**Handwritten Digit Recognition Machine Learning Project Writeup**

* **Problem Statement:** The goal of this project is to build a machine learning classifier that is able to correctly identify handwritten digits.
* **Potential Methods and Models**: This project will rely primarily on K-Nearest Neighbors (KNN) and Random Forest classifier modeling to distinguish the digits in the images. Convolutional Neural Networking (CNN) may potentially be incorporated to improve accuracy.
* **The Data**: The data set is provided by Kaggle for their [Digit Recognizer Machine Learning Competition](https://www.kaggle.com/c/digit-recognizer). The data is broken into two sets: a training set and a test set with 42,000 and 28,000 images, respectively. The data sets are derived from the MNIST database of handwritten digits.
  + [The data files contain gray-scale images of hand-drawn digits, from zero through nine. Each image is 28 pixels in height and 28 pixels in width, for a total of 784 pixels in total. Each pixel has a single pixel-value associated with it, indicating the lightness or darkness of that pixel, with higher numbers meaning darker. This pixel-value is an integer between 0 and 255, inclusive.](https://www.kaggle.com/c/digit-recognizer/data)
  + [Each pixel column in the training set has a name like pixelx, where x is an integer between 0 and 783, inclusive. To locate this pixel on the image, suppose that we have decomposed x as x = i \* 28 + j, where i and j are integers between 0 and 27, inclusive. Then pixelx is located on row i and column j of a 28 x 28 matrix, (indexing by zero).](https://www.kaggle.com/c/digit-recognizer/data)
* **Challenges:** The vast disparity of the handwriting samples will present some challenges to any classification algorithm. The data, however, is complete and has been widely used.
* **Benchmarks**: This data set is widely cited in data science and machine learning literature as it is a benchmark for classification algorithms. Benchmarks for KNN algorithms range from 95-99.4 accuracy. CNN algorithms generally perform better ranging from 98.3-100% accuracy.
* **Goals**: My goal is to build a classifier with at least 95% accuracy.