Classification of Rice Varieties Using Machine Learning Algorithms

Amy Zhao
azhao150@gmail.com
Fremont, CA, USA

Spencer Morga
spencer.morga@gmail.com
Los Angeles, CA, USA

Cassidy Xu
cassidyh.xu1@gmail.com
Lexington, MA, USA

Marios Tsotras
kirbynin10do@gmail.com
Los Angeles, CA, USA

ABSTRACT

Rice is one of the most important and widely consumed food crop in the world, providing over 20% of calories consumed by the average individual. With over 120,000 different varieties of rice in the world, manually classifying rice grains is neither efficient nor practical. An automation process could replace human judgement and correctly identify individual rice grains according to their respective classes. The classes utilized in this project are: Arborio, Basmati, Ispala, Karacdag, and Jasmine. Features such as area, perimeter, eccentricity, etc were measured. Rice samples were classified through two methods: image and feature classification. The resulting accuracies averaged around 98%.

1 INTRODUCTION

1.1 Background

Rice is a staple food for more than half of the people on Earth. It is cultivated in over 100 countries with 90% of rice production being centralized in Asia, where rice takes reign as the staple food for many.

Rice varieties are distinguished by specific characteristics such as texture, form, and color. However, these distinct characteristics are often subtle due to the diminutive nature of a grain of rice along with multiple overlapping characteristics between the 120,000 different rice varieties, so much so the average consumer would likely be not able to differentiate between different rice varieties [1]. Still, it is feasible to categorize and evaluate the quality of seeds using these characteristics that separate rice kinds.

Rice quality inspections performed by humans are not reliable. Even inspectors who are industry experts are prone to human error [2]. Additionally, analysis on tiny grains of rice would take up an inordinate amount of time which can be put to better use in other sectors. Rice automation utilizing machine learning would optimally classify large amounts of rice data reliably.

1.2 Related Work

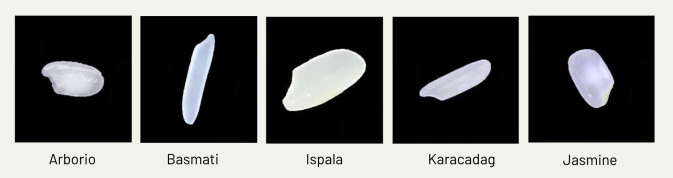
In 2021, Murat Koklu classified rice varieties utilizing Artificial Neural Network (ANN), Deep Neural Network (DNN), and Convolutional Neural Network (CNN) models. The CNN model classified with an accuracy rate of 100% [3]. This is due to the CNN method using numerous hidden factors such as size, color, etc. The DNN method had the greatest average classification rate of 99.95 percent as it can execute a wide range of learning in huge data sets. The ANN performed performed similarly to the other methods, with a classification accuracy of 99.87%.

Bhupinder Verma proposed a computer vision system to sort rice by their quality such as whether or not kernels were damaged or broken [4]. For their preprocessing, they used a smoothing filter to enhance and smooth images which removed noise and sharpened the edges. Then a binarization operation reduced the images to two grey-scale values. The resulting computer vision system designed helped grade and classify rice kernels accurately (better than 90%) and that too at a nominal cost.

2 DATA PREPARATION AND REPROCESSING

The rice image dataset is composed of 75,000 images and 5 varieties of rice: Arborio, Basmati, Ipsala, Jasmine and Karacadag. The dataset is divided evenly with 15,000 images pertaining to each species.

Figure 1: Rice samples used in the study.



A second rice dataset is used for feature classification [5]. This dataset corresponds to the same rice grains as the rice image dataset and is also composed of 75,000 entries. The dataset has 12 morphological, 4 shape and 90 color features.

2.1 Feature Classification

Features such as area, perimeter, roundness, major axis, and minor axis were measured from the images of each rice grain. With these features and more a total of three models were created: Logistic Regression without penalty, Logistic Regression with l2 penalty, and K-Nearest Neighbors Classifier with 5 neighbors. Undecipherable features such as kutosis, skewRR, and ALLdaub4RB were removed. Once the numerical features were scaled in the pipeline, the data was ready. The data was split with 15,000 data points as testing and the other 60,000 as training data.

2.2 Image Classification

For image classification models, we trimmed down the dataset to 500 images per rice variety for a total of 2,500 images. This is due to the original size of the dataset being to large for the capabilities of our limited resources, and lack of time to run the entire dataset. Each image was converted to tensors, an end dimensional array representing each pixel of the respective rice images.

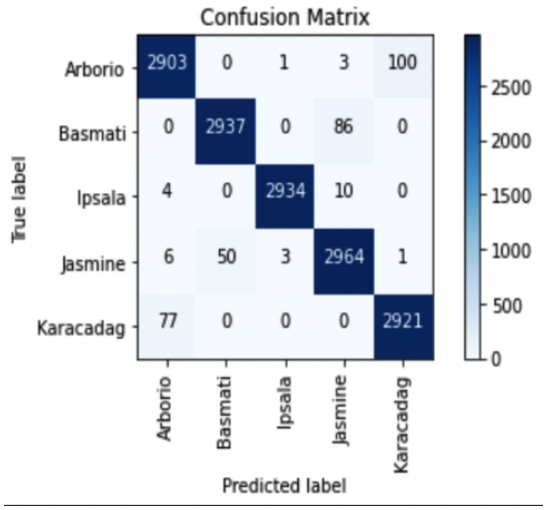
3 MODEL DESCRIPTIONS

3.1 Feature Classification Models

3.1.1 Logistic Regression (LR). Logistic Regression is a machine learning algorithm that is commonly used for predicting probabilities regarding one or more variables. For our project, we used a multinomial logistic regression. This model would use a lbfgs solver and would not place a penalty on the beta values. We chose to do a LR model without a penalty as the model may have been fitted properly and regularization may not have been necessary. This model would use a one vs. all method of classifying each class of rice.

In a multinomial logistic regression, a multidimensional linear model can be turned into a classification model, essentially turning a continuous model into a more discrete one. This model ranges from 0-1, 0 meaning a negative class or no classification, and 1 meaning a positive class. When such a model uses a one vs all method, instead of separating each class against each other, the model separates each class against the rest of them, somewhat like picking a suspect from a lineup. Lastly, since this model would not have a penalty on it, that means that its beta values would not be lowered if the model overfitted. When overfitted, a model's training accuracy is extremely high, but its validation accuracy would be much lower since the model would not to be able to predict extraneous data points.

Figure 2: Confusion Matrix of Logistic Regression - No Penalty.



3.1.2 Logistic Regression with L2 Penalty. A second multinomial logistic regression was created with the addition of an L2 (Ridge Regression) penalty on the beta values, the forumla of which can be found in Figure 4. This penalty typically prevents overfitting in a logistic regression. If this model became too accurate, meaning that it wasn't recognizing trends but rather getting its accuracy from a "connecting the dots" way, it would overfit the data and would not be able to perform well on validation data. But an l2 penalty ensures that the model recognizes trends and can actually predict extraneous data points.

Figure 3: Ridge Regression Formula

$$\|\mathbf{w}\|_2 = \left(|w_1|^2 + |w_2|^2 + \ldots + |w_N|^2\right)^{\frac{1}{2}}$$

Figure 4: Confusion Matrix of Logistic Regression - L2 Penalty.

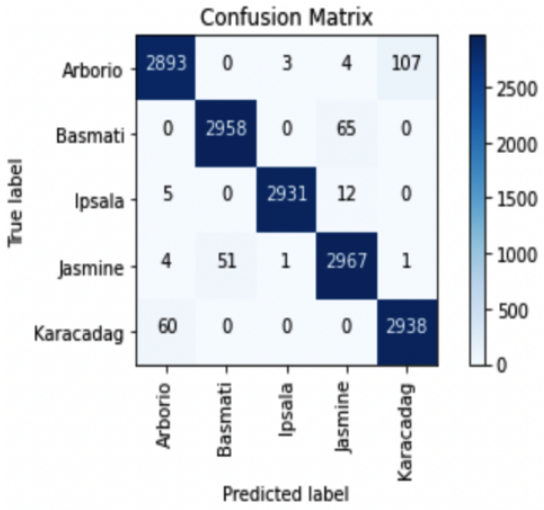
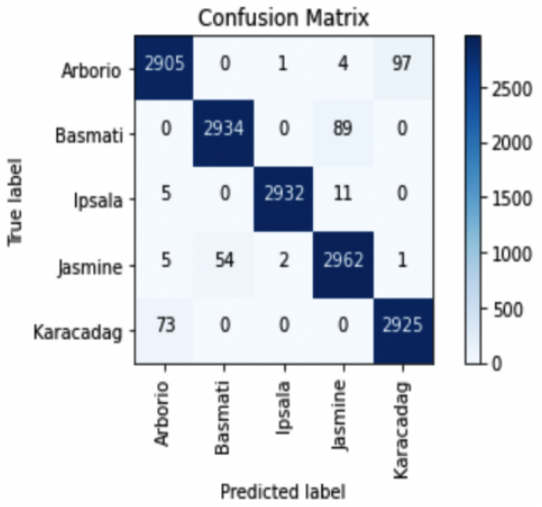


Figure 5: Confusion Matrix of k-NN Model.



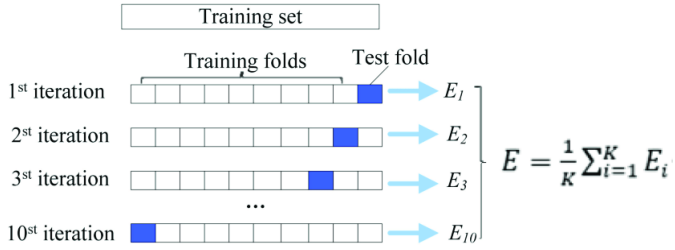
3.1.3 K-Nearest Neighbor(k-NN). A K Nearest Neighbors Classifier is a type of model which looks towards the nearest data points to predict a new data point. If the majority of points around a new point are part of a certain class, K-Nearest Neighbors will use majority voting to say that the new data point is also a part of this class. The K in the K Nearest Neighbors is a hyperparameter that lets the model know that it needs to find the k nearest points to make its prediction. After testing on a small range of k values up to 200 neighbors, a k value of 5 returned the highest accuracy, so this value was selected for all nearest neighbors models. Choosing the right k value can be problematic: too small and the model becomes extremely variable and overfit, but too large and the model becomes overly simplified and eventually merely represets the mean of all the data points.

3.1.4 K-Fold Cross Validation. A k-fold cross validation was also used to find the mean accuracy across folds of the highest performing logistic regression model as well as the k-nearest neighbors model. This means that the data is split across k equal measure groups and a model is trained on each one such that k-1 folds are training data and 1 fold is testing data. Then, the cross validation algorithm switches up which folds are test and train such that it eventually can account for all possibilities of train and test splits in the data across those folds. For the KNN, the accuracy across 10 folds was 0.9794. For the logistic regression model without a penalty the accuracy across 10 folds was 0.9776.

3.2 Image Classification Models

3.2.1 Convolutional Neural Network (CNN). CNN is a deep learning model that is commonly used in image processing, natural language processing, speech recognition, and data sets with a large amount of data. Typically, when neural networks are brought one first thinks about matrix multiplications but that is not the case with CNN, which uses a different technique called Convolutions, a mathematical operation on two functions which results in a third

Figure 6: 10-Fold Cross Validation.



function that indicates how the form of one is changed by the other. CNNs are composed of many layers of artificial neurons.

In the convolution layer, filters are applied to extract image feature information. This layer allows for a customized specified number of steps and filters, as well as the ability to raise and decrease the amount of characteristics. However, too many features may make it more difficult for the network to learn. In the pooling layer, operations reduce the complexity from the large amount of data coming from the convolution layer.

Following these operations, features are lowered to the level of the neural network in the fully connected layer as a classification layer, and learning operations are done to form inferences. A softmax activation function (Figure 9) is then used to parse the classes. A general model of the CNN structure can be found in Figure 7 [6].

3.2.2 Multiplayer Perceptron (MLP). A multi layer perceptron is one of the most widely used artificial neural network models. MLPs are intended to approximate any continuous function and to address issues that cannot be solved linearly. MLP neuron sequencing comes in the form of layers: the input layer, the output layer, and multiple hidden layers. The input layer receives information to be processed and solved. The output layer produces the information that was processed in the neural network. The real computational engine of the MLP is an arbitrary number of hidden layers inserted between

Figure 7: CNN General Structure

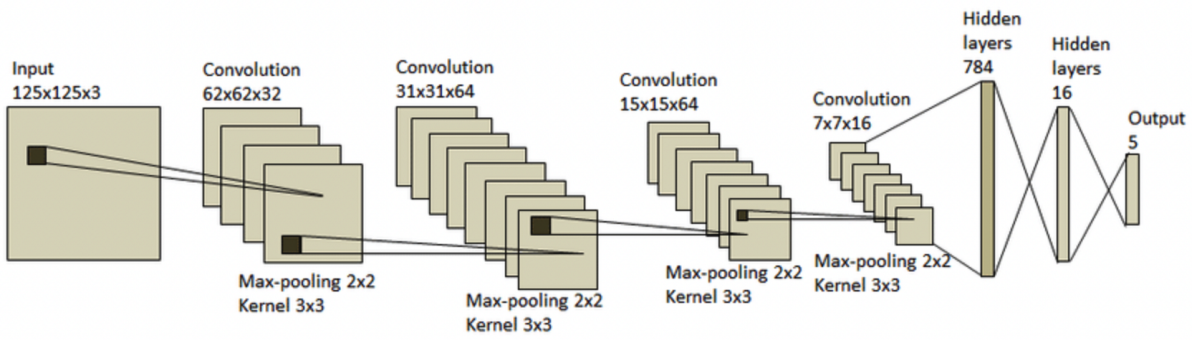


Figure 8: Confusion Matrix of CNN Model

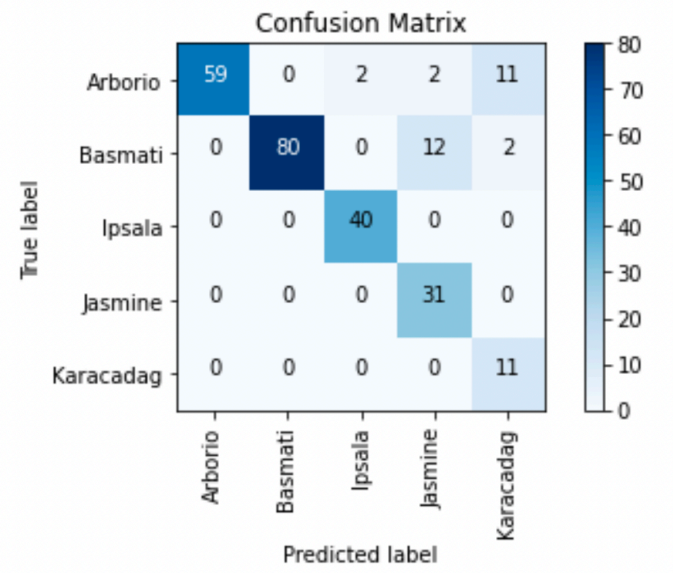
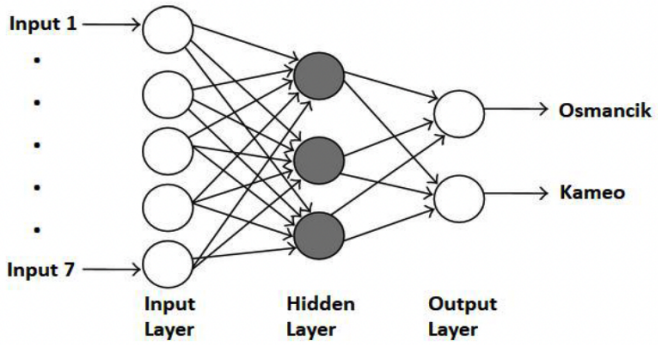


Figure 9: softmax activation function

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

the input and output layers. In an MLP, data travels in the forward direction from input to output layer, similar to a feed forward network. The back propagation learning technique is used to train the neurons in the MLP. A general model of MLP can be found in Figure 10.

Figure 10: MLP General Structure.



4 EVALUATIONS

4.1 Feature Models

In order to see how well the models did, Accuracy, Precision, Recall, and F1-score were all used. Accuracy was measured as the number of correct predicts over the total number of test data points. Precision was measured as the number of correct predictions for each class over the number of data points that was predicted to each class, respectively. Each of those were then averaged to get the final result. Recall was calculated similarly where the number of correctly predicted data points for each class was divided by the total number of points that were supposed to be predicted positive in that class. Again, the final value was the average between the five types of rice grains. F1-score was calculated using the average values for the Precision and Recall. Formulas for each metric are shown below.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1\text{-}score = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Measure	LR	LR-L2	k-NN
Accuracy	0.977267	0.97720	0.979133
Precision	0.977412	0.97736	0.979272
Recall	0.977338	0.97727	0.979191
F1-score	0.977338	0.97729	0.979204

Since the training data for each class was evenly split, using accuracy, precision, recall, and f1-score would all provide results that correctly represented the models. The k-NN model performed the best with all measures showing the highest scores. LR with an L2 penalization performed similar to but not as well as the LR which meant that the model was not overfit.

Of all of the feature classification models, the k-NN model with 5 neighbors worked best. This is most likely due to some inefficiencies in logistic regression. In logistic regression, the model essentially tries to find lines that can best split the data into two groups. In this case, it would be each type of rice grain against all the other types. Logistic regression, however, is not perfect. Sometimes, data points will be left out, costing the model some accuracy. K-NN does not have this problem. Instead, it looks around the model for the nearest 5 points, in this case, and essentially reports back the grain of rice that is found the most often. This most likely works better since classes will be in groups that are slightly mixed in with one another. This makes it hard for logistic regression to get an accurate model, but k-NN can perform better in this situation.

4.2 Image Classification Models

For the first test of the model, the learning rate was set at 0.01. This resulted in a low accuracy of around 20%. However, the CNN method was thought to produce a high classification success due to it directly processing images and utilizing features that would not be caught by other models such as size or color. To try and improve the accuracy, the learning rate was decreased to 0.001, which resulted in a much better model at 97.6% accuracy.

Epoch indicates the number of the passes the CNN model completed. For every epoch processed in the CNN model, both training and validation data loss decreased and accuracy increased logistically. The optimal amount of epochs peaks at around 4, as visualized in Figures 11 and 12.

5 CONCLUSION

The best performing feature model and the best performing image classification model compares at 97.91% and 97.6%, which the CNN model performing slightly worse than the k-NN model.

However, in terms of practicality, the CNN model trumps k-NN. Having access to a large number features requires more difficulty than utilizing a photo image. The convenience of automatic systems can be designed not only the ability to distinguish between rice varieties, but also detect blemishes and separate unwanted substances that may be mixed during rice production.

Figure 11: CNN Training and Validation Loss

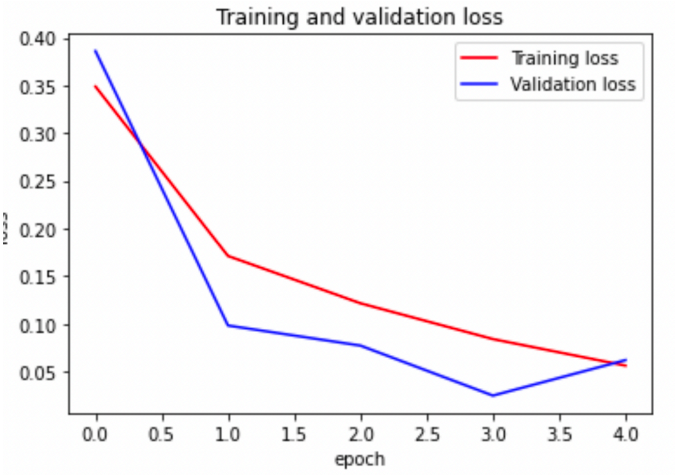
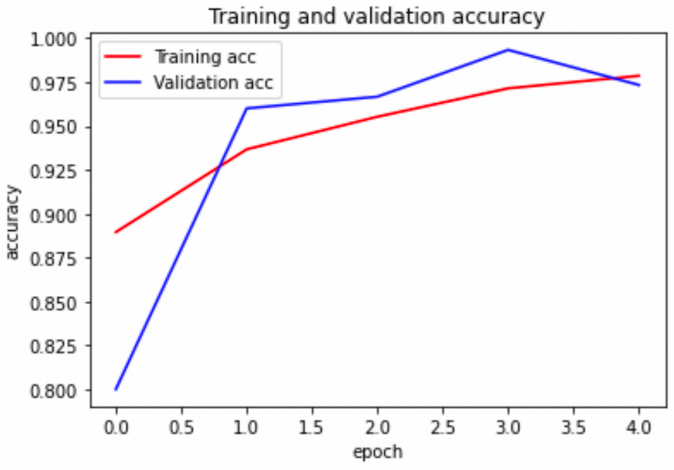


Figure 12: CNN Training and Validation Accuracy



REFERENCES

[1] Koklu M. I. Cinar. Determination of effective and specific physical features of rice varieties by computer vision in exterior quality inspection. *Selcuk Journal of Agriculture and Food Sciences*, pages 229–243, 2021.

[2] L. R. Saritha S. Arcot B. Arora, N. Bhagat. Rice grain classification using image processing & machine learning techniques. *2020 International Conference on Inventive Computation Technologies (ICICT)*, pages 205–208, 2020.

[3] Y.S. Taspinar M. Koklu, I. Cinar. Classification of rice varieties with deep learning methods. *Computers and Electronics in Agriculture*, 2021.

[4] B. Verma. Image processing techniques for grading & classification of rice. *2010 International Conference on Computer and Communication Technology (ICCCT)*, pages 220–223, 2010.

[5] I. Cinar M. Koklu. Classification of rice varieties using artificial intelligence methods. *International Journal of Intelligent Systems and Applications in Engineering*, 2019.

[6] I. Cinar M. Koklu. Identification of rice varieties using machine learning algorithms. *Journal of Agricultural Sciences*, 2022.