Optimal Artificial Neural Network Depth For Noisy Signal Forecasting

Andrew Raine 2024 (andrewlr@seas.upenn.edu)
Dr. Jianbo Shi (jshi@seas.upenn.edu)
CURF

Abstract

- The goal of this project was to determine whether there exists an optimal depth to a particular artificial neural network architecture for noisy signal forecasting.
- For several datasets, each with a set amount of noise, we examined the performance of similar neural networks with varying depths, while controlling for training parameters (learning rate and training epochs).
- We generally found that shallower networks outperformed deeper ones for this specific architecture and task.
- Applications include any noisy signal forecasting task (e.g. electricity demand forecasting).

Signal Generation

- We generated ten sinusoidal waves by sampling their amplitude, frequency, and shift from uniform distributions and then combined them to form a single signal.

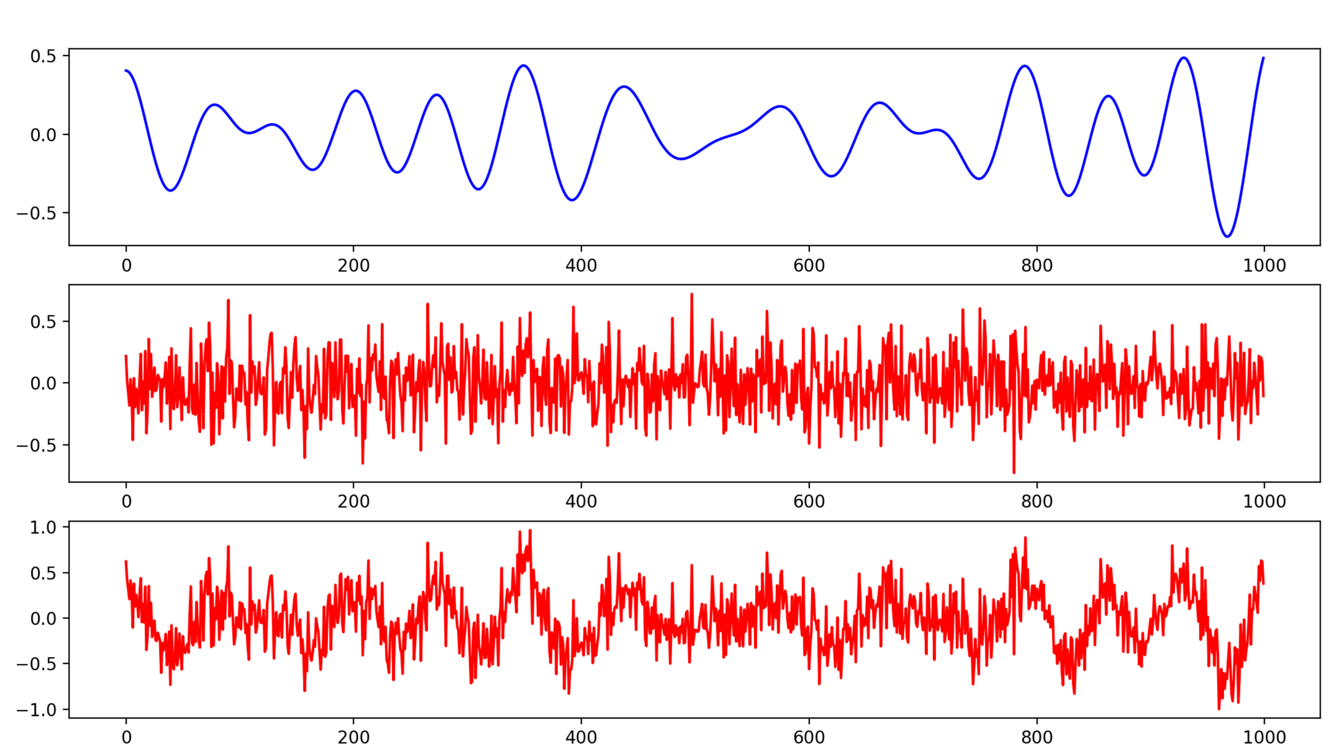
$$S_{clean} = \sum_{i=1}^{10} a\,sin(fX + s) \qquad a \sim U(0.2, 1.0),\ f \sim U(0.2, 1.0),\ s \sim U(0.2, 1.0),\ X \in \mathbb{R}$$

- Then we generated gaussian noise with the same size as the signal, $|X|$, and added it to the generated signal above. Each noise vector is scaled by the coefficient alpha such that when added to the clean signal, the result achieves a predetermined signal to noise ratio.

$$N \sim \mathcal{N}(0, 1.0) \qquad S_{noisy} = S_{clean} + \alpha N \qquad \alpha = \sqrt{\sum_{i=1}^{1000} \frac{{S_{clean_i}}^2}{SNR \times {N_i}^2}} \qquad SNR \in \{0.25, 0.5, 1, 2, 3, 4\}$$

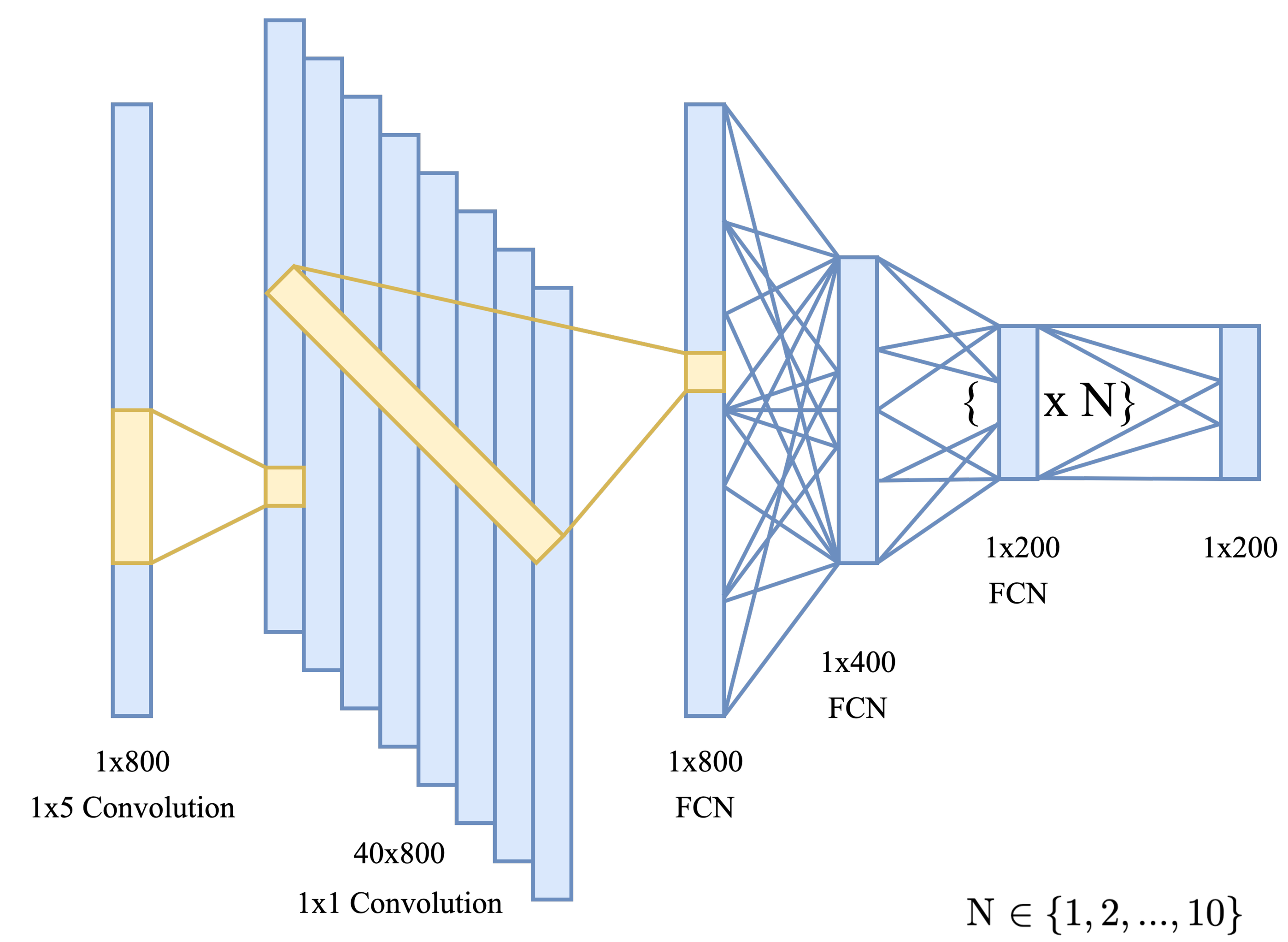
- For each of the six signal to noise ratios, we generated 50,000 sample signals. These are the datasets that we used to train each model architecture.

Signal Generation Example



Network Architecture

- Each model shared a base feature extraction network, which is composed of [1D CNN with 40 channels and a 1x5 kernel, 1D CNN with 1 channel and a 1x1 kernel, FCN that reduces the vector size from 800 to 400 to 200 (with ReLUs in between layers)].



Procedure

1) Generated 50k noisy signals for each signal to noise ratio, scaling each to [-1.0, 1.0] for regularity. Split each dataset into a 70-30 split for training and test sets.

2) For each SNR dataset:

   Instantiated ten neural networks with increasing depth, from 1 to 10.

   Trained each network for three epochs using the Adam gradient descent algorithm, with a learning rate of 3e-4[1], minimizing the MSE error between forecasted signal and the underlying clean signal.

   Each model's input is the first 800 timesteps of a signal, with the goal of forecasting the subsequent 200 timesteps.

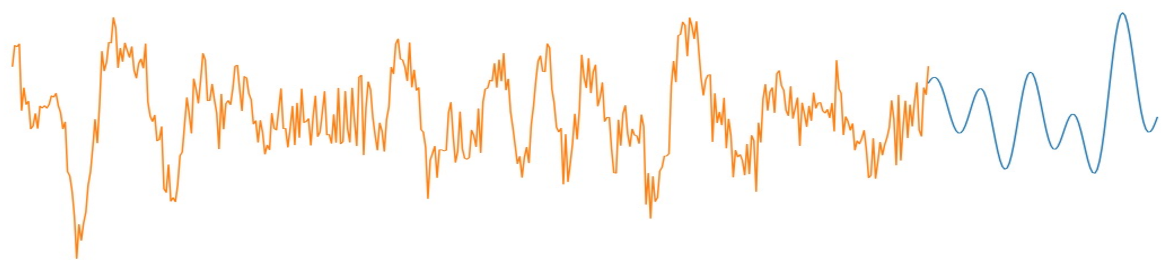
   Repeated step 2 three times.

3) Averaged final loss values among each group of three trials, plotted the results and observed the optimal depth of network.

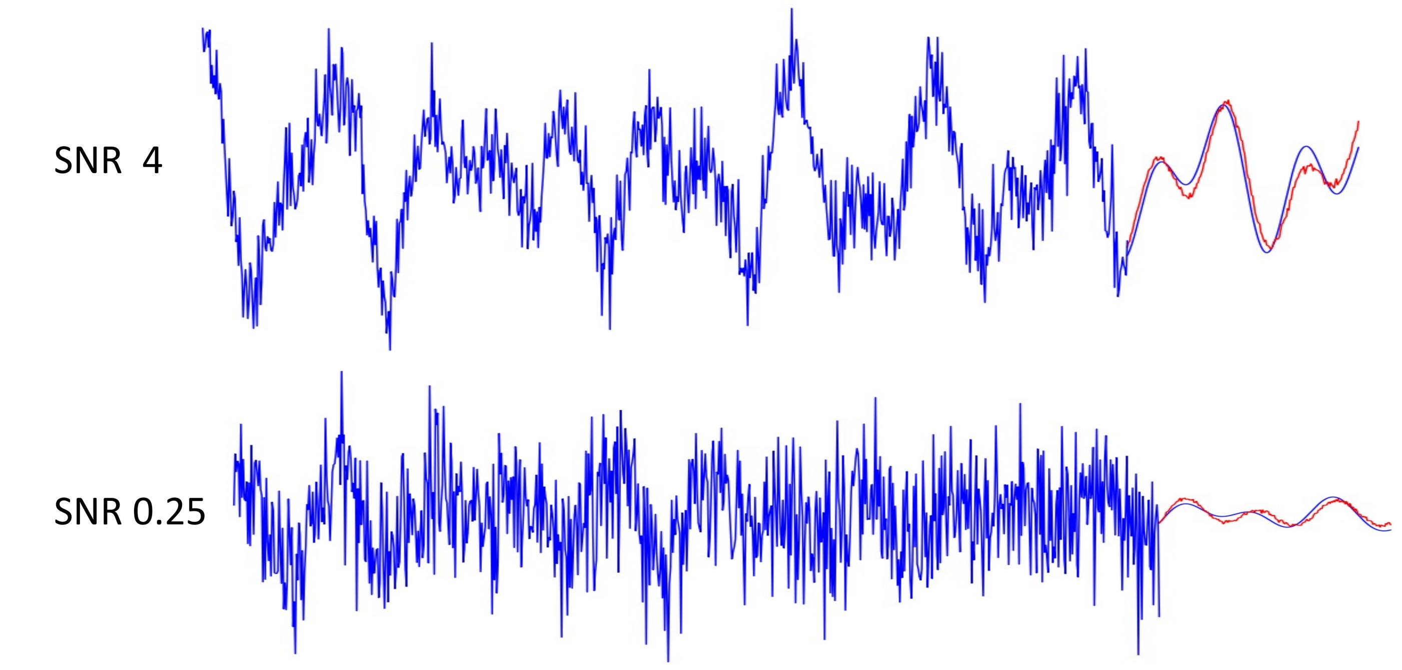
[1] These two hyperparameters in addition to the CNN channel depth were experimentally determined to ensure convergence in all models compared.

Example Model Forecasts

- The goal of each model is to forecast the last 200 timesteps of a signal given the preceding 800 timesteps of the signal's noisy counterpart:

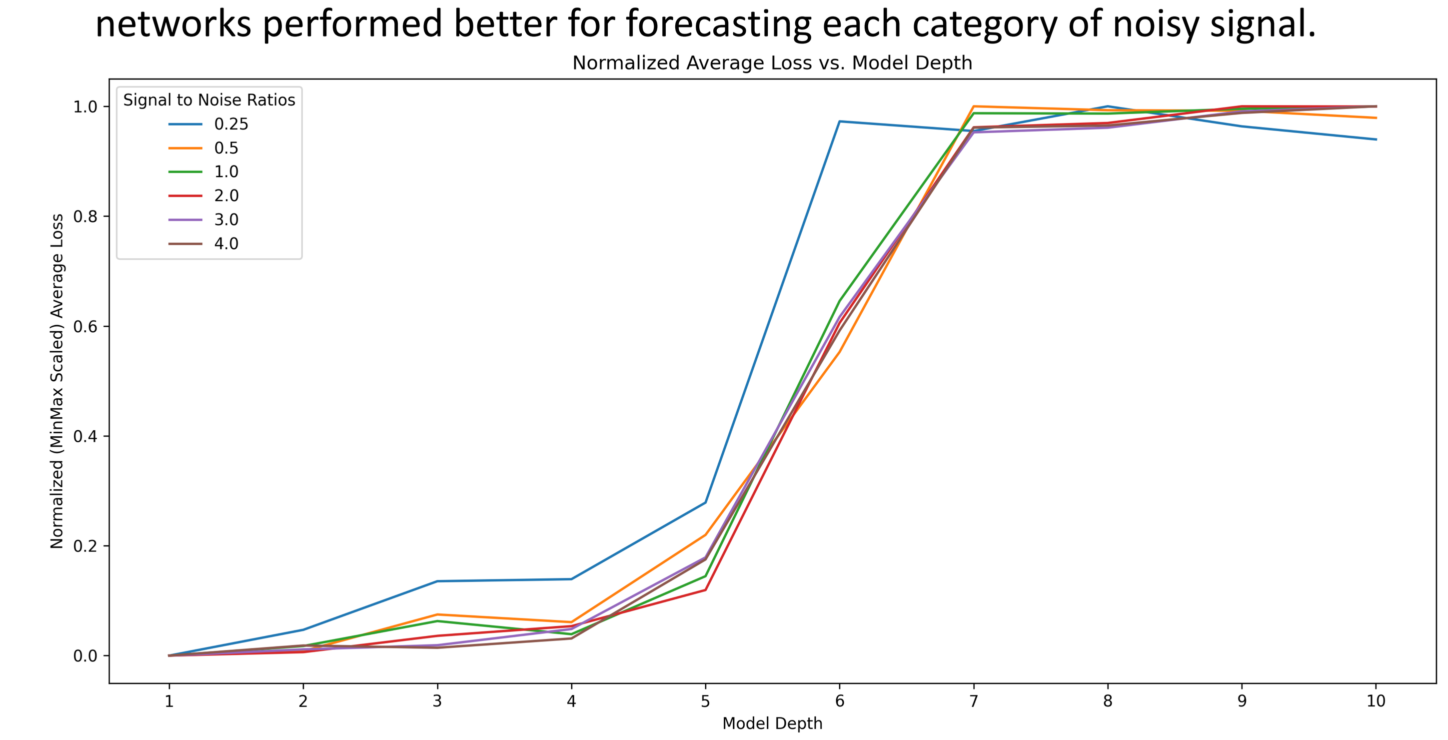


- The model inputs, the base clean signal, and the forecast:



Results & Implication

- Shallow neural networks with depths of up to 5 layers performed generally well, while any deeper resulted in non-optimal loss values. In this study, shallower networks performed better for forecasting each category of noisy signal.



- Practitioners using similar architectures may use this case study to guide their early model development work. Starting the network prototyping stage with shallower models may potentially reduce training, tuning, and compute resources.

Acknowledgements

I would like to thank Dr. Jianbo Shi for his unwavering support and mentorship, as well as the Center for Undergraduate Research and Fellowships.