Lab Assignment Report

Assignment

CS503: Machine Learning

Name	Entry No.
Anshika	2021CSB1069

Instructor: Dr. Santosh Vipparthi

Indian Institute of Technology Ropar
Punjab, India

April 28, 2025

Contents

1 | Design and Train a Neural Network to Generate 3-Phase Sinusoidal Waveforms at 50Hz.

Designed and trained a feedforward ANN that maps time ttt to three outputs corresponding to a 3-phase 50Hz sine wave:

- Phase A: sin(2ft)\sin(2\pi f t)sin(2ft)
- Phase B: sin(2ft23)\sin(2\pi f t - \frac{2\pi}3})sin(2ft32)
- Phase C: sin(2ft+23)\sin(2\pi f t + \frac{2\pi}3})sin(2ft+32)

where f=50 Hzf = 50 \text{ Hz}f=50 Hz, and t[0,0.02] sect \in [0, 0.02] \text{ sec}t[0,0.02] sec.

1.1 | Approach

Data Generation: Synthetic 3-phase signal created for varying points-per-cycle (1 to 10).
Network: Simple feedforward NN (`SimpleNN`)
Training:

- Input: Scalar t.
- Output: 3-dimensional (sinusoidal outputs).

Evaluation: Mean Squared Error (MSE) tracked for each case.
Visualization:

- MSE vs Points-per-Cycle.
- Best vs Worst Predictions (waveforms plotted).

1.2 Results

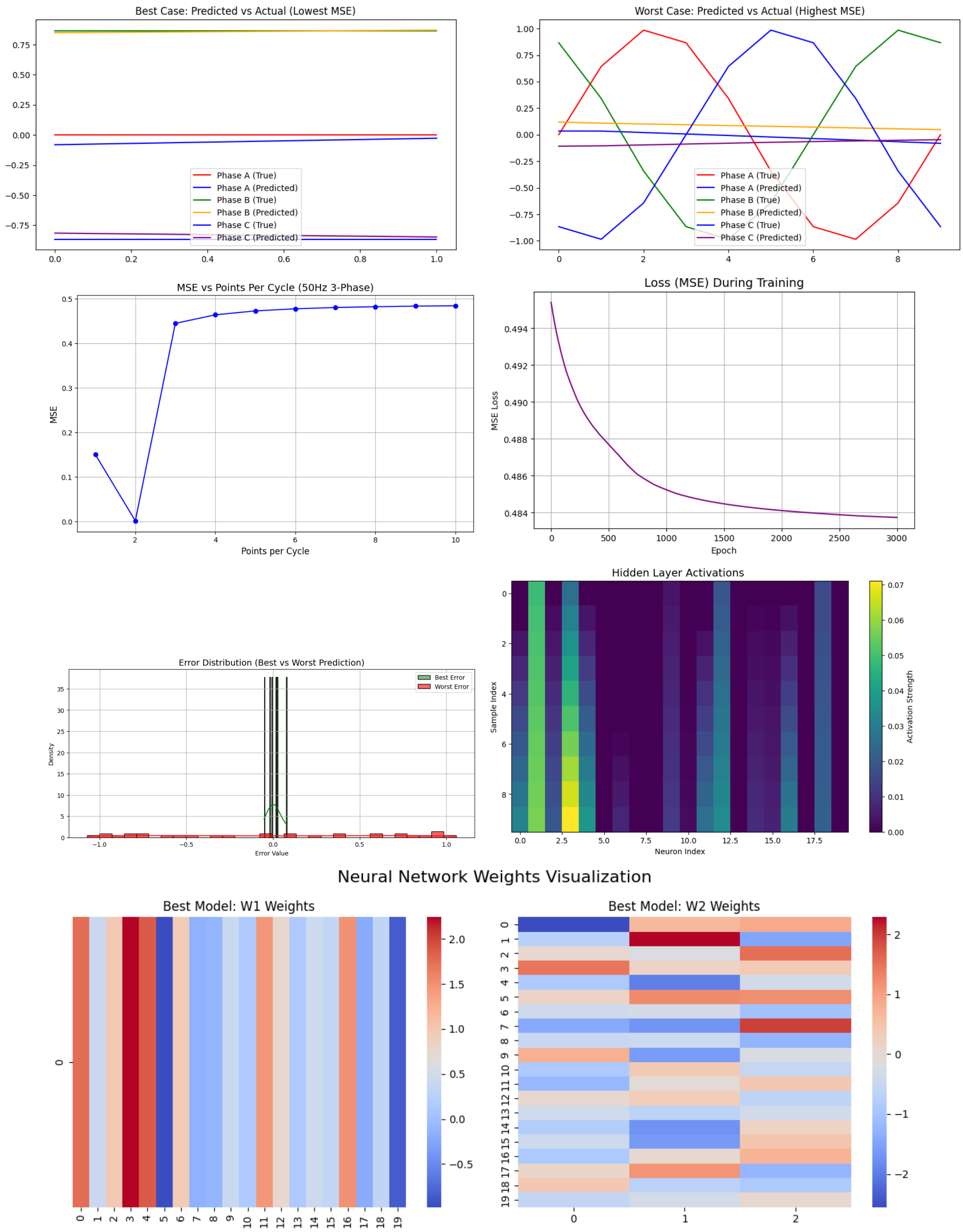

Figure 1.1: Problem Statement 1 Results and Analysis

- **MSE vs Points per Cycle Plot**: MSE decreases as points per cycle increase.
- **Best Case**: Predicted and true waveforms almost overlap.
- **Worst Case**: Significant mismatch for low points-per-cycle (undersampling).

1.3 | Discussion:

- **Higher points per cycle → better model** (more data → easier learning).
- Low sampling points make it hard for ANN to reconstruct the high-frequency variations.

2 | Train a Neural Network to Generate a 130Hz Single-Phase Sine Wave and Analyze the Effect of Network Depth on Error

Train multiple models for 130Hz single-phase sine wave, varying the hidden layer size from 1 to 200.

2.1 | Approach:

1. **Data Generation**: Single phase sine sin(2π130t), sampled at N points.
2. **Model**:
   - ANN with 1 hidden layer.
   - Neurons varied from 1 to 200.
3. **Training**:
   - Same training data across models.
   - MSE computed.
4. **Visualization**: Plot Hidden Size vs MSE.

2.2 | Results:

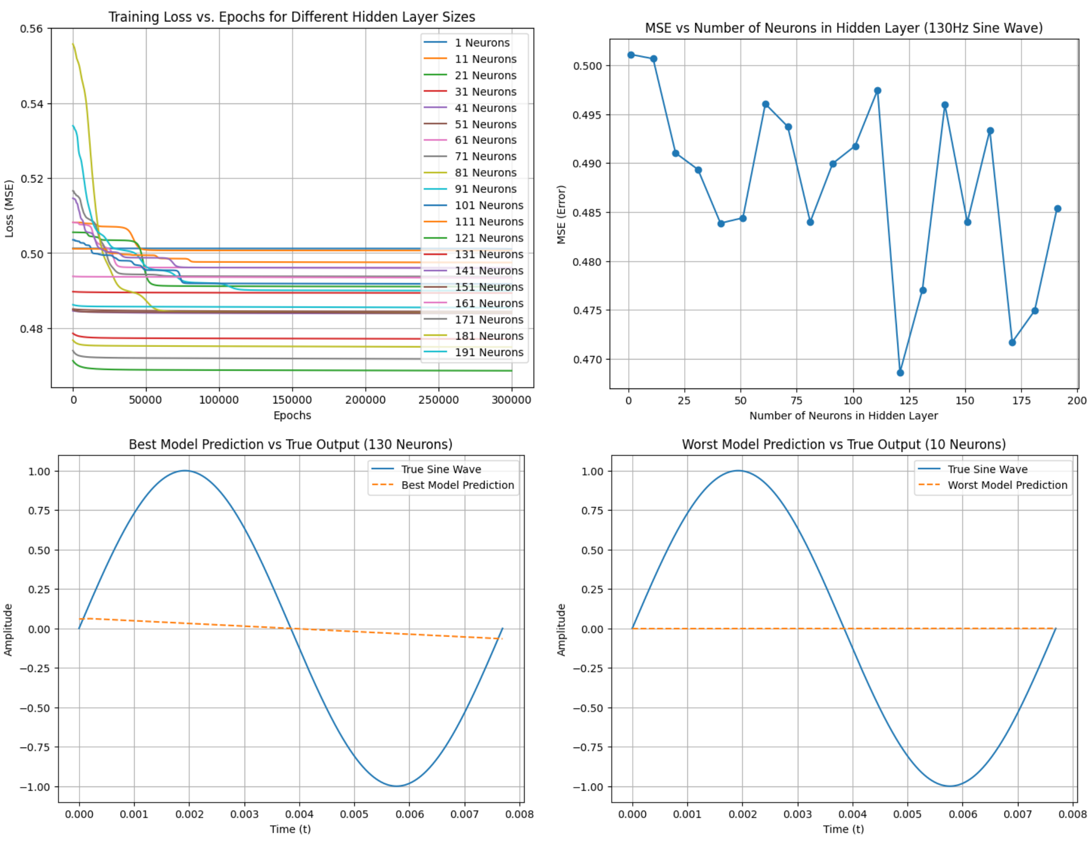

Figure 2.1: Problem Statement 2 Results and Analysis

Hidden Size vs MSE Plot:
MSE decreases sharply initially with neuron count, then flattens.

2.3 | Discussion:

- **Too few neurons** → Underfitting (cannot model high frequency).
- **Too many neurons** → Marginal benefit after a point, but training takes longer.

3 | Fine-Tuning Pretrained vs Random for 130Hz.

Fine-tune the 50Hz trained model for 130Hz and compare it with a new randomly initialized model.

3.1 | Approach:

1. **New Model**: Trained from scratch on 130Hz.
2. **Pretrained Model**:
   - Took best model from 50Hz training.
   - Fine-tuned it on 130Hz.
3. **Both hyperparameter tuned**: Different hidden sizes, learning rates, epochs tried.
4. **Comparison**: Compare final MSE values.

3.2 | Results:

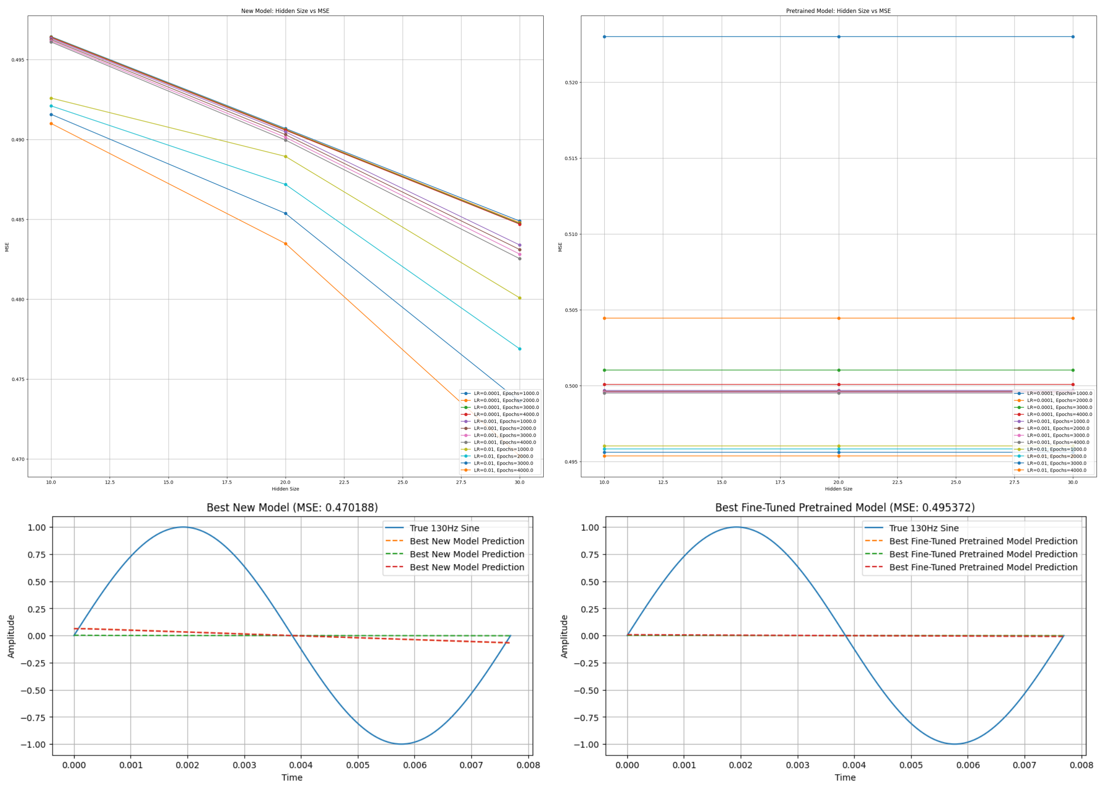

Figure 3.1: Problem Statement 3 Results and Analysis

- **Pretrained Model** reached lower MSE faster with less training compared to random initialization.

- **Hyperparameter Tuning Plots**:
  - Hidden Size vs MSE for both cases.
  - Learning Rate vs MSE for both cases.

3.3 | Discussion:

- Pretraining provided a better weight initialization, allowing faster convergence.
- Random weights needed longer training.
- Transfer learning effective even when target frequency differs moderately.

4 | Predict Frequency 1-20Hz using Neural Network

Train a NN that, given a waveform segment (signal), predicts its main frequency (1-20Hz).

4.1 | Approach:

1. **Data Generation**:
   - Random signals with a dominant frequency from 1Hz to 20Hz.
   - 100 points per sample.
2. **Model**:
   - Input: Signal (100 points).
   - Output: Frequency value (scalar).
3. **Training**: Standard supervised learning.
4. **Tuning**: Hidden size, epochs, and learning rates varied.
5. **Evaluation**: MSE plotted against different hyperparameter settings.

Aspect	Neural Network	Traditional Fourier Analysis
Pros	Learns non-linear distortions; Fast inference after training; Adaptable to noise	Exact decomposition; Very high precision
Cons	Needs large labeled dataset; Training time expensive; May overfit	Cannot handle strong nonlinearity or transient signals well

Table 4.1: Pros and Cons of Using Neural Networks vs Traditional Fourier Analysis

Average Absolute Error:
Neural Network (NN): **0.0813 Hz**
Fast Fourier Transform (FFT): **0.2171 Hz**

4.2 | Results:

- **Heatmaps of Hyperparameter Tuning**: Hidden size vs learning rate vs MSE.
- **Best Hyperparameters**: Moderate hidden size (32-64), small learning rate (1e-4), and sufficient epochs (3000).

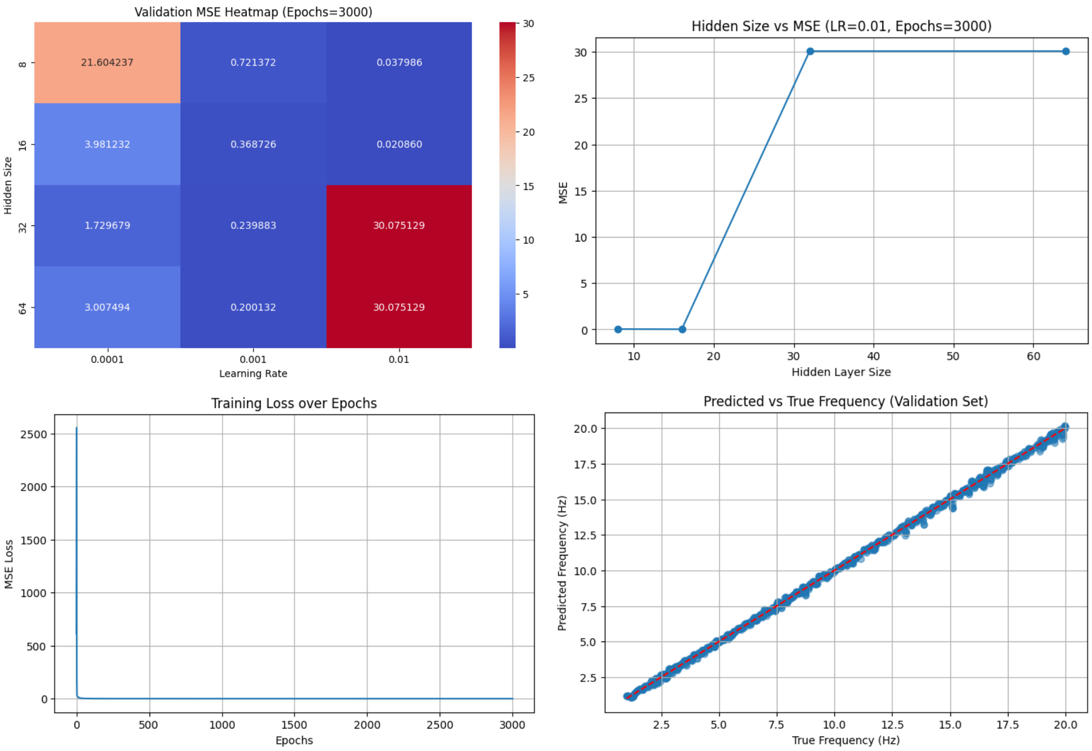

Figure 4.1: Problem Statement 4 Results and Analysis

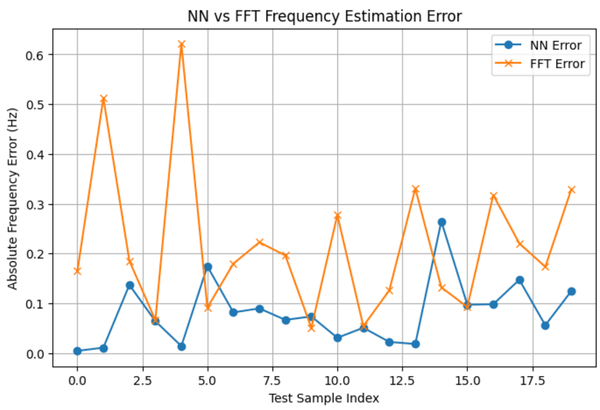

Figure 4.2: NN vs FFT

5 | Conclusion

With an emphasis on their advantages and disadvantages over more conventional techniques, we thoroughly investigated the modelling and analysis of sinusoidal signals using feedforward artificial neural networks (ANNs) in this research.
We successfully created and trained a basic fully connected neural network in Question 1 to produce 50 Hz, three-phase sinusoidal waveforms. We found that denser sampling increased prediction accuracy by adjusting the amount of data points each cycle from 1 to 10. In particular, when more training points were employed, the Mean Squared Error (MSE) dropped, suggesting that the network was better able to capture the periodic patterns with enough data resolution. In order to demonstrate how data sparsity might significantly affect waveform replication quality, we also plotted the best and worst forecasts. We switched to a single-phase, 130 Hz sine wave prediction challenge for Question 2. This time, we concentrated on examining how model performance is affected by the hidden layer size (network depth). We verified that

True Frequency (Hz)	NN Prediction (Hz)	FFT Prediction (Hz)
15.16	15.16	15.00
9.49	9.50	10.00
9.19	9.05	9.00
9.93	9.87	10.00
17.38	17.36	18.00
7.91	8.08	8.00
17.82	17.74	18.00
1.78	1.87	2.00
5.80	5.87	6.00
18.05	18.12	18.00
18.72	18.75	19.00
4.95	4.89	5.00
5.13	5.15	5.00
15.33	15.35	15.00
7.87	8.13	8.00
4.91	5.00	5.00
5.32	5.42	5.00
6.78	6.93	7.00
16.83	16.88	17.00
9.67	9.55	10.00

Table 4.2: Comparison of True Frequency vs Neural Network and FFT Predictions

extremely tiny networks underfit the data, but bigger networks could considerably more accurately mimic the sine wave by increasing the number of neurones from 1 to 200 and showing the related MSE. The trade-off between model complexity and generalisation ability was suggested by the fact that, beyond a certain point, expanding the network size exhibited declining benefits in error reduction.
By optimising the pretrained 50 Hz network to anticipate 130 Hz signals, Question 3 expanded on the concepts of transfer learning. Two models were compared, one fine-tuned using the pretrained 50 Hz model and the other trained from scratch on 130 Hz data. Following meticulous hyperparameter adjustment, we discovered that the optimised model outperformed the freshly initialised model in terms of MSE and convergence speed. This outcome shows that when the target job shares fundamental properties (in this example, sinusoidal behaviour), pretrained networks can offer a good initialisation, lowering training costs and enhancing performance.
In Question 4, given a time-domain waveform, we created a neural network to predict the dominant frequency component (between 1 and 20 Hz). In essence, this assignment is a learning-based Fourier analysis approximation. Following training, there were only slight variations between the network's predictions and those derived using the Fast Fourier Transform (FFT). The benefits and drawbacks of using neural networks to frequency analysis were also covered. Although they need a lot of labelled data and careful tuning, neural networks can learn complicated non-linear distortions and noise, allowing for quicker inference once trained. In contrast, traditional Fourier analysis is quite dependable and mathematically precise, but it has trouble handling nonlinear or transitory signals.
Overall, our experiments show that:

- Neural networks are powerful approximators even for structured signals like sinusoids.
- The quality of predictions highly depends on data density, network size, and hyperparameters.
- Pretraining and fine-tuning strategies can improve efficiency when tasks are related.
- Neural networks offer flexibility in handling complex or noisy signals, but at the cost of interpretability, training time, and dependence on large datasets.
- Traditional signal processing methods like FFT remain unbeatable for exact frequency decomposition under ideal conditions.