae, 2026, pp. 89–97

SMART BEAN ANALYSIS: RAPID DETECTION WITH SPECTRAL DATA AND DEEP LEARNING

Raziyeh POURDARBANI^{*1}, Sajad SABZI^{*2}, Dorrin SOTOUDEH³, Mohammadreza AHMADITESHNIZI³,
Nadia SAADATI¹, Mario HERNANDEZ-HERNANDEZ⁴

¹University of Mohaghegh Ardabili, Faculty of Agricultural and Natural Resources, Department of Biosystems Engineering, Ardabil, Iran

²Gorgan University of Agricultural Sciences and Natural Resources, Faculty of Water and Soil Engineering, Department of Biosystems Engineering, Gorgan, Iran, s.sabzi@gau.ac.ir

³Sharif University of Technology, Faculty of Engineering, Department of Computer Engineering, Tehran, Iran, dorrin.sotoudeh@sharif.edu (D.S.), mohammadreza.ahmaditeshnizi@sharif.edu (M.A.)

⁴Autonomous University of Guerrero, Faculty of Engineering, Department of Computing, Chilpancingo, Guerrero, México, mhernandezh@uagro.mx

*correspondence: r_pourdarbani@uma.ac.ir

Beans have become a popular food choice in most countries around the world due to their nutritional value and their high protein, fibre, and mineral content. However, like other high-consumption products, beans are not immune to food adulteration. Fresh beans are more popular and more expensive due to their high quality and short cooking time, which results in reduced energy consumption that is a tempting reason for adulteration. Machine learning is one of the emerging approaches to non-destructive detection of food adulteration. This study investigates the application of supervised learning algorithms – AdaBoost, random forest, and ridge classifiers – to distinguish fresh and stale beans based on spectral data. The data were collected using a VIS-NIR spectrometer, and the models were evaluated under different noise conditions using metrics such as precision, accuracy, recall, F1 score, and ROC/PR curves. Ridge is the most robust model to noise in all four metrics (Accuracy > 0.95, Precision > 0.92, Recall > 0.97, and F-Score > 0.95). Adaboost and RF perform well in low noise, but their performance degrades as noise increases. Ridge is a better choice for applications with noisy environments or data quality issues.

Keywords: bean; deep learning; spectroscopy; spectrum; artificial intelligence

Beans are among the most consumed legumes in many countries around the world and hold a special place in the food basket alongside widely consumed cereals such as wheat, rice, and corn due to their high protein content and their value as a good substitute for meat. The popularity of this product is attributed to its affordability compared to animal protein sources, making beans a popular option in low-income countries. They have also gained a special place in the diets of healthy eating advocates (due to their low-calorie content) and vegetarians. In addition to being rich in protein, beans are high in fibre, carbohydrates, vitamins, and minerals, including iron, potassium, and magnesium (Messina, 2014). Due to their high fibre and protein content, beans create a feeling of satiety for longer periods and can assist with food management in critical situations (Rebello et al., 2014).

Among the countries of the world, Brazil is one of the largest producers and consumers of beans, primarily consuming black beans. In contrast, countries such as Mexico, India, China, and Iran prefer red beans and pinto beans, while Africa and America consume a significant amount of beans in general.

Despite the importance of beans, the cooking time of legumes and grains is also a key consideration. Fast cooking has become the preferred option in the modern lifestyle due

to reduced energy consumption, preservation of nutritional value, and time savings (eliminating the need for soaking). Furthermore, the ability of legumes to cook quickly is an indicator of their freshness and high quality. These factors have opened the door for profiteers in the food industry, as the price difference between fresh and stale foods creates opportunities for fraud. Fraudsters can make significant profits by mixing slow-cooking beans with quick-cooking ones.

Factors such as consumer demand for cheaper items, economic pressures in society, weak laws and oversight, low public awareness, and lack of sorting equipment make food fraud inevitable. One solution to combat this issue is the development of real-time identification technologies that are portable, user-friendly, and accessible to consumers. These devices could help identify quality products from substandard ones during purchase. The prerequisite for designing such technologies is identifying product characteristics that can serve as thresholds for classification. For instance, to distinguish between fresh and stale beans, it is necessary to identify characteristics of each category. Fresh beans typically have denser starch, higher moisture content, and shiny, smoother skin (Ovando-Martínez et al., 2011). The knowledge of these traits can be crucial in separating beans.

Today, significant advances in training algorithms with large datasets and increasing computational capacities have expanded the use of modern intelligent technologies based on machine learning. Combining machine learning with artificial intelligence can improve performance in addressing complex problems. Numerous studies have been conducted in the field of detecting food fraud using machine learning (Tigga et al., 2025; Gbashi and Njobeh, 2024; Vinothkana et al., 2024; Das et al., 2025). Besharati et al. (2024) investigated fraud detection in Iranian tea. They tested several deep-learning algorithms, including RegNetY, MobileNet V3, EfficientNet V2, ShuffleNet V2, and Swin V2T, to identify four classes of adulteration levels (0%, 15%, 45%, and 100%). Their results demonstrated the potential of image processing and deep learning in tea adulteration detection. Amponsah et al. (2022) used machine learning techniques and blockchain technology to detect healthcare fraud, achieving a high algorithmic performance with 97.96% accuracy and 98.09% recall. Zhang et al. (2024) studied the identification of superior corn varieties using hyperspectral imaging and deep learning. By combining spectral and spatial grain information, they achieved a detection accuracy above 93%. Momeny et al. (2023) investigated saffron adulteration detection using a dataset of 1869 images across six classes. They developed the LAIL-v4 convolutional neural network, which achieved 99.5% accuracy. Jahanbakhshi et al. (2021) explored turmeric adulteration detection using image processing and deep learning. Their dataset included 6240 images across 13 classes (e.g., pure turmeric powder, chickpea powder, chickpea powder mixed with food colouring, and 10–50% adulteration). Their results showed that CNN outperformed other classifiers (e.g., MLP, Fuzzy, SVM, GBT, and EDT) with a classification accuracy of 99.36%.

There have been many studies implemented in the field of detecting fraud, but there is no study available in the field of beans. The aim of the present study is to identify old beans from fresh beans, as fraudsters mix these two types of beans for greater profitability. To meet this goal, three artificial intelligence algorithms were developed: AdaBoost, Random Forest, and Ridge.

Material and Methods

Data collection

Since product classification requires supervised learning algorithms, it was necessary to collect sufficient data from each class for this study. Accordingly, 200 individual fresh beans and 200 individual old, slow-cooked beans of the Red Kidney Bean variety (*Phaseolus vulgaris*) were obtained from a trusted farmer. This variety is one of the most common beans consumed in Iran. Each bean was individually labelled, and its spectral information was extracted and recorded using a VIS-NIR spectrometer. The process involved placing the spectrometer's probe directly onto the surface of each bean. For each bean, three spectra were collected from different points on its surface, and the average spectrum was used for analysis to ensure representativeness and minimise local variations.

Hardware and software

The utilised spectrometer (Made in Noor Iman Tajhiz Co., Kashan, Iran) operated within the spectral range of 900 nm to 1700 nm with a spectral resolution of 3 nm. The system used a three-pronged Y-type fibre optic probe, with prong 1 connected to a tungsten light source that illuminated the bean sample. Prong 2 transmitted the reflected light from the sample surface to the spectrometer for analysis. Prong 3 served as the probe head, which was in direct contact with the sample. This direct contact minimised interference from ambient light and standardised the distance between the probe and the sample. The calibration of the spectrometer was also performed using a reflectance disk (a white reference standard). This standard method is essential for converting raw intensity to reflectance values and correcting for the spectral effect of the light source and any instrument drift. For each of the 400 beans, the spectrometer probe was placed directly on the surface of the bean, and its spectrum was repeated at three different, randomly selected points on the surface of the same bean. The goal of this multi-point measurement was to create a more representative spectral trace for the entire bean, averaging out minor local variations in texture, colour, or surface curvature. The three separate spectra collected from each bean were then computationally averaged. This resulting average spectrum was used as the final data point representing that particular bean in the dataset for all subsequent analyses and model training. This rigorous protocol ensures that each bean is represented by a robust spectral signature that minimises local anomalies and is suitable for supervised machine learning classification. Then, the spectral data were pre-processed using the Standard Normal Variate (SNV) method to correct for scattering effects caused by variations in particle size and path length.

For data processing and analysis, the Python platform was employed. Specifically, Python version 3.9 was used along with key libraries such as Scikit-learn for implementing the machine learning models and Pandas for data manipulation. Three classification algorithms were utilised: Ridge, Random Forest, and Ada Boost. These algorithms were selected to facilitate a fair comparison of machine learning approaches.

Quantitative and mathematical description of the NIR signal

The obtained signal was a reflectance spectrum. Mathematically, the reflectance (R) at a given wavelength (λ) was the ratio of the intensity of light reflected from the sample (I) to the intensity of light reflected from a perfect white reference standard (I_0) measured under identical conditions: $R(\lambda) = I(\lambda)/I_0(\lambda)$. The output was a set of these reflectance values.

Dimensions and structure included a spectral range of 900 nm to 1700 nm and a spectral resolution of 3 nm. The number of data points (wavelengths) in each spectrum was calculated as follows:

$$\frac{1700 \text{ nm} - 900 \text{ nm}}{3 \text{ nm}} + 1 = \frac{800 \text{ nm}}{3 \text{ nm}} + 1 = 266.67 + 1$$

This results in approximately 267 discrete wavelength bands. Therefore, each recorded spectrum was a one-dimensional vector (or array) of approximately 267 floating-point numbers. Each element in the vector represents the reflectance value at a specific wavelength (e.g., 900 nm, 903 nm, 906 nm..., 1700 nm).

The mathematical representation of a sample

A single average spectrum for a bean sample can be represented as a vector S :

$$S = [r_1, r_2, r_3, \dots, r_n] \quad (1)$$

where: n – the number of wavelength bands (approximately 267); r_i – the reflectance value at the i -th wavelength

The raw spectral data were pre-processed using the SNV method. This was a critical mathematical transformation that was applied to each spectrum individually.

SNV was used to correct for physical effects such as light scattering that can be caused by variations in grain surface texture, curvature, or slight differences in probe contact pressure. These effects often appear as a baseline shift (additive effect) or scaling of the entire spectrum (multiplicative effect). SNV normalises each spectrum to remove these artifacts and thereby emphasise spectral features related to chemical composition (such as moisture and starch content).

For a raw spectrum vector S , the SNV-transformed spectrum; S_{SNV} – calculated as:

$$S_{snv} = \frac{(S - \text{mean}(S))}{\text{std}(S)} \quad (2)$$

where: $\text{mean}(S)$ – the average of all reflectance values in that single spectrum; $\text{std}(S)$ – the standard deviation of all reflectance values in the same spectrum

After SNV, each spectrum vector had a mean of 0 and a standard deviation of 1. This standardised data was then used as input for machine learning models (AdaBoost, Random Forest, and Ridge).

Models of classification

Ada Boost Classifier

AdaBoost stands for Adaptive Boosting. It was the first successful boosting algorithm for binary classification and serves as an excellent starting point for understanding the concept of boosting. Modern boosting methods, such as stochastic gradient boosting machines, are built upon the foundation of AdaBoost (Wang, 2012).

Typically, AdaBoost is used with small decision trees. The first tree is constructed, and its performance on each training example is evaluated. This evaluation determines how much attention the next tree should give to each example. As a result, the algorithm focuses more on training examples that are difficult to predict while giving less weight to those that are easier to predict.

Random Forest Classifier

Random Forests are an ensemble learning method used for classification and regression tasks. They consist of a large number of decision trees that operate independently during training and make predictions based on either majority voting (for classification) or averaging (for regression).

Random Forests are particularly effective for decision trees that tend to overfit on the training set. By combining the predictions of multiple trees, they reduce the risk of overfitting and improve generalisation. While the performance of Random Forests is generally better than that of individual decision trees, the extent of this improvement depends on the characteristics of the dataset (Vergni and Todisco, 2023).

Ridge Classifier

The Ridge Classifier is a machine learning algorithm that merges traditional classification techniques with ridge regression, offering a unique approach to classification tasks. It utilises a loss function similar to the mean squared loss, which helps in minimising errors while training the model.

The regularisation strength is controlled by the alpha parameter. This parameter determines how much penalty is applied to the model coefficients, effectively balancing the trade-off between model complexity and its ability to generalise (Grüning and Kropf, 2006).

Adversarial stage

Adversarial training is used to increase the robustness of a deep learning model against unusual or destructive inputs. In fact, adversarial data is similar to normal data but contains small changes that have been introduced through mathematical or learning techniques and the model cannot recognise them correctly. For example, a human can easily recognise an image of a cat, but if noise is added to the model and minor changes are made, the model makes a wrong diagnosis. An image of a cat that a human can easily recognise may be misclassified by the model due to minor changes, such as added noise (Goodfellow et al., 2015).

In the adversarial stage, to simulate the potential sensor errors or noisy conditions during data acquisition, "Salt & Pepper" noise was added to the spectral data of each sample at different levels (0.05, 0.1, 0.15, 0.2) by randomly selecting 5% of the data points in the vector and setting them to either the minimum (0) or maximum (1) value of the normalised spectral range.

Training stage

The models were trained using the parameters detailed in Table 1. The dataset was split into three parts: 60% for training, 20% for validation, and 20% for testing. This allocation ensures that the model is trained on a majority of the data, tuned on a separate validation set to prevent overfitting, and finally evaluated on an unseen test set for an unbiased performance assessment. The training process was conducted for 100 epochs, meaning the entire training dataset was passed through the algorithm 100 times. A learning rate of 0.0001 was chosen, which controls how much the model's weights are adjusted with respect to the loss gradient. A small learning rate helps ensure a stable

convergence. The Adam optimiser was used, which is an efficient optimisation algorithm well-suited for problems with large datasets or parameters. To increase the diversity of the training data and improve model robustness, data augmentation techniques such as ColorJitter, RandomHorizontalFlip, and VerticalFlip were applied. Although these are typically used for images, analogous transformations were applied to the spectral data vectors to simulate variations.

Table 1 Model training configuration specifications and parameters

Parameter	Number
Epoch	100
Learning rate	0.0001
Classes	2
Data allocation	60% training, 20% validation, 20% testing
Optimiser	Adam
Train transform	ColorJitter, RandomHorizontalFlip, VerticalFlip

Models evaluation criteria

Confusion matrix

A confusion matrix provides a clear picture of the model's strengths and weaknesses by summarising results into these four categories. True Positives (TP) are cases where the model correctly predicts the positive class. True Negatives (TN) are cases where the model correctly predicts the negative class. False Positives (FP) occur when the model incorrectly predicts the positive class (a "false alarm"). False Negatives (FN) occur when the model incorrectly predicts the negative class (a "miss"). Correct classifications are represented on the main diagonal of the matrix (TN and TP), whereas incorrect classifications are found outside the diagonal (FP and FN). For example, a high True Positive and True Negative count indicates the model is performing well.

The confusion matrix is crucial for calculating several key performance metrics, such as accuracy, precision, recall, and F1-score, which are used to further analyse the classification performance.

Accuracy

Accuracy measures the overall correctness of a model. It represents the proportion of correctly classified positive and negative samples out of the total number of samples (Eq. (3)) (James et al., 2013).

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (3)$$

Precision

Precision measures the percentage of positive predictions made by the model that are actually correct. It reflects the model's ability to avoid false positives (Eq. (4)) (James et al., 2013).

$$precision = \frac{TP}{TP + FP} \quad (4)$$

Recall

Recall, on the other hand, measures the percentage of actual positive samples correctly classified by the model. It reflects the model's ability to avoid false negatives (Eq. (5)) (James et al., 2013).

$$recall = \frac{TP}{TP + FN} \quad (5)$$

Balancing precision and recall is critical, particularly in highly imbalanced datasets (where the number of samples in each class is very different). For example, if the model has high precision but low recall, it tends to classify samples as negative more frequently.

F1-Score

The F1-Score is a metric that combines precision and recall into a single value. It is the harmonic mean of these two metrics, providing a balanced evaluation of the model's performance (Eq. (6)) (James et al., 2013).

$$F1\text{-Score} = \frac{2 \times precision \times recall}{precision + recall} \quad (6)$$

The F1-Score is especially useful when the dataset is imbalanced, as it provides a more comprehensive view of the model's performance than precision or recall alone (Pourdarbani et al., 2021).

Precision-recall (PR) curve

The PR curve is a graphical representation that evaluates the model's performance by examining how changes in the decision threshold affect the balance between precision and recall. An ideal PR curve adheres closely to the top-right corner of the graph, where both precision and recall approach 1. The Average Precision (AP) score, ranging from 0 to 1, quantifies the PR curve, with higher values indicating better performance (Ali et al., 2022).

Receiver operating characteristic (ROC) curve

The ROC curve shows the trade-off between the True Positive Rate (TPR) and the False Positive Rate (FPR) at different classification thresholds (Ali et al., 2022). The Area Under the Curve (AUC) is a key metric derived from the ROC curve. It represents the probability that the model will rank a randomly chosen positive instance higher than a randomly chosen negative one. An AUC of 1.0 indicates a perfect classifier, while an AUC of 0.5 suggests a classifier with no discriminative ability (equivalent to random guessing).

Results and Discussion

AdaBoost Classifier

Figure 1 examines the performance of the AdaBoost classifier under different Salt & Pepper noise conditions. Each matrix

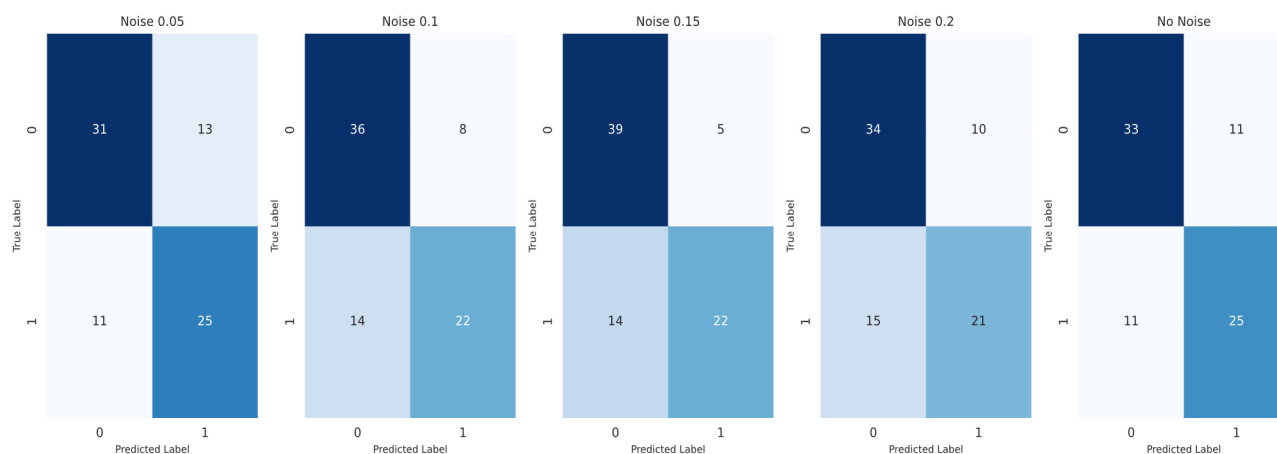


Fig. 1 The performance of the AdaBoost classifier under different noise conditions

shows the predicted class on the x-axis versus the true class on the y-axis. The numbers in the cells represent the count of samples, and the colour intensity (darker blue) corresponds to a higher count. Each matrix corresponds to a specific noise level of 0, 0.05, 0.1, 0.15, and 0.2 noise condition. At 0.05 noise, the model error is low, but some samples are misclassified. At 0.1 noise, the increase in error due to noise is more evident. As the noise increases, the errors increase. The lowest error is associated with the No Noise condition, which indicates the model's sensitivity to noise in the data. To improve the model's performance in noisy conditions, adversarial training or data preprocessing can be used.

According to ROC curve (Fig. 2), the model performs relatively well in the noise-free condition (AUC = 0.79) but is slightly weaker than the noise of 0.1. This phenomenon can occur when a small amount of noise acts as a form of regularisation, preventing the model from overfitting to the training data and thus improving its generalisation on the test set. At noise of 0.05 and 0.15, the model performance in these conditions is average, indicating that the model still performs acceptably in the face of this noise. The worst performance is related to the noise of 0.2. High noise levels corrupt the underlying patterns in the spectral data, making

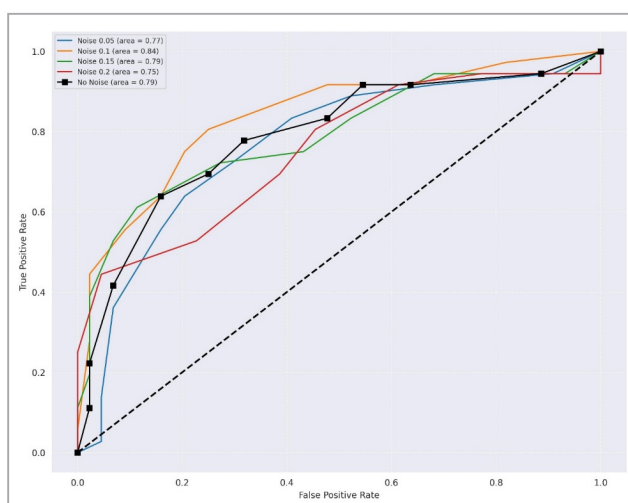


Fig. 2 The ROC curve of AdaBoost Classifier at different noise levels

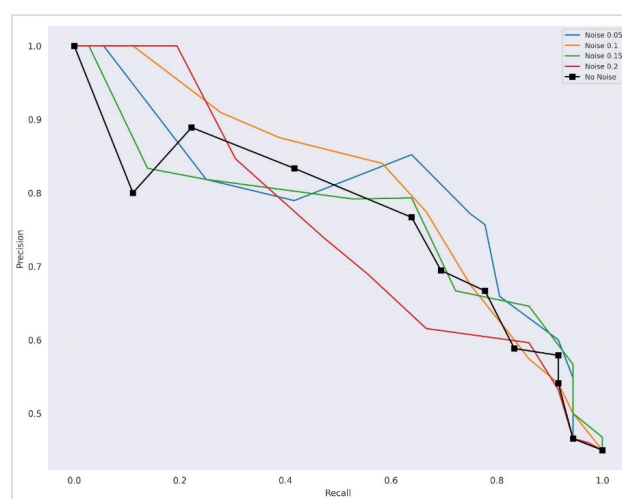


Fig. 3 The PR curve of AdaBoost Classifier at different noise levels

it difficult for the model to distinguish between the classes, hence the significant drop in AUC. The high noise has reduced the resolution of the model between classes.

Figure 3 gives the PR curves of AdaBoost classifier at different noise levels. The best performance of the model is for noise 0.1, which is generally higher than the other curves, indicating a better balance between Precision and Recall at this noise level. The performance of the model without noise is in a similar range to Noise 0.1, but in some ranges Precision decreases. This behaviour may be due to the model settings or the data. At noise levels of 0.05 and 0.15, the model performance is average, but the worst performance occurs at noise level 0.2 where precision decreases more compared to recall at some points, indicating significant increases at number of False Positives and False Negatives.

Random Forest Classifier

The confusion matrix of Random Forest classifier is shown in Fig. 4. As with the previous figure, the numbers represent sample counts, and the colour intensity reflects the magnitude of these counts. False Positives are zero at all noise levels, indicating the model's ability to prevent misclassification of class 0 (stale beans) samples as class

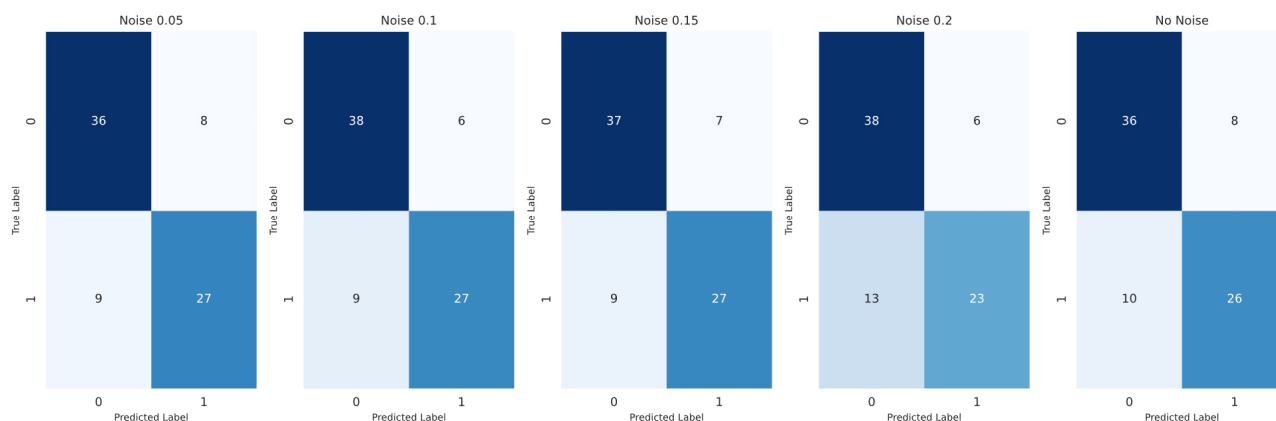


Fig. 4 The confusion matrix of Random Forest classifier at different noise levels

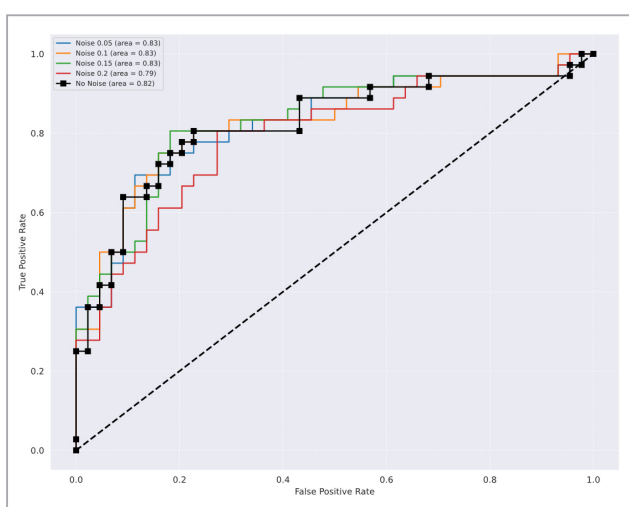


Fig. 5 The ROC curve of Random Forest classifier at different noise levels

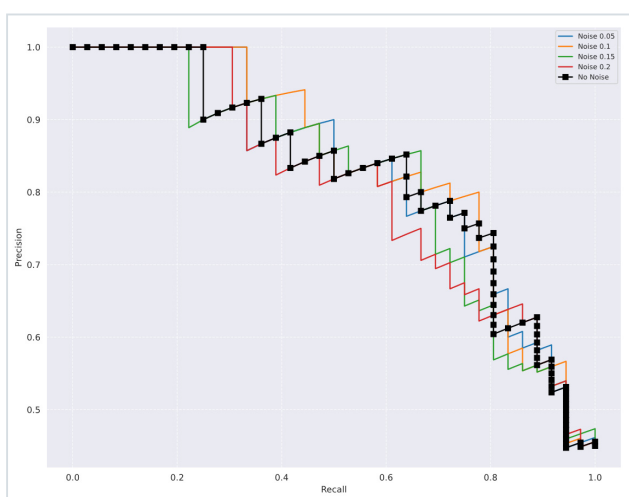


Fig. 6 The PR curve of Random Forest classifier at different noise levels

1 (fresh beans). As noise increases, the number of class 1 samples that are misclassified as class 0 increases. The model performs very well without noise and also at low noise levels (0.05 and 0.1). However, the noise of 0.2 has a strong negative impact on class 1 identification.

From the ROC curve perspective, the model performs well without noise, but slightly lower than with low noise, which may be due to the specific settings of the model. In low noise conditions (0.05, 0.1, and 0.15), there is no significant impact on the model performance, and the model can distinguish between positive and negative samples well. However, high noise (0.2) has reduced the model performance, as the model makes mistakes in accurately separating classes (Fig. 5).

According to PR curve (Fig. 6), the model performance is very close in low and no noise, with a significant difference only seen in the noise of 0.2. In no noise and low noise conditions, the model strikes a better balance between Precision and Recall, and the model performs best in low noise conditions (0.05, 0.1, and 0.15). Increasing Recall in high noise has reduced Precision, indicating an increase in false positive errors.

Ridge Classifier

The confusion matrix of Ridge classifier at different level of noise is given in Fig. 7. The axes, numbers, and colours follow the same interpretation as in previous confusion matrices. The model performance is ideal in the noise-free and noise-free state of 0.1, but as the noise increases to 0.15, it starts to make errors. The effect of noise on the model performance is more visible at the noise level of 0.2. To increase the noise resistance, it is necessary to improve the model settings or use adversarial training.

From the ROC curve perspective (Fig. 8), the model was able to correctly classify all samples at all noise levels. The model still performs flawlessly even at high noise (0.2). From the perspective of the PR curve (Fig. 9), the model is very robust to low noise (0.05 and 0.1) and even with increasing noise, the model still maintains high Recall, but Precision decreases slightly but is still acceptable.

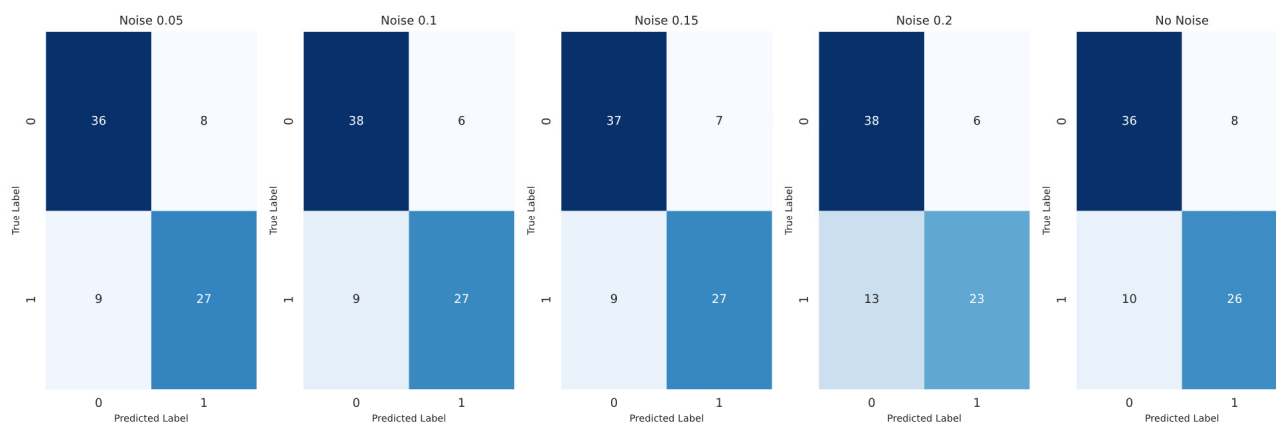


Fig. 7 The confusion matrix of Ridge classifier at different level of noise

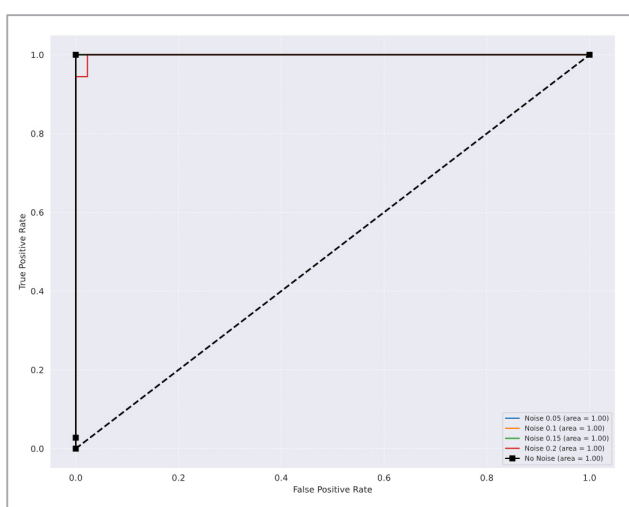


Fig. 8 ROC curve of Ridge model at different noise levels

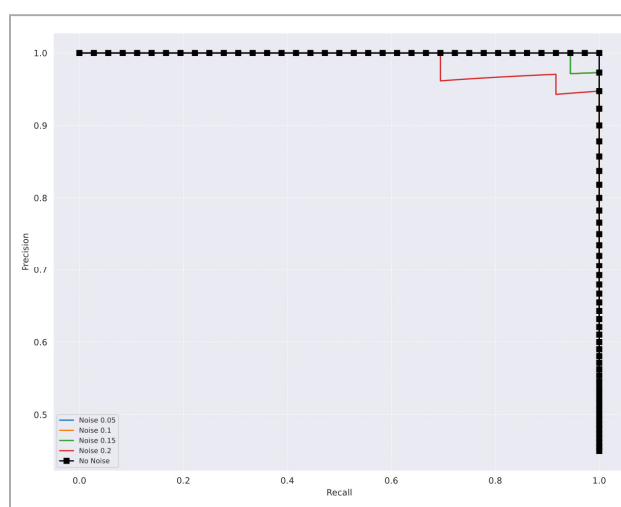


Fig. 9 PR curve of Ridge model at different noise levels

Comparison of AdaBoost, Random Forest, and Ridge classifiers

Figure 10a–b compares the proposed classifiers in terms of assessment metrics. Ridge appears to be the most robust model to noise. The performance of Adaboost

and Random Forest is similarly affected by noise, but depending on the noise level, one may be slightly better. The choice of the model depends on the specific application and the need to balance these metrics.

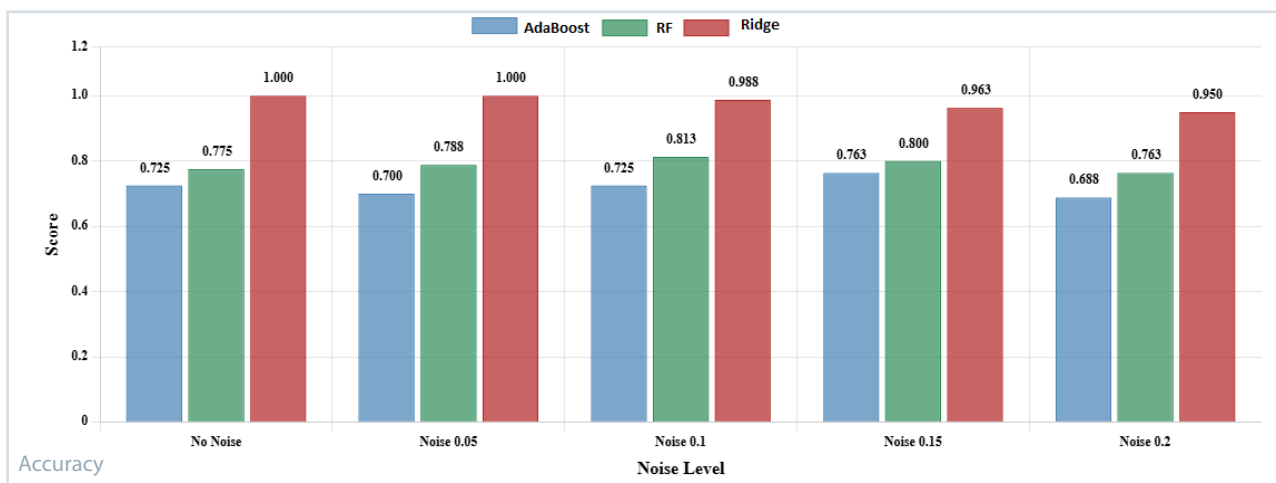


Fig. 10a Comparison of proposed classifiers in terms of evaluation criteria

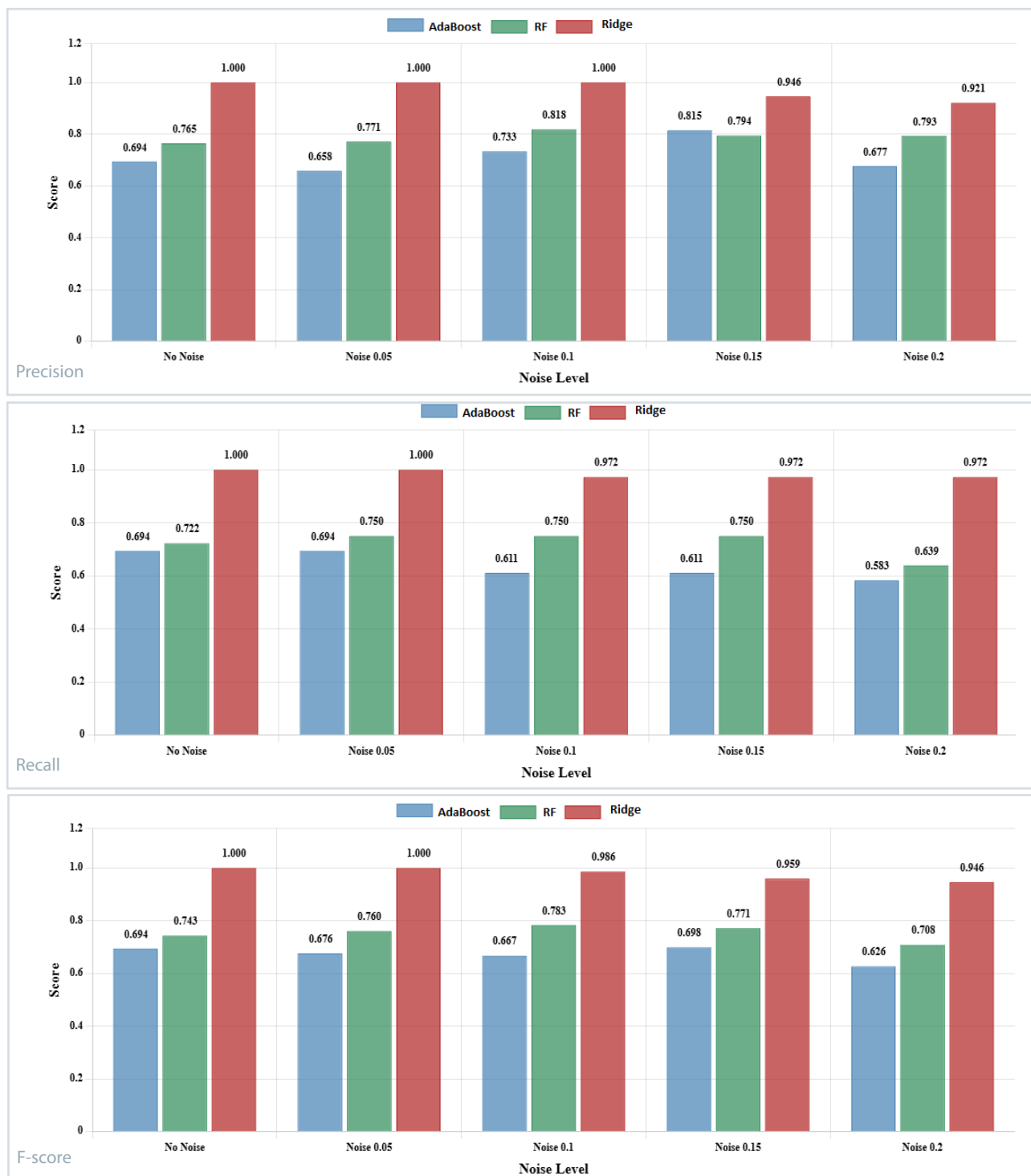


Fig. 10b Comparison of proposed classifiers in terms of evaluation criteria

Conclusion

This study reported the successful application of machine learning algorithms using VIS-NIR spectroscopy in non-destructive detection of bean adulteration by mixing stale beans with fresh beans. The implications of the results in the food industry can be thought-provoking and promising. Among the implemented classifiers, the Ridge classifier showed outstanding performance, maintaining high accuracy (>0.95) and stability even

under high noise conditions. This can be attributed to the L2 regularisation by penalising large coefficients and preventing overfitting. Such stability is recommended for real-world applications. In contrast, AdaBoost and Random Forest were effective in low noise conditions, but the performance decreased significantly as the noise level increased, indicating the sensitivity of these models to disturbances in spectral data. The results were consistent with the findings of other studies in the same field. For

example, Zhang et al. (2024) achieved high accuracy in corn seed identification using deep learning, and Momeny et al. (2023) successfully detected saffron adulteration. The robustness of the Ridge model in our study is a key finding, as many real-world settings for food inspection are not pristine and may introduce noise. Similar robustness has been observed in other agricultural applications where spectrometry is used, for example, in determining wheat quality parameters.

Future work should focus on several key areas for advancing this technology. First, expanding the dataset to include bean varieties from diverse geographical origins and storage conditions and making the model generalisable. Second, implementing and developing these models on low-cost, portable IoT-based spectrometers could provide consumers with a user-friendly and real-time tool for on-site quality verification. Ultimately, a multimodal approach that combines spectral data with other data sources, such as image analysis or textural information, could create an even more powerful and accurate system for ensuring food safety.

Acknowledgement

This research was funded by project 1403/D/9/19325 financed by the University of Mohaghegh Ardabili. We wish to express our gratitude to the University of Mohaghegh Ardabili for providing the necessary facilities and a conducive academic environment for this research.

References

- Ali, R., Balamurali, M., Varamini, P., 2022. Deep learning-based artificial intelligence to investigate targeted nanoparticles' uptake in TNBC cells. *International Journal of Molecular Sciences* 23(24), 16070. <https://doi.org/10.3390/ijms232416070>
- Amponsah, A.A., Adekoya, A.F., Weyori, B.A., 2022. A novel fraud detection and prevention method for healthcare claim processing using machine learning and blockchain technology. *Decision Analytics Journal* 4, 100122. <https://doi.org/10.1016/j.dajour.2022.100122>
- Besharati, M. S., Pourdarbani, R., Sabzi, S., Sotoudeh, D., Ahmadireshnizi, M., Garcia-Mateos, G., 2024. Counterfeit detection of Iranian black tea using image processing and deep learning based on patched and unpatched images. *Horticulturae*, 10(7), 665. <https://doi.org/10.3390/horticulturae10070665>
- Das, P., Altemimi, A.B., Nath, P.C., Katyal, M., Kesavan, R.K., Rustagi, S., Panda, J., Kumar Avula, S., Kumar Nayak, P., Mohanta, Y.K., 2025. Recent advances on artificial intelligence-based approaches for food adulteration and fraud detection in the food industry: Challenges and opportunities. *Food Chemistry*, 468, 142439. <https://doi.org/10.1016/j.foodchem.2024.142439>
- Gbashi, S., Njobeh, P.B., 2024. Enhancing food integrity through artificial intelligence and machine learning: A comprehensive review. *Applied Sciences* 14(8), 3421. <https://doi.org/10.3390/app14083421>
- Goodfellow, I.J., Shlens, J., Szegedy, C., 2015. Explaining and harnessing adversarial examples. *International Conference on Learning Representations (ICLR)*. arXiv:1412.6572v3 [stat.ML]. <https://doi.org/10.48550/arXiv.1412.6572>
- Grüning, M., Kropf, S., 2006. A ridge classification method for high-dimensional observations. In Spiliopoulou, M., Kruse, R., Borgelt, C., Nürnberger, A., Gaul, W. (eds), *From Data and Information Analysis to Knowledge Engineering. Studies in Classification, Data Analysis, and Knowledge Organization*. Springer, Berlin, Heidelberg, Germany, pp. 684–691. https://doi.org/10.1007/3-540-31314-1_84
- Jahanbakhshi, A., Abbaspour-Gilandeh, Y., Heidarbeigi, K., Momeny, M., 2021. A novel method based on machine vision system and deep learning to detect fraud in turmeric powder. *Computers in Biology and Medicine* 136, 104728. <https://doi.org/10.1016/j.compbiomed.2021.104728>
- James, G., Witten, D., Hastie, T., Tibshirani, R., 2013. *An Introduction to Statistical Learning. With Applications in R*. 1st ed. Springer, New York, USA. eISBN 978-1-4614-7138-7. <https://doi.org/10.1007/978-1-4614-7138-7>
- Messina, V., 2014. Nutritional and health benefits of dried beans. *The American Journal of Clinical Nutrition* 100(1), 437S–442S. <https://doi.org/10.3945/ajcn.113.071472>
- Momeny, M., Neshat, A.A., Jahanbakhshi, A., Mahmoudi, M., Ampatzidis, Y., Radeva, P., 2023. Grading and fraud detection of saffron via learning-to-augment incorporated Inception-v4 CNN. *Food Control* 147, 109554. <https://doi.org/10.1016/j.foodcont.2022.109554>
- Ovando-Martínez, M., Bello-Pérez, L.A., Whitney, K., Osorio-Díaz, P., Simsek, S., 2011. Starch characteristics of bean (*Phaseolus vulgaris* L.) grown in different localities. *Carbohydrate Polymers* 85(1), 54–64. <https://doi.org/10.1016/j.carbpol.2011.01.043>
- Pourdarbani, R., Sabzi, S., Arribas, J.I., 2021. Nondestructive estimation of three apple fruit properties at various ripening levels with optimal Vis-NIR spectral wavelength regression data. *Heliyon* 7(9), e07942. <https://doi.org/10.1016/j.heliyon.2021.e07942>
- Rebello, C.J., Greenway, F.L., Finley, J.W., 2014. A review of the nutritional value of legumes and their effects on obesity and its related co-morbidities. *Obesity Reviews* 15(5), 392–407. <https://doi.org/10.1111/obr.12144>
- Tigga, S.L., Burman, R.K., Kumar, A., Kumar, N., Kumar, B., Kamaldeep, K., 2025. A comparative study of fraud detection algorithm in food delivery systems. *AIP Conference Proceedings* 3327, 020005. <https://doi.org/10.1063/5.0289764>
- Vergni, L., Todisco, F., 2023. A random forest machine learning approach for the identification and quantification of erosive events. *Water* 15(12), 2225. <https://doi.org/10.3390/w15122225>
- Vinothkanna, A., Dar, I.O., Liu, Z., Jia, A.-Q., 2024. Advanced detection tools in food fraud: A systematic review for holistic and rational detection method based on research and patents. *Food Chemistry* 446, 138893. <https://doi.org/10.1016/j.foodchem.2024.138893>
- Wang, R., 2012. AdaBoost for feature selection, classification and its relation with SVM, a review. *Physics Procedia* 25, 800–807. <https://doi.org/10.1016/j.phpro.2012.03.160>
- Zhang, L., Wei, Y., Liu, J., An, D., Wu, J., 2024. Maize seed fraud detection based on hyperspectral imaging and one-class learning. *Engineering Applications of Artificial Intelligence* 133(B), 108130. <https://doi.org/10.1016/j.engappai.2024.108130>

