Time Series Prediction Task

This task involves studying and implementing neural networks to predict various time series.

Preview

Question 1: What is the advantage of weather data when compared to a house pricing dataset?

Architecture overview

Question 2: • **Recurrent Neural Network (RNN)** - describe the concept in general. Describe the architecture of Long Short Term Memory (LSTM) and the advantages it has over other implementations of RNNs.

Question 3: **Attention-based Networks** - describe how attention works (elaborate on self-attention and cross-attention). Draw the Transformer architecture and describe the advantages it has over RNNs.

Implementation

Question 4.1: Use PyTorch/Tensorflow to create or use an existing template for experiments training and testing.

Question 4.2: Create your time series data - create an Autoregressive (AR) process with time dependency. For example:

$$Y_t = \alpha Y_{t-1} + \beta Y_{t-4} + \gamma$$

$$\gamma = \mathcal{N}(0, \sigma^2)$$

Make sure to use valid coefficients to obtain stationarity.

Question 4.3: Create three different networks and compare their forecasting quality. You may need to pick a metric to evaluate the prediction. Each prediction is only one value into the future:

$$Y_{t+1} | \{Y_t, Y_{t-1} ...\}$$

1. **None time-dependent network** - create a simple neural network that only has as input the last time step (t).

2. **RNN** - create a version of RNN (you may use the LSTM module from the library you are using).
3. **Attention** - create a network based on attention. Implement the network by hand. You may use more than 1 step from the history of the process, i.e. {t, t-1, t-2, t-3, t-4, t-5}.

Question 4.4: Train the network on the training set, and evaluate it on the validation set. Test the network on the test set and create appropriate graphs representing the prediction versus the ground truth. Explain what we can infer from the results.