Christopher Gjersing
CS 422 1002 | Fall 2021
05 November 2021

Assignment 4: Predicting Money Authenticity Using Logarithmic Regression

While browsing the UCI Machine Learning Repository I happened upon the banknote authentication dataset. This dataset will allow simple binary classification, the banknote is either guessed to be real or fake. Using 1372 images of real and forged bills, the images were processed into 4 input features of the wavelet transformed image. The entropy of the original image, as well as the variance, skewness, and kurtosis of the transformed image make up the 4 input features used to predict authenticity. These inputs measure things like asymmetry, tailedness, and pixel spread. Kurtosis specifically helps us identify outlier information, something vital in determining whether or not the banknote is authentic.

Processing the data was very simple, separating the x and y vectors and then appending the bias term (all 1's) to the parameter vector (x). The intercept is the last value of the parameter vector. I was prepared to scale the data, something that improved my linear regression in the last assignment, but found successful results with the data as it is. For the algorithm I used the SGDClassifier from sklearn. This algorithm is preferred as it uses gradient descent. In the algorithm parameters I was able to set the loss function to 'log', which made the algorithm into a logarithmic regression algorithm that uses gradient descent with customizable learning rate and learning schedule. For this assignment I kept the algorithm simple, using the 'constant' learning rate schedule and a learning rate of 0.01. Experimentation with learning rate did not do much to impact results. The solution, evaluation metrics, and further discussion continue on the next page.

==== Logistic Regression with Gradient Descent Results ====

Solution (w): [[-2.78594305 -1.63661293 -1.98193051 -0.16140453 3.02425443]]

Convergence Iterations: 26

Learning Rate (eta0): 0.01

Evaluation Metrics:

<table>
<tr><th>Training Dataset</th><th>Test Dataset</th></tr>
<tr><td>Confusion Matrix<table><tr><td>604</td><td>8</td></tr><tr><td>4</td><td>481</td></tr></table></td><td>Confusion Matrix<table><tr><td>148</td><td>2</td></tr><tr><td>0</td><td>125</td></tr></table></td></tr>
<tr><td>Accuracy: 0.9890610756608933</td><td>Accuracy: 0.9927272727272727</td></tr>
<tr><td>Sensitivity: 0.9917525773195877</td><td>Sensitivity: 1.0</td></tr>
<tr><td>Specificity: 0.9869281045751634</td><td>Specificity: 0.9866666666666667</td></tr>
<tr><td>F1 Score: 0.9876796714579056</td><td>F1 Score: 0.9920634920634921</td></tr>
<tr><td>Log Loss: 0.3778228929064456</td><td>Log Loss: 0.25119691630793467</td></tr>
</table>

As evident by the metrics, the algorithm performed very well, and finished very quickly. Logarithmic regression converged in only 26 iterations while achieving ~99% accuracy on the training and test dataset. In the test shown here, the model got a 1.0 on sensitivity. This means there were no false negatives and the model correctly identified every authentic banknote as real. There were some false positives, but for a model trying to predict authenticity I find that having a model more likely to guess a fake bill as real is better than guessing a real bill is fake. Overall I was very impressed by the results of the logarithmic regression with gradient descent algorithm on this dataset. Other classification algorithms may be able to reach similar numbers to this model, but will find it difficult to improve on the ~99% accuracy that this model achieved.