# Project Report for CS591 Programming Project 3(Final Project)

Team Contributors:

1. Mohammed Nurul Abedin Ashraf (S02036740) (Undergraduate)
2. Habibur Rahman (Undergraduate)

## Abstract

This report describes the implementation of Stochastic Gradient Descent (SGD) and Gradient Descent (GD) optimizers for curve fitting, as well as the development of a neural network to classify handwritten digits from the MNIST dataset. The project demonstrates an understanding of machine learning principles, optimization techniques, and neural network design while tackling practical computational challenges.

## Problem Description

The project comprises two tasks:  
  
1. Task 1: Stochastic Gradient Descent and Gradient Descent for Curve Fitting  
 - Implement GD and SGD optimizers to fit a curve to 100 data points using a polynomial model with at least three features. Experimentation with additional features and learning rates was conducted.  
  
2. Task 2: Neural Network for Handwritten Digit Classification  
 - Build and train a neural network to classify handwritten digits from the MNIST dataset. The model is evaluated based on its accuracy, with a focus on simplicity and performance.

## Algorithm

Task 1: GD and SGD  
- GD and SGD algorithms were implemented to minimize the Mean Squared Error (MSE) between predicted and actual values.  
- Models with three features (Y = aX^2 + bX + c) and additional features (up to 5) were used.  
- Learning rate experiments (0.005, 0.01, 0.02) were conducted.  
- Feature weights were updated iteratively, and their evolution was visualized.  
  
Task 2: Neural Network  
- A neural network was designed with the following layers:  
 - Flatten layer: Converts 28x28 images into a 1D vector.  
 - Dense layers: Two hidden layers with 128 and 64 neurons using ReLU activation.  
 - Output layer: 10 neurons with softmax activation.  
- The model was trained with the Adam optimizer and sparse categorical crossentropy loss.

## Implementation Details

1. Task 1 Implementation:  
- Gradient Descent:  
 - Iteratively updated weights by computing gradients of the loss function w.r.t. features.  
- Stochastic Gradient Descent:  
 - Performed weight updates for each data point to ensure faster convergence.  
- Experiments:  
 - Used models with 3, 4, and 5 features.  
 - Tested learning rates of 0.005, 0.01, and 0.02.  
- Results:  
 - Visualized the fitting curve for every epoch.  
 - Plotted the convergence of feature weights and loss history.  
  
2. Task 2 Implementation:  
- Dataset: MNIST handwritten digit dataset, normalized to enhance convergence.  
- Model Training:  
 - Trained for 10 epochs with a batch size of 32.  
 - Used a validation split of 20% to monitor performance.  
- Visualization:  
 - Displayed training and validation accuracy over epochs.  
 - Sample test images were shown with predicted labels.

## Results and Analysis

1. Task 1 Results:  
- Models with higher numbers of features (4 and 5) achieved better curve fitting at the cost of increased complexity.  
- Lower learning rates (0.005) required more epochs for convergence, whereas higher rates (0.02) showed instability.  
- Visualization demonstrated effective convergence with the chosen configurations.  
  
2. Task 2 Results:  
- Achieved ~97% test accuracy on the MNIST dataset.  
- Validation accuracy trends closely matched training accuracy, indicating low overfitting.  
- The network effectively captured patterns in handwritten digits.

## Strengths and Limitations

1. Strengths:  
- The implementation of GD and SGD showed clear visualization and adaptability to different configurations.  
- The neural network achieved high accuracy with a simple architecture.  
- Efficient training due to proper normalization and hyperparameter choices.  
  
2. Limitations:  
- The GD/SGD implementation was computationally intensive for larger datasets.  
- The neural network lacked advanced techniques like convolutional layers, which could further improve accuracy on more complex datasets.  
- Results were specific to MNIST and may not generalize to other tasks without modifications.

## Conclusions

This project demonstrated practical applications of machine learning techniques. GD and SGD were successfully applied for curve fitting with insightful experiments, while the neural network achieved impressive performance on digit classification. These results showcase the effectiveness of optimization methods and neural networks in solving real-world problems.

## References

1. LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-Based Learning Applied to Document Recognition. Proceedings of the IEEE.  
2. Kingma, D. P., & Ba, J. (2015). Adam: A Method for Stochastic Optimization. arXiv preprint arXiv:1412.6980.