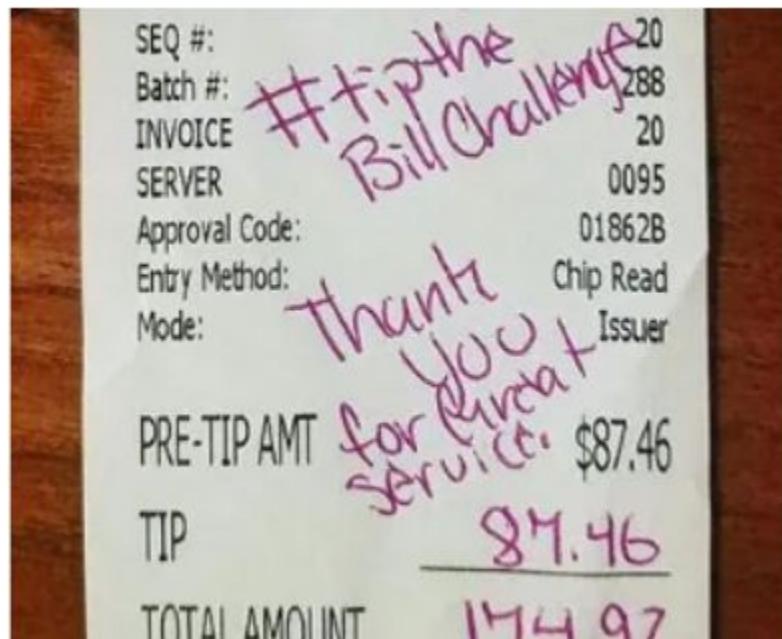


Tip Recognition

Jacob Alamparambil

Main Problem



Overview

- Load csv files
- Divide data into features and labels
- Normalize pixel values
- Gaussian Non Naive Bayes classifier
- Fit method
- Predict method
- Confusion matrix
- Results
- Conclusion

Step1: Load tip data

```
# Load CSV data
train_df = pd.read_csv("/content/drive/MyDrive/Colab Notebooks/MNIST_train.csv")
test_df = pd.read_csv("/content/drive/MyDrive/Colab Notebooks/MNIST_test.csv - MNIST_test.csv.csv")
```

Step2: Divide training and testing data from csv file

- Identified features for training set and updated dataframe to only include these columns
- Identified labels for training set
- Repeated this same procedure for testing set

Step 3: Normalize pixel values

- Make sure digits are in black and white for digit recognition

Gaussian Non Naive Bayes Classifier

- Definition
- Fit method
- Predict Method

Step4: Fit training data

1. Identify which samples belong to each class
 - a. Find all training images for our condition
2. Compute class mean vector
 - a. Take average value among all images of that class
3. Compute class covariance matrix
 - a. How pixel intensities vary together within class K
4. Compute class prior
 - a. Probability of randomly seeing class k
5. Store all parameters
 - a. Load up parameters to use in predict function

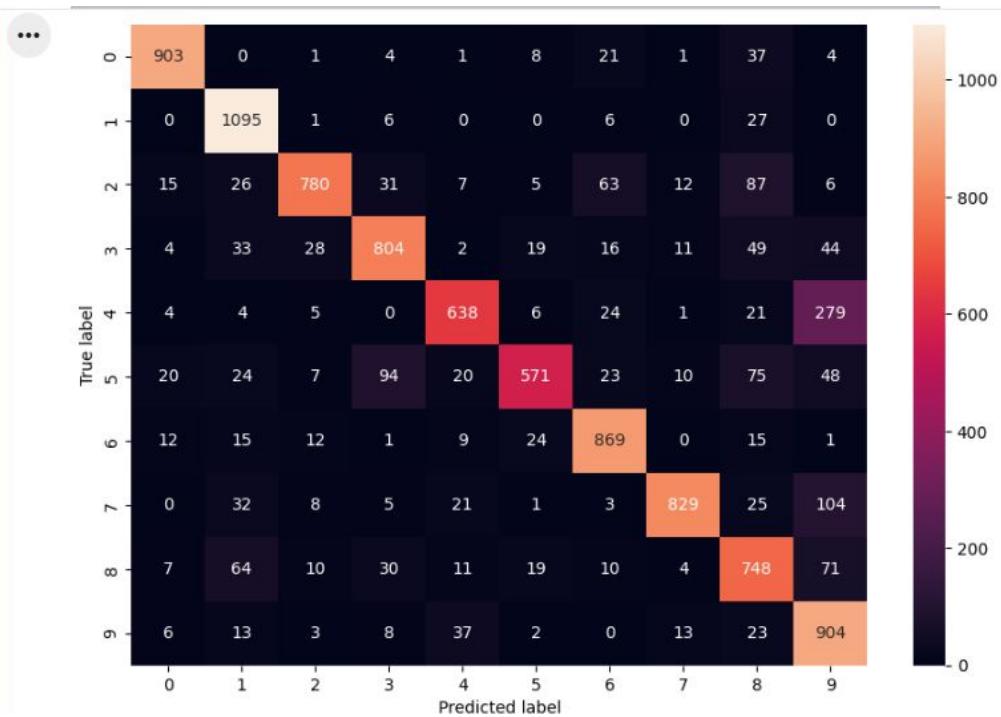
Step5: Predict

1. Compute how well each digit model matches input image
2. Compute how common digit occurs
3. Combine steps 1 and 2 to compute total score for digital k value
4. Return the highest score

Step 6: Confusion Matrix (Naive Bayes)

This matrix displays the frequency count of each digit in a heatmap format

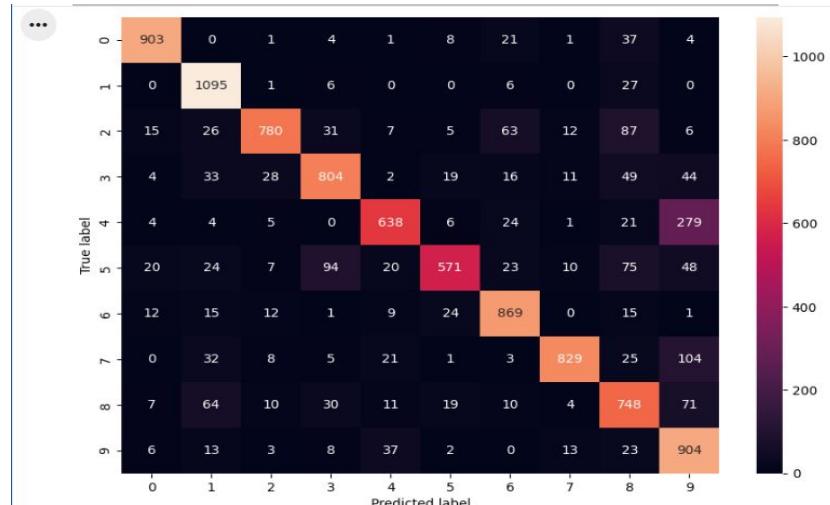
Model Accuracy: 81.41%



Step 6: Digit Accuracy (Naive Bayes)

Accuracy=diagonal value / rowsum

- Naive Bayes Classifier
 - Min:5 Max:1 Median:3/7

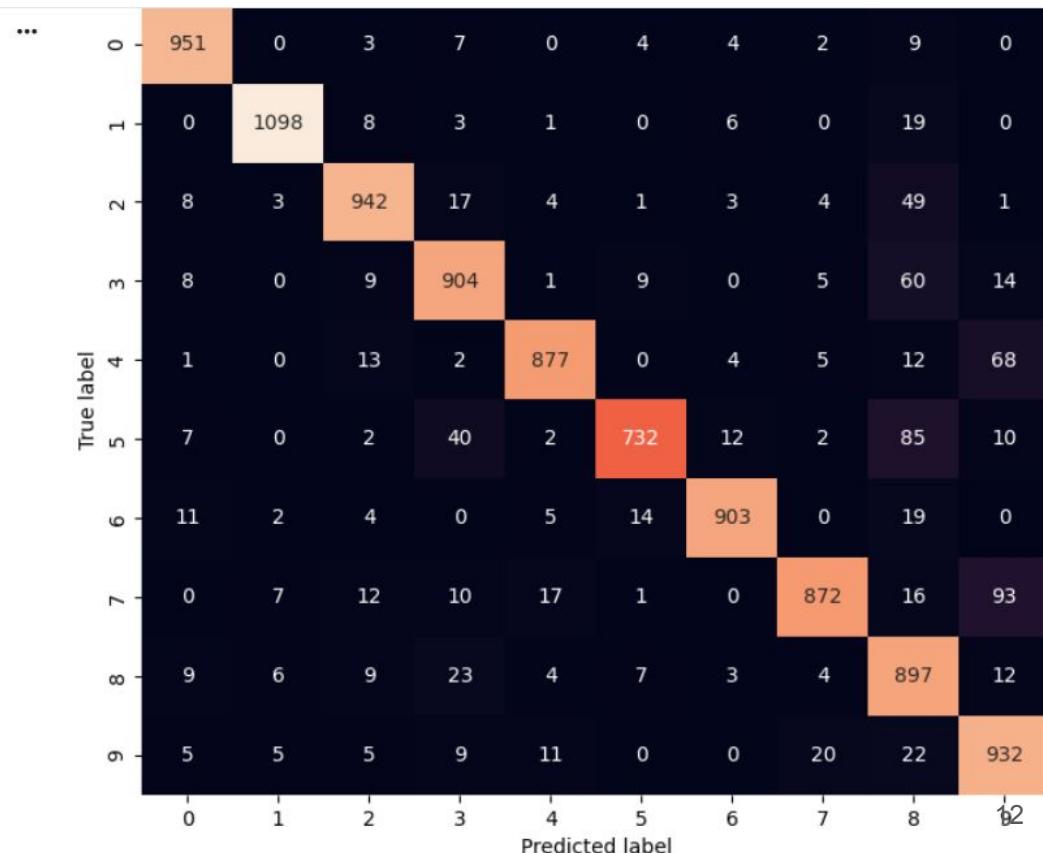


0	1	2	3	4	5	6	7	8	9
903/980	1095/1131	780/ 1032	804/1 010	638/982	571/892	869/958	829/102 9	748/974	904/994
0.9211	0.968	0.75 58	0.796	0.65	0.64	0.907	0.806	0.768	0.91

Step 7: Confusion Matrix (Non-Naive Bayes)

This matrix displays the frequency count of each digit in a heatmap format

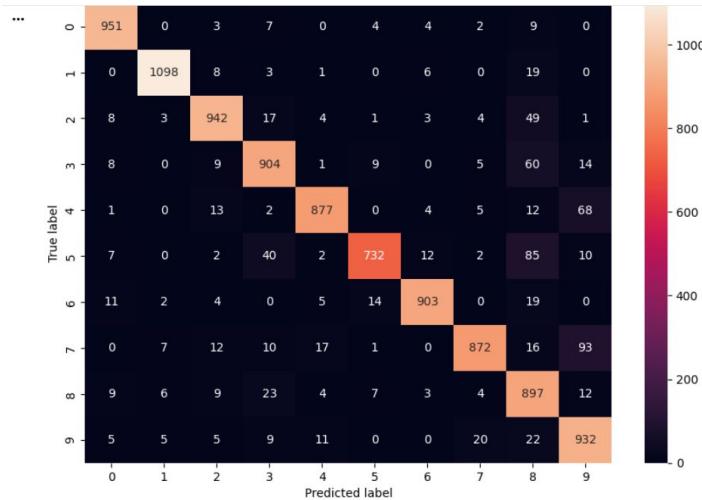
Model Accuracy: 91.08%



Step 7: Digit Accuracy (Non-Naive Bayes)

Accuracy=diagonal value / rowsum

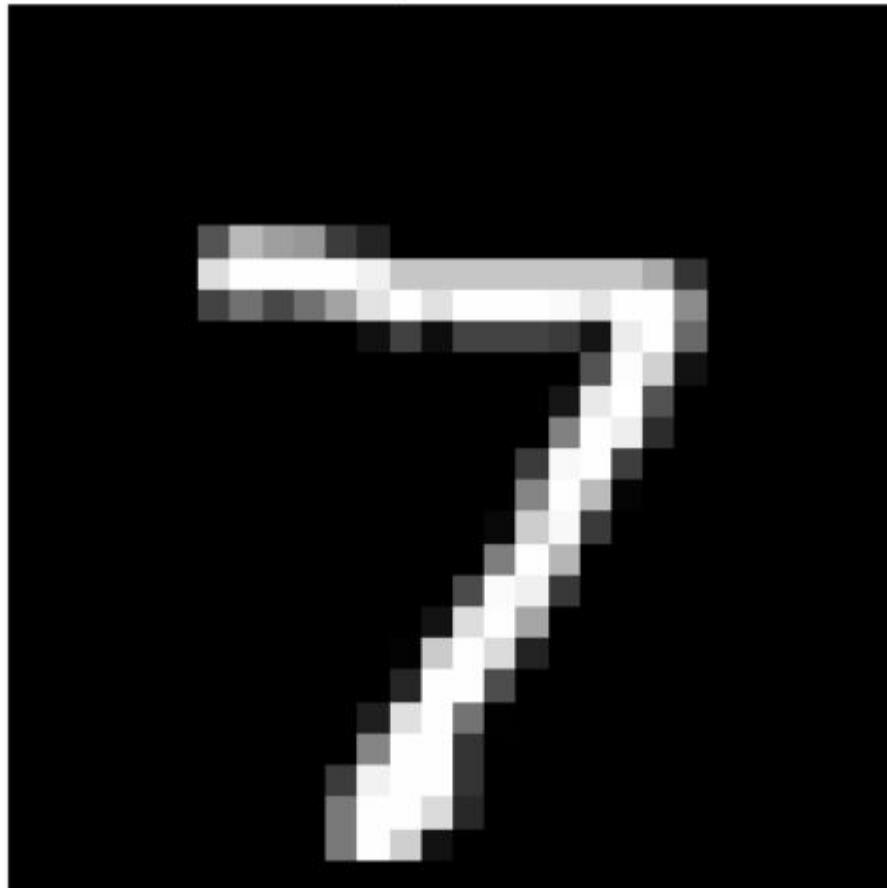
- Min:5
- Max:0
- Median:2/9



0	1	2	3	4	5	6	7	8	9
951/976	1098/1136	942/ 1030	904/1 010	877/982	732/892	903/958	872/102 9	897/965	932/100 4
0.9744	0.9665	0.91 46	0.895 0	0.8931	0.8206	0.9426	0.8474	0.9295	0.9283

...

True: 7 | Predicted: 7



Results

- Reshape each image as 28x28 pixel
- Return results of first 10 images

Conclusion

- Automate receipt and tip digitization
- Reduce manual data entry
- Increase accuracy
- Enhance operational efficiency in the food service industry