**Summary**

**Dataset**

* This project aims to recognize handwritten digits using the MNIST dataset, which consists of 60,000 training images and 10,000 test images of digits (0-9).

# **Data Preprocessing:**

# **Data Loading:** The MNIST dataset was loaded from CSV files containing pixel values (0-255) and corresponding labels.

# **Normalization:** The pixel values were normalized to the range of 0 to 1 to improve the efficiency of the algorithm.

# **Reshaping:** The images, originally represented as 784-dimensional vectors, were reshaped into 28\*28 matrices for visualization purposes.

**Implementation:**

* **Naives Bayes:** A manual implementation of the NB algorithm was used, which calculates the Euclidean distance between a test image and all training images. The algorithm then predicts the label of the test image based on the most frequent label.
* **Prediction: Model Performance:** The classifier performs well, but there are specific classes where misclassification is more frequent.**Potential Improvements:** Focus on improving the classification of classes with significant off-diagonal elements by possibly fine-tuning the model or incorporating additional features.Further tuning makes the model overfitting.
* **Gauss Bayes:** A manual implementation of the Gauss algorithm was used, which calculates the Euclidean distance between a test image and all training images. The algorithm then predicts the label of the test image based on the most frequent label .visualized by the confusion matrix
* **Prediction:** This Model gives a detailed breakdown of the model's performance for each class, highlighting both its strengths and areas where it may need improvement. Overall, the model seems to perform well, with relatively few errors compared to the total number of instances.visualized by the confusion matrix.
* **K-Nearest Neighbors (KNN):** A manual implementation of the KNN algorithm was used, which calculates the Euclidean distance between a test image and all training images. The algorithm then predicts the label of the test image based on the most frequent label among the K nearest neighbors.
* **Prediction:** The classifier performs well. The predicted labels were compared with the true labels to evaluate the model's performance.The confusion matrix says the model performs with high diagonal elements.visualized by the confusion matrix

**Tools Used:**

**Python Libraries** used Numpy,Pandas used for Data processing and Normalization process and Matplot used for Data visualization. Then KNN, Gauss Bayes and Naives bayes algorithm written Manually without using Sklearn.

# **Conclusion:**

* The project successfully demonstrated handwritten digit recognition using the Naive Bayes ,Gauss Bayes and KNN algorithm.
* The manual implementation provided a deep understanding of how Naive Bayes,Gauss Bayes and KNN algorithm works, including its computational challenges with large datasets.
* Visualization of the results enabled a straightforward comparison between true and predicted labels, showcasing the effectiveness of the model and identifying areas for further optimization.