11310CS460200 Intro 2 ML Lab 6

Name	Date
110020007 施淙綸	2024/12/18

When predicting values using sine wave data, is there a performance difference between the model that only contains Dense layers and one that includes an RNN layer? Which performs better?

- Only `Dense` layers can also fit the sine wave very well, but the number of layers and parameters is higher than `RNN` one. In other word, model that contains `RNN` layer can use more shallow and less parameters to reach the same performance of only `Dense` model.

Have you tried stacking two consecutive RNN layers in the model? How would you configure the parameters for the second RNN layer if the first RNN layer is defined as RNN(1, 16)? Briefly explain your reasoning.

- Yes! I have tried to stack two to three `RNN` layers in my model. We can view the last timestamp hidden layer as the input of next `RNN` layer, then continue to train next `RNN` layer. That is, if first `RNN` layer is defined as `RNN(1, 16)`, the next layer can be `RNN(16, *)` where `*` represents arbitrary dimension. Similarly, we can stack more `RNN` layers like `RNN(1, 16) -> RNN(16, *) -> RNN(*, **) -> ...`, so and so on.

What would be the effects with the larger size of hidden units in RNN layer?

- More hidden units in `RNN` layer provide more dimensions and trainable parameters that can "capture" more complex hidden features in the data, while consuming more computation resources. Like parameters in `Dense`, if the dimensions set too high, the model would probably overfit the training data. Moreover, too large hidden units may result in gradient exploding or vanishing in `RNN` model. So find the most suitable size of hidden units rather than increasing it infinitely.