Applications of Machine Learning: Classifying Wheat Seeds through Computer Neural Networks

By Tianle Liang

AUTHOR BIO

Tianle Liang, a junior at Zionsville Community High School in Indiana, is strongly passionate about pursuing a career in STEM and technology. He is involved in the Technology Student Association, where he won the 2024 Indiana State Conference section of debating technology issues. Tianle also loves teaching and connecting with others in his community, founding his own online mathematics courses with the nonprofit Youth Development Resource Center. In late 2023, he developed an interest in machine learning, and worked with Georgia Tech professor Guillermo Goldsztein to program his first neural network model. This summer, Tianle will research machine learning in depth at a summer program at the University of Chicago.

ABSTRACT

In the past few decades, the incompetent processes of traditional mechanical seed sorting have given rise to new, innovative methods, along with the integration of smart technology in agriculture. The present study created a multi-classification machine learning model that classifies wheat seeds based on given characteristics by giving highly accurate predictions for future samples based on given data. Due to various uses of different types of seeds, effective sorting is an integral preparatory stage for food processing. The dataset used in the study included seven features (directly measured characteristics of seeds), three labels (types of seeds), and around two hundred examples. The seven features include area, perimeter, compactness, kernel length, kernel width, asymmetry coefficient, and kernel groove. The first model was created through neural networks trained on all seven characteristics, reaching an overall accuracy of 0.94. Implementing a decision tree method, feature selection was applied to the data, which distinguished kernel groove, kernel width, and asymmetry coefficient as the three most important features. A new model was constructed with only the three selected features, which reduced the risk of overfitting, eliminated noise, and yielded a higher accuracy (0.96) than the previous model. Both multi-classification models used the softmax activation function and results were compared; feature selection noticeably reduced any inaccuracies, including categorical cross entropy loss. The main incentive of the study was to develop an effective model to sort wheat seeds (to replace traditional mechanical sorting) and provide reasonable recommendations for practical use through an analysis of feature selection.

Keywords: machine learning, dataset, feature, label, one-hot encoding, training and validation, neural networks, overfitting, feature selection, decision tree, scaling, softmax, categorical cross entropy

INTRODUCTION

Historical Context

Since its initial domestication over ten thousand years ago in the Fertile Crescent of Mesopotamia, wheat has rapidly grown to be one of the world's most prominent staple crops. Wheat grows exceptionally well in most regions, especially temperate and subtropical climates. The versatility and practicality of wheat, along with maize and rice, fueled most of the world's major population growth. The cultivation of wheat is also a pivotal part of modern agriculture. Efficient and accurate methods to classify wheat seeds become crucial to optimize agricultural processes, ensuring quality control, and facilitating breeding programs. For centuries, seeds were sorted traditionally by hand, an excruciatingly slow and often ineffective method. The 19th and 20th centuries introduced the first mechanical machines applied to seed sorting. The early machines still required hand assistance and relied on the approximate measurements of only the most basic characteristics, such as weight and length ("Wheat", n.d.).

Since then, much more sophisticated technologies have been developed, and the revolutionary appearance of machine learning in recent decades opens a new wave of possibilities. The earliest "intelligent" machines included popular game engines, such as the first chess engine that trumped World Chess Champion Gary Kasparov in the 1990s. Although the initial concepts of machine learning and artificial intelligence were first conceived in the mid-1900s, the implementation of machine learning in technology was popularized in the 1980s, when the first AI models infatuated scientists (Karjian, 2023). Predictive machine learning is a major branch of artificial intelligence and data analysis that uses computer algorithms to project new data using old training data. Machine learning is advantageous over traditional learning methods because algorithmic learning eliminates the need for explicit programming. Machine learning models become more accurate with more data, and the automation and efficiency of data analysis through machine learning has skyrocketed its popularity. With modern information technology, classification models, which sort data into predefined categories, can detect significantly more features facilitating the sorting process while giving insight on the science of wheat itself. The current model aims to classify wheat seeds and replace human labor with reliable computer processes to improve cultivation.

Dataset Description

The data used in the present study is a real-world dataset obtained from the UCI Center for Machine Learning and contains seven seed features: area, perimeter, compactness, kernel length, kernel width, asymmetry coefficient, and kernel groove. Using soft X-ray imaging, scientists at the Institute of Agrophysics of the Polish Academy of Sciences in Lublin created high quality visualizations of internal kernel structures. The seven features were then extracted from the images, which were recorded on 13x18 cm X-ray KODAK plates (Charytanowicz et al, 2012). See Table 1 for a short description of the seven features.

Feature (Measured in Millimeters)	Description

Area (A)	Total surface area, measured in square millimeters
Perimeter (P)	Length of the outer boundary
Compactness (C)	How closely a seed resembles a spherical and dense shape $C = \frac{4\pi \cdot A}{P^2}$
Kernel Length	Longitudinal length
Kernel Width	Transverse length
Asymmetry Coefficient	Deviation from perfect symmetry
Kernel Groove	Length of groove indentation

Table 1: Description of Features

In the context of wheat seeds, the "kernel" refers to the embryo, the actual edible part used to produce food, which is found within multiple seed walls. The "area" feature refers to the total surface area of the seed, measured in square millimeters, and the "perimeter" feature measures the length of the outer boundary of the seed. "Compactness," a calculated value involving a certain ratio of surface area and perimeter, measures how closely the seed resembles a spherical and dense shape. A dense, regularly shaped seed would obtain a higher compactness score as opposed to an elongated or irregularly shaped seed. The "kernel length" and "kernel width" features measure the longitudinal and transverse lengths of the seed, respectively. "Asymmetry coefficient" measures the amount of deviation from a perfectly symmetrical shape (a very asymmetrical seed would therefore obtain a high asymmetry coefficient). Finally, the "kernel groove" feature refers to the length of the groove indentation present on the seed, often a defining characteristic of the different seed varieties. Figure 1 shows examples of kernels analyzed

in this study.

Figure 1: Example of Wheat Kernels

The picture above shows examples of the individual wheat kernels analyzed the dataset. The kernels are from the genus Tricitum, known as common wheat ("Traditional grains, a blend of the best known grains", n.d.).

After preprocessing (discussed below), the model classified each example based on the features into three varieties of wheat: Kama, Rosa, and Canadian (represented by Type 1, Type 2, and Type 3 in the dataset for convenience). Each type is established based upon distinct ranges in characteristics; for example, Type 3 may be distinguished from Type 1 because of a higher average asymmetry coefficient and other unique differences. Machine learning implements complex pattern recognition to distinguish the types, but in this model, specific scientific names and properties would be determined after sorting. See Table 1 for the dataset before preprocessing.

	Area	Perimeter	Compactness	Kernel.Length	Kernel.Width	Asymmetry.Coeff	Kernel.Groove	Type
0	15.26	14.84	0.8710	5.763	3.312	2.221	5.220	1
1	14.88	14.57	0.8811	5.554	3.333	1.018	4.956	1
2	14.29	14.09	0.9050	5.291	3.337	2.699	4.825	1
3	13.84	13.94	0.8955	5.324	3.379	2.259	4.805	1
4	16.14	14.99	0.9034	5.658	3.562	1.355	5.175	1
...	...	...	...	...	...	...	...	...
194	12.19	13.20	0.8783	5.137	2.981	3.631	4.870	3
195	11.23	12.88	0.8511	5.140	2.795	4.325	5.003	3
196	13.20	13.66	0.8883	5.236	3.232	8.315	5.056	3
197	11.84	13.21	0.8521	5.175	2.836	3.598	5.044	3
198	12.30	13.34	0.8684	5.243	2.974	5.637	5.063	3

199 rows × 8 columns

Table 2: Dataset before Preprocessing

METHOD

Preprocessing

In machine learning, the computer is given a large set of examples, each with a set of features and labels; then, a model is trained on the examples to predict labels based on the features. With each iteration of the training data (epochs), the model will become increasingly accurate. In this particular study, the features are the recorded characteristics of the seed, and the label is the type of seed. Sometimes it is

necessary to check the distributions of data for gaps or outliers and drop unsuitable data, but there were no anomalies in this dataset. See Figure 1 for the preliminary histogram distributions of each of the seven features.

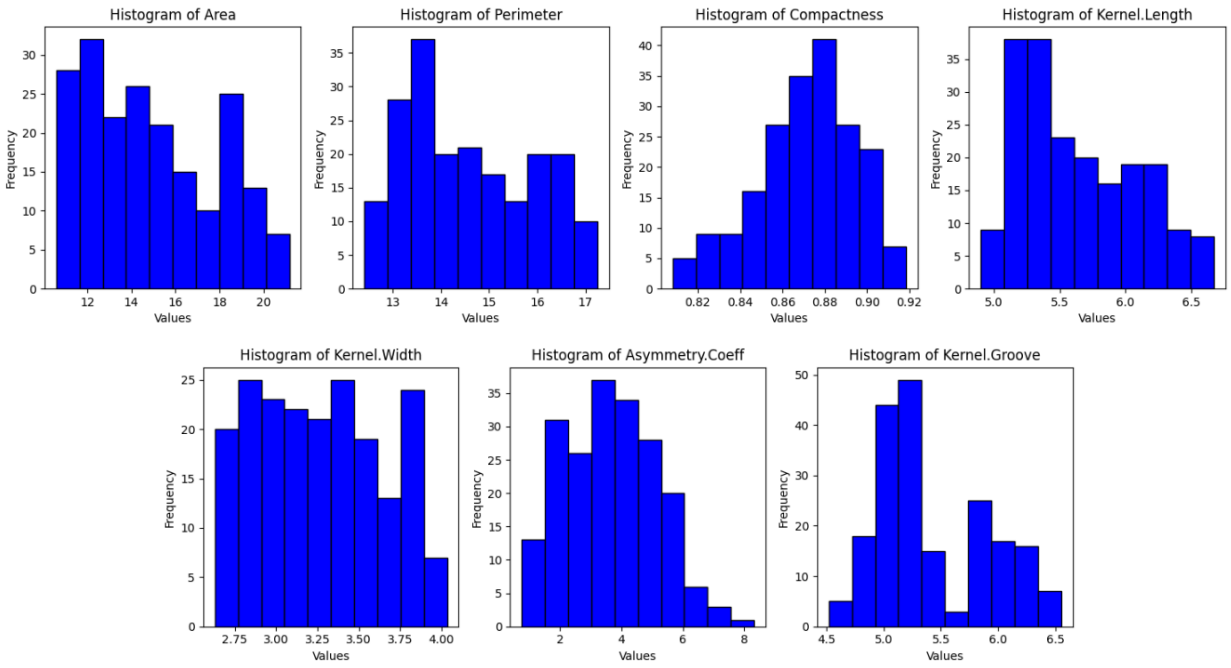


Figure 1: Histogram Distributions of Features

All measurements are in millimeters; Compactness and Asymmetry Coefficient are calculated values.

Before fitting the model, the data needed to be preprocessed. The raw dataset did not have any elements of "NaN" (not a number) and did not have any apparent abnormalities, so there was no need to drop any parts of the data. With a multi-classification study such as wheat classification, it is necessary to one-hot encode the label. All categorical data must be one-hot encoded before being fit into a numerical model. One-hot encoding is a preprocessing technique that converts categorical labels (which usually include letters) into a binary matrix for algorithms that prefer numerical input (DelSole, 2018). Binary code is much simpler for the computer to read, thus improving efficiency and runtime. The label of each example is split into multiple categories, one for each label. In such categories, the true label of the example receives a 1 while the others receive a 0. See Table 2 for the dataset after one-hot encoding.

	Area	Perimeter	Compactness	Kernel.Length	Kernel.Width	Asymmetry.Coeff	Kernel.Groove	Type 1	Type 2	Type 3
0	15.26	14.84	0.8710	5.763	3.312	2.221	5.220	1	0	0
1	14.88	14.57	0.8811	5.554	3.333	1.018	4.956	1	0	0
2	14.29	14.09	0.9050	5.291	3.337	2.699	4.825	1	0	0
3	13.84	13.94	0.8955	5.324	3.379	2.259	4.805	1	0	0
4	16.14	14.99	0.9034	5.658	3.562	1.355	5.175	1	0	0
...	...	...	...	...	...	...	...	...	...	...
194	12.19	13.20	0.8783	5.137	2.981	3.631	4.870	0	0	1
195	11.23	12.88	0.8511	5.140	2.795	4.325	5.003	0	0	1
196	13.20	13.66	0.8883	5.236	3.232	8.315	5.056	0	0	1
197	11.84	13.21	0.8521	5.175	2.836	3.598	5.044	0	0	1
198	12.30	13.34	0.8684	5.243	2.974	5.637	5.063	0	0	1

199 rows × 10 columns

Table 3: Dataset after One-Hot Encoding

Above, it can be seen that the first example's true label is Type 1, so only the Type 1 category receives a 1 while the others receive a 0. This process, one-hot encoding, can be similarly observed in the last example, with the true label of Type 3.

The original dataset is a CSV (Comma-Separated Values) file, which was subsequently converted into an array for easier manipulation of elements. Next, the array was split into array X and array Y, representing the feature array (input) and label array (output). To effectively fit the model, the arrays need to be split again into X-train, Y-train, X-validation, and Y-validation. The training sets will be used to fit and train the model while the validation sets will be used to assess the model's performance and ability to generalize to new data (the validation accuracy is usually considered the overall accuracy of the model). Additionally, the accuracy assessed by validation sets will show any overfitting, a case where the model learns the data too specifically and fails to adequately generalize to new data (Bhande, 2018). See Figure 2 for an example of overfitting.

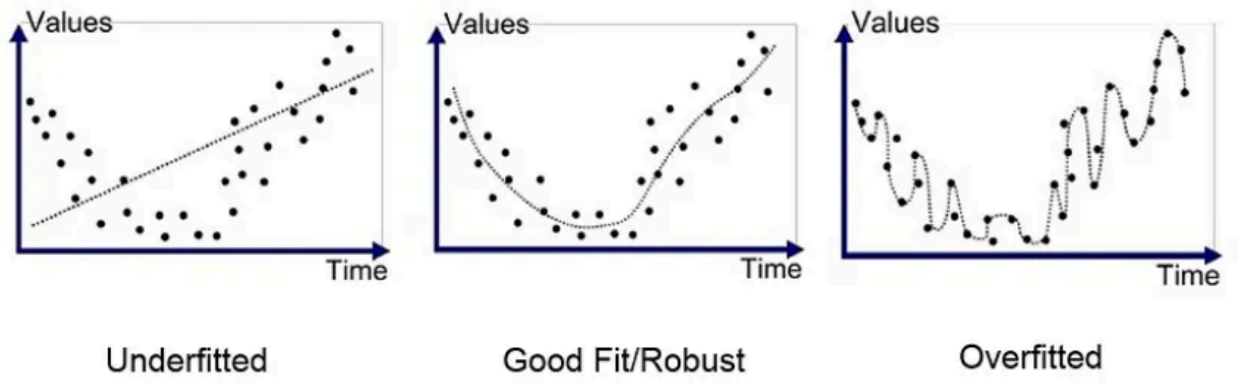


Figure 2: Example of Overfitting (Bhande, 2018)

Models generate an approximate curve to predict new samples. When the curve is too specific to the existing training points, the model cannot generalize well to new data.

Finally, the data was scaled in preparation for sensitive model algorithms. Scaling the data is important to ensure all features have similar ranges, leading to more effective performance of models. Additionally, scaling establishes equal treatment of features during the computer training process and may facilitate efficiency (leading to better runtime). Typically, in multi-classification models, only the features need to be scaled; in this case, the X-train and X-validation sets were scaled. The scaler package utilized in this study implements the standardization scaling (Z-score normalization). See Equation 1 for details on how Z-score is calculated.

$$z = \frac{x-\mu}{\sigma}, \mu = \frac{1}{n}\sum_{i=1}^{n}(x_i), \sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2}$$

Equation 1: Z-score Calculation

z = *Z-score,* μ = *mean,* σ = *standard deviation,* x = *specific element in set,* n = *total number of elements in set*

Initial Model

The initial model factored in the values of all seven features to predict the wheat type. The model utilized the Sequential neural network model from the Keras library (Chollet, 2023). Neural networks are a class of machine learning models designed to process information similarly to the human brain; because neural networks specialize in complex pattern recognition and representation, they are ideal for multi-classification problems. In a neural network model, the interconnected nodes receive input, process information through an activation function, and generate output - imitating the connections of the billions of neurons in the human brain. Activation functions are used to process information in a model; numerical output has no activation function, binary classification uses the sigmoid function, and multi-classification like the present study uses the softmax function.

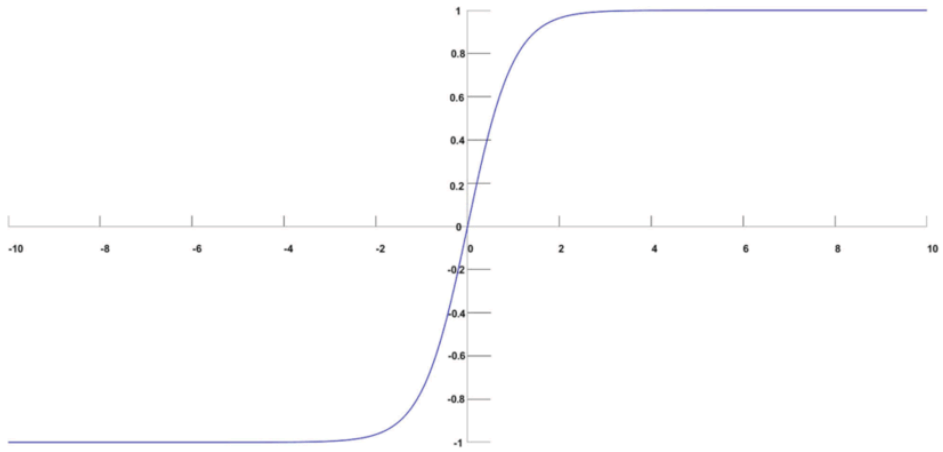


Figure 3: Example Graph of Softmax Function

The above graph shows a probability distribution based on input values.

The softmax function (see Figure 3 above) is a pivotal activation function because of the nature of multi-classification models ("Softmax Function," n.d.). Machine learning models are trained to be exceptionally accurate, but predicted outputs are never absolute; thus, the softmax function returns the probabilities of each label (Bhardwaj, 2020b). For example, given a new random sample of a seed, the present model may predict the probabilities of Type 1, Type 2, and Type 3, to be 0.1, 0.8, and 0.1, respectively. The label with the highest probability is typically the true value, meaning that Type 2 is most likely the label of the sample. Essentially, the softmax function takes an input vector of numerical scores (the features) and returns a probability distribution (predicting the label). A notable characteristic of softmax is scale-invariance, meaning the probability distributions are unaffected when the input vector is multiplied by a constant. This provides numerical stability during the training stage and improves interpretability. See Equation 2 for the defining equation of softmax.

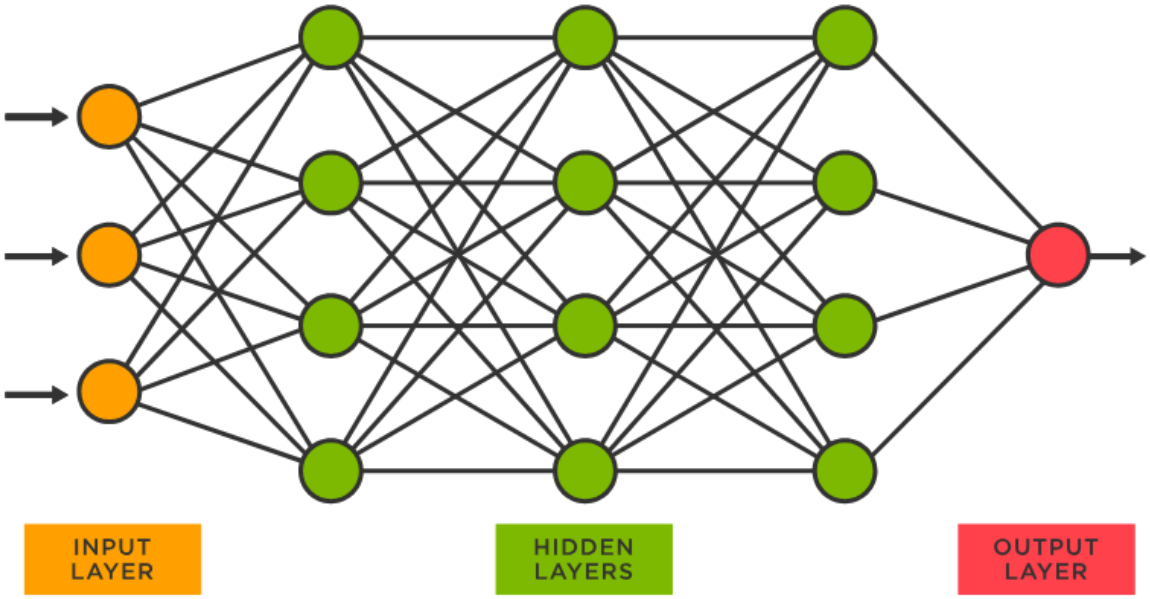
$$\sigma(\vec{z}_i) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

Equation 2: Softmax Function

σ = *softmax,* $\vec{z}$ = *input vector,* K = *number of classes in multi-classifier,* e^{z_i} = *standard exponential function for input vector,* e^{z_j} = *standard exponential function for output vector*

Nodes in a neural network are separated into three types of layers: an input layer (input from features), hidden layers (information processing), and an output layer (final prediction) ("What is," n.d.). See Figure 4 for an example neural network. The present model used seven nodes in the input layer (one for each feature) and three nodes in the output layer (one for each type of seed). Due to the simplicity of the problem, hidden layers were unnecessary and did not increase the overall accuracy of the model.

Figure 4: Example of Neural Network



The above example has 3 input nodes, three hidden layers with 4 nodes each, and one output node.

Feature Selection and Revised Model

After the initial model was created, the dataset was processed through feature selection. Feature selection is a method to select the most important features to determine the labels and facilitates better understanding of the topic, as well as relations between features and labels (Sudharsan, 2018). Clarifying the feature array to only the most important features reduces noise unintentionally caused by other, less relevant features, effectively preventing overfitting. Of course, less features also results in less runtime and computational resources and improves the interpretability of the model. The present study implemented a decision tree method to determine the importance of each feature. Decision trees, like neural networks, use interconnected nodes, but the structures differ fundamentally. See Figure 5 for an example decision tree.

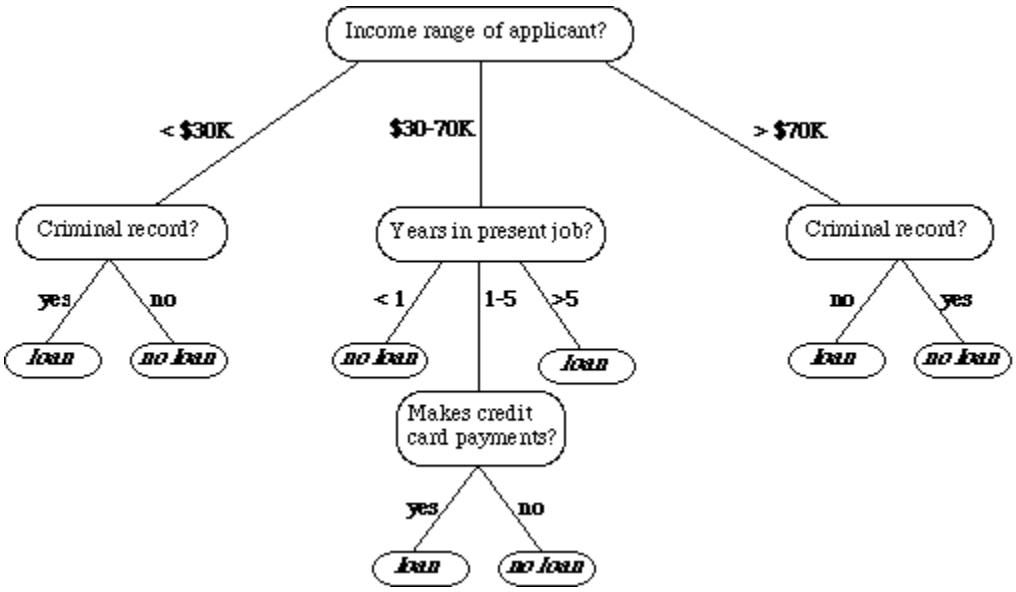


Figure 5: Example Decision Tree

The above decision tree is used by banks to determine whether or not to offer someone a loan using a sequential list of questions including income, criminal record, and credibility.

Decision trees start with a single node, called the root (in this case, all seven features), and then split into child nodes based on Gini node impurity. Gini impurity is essentially the probability of the misclassification of a set of data after a node split. Through splitting based on maximum reduction of Gini impurity, decision trees group the most relevant features together (Plapinger, 2017). Importance scores range from 0 to 1, and feature selection yielded the most important features to be kernel groove, kernel width, and asymmetry coefficient, with approximate scores of 0.5, 0.4, and 0.1, respectively. The second, revised model was created with the feature selection cut to only include these three features. The revised model was created using the same parameters as before (activation function and output layer) and only changed the nodes in the input layer for effective comparison. See Figure 6 for the graph of feature importance after decision tree processing.

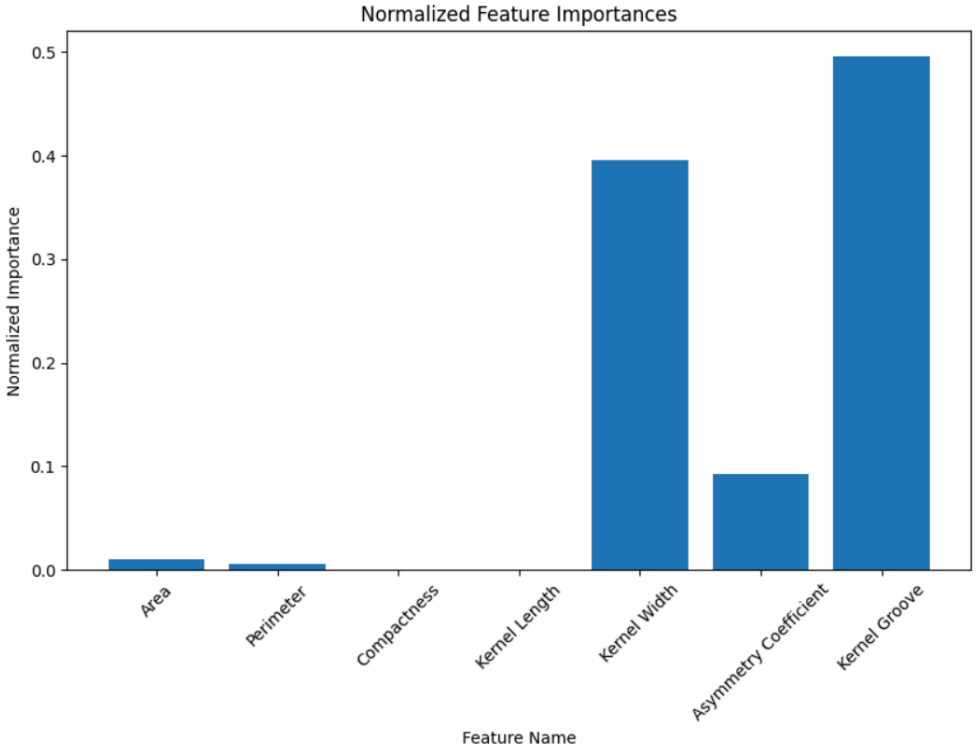


Figure 6: Bar Graph of Feature Importances

The three most important features after decision tree processing were kernel groove (0.5), kernel width (0.4), and asymmetry coefficient (0.1).

RESULTS

Model Accuracies

To assess the accuracy of the models, the study implemented categorical cross entropy, a common loss function used for multi-classification. Categorical cross entropy is sometimes known alternatively as softmax loss and is typically used together with the softmax activation function. Essentially, cross entropy functions measure the difference between model predictions and the true values, similar to residuals in linear regression (Bhardwaj, 2020a). Intuitively, when the predicted probability for the true value is high, the loss is low, and vice versa. The goal of revising a model is to minimize the loss, thus improving the overall accuracy. See Equation 3 for the commonly used equation for categorical cross entropy.

$$CELoss = -\sum_{i=1}^{K} y_i \cdot log(\hat{y}_i)$$

Equation 3: Categorical Cross Entropy Equation

Given true probability distribution $y = [y_1, y_2, ..., y_K]$ *where* $y_i = 1$ *for the true class and 0 for other classes (one-hot encoding) and predicted probability distribution* $\hat{y} = [\hat{y}_1, \hat{y}_2, ..., \hat{y}_K]$.

A major factor that changes the model accuracy is the number of epochs - the number of iterations through the complete training set. As the data is repeatedly processed through the model, the accuracy initially increases; however, too many epochs can cause overfitting, which decreases model accuracy. The models yield four values (training loss, training accuracy, validation loss, and validation accuracy) corresponding with the number of epochs. Training accuracy pertains to the model accuracy while iterating through the given training data; the overall accuracy of the model is represented by validation accuracy, the model accuracy when adapting to new samples. The first model reached its maximum accuracy at around 350 epochs, with a validation accuracy of 0.94 (94% accurate). For each model, the approximated epoch threshold was verified by testing higher epochs in intervals, such as 400, 600, 800, and 1000 epochs. The validation accuracies were confirmed to have plateaued; while loss continued to decrease very slightly, the change was negligible. See Figure 7 for the results of the initial model.

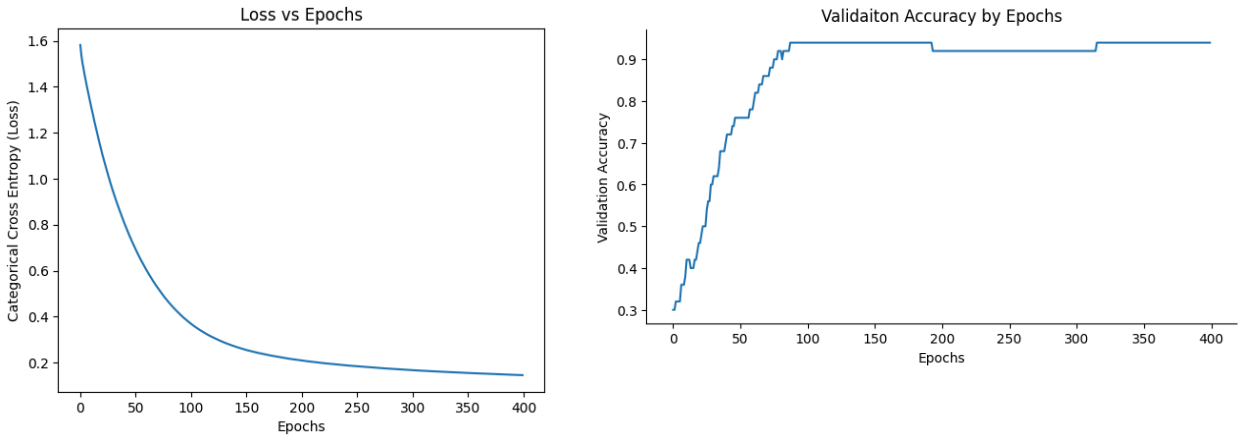


Figure 7: Results of Initial Model

The graph on the left shows the cross-entropy loss continues to decrease with more epochs, but gradually flattens at 350-400 epochs; at around 350 epochs, the validation accuracy line becomes constant at 0.94, shown on the right graph. There is a slight dip in at around 200-300, which can be attributed to model instability.

Revised Model

The revised model, using the subset of the original feature array obtained after feature selection, generated clearly superior results. The second model reached the maximum model accuracy at around 150 epochs, with a higher validation accuracy of 0.96. A comparison of the two models yields multiple notable observations. A significant increase in performance is seen in the second model, with the overall accuracy increasing from 0.94 to 0.96, due to feature selection optimizing the feature array. However, while the validation accuracy increased, there was a drop in training accuracy from approximately 0.97 to 0.93; this drop is likely attributed to elimination of additional features during feature selection. Another advantage of the revised model is the decrease in epochs from 350 to 150, resulting in better runtime and conservation of resources. Feature selection eliminated unintentional noise in the model, which helped the revised model reach the maximum accuracy with significantly fewer epochs. See Figure 8 for a comparison of results.

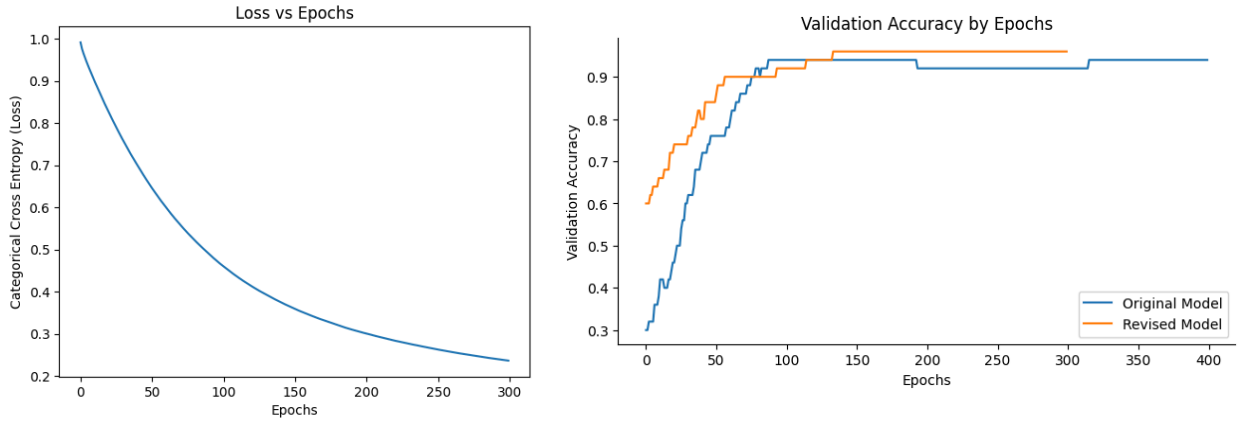


Figure 8: Results of Revised Model and Comparison

The graph on the left shows that the revised model reaches a low cross entropy loss significantly faster than the initial model. On the right, the Validation Accuracy vs Epochs graphs of the initial (blue) and revised (orange) models are plotted on the same graph.

From Figure 8, the accuracy of the second model starts at around 0.6, as opposed to the initial model, which starts at around 0.3; the immediately superior starting fit of the two models reinforces evidence on the reduction of noise from feature selection. Additionally, the graph of the first model exhibits unpredictable dips and deviations from the positive trend of accuracy, which vary each time the model is run. The revised model's graph appears cleaner, and decidedly plateaus at around 150 epochs, showing overall model stability.

DISCUSSION

The present study investigated wheat seed properties and classification through computer neural networks. By analyzing the association of the seven features (area, perimeter, compactness, kernel length, kernel width, asymmetry coefficient, and kernel groove) and the three labels (Type 1, Type 2, or Type 3),

the model can accurately predict the classification of new samples. The noticeable increase in performance observed with the second model justifies the importance of feature selection and supports that feature selection reduces unnecessary noise. The exceptionally low importances of the other features contrasted with the three selected features (kernel width, asymmetry coefficient, and kernel groove) form the conjecture that the variation in features like area, perimeter, compactness, and kernel length is not consequential to the sorting process. The selected features seem to be the main distinctive characteristics in the seed that the model detects which differentiate type. Thus, the results of feature selection incentivize future biological studies and modification of technology focusing on the three selected features, which can save time and resources.

However, the study has limitations, considering how the dataset contains only a few hundred samples. Future studies can be improved with larger datasets and deeper analysis of features; more comprehensive neural networks may require hidden layers. Other models can be implemented for deeper visualizations of the data and may yield additional analysis points. For example, models like K-nearest neighbors and cluster analysis explore how the distribution of samples affect model predictions, potentially reinforcing or complicating the results of feature selection. Models similar to the one implemented in this study can be applied to a broad range of crops, including other popular crops like maize and rice. As almost every major industry expeditiously digitalizes, the use of neural networks and machine learning is a crucial step toward replacing human shortcomings in the thousand-year-old art of agriculture.

REFERENCE

Wikimedia Foundation. (2024, August 6). Wheat. Wikipedia. https://en.wikipedia.org/wiki/Wheat

Waldkorn. (n.d.). Traditional grains, a blend of the best-known grains: Waldkorn®. Traditional grains, a blend of the best-known grains | Waldkorn®. https://www.waldkorn.com/en/our-grains/waldkorn-traditional-grains

Bhande, A. (2018). What is underfitting and overfitting in machine learning and how to deal with it. Medium. https://medium.com/greyatom/what-is-underfitting-and-overfitting-in-machine-learning-and-how-to-deal-with-it-6803a989c76

Bhardwaj, A. (2020a). What is cross entropy? Towards Data Science. https://towardsdatascience.com/what-is-cross-entropy-3bdb04c13616

Bhardwaj, A. (2020b). What is the softmax function? - Teenager explains. Towards Data Science. https://towardsdatascience.com/what-is-the-softmax-function-teenager-explains-65495eb64338

Charytanowicz, M., Niewczas, J., Kulczycki, P., Kowalski, P., & Lukasik, S. (2012). Seeds. UCI Machine Learning Repository. https://archive.ics.uci.edu/dataset/236/seeds

Chollet, F. (2023). The sequential model. Keras. https://keras.io/guides/sequential_model/

DelSole, M. (2018). What is one hot encoding and how to do it. Medium. https://medium.com/@michaeldelsole/what-is-one-hot-encoding-and-how-to-do-it-f0ae272f1179

Karjian, R. (2023). History and evolution of machine learning: A timeline. TechTarget. https://www.techtarget.com/whatis/A-Timeline-of-Machine-Learning-History

Plapinger, T. (2017). What is a decision tree. Towards Data Science. https://towardsdatascience.com/what-is-a-decision-tree-22975f00f3e1#:~:text=The%20decision%20to%20split%20at,belongs%20to%20a%20single%20class

Softmax function. (n.d.). BotPenguin. https://botpenguin.com/glossary/softmax-function

Sudharsan, A. (2018). Why, how and when to apply feature selection. Towards Data Science. https://towardsdatascience.com/why-how-and-when-to-apply-feature-selection-e9c69adfabf2

What is a neural network? (n.d.). TIBCO Software. https://www.spotfire.com/glossary/what-is-a-neural-network