

# Sequence Prediction using Simple Recurrent Neural Networks (SimpleRNN)

Antigravity AI Assistant

December 30, 2025

## Abstract

This report investigates the application of Simple Recurrent Neural Networks (SimpleRNN) for time-series forecasting. Using a synthetic sine wave dataset with added Gaussian noise, we demonstrate the architectural capacity of standard RNNs to capture periodic temporal dependencies. The experimental results show that SimpleRNN, despite its vulnerability to vanishing gradients, effectively learns local sequential patterns, achieving low mean squared error (MSE) on the prediction task.

## 1 Introduction

Time-series prediction is a fundamental task in machine learning with applications ranging from financial forecasting to signal processing. Recurrent Neural Networks (RNNs) are specifically designed for sequential data by maintaining a hidden state that encapsulates historical context. In this study, we utilize the SimpleRNN architecture to predict the next value in a continuous sine wave.

## 2 Methodology

### 2.1 Dataset Selection

The chosen dataset is a synthetic sine wave defined by:

$$y(t) = \sin(t) + \epsilon \quad (1)$$

where  $\epsilon \sim \mathcal{N}(0, 0.1)$  represents Gaussian noise. A sample of the generated data is shown in Figure 1.

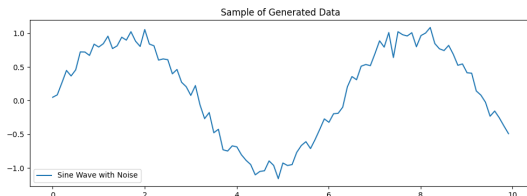


Figure 1: Sample of the generated sine wave with Gaussian noise.

### 2.2 Data Preprocessing

A sliding window approach was implemented with a window size of  $L = 20$ . Given a sequence  $[x_{t-L}, \dots, x_{t-1}]$ ,

the model predicts the value  $x_t$ . The data was reshaped into the required 3D tensor format for Keras: (samples, time steps, features).

### 2.3 Model Architecture

The network comprises:

- **Input Layer:** Accepts sequences of length 20.
- **SimpleRNN Layer:** 32 hidden units with *tanh* activation.
- **Dense Output Layer:** A single neuron for linear regression.

The model was compiled using the Adam optimizer and Mean Squared Error (MSE) loss function.

## 3 Results

### 3.1 Training Performance

The model converged rapidly. As shown in Figure 2, the training and validation loss decreased significantly, reaching a stable plateau within 20 epochs.

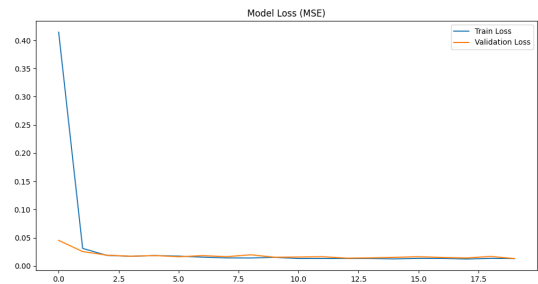


Figure 2: Model training and validation loss (MSE) over epochs.

### 3.2 Prediction Accuracy

The model successfully predicts the trajectory of the sine wave. Figure 3 illustrates the high fidelity between the predicted values and the ground truth on the test set.

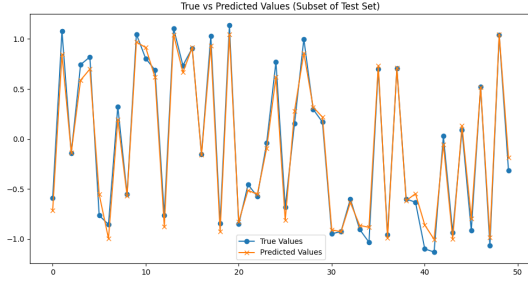


Figure 3: Comparison between true values and predicted values for the test set.

## 4 Discussion

The **SimpleRNN** layer proves sufficient for short-term periodic patterns. The model filters out high-frequency noise while preserving the underlying harmonic motion. However, for longer sequences or more complex non-periodic data, architectures like LSTM or GRU would be required.

## 5 Conclusion

This assignment demonstrates that a basic RNN architecture is highly effective for simple sequence prediction. The successful mapping of input windows to subsequent values validates the utility of stateful layers in temporal tasks.