# TRAINING OF MNIST DATASET USING VARIOUS CLASSICAL MACHINE LEARNING ALGORITHMS:

Shahira Aslam Khan

September 2025

**Introduction:**

The MNIST (Modified National Institute of Standard and Technology) dataset is a collection of 70,000 handwritten digits (0-9). Each image is 28x28 pixels, i.e.,784 features (grayscale intensity). It is generally divided in a 9:1 ratio with 60,000 training data and 10,000 test data. Here we are implementing various classical Machine Learning (ML) algorithms, to compare their performance and find the most suitable algorithm.

**About the Dataset:**

The MINST dataset is in ubyte format and follows a well-defined binary file structure. There are four files, two for training and two for testing:

Training set images -> train-labels-idx3-ubyte

Training set labels -> train-labels-idx1-ubyte

Test set images -> t10k-images-idx3-ubyte

Test set labels -> t10k-labels-idx1-ubyte

Each image is 28x28 with 784 bytes and pixel intensity of 0 (white background) and 255 (black foreground).

**Classical ML Algorithms used:**

* Decision tree
* Naïve bayes gaussian algorithm
* Naïve bayes multinominal algorithm
* K Nearest Neighbour algorithm
* Support Vector Machine – Linear kernel
* Support Vector Machine – RBF kernel
* Bagging with Decision tree as base estimator
* Boosting – AdaBoost and XG Boost
* Random forest

**Inference:**

Based on the observations made on the algorithm’s performance metrics:

1. SVM with RBF is the most accurate but not practical for large-scale prediction (slow).
2. XG Boost and Random Forest give nearly the same accuracy (~97%) with much faster inference, making them more scalable.
3. KNN performs excellently in accuracy but is computationally infeasible for real-time use.
4. Naïve Bayes Gaussian struggles because MNIST pixel distributions don’t fit Gaussian assumptions; Multinomial NB works better.
5. AdaBoost is not effective on MNIST, likely due to its reliance on weak learners that can’t capture image complexity.

**Graph:**

DT – Decision Tree

NBG – Naïve Bayes Gaussian

NBM – Naïve Bayes Multinominal

KNN – K Nearest Neighbour

SVM – Support Vector Machines (Linear kernel)

SVM\_RBF – Support Vector Machine (RBF kernel)

BAG – Bagging with Decision Tree as base estimator

RF – Random Forest

AB – AdaBoost with Decision Tree as base estimator

XG – XG Boost

A graph with a line

AI-generated content may be incorrect.

A graph with numbers and lines

AI-generated content may be incorrect.

A graph with blue lines

AI-generated content may be incorrect.

**Conclusion:**

* For best balance of accuracy & efficiency, we should use XG Boost or Random Forest.
* For highest possible accuracy we should choose SVM RBF.
* For very fast baseline models, use Decision Tree or Multinomial Naïve Bayes.