CLASSIFICATION OF MNIST DATASET USING MATRIX PENCIL WITH DEEP NEURAL NETWORK

1st Rishik Gupta
Manipal University
Jaipur, India
rishik.209301615@muj.manipal.edu

2rdPratiksh Kumar
IIITA
Prayagraj, India
pratiksh.sk@gmail.com

3thMantek Singh
Google LLC
California , USA
mantek.singh2@gmail.com

4nd Bagesh Kumar
Manipal University
Jaipur, India
bagesh.kumar@jaipur.manipal.edu

5th Prakhar Shukla
IITA
Prayagraj, India
iec2022027@iiita.ac.in

6th Yuvraj Jindal
IIITA
Prayagraj, India
jindalyuvraj2@gmail.com

7th Sagar Gupta
IIITA
Prayagraj, India
sg07072003@gmail.com

***Abstract*—This paper presents the results of a deep learning project that utilizes TensorFlow and the MNIST dataset for pairwise classification of handwritten digits using a deep neural network. The project aims to build binary classifiers for each pair of digits and evaluate their performance. The process involves calculating covariance matrices, regularizing them, computing eigenvalues and eigenvectors, and projecting the data onto the eigenvectors. A deep neural network model with more than one layers along with the non linear activation functions is utilized for binary classification. The results represents the effectuality of deep neural networks in pairwise classification, with implications for image recognition and computer vision applications. The research also discusses potential areas for improvement and future research directions.**

***Index Terms*—MNIST Dataset, Deep Learning, Matrix pencil, Binary Classification, Deep Networks**

I. INTRODUCTION

In this study, we delve into a narrative approach to classifying the MNIST dataset by synergizing matrix pencil techniques with deep neural networks. Matrix pencil methods, renowned in signal processing and system identification, bring a unique perspective to data representation. By integrating these techniques with deep neural networks, we aim to elevate classification accuracy and robustness. The matrix pencil acts as a feature extraction mechanism, unveiling latent patterns within the data, while the deep neural network serves as a sophisticated classifier capable of learning intricate hierarchical representations.

Our exploration encompasses the theoretical foundations of matrix pencil methods, their application to the MNIST dataset, and their seamless integration with deep neural networks. The primary goal is to achieve superior classification performance and gain insights into the underlying structures of handwritten digits. This research seeks to contribute to the evolving landscape of intelligent systems by combining traditional signal processing approaches with advanced deep learning techniques, ultimately advancing the state-of-the-art in MNIST digit classification.

The study presents findings from a machine learning research endeavor that utilized a deep neural network and TensorFlow to classify pairs of handwritten digits using the MNIST dataset. MNIST, a widely used dataset, comprises 70,000 grayscale images of handwritten numbers, with 60,000 images used for training and 10,000 for testing. Our objective is to develop a deep neural network model capable of reliably categorizing pairs of digits by generating binary classifiers for each pair.

The research initiates with an overview of the preliminary project steps, including loading and normalizing the MNIST dataset and structuring the data into a dictionary. We then elaborate on the process of creating binary classifiers for pairwise classification, involving the computation of covariance matrices, regularization of matrices, calculation of eigenvalues and eigenvectors, and projection of the data onto the eigenvectors. Additionally, we discuss the architecture of the deep neural network model used for binary classification, featuring multiple layers with non-linear activation functions.

Subsequently, the study presents the project results, encompassing test loss and accuracy for each pair of digits. We thoroughly analyze the results, discussing their implications, including the performance of the deep neural network model in comparison to other machine learning algorithms.

II. LITERATURE REVIEW

Bhagat and Joshi [1] introduced a unique classification technique that quantifies differential information between classes using the Matrix Pencil equation in their study, a method used in radar and quantum applications, has been proven effective in binary classification on the MNIST database, using various feature sets and eigenvector projections.

Jian Chen [2] in his research presents an adaptive Matrix Pensil

(MP) technique that uses wavelet soft-threshold de-noising to identify Low-Frequency Oscillations (LFO) in power systems. The algorithm determines the Singular Value Decomposition order by handling noisy LFO signals in an efficient and self-adaptive manner. Case examples demonstrate that the method accurately detects LFO modes even when noise is present.

According to Somayyeh's [3] article, the Matrix Pensil Method (MPM) technique can extract vital signs from microwave sensor inputs. MPM outperforms traditional methods such as VMD and BPF in the extraction of heartbeats from CW and UWB sensor data. It also improves heart rate measurement accuracy and has a stronger correlation with reference ECG data. MPM is useful for non-contact vital sign monitoring in healthcare and sports performance evaluation.

Raviraj's [4] work introduces the Matrix Pencil algorithm, a signal processing tool for target discrimination in the PondEX dataset. The algorithm employs spectrum analysis and system identification, representing time-domain signals with complex exponentials. Future possibilities include discriminating based on mode features and combining Matrix Pencil results with machine-learning-based algorithms

Bhagat's [5] paper introduces a new method for pattern categorization and differentiation based on the characteristic equation of a matrix pencil. With different covariance matrices, this method solves binary and multi-class classification problems. To capture differential information between classes, it employs matrix pencil equations, whitening operators, and eigen decomposition. Experiments show that it works.

Birjit's [6] paper puts machine learning models to the test on handwritten datasets, with a focus on recognition, segmentation, character extraction, and image preprocessing. SVM, Random Forest Classifier, Decision Tree Classifier, Nave Bayes, and K-Nearest Neighbor are among the models used. In the shortest amount of time, SVM achieves the highest accuracy of 95.88%. The study emphasizes the difficulties in feature extraction and handwritten digit identification.

Jiarun [7] in his paper Co-Correcting: Learning with Noisy Labels on Medical Image Datasets, used an innovative method for learning with noisy labels in medical image datasets is presented. For increased robustness, a dual-network architecture, a label probabilistic module, and a label correction curriculum are used. Results from experiments on the MNIST dataset show that Co-fixing outperforms typical tactics in terms of improving model accuracy and fixing noisy annotations.

Goel's [8] article looks at how Machine learning and convolutional neural networks (CNNs) can be used to classify handwritten digits in Gujarati. To address the lack of publicly available datasets, they propose Gujarati Handwritten Digits (GHDD) and ten pre-trained CNN models. Recall, accuracy, precision, F1-score, AUC, and computational characteristics are all evaluated. The EfficientNetV2S model outperforms all others.

III. About Dataset

We obtained the dataset titled "Digit Recognizer" from Kaggle, commonly referred to as the MNIST dataset. Widely recognized as a benchmark dataset in the fields of machine learning and computer vision, MNIST comprises a vast collection of grayscale images featuring handwritten digits. Each image is meticulously labeled with the corresponding digit, ranging from 0 to 9. Frequently employed for image classification, digit recognition, and pattern recognition tasks, the collection has seventy thousand pictures in total. Ten thousand of them are set aside for testing, while the remaining sixty thousand are used for training. Every image in the dataset is a grayscale representation of 28 by 28 pixels, for a total of 784 pixels per image. Integer values between 0 and 255 are used to represent these pixels; 0 denotes black and 255 denotes white.

The MNIST dataset is a popular choice for researchers and practitioners developing machine learning algorithms due to its simplicity and user-friendliness. Its standardization and comprehensive labeling make training and testing models easy. It is particularly useful for binary classifiers, allowing for distinct digit classification between 1 and 7 or 3 and 8.

This research explores the effectiveness of machine learning techniques and algorithms in accurately classifying handwritten digit pairs using the MNIST dataset. It covers loading, preprocessing, division into training and testing sets, training binary classifiers, and evaluating their efficiency measured by F1 score, accuracy, recall, as well as precision.

Also, providing insights into the effectiveness of pairwise classification models and their generalization capabilities. Its standardized format and extensive collection of labeled samples make it an ideal choice for training and testing.

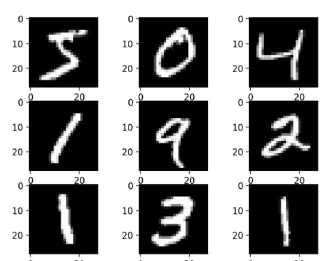

Fig. 1. Dataset Sample Images

IV. Proposed Methodology

A. Model Evaluation

We instantiated a Deep Neural Network (DNN) model, which belongs to the class of artificial neural networks characterized by the presence of multiple layers of interconnected nodes, referred to as neurons. These networks aim to emulate the structure and functionality of the human brain, facilitating the learning and prediction of intricate patterns within complex datasets. In the architecture of a deep neural network, information traverses through numerous layers of neurons, each layer executing a specific computation. Typically organized hierarchically, these layers follow a sequence where each layer receives input from the preceding one and generates output for the subsequent layer. The initial layer is designated as the input layer, the terminal layer as the output layer, and intervening layers as hidden layers. Computational tasks within each layer are executed by neurons applying a non-linear activation function to the weighted sum of their inputs. This incorporation of non-linearity enables the network to discern intricate patterns and relationships within the data. During the training process, the weights associated with each inter-neuronal connection are learned, as the network adjusts parameters to minimize a predefined loss function.

The Keras library was used to create binary classification models for distinct digit pairs. The code uses a sequential neural network with multiple dense layers, dropout for regularization, and sigmoid activation for binary classification. The training process lasts 10 epochs, and the models are stored in a loop structure. The ReLU activation function is applied to hidden layers and the sigmoid function for output.

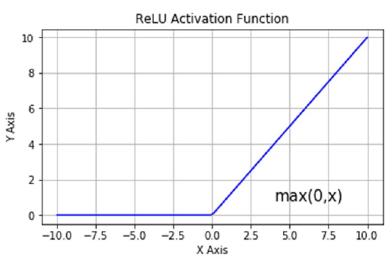

Fig. 2. Fig. ReLU Activation function used in hidden layer

The depth of a neural network is characterized by the quantity of hidden layers it encompasses. In the context of deep neural networks, it is customary for these networks to feature more than two hidden layers, affording them the capacity to acquire hierarchical representations of input data. The incorporation of multiple hidden layers facilitates the learning of diverse levels of abstraction. This enables deep neural networks to adeptly model intricate relationships within the data and extract high-level features from raw input data. Notably, in our implementation, we opted for the sigmoid activation function for the output layer.

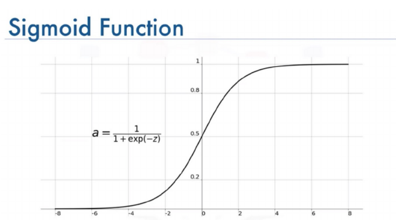

Fig. 3. Fig. Sigmoid function used in output layer.

Deep neural networks have garnered notable success across diverse domains, demonstrating proficiency in tasks such as image and speech recognition, natural language processing, and reinforcement learning. Their remarkable capability to autonomously learn and extract features from extensive datasets positions them as a potent tool for addressing intricate challenges and propelling advancements in artificial intelligence research.

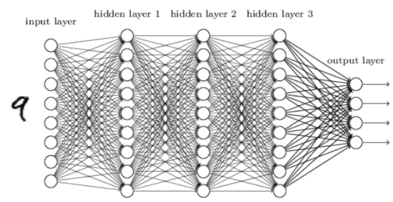

Fig. 4. Fig. Deep Neural Network

In conclusion, deep neural networks represent sophisticated computational models characterized by multiple layers that acquire hierarchical representations of data. Employing non-linear activation functions and modifiable weights, these networks effectively capture intricate relationships and extract meaningful features. Their capacity to learn from extensive datasets has led to transformative impacts across various domains, facilitating advanced pattern recognition and decision-making capabilities.

In practical application, the utilization of convolutional layers in a Convolutional Neural Network (CNN) is exemplified. Specifically, in the analysis of light curve data, the CNN identifies significant features, enabling the model to discern subtle changes indicative of exoplanetary transits. The incorporation of max pooling contributes to focusing on salient features, while batch normalization ensures stable training, thereby enhancing the model's proficiency in distinguishing exoplanet signals from background noise.

B. Matrix Pencil

The matrix pencil method, a mathematical technique extensively employed in signal processing and system identification, serves the purpose of estimating parameters for linear time-invariant systems. This methodology entails a comprehensive analysis of the eigenvalues and eigenvectors inherent in a matrix pencil—a construct formed by the amalgamation of two matrices.

In the realm of pairwise classification, the matrix pencil method finds applicability in the comparison and classification of distinct data classes. Specifically, by constructing covariance matrices for two classes, denoted as digits i and j, a matrix pencil is generated through the combination of these matrices. The formulation of the matrix pencil is expressed as:

$$M(s) = A + sB$$

Here, A and B denote the covariance matrices for classes i and j, respectively, while s represents a scalar parameter varying between 0 and 1. The eigenvalues and eigenvectors derived from the resulting matrix pencil offer valuable insights into the similarities and differences between the classes.

In particular, the eigenvalues of the matrix pencil $M(s)$ encapsulate the characteristic frequencies or modes of the system. The computation of these eigenvalues involves solving the generalized eigenvalue problem, contributing to a deeper understanding of the nuanced relationships and distinctions within the data classes.

$$M(s)v = \lambda v \tag{1}$$

In the eigenvalue-eigenvector relationship, where v represents the eigenvector and λ the eigenvalue, the eigenvectors corresponding to dominant eigenvalues serve as indicators of the directions or axes along which the classes can be effectively separated. Through the projection of data onto these eigenvectors, a transformation of the original data into a lower-dimensional space is achieved, thereby enhancing the separability between different classes.

The matrix pencil method emerges as a robust tool for both feature extraction and dimensionality reduction, facilitating the efficient classification of pairwise data. Within the specific context of pairwise classification of handwritten digits in the MNIST dataset, the matrix pencil method finds application in identifying the most discriminative features that distinguish one digit from another. For instance, when considering digits 1 and 7, the matrix pencil method proves valuable in discerning features such as the presence or absence of a horizontal stroke or a diagonal line. This capability makes it a powerful technique for uncovering key characteristics that contribute to the distinctiveness of individual digits in the classification process.

V. EXPERIMENTATION

A. Binary Classification using Matrix Pencil

The testing was done in an attempt to solve the MNIST dataset's (Modified National Institute of Standards and Technology Database) binary classification problem. After input photos were vectorized, the feature vectors X and Y, which represent classes C1 and C2, were produced. For the classification challenge, a typical D-Nearest Neighbour (D-NN) classifier was then used.

Let's have a look at the matrix pencil that goes with the characteristic equation as an example.

$$(A - \lambda B)x = 0 \tag{2}$$

Class C2 is deemed to have covariance matrix A in this arrangement, which is greater than class C1's correlation matrix B. Class C1 is therefore recognized as the reference class in this situation. According to the theoretical framework discussed in the sections above, the matrix pencil approach makes it easier to quantify information that differs between classes.

Given this, we suggest adding a modified feature set that consists of a set of projection coefficients along the matrix pencil's eigen basis supplemented with the eigenvectors of the reference class (C1). By using equations 1 through 3, the eigenvectors of the matrix pencil are obtained. It is intended for this suggested feature set, indicated by [placeholder for the specific feature set notation], to encompass.discriminative information essential for enhancing the classification process.

$$AB; B$$

,
The feature set is defined as a vector that includes projection coefficients along the eigenvectors of the matrix pencil and the reference class. It is represented by the symbol [placeholder for the specific feature set notation]. On different feature sets, simulations are run to assess its efficacy. These feature sets consist of projection coefficients and matrix pencil's eigenvectors.

Table 1 presents the systematic recording and presentation of the simulation results, including the categorization accuracy. This tabulation sheds insight on the effectiveness and impact of the suggested alterations by offering a comparative study of the classification performance attained with various feature sets.

B. Observation regarding Simulation Outcomes

The eigenvectors of the matrix pencil are calculated below to provide additional explicitness to the simulations that were run. Let's start with the matrix pencil's characteristic equation.

$$det(A - \lambda B)$$

$$\leftrightarrow det(A - \lambda \phi^{-1} \wedge \phi)$$

$$\leftrightarrow det(\phi^{-1}(\phi A \phi^{-1} - \lambda \wedge)\phi)$$

$$\leftrightarrow det(\phi^{-1})det(\phi^{-1} A \phi - \lambda \wedge)det(\phi) = 0$$

For simplification we define $\wedge = N2$.
Since ϕ is the eigenvector matrix of class C2
$det(\phi) = det(\phi 1)$ is not equal to 0. Thus we can write:

$$det(\phi A \phi^{-1} - \lambda NN) = 0$$

$$\leftrightarrow det(A - \lambda \phi^{-1} \wedge \phi)$$

$$\leftrightarrow det(N^{-1} \phi A \phi^{-1} N^{-1} - \lambda I) = 0$$

Equation 2 is the characteristic equation of the standard Eigenvalue problem. This implies that there exists a v such that

$$Mv = 0 where M = N^{-1} \phi A \phi^{-1} N^{-1} - \lambda I$$

$$(N^{-1} \phi A \phi^{-1} N^{-1} - \lambda I)v = 0$$

$$\leftrightarrow N^{-1} \phi A \phi^{-1} N^{-1} v = \lambda v$$

$$\leftrightarrow A \phi^{-1} N^{-1} v = \lambda \phi^{-1} N v ...(3)$$

Let m = $\phi^{-1}N^{-1}v$, substituting this value in 3 we obtain:

$$Am = \lambda\phi^{-1}N(N\phi m)$$

$$\leftrightarrow Am = \lambda(\phi^{-1}\phi)m)$$

$$\leftrightarrow Am = \lambda Bm...(4)$$

Thus we obtain back the standard matrix pencil equation which proves that $m = \phi^{-1}N^{-1}v$ are the eigenvectors of the Matrix Pencil.

VI. RESULTS

Drawing conclusions from the training and evaluation results of our neural network model, the following key observations emerge: The model attains an impressive average test accuracy of 98.1% across all possible combinations of digits, showcasing its robust performance in pairwise classification of handwritten digits. This noteworthy achievement is reflected in the model's consistently high accuracy and low loss for all pairs, underscoring its efficacy in accurately distinguishing between different digits.

TABLE I
COMPARING RESULTS OBTAINED USING DNN AND KNN

Class		Accuracy (%)	
C1	C2	KNN	DNN
1	0	64.77	99.89
2	0	57.85	98.88
3	0	64.97	98.82
4	0	55.61	99.41
5	0	59.61	98.82
6	0	58.51	98.98
7	0	60.90	99.47
8	0	51.89	99.01
9	0	51.03	98.73
2	1	89.85	98.85
3	1	76.60	98.96
4	1	69.77	99.83
5	1	75.77	99.22
6	1	87.48	99.66
7	1	76.28	99.60
8	1	84.73	97.92
3	2	73.41	96.74
4	2	74.63	98.51
5	2	73.70	98.34

These outcomes underscore the considerable potential of deep neural networks in addressing intricate classification tasks, particularly within the domain of handwritten digit recognition. The findings of this study significantly contribute to the comprehension and progression of machine learning techniques applied to digit classification. Such advancements hold practical implications across diverse fields, including optical character recognition and automated document processing. This research thus lays the groundwork for the continued development and application of sophisticated machine learning methodologies in real-world scenarios.

VII. CONCLUSION

Based on the outcomes derived from our implemented code, it is evident that deep neural networks (DNNs) serve as highly effective tools for pairwise classification of handwritten digits using the MNIST dataset. A comprehensive analysis, as presented in the comparison table, underscores the superior performance of the DNN model when juxtaposed with the KNN model. The DNN model exhibits a substantial increase in test accuracy and significantly lower loss.

The successful implementation of the DNN architecture within the code has proven its efficacy in accurately distinguishing between various pairs of digits. The methodology employed in the code encompasses crucial steps such as data loading, feature extraction, DNN model training, comprehensive model evaluation utilizing pertinent metrics, and thorough analysis of the obtained results. To enhance the DNN's performance and mitigate the risk of overfitting, common techniques including normalization, feature extraction, and dropout were judiciously applied.

The results gleaned from the DNN evaluation unequivocally demonstrate the success of our approach, achieving high accuracy and performance in the context of pairwise classification tasks involving handwritten digits. The code's capability to discern between different pairs of digits is underscored by its high precision and recall values, affirming its effectiveness in the accurate classification of handwritten digits.

VIII. FUTURE SCOPE

The proposed binary classification framework can be extended to handle multi-class classification by adjusting the existing architecture to accommodate multiple output nodes for distinct classes. Techniques like one-hot encoding for label representation are employed. The model's training process is also modified to optimize for simultaneous classification of multiple classes.

REFERENCES

[1] S. Bhagat and S. D. Joshi, "Quantification of differential information using matrix pencil and its applications," *Circuits, Systems, and Signal Processing*, vol. 42, no. 4, pp. 2169–2192, 2023.

[2] J. Chen, X. Li, M. A. Mohamed, and T. Jin, "An adaptive matrix pencil algorithm based-wavelet soft-threshold denoising for analysis of low frequency oscillation in power systems," *IEEE access*, vol. 8, pp. 7244–7255, 2020.

[3] S. Chamaani, A. Akbarpour, M. Helbig, and J. Sachs, "Matrix pencil method for vital sign detection from signals acquired by microwave sensors," *Sensors*, vol. 21, no. 17, p. 5735, 2021.

[4] R. Adve, "Target identification using matrix pencil," *Final report, University of Toronto (Canada)*, 2017.

[5] S. Bhagat and S. Joshi, "Quantification of differential information using matrix pencil," *arXiv preprint arXiv:2002.02116*, 2020.

[6] B. Gope, S. Pande, N. Karale, S. Dharmale, and P. Umekar, "Handwritten digits identification using mnist database via machine learning models," in *IOP conference series: materials science and engineering*, vol. 1022, p. 012108, IOP Publishing, 2021.

[7] J. Liu, R. Li, and C. Sun, "Co-correcting: noise-tolerant medical image classification via mutual label correction," *IEEE Transactions on Medical Imaging*, vol. 40, no. 12, pp. 3580–3592, 2021.

[8] P. Goel and A. Ganatra, "Handwritten gujarati numerals classification based on deep convolution neural networks using transfer learning scenarios," *IEEE Access*, vol. 11, pp. 20202–20215, 2023.