Eric Gagliano (egagli)
Inferring Structure from Complex Systems

Lab 6

Github link: https://github.com/egagli/amath563/blob/main/labs/6/Lab6.ipynb

1. Describe how you generated your data for the Cosine reconstruction task. How many data points did you generate? Was this sufficient to cover the possible signals the network would need to see? How did you generate a test set? Did you use the same distribution as the training data, or did you focus on a particular range of amplitudes?

I generated the data for the cosine reconstruction task using the following code:

```
amp = np.random.rand()+0.5
shift = 2*np.pi*np.random.rand()
data = amp*np.cos(np.linspace(0,10,100)+shift)
```

This allows the amplitude to vary from (0.5,1.5) and the shift to be (0,2pi). I generated 10,000 data points. This should have been sufficient to cover all combinations of amplitude and shift that the model might see. Assuming true randomness, that's about 1600 points per square unit in the amplitude / phase shift domain. To create the test set, I used the same range of possible amplitudes and shifts (I did this so the testing and training domain distributions are as similar as possible) but with 10 times the length. The longer sequences allow for more distant predictions.

2. For your network, what did you use for your:

A. RNN Hidden Size

RNN with size 128.

B. Nonlinearity

I used ReLU nonlineaity.

C. Other Network Layers (FC, dropout, etc.)

4 RNN layers (overkill, but wanted to see what it looked like), dropout = 0.5.

D. Training Epochs

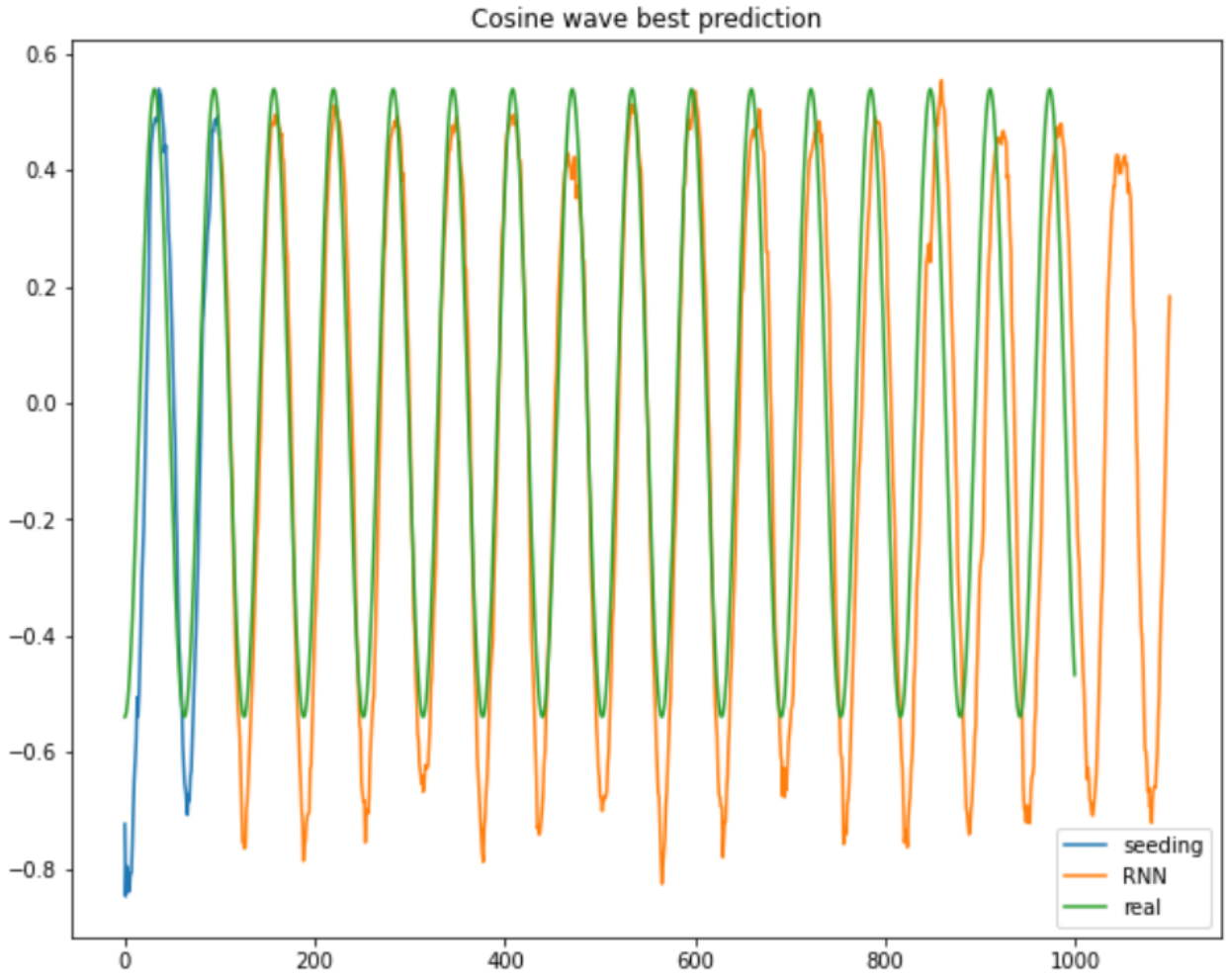
10,000 iterations

E. Learning Rate

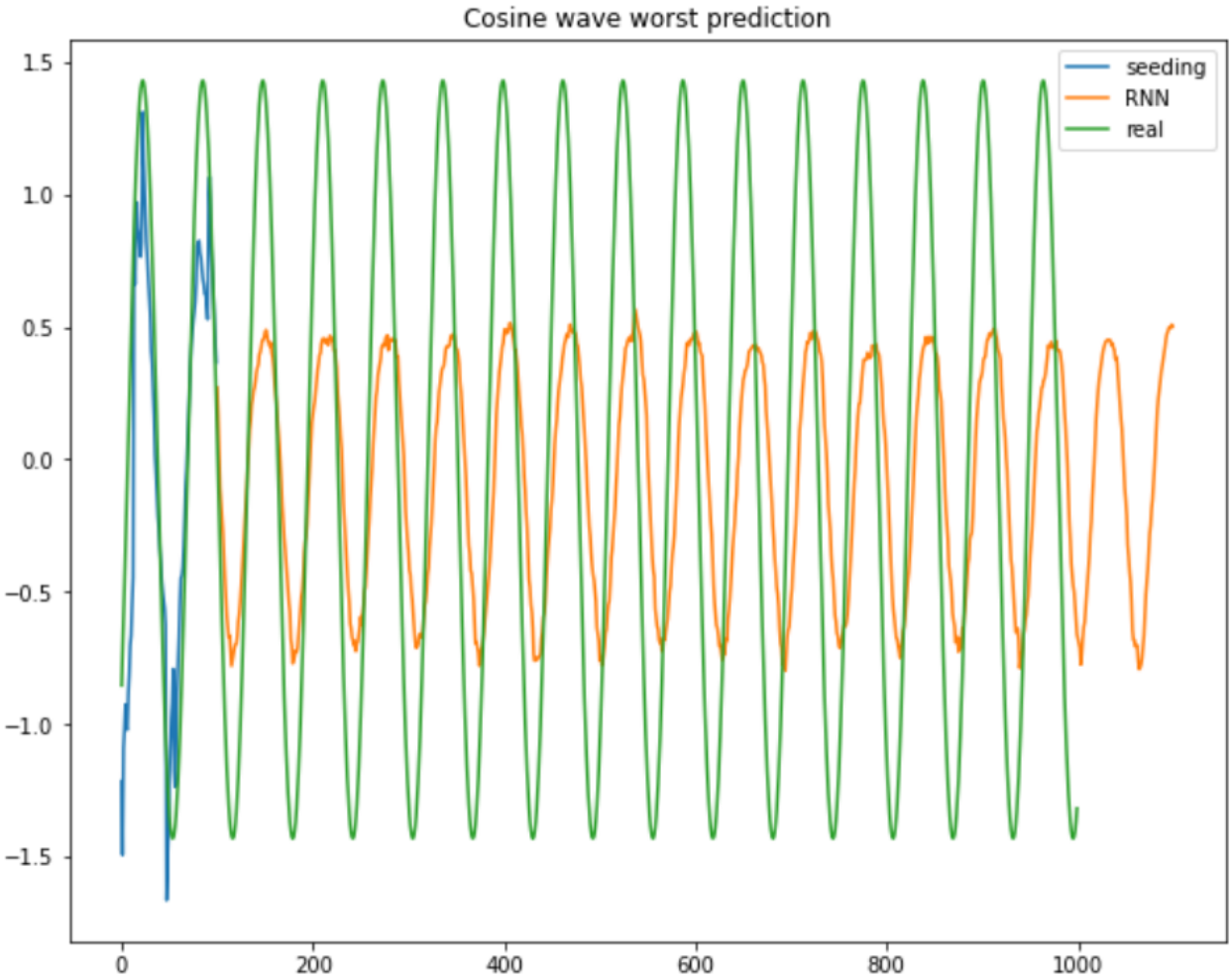
I used Adam with a learning rate of 0.001.

3. Please plot at least one example each of signals in your test which were accurately and inaccurately reconstructed. You can use the MSE loss to determine the best (low-error) and worst (high-error) samples. From those, plot the true signal (the input) and the signal your network predicted. In the comments, please describe what you notice about what your network is good at capturing and what it struggles with. Try explaining why this might be.

Best prediction (MSE=0.0103):



Worst prediction (MSE=1.1687):



My network seems to be really good at predicting period and shift, but bad at predicting amplitude. Also, there seems to be a bias towards more negative values. I think the strength of period prediction makes sense because we never varied the period in the training data. Also, the shift prediction is good because it is mathematically just a simple translation. I think the amplitude is bad because of the limited training data. Strangely enough, even in the best predictions the peak amplitudes predicted by the RNN seem to oscillate in time. One thing I wonder is why we don't use truth data for the seeding? From examining multiple predictions, it seems the RNN accuracy is very sensitive to the accuracy of the seeding.