# Deep learning for Commodity Price Prediction

1. Knowing about the dataset (from Kaggle)

[https://www.kaggle.com/datasets/prasertk/historical-commodity-prices-from-20002022]

表格

描述已自动生成

1. Aim: predict the exact close price for every commodity (6) from dataset
   1. Predict the close price only using the close price before.
   2. Predict the close price using all data of previous day.

# Need to decide how many day’s data will be need in one prediction.

1. Visualize dataset.

图形用户界面, 图表

描述已自动生成 图形用户界面, 图表, 直方图

描述已自动生成

图形用户界面, 图表

描述已自动生成 图形用户界面

描述已自动生成

图形用户界面, 图表, 直方图

描述已自动生成 图形用户界面

描述已自动生成

As we can see, here are the close price and the volumn for each commodity. To solve the question in 2, ‘how to select features to train’, we seek the answear from these plots.

In my opinion, the volumn is not a good feature for close price prediction, since the relavativity is really not stable. It is hard to say that some performance is noise or not. For example, Gold have a greatest peak volumn than usual. Nickel’s volumn keep nearly zore and suddenly has its highest peak. Palladium and US Wheat both have a period of volumn which is much lower than usual. I not only can not say it is noise 100%(maybe it is real situation) but also don’t have a good technic to preprocess or clean it.

However, this conclusion is only based on my own thinking. We need to really implement the model to tell the truth.

1. Possible model
   1. Linear regression
      1. Different lookback window size
      2. Using only close price
      3. Using other features together (open, …, close)
   2. Decision tree (not suitable) data type not num.
   3. Random forest (not suitable) data type not num.
   4. Logistic regression (not suitable) bi-classify for direction.
   5. SVC from scikit-learn (not suitable) bi-classify.
   6. Neural network model
      1. 1
      2. 2
      3. 3
2. Plan:
   1. Import the dataset.
   2. Preprocess the data.
   3. Build the model.
   4. Train.
   5. Predict.
   6. Evaluate (MSE: Mean square Error)
3. Evaluation.
   1. Linear regression
      1. (Why the plot is not monotonous) Since random split the train and test dataset, it will break the different period performance (or rules) based on the real market. Apply lookback before data split. The advantage of doing this is to maximize the use of the dataset to improve the accuracy of the model. However, it increases the correlation in the dataset and train and test dataset may have intersection.
         1. Only using close price can reach the best MSE with lookback window is 75 days.
         2. 图表, 折线图

            描述已自动生成
      2. Using the data split without shuffle, it may be helpful to learn the period performance of real market.
         1. The best performance is both for smallest lookback. And train with all prices with lookback = 10 is best. It is monotonous basically.
         2. 图表, 折线图

            描述已自动生成
         3. That means, bigger lookback window size will not be helpful for training the linear regression model to predict the commodity price.
      3. I decide to plot the true price and prediction for test dataset to visualize the model performance.

图表, 直方图

描述已自动生成图表, 直方图, 散点图

描述已自动生成图表, 散点图

描述已自动生成

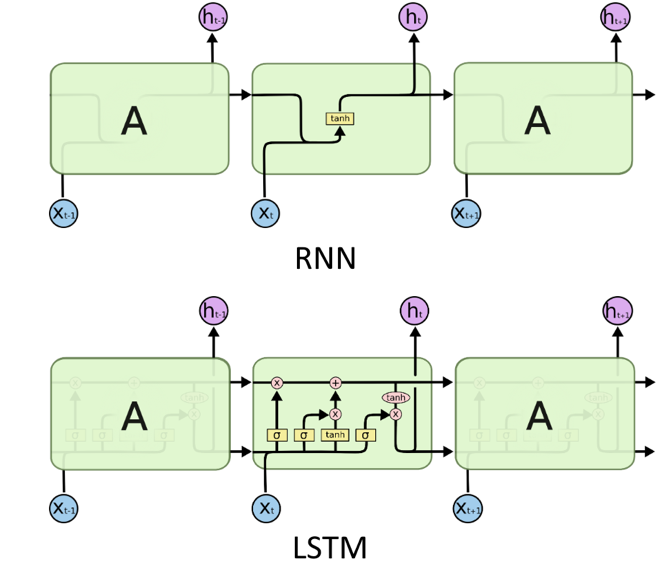
Figure1: model trained with lookback = 75 and random split data.

Figure2: model trained with lookback = 10 and unshuffled data

Figuer3: model from figuer1 but with unshuffled data.

I plot the figure3 to try to find that whether the model train without period information (shuffled data) can predict well on unshuffled dataset.

In conclusion, the model trained with shuffled data also can predict the unshuffled data well. It is hard to tell which model train with different dataset (different features) is better. However, the mean squared error can.

* 1. Neural network model
     1. 
     2. Build the LSTM model for different structure, different parameter combination (lookback, features, epoch, batch\_size) to find a model having best performance (lowest mean squared error) for this dataset.

图表, 折线图

描述已自动生成 图表, 折线图

描述已自动生成 图表, 折线图

描述已自动生成 图表, 折线图

描述已自动生成