HW4

110034009 蕭禮英

1. Experiment with different window sizes and steps.

組別	Window size	step	Train MSE	Val MSE	Test MSE
1	5	10	29.8438	0.6244	98.742035
2	10	15	186.0125	19.9305	322.0287
3	20	30	295.0580	99.4492	1358.834839

We observe that smaller window sizes and step sizes lead to better model performance. Group 1, with the smallest window and step size (5, 10), achieved the lowest MSEs across training, validation, and testing. A smaller window allows the model to focus on recent, short-term patterns, which are easier to predict. Additionally, a smaller step size results in more training samples, helping the model generalize better. As the window and step sizes increase, the model faces greater difficulty capturing long-term patterns and suffers from reduced data availability, leading to significantly higher errors.

2.

(i) Include 'Volume' as an additional input feature in your model.

Window size	step	Train MSE	Val MSE	Test MSE
5	10	878.6817	998.4980	1357.9923

In this experiment, we included 'Volume' as an additional input feature. However, the model's performance significantly worsened after incorporating 'Volume'.

Trading volume can be noisy and does not always directly correlate with price movements, further introducing irrelevant information. The increased input complexity may have required a larger model or more training data to fully capture the patterns.

(ii) Explore and report on the best combination of input features that yields the best MSE.

方法一：一組一組測試

組別	特徵組合	**Train MSE**	**Val MSE**	**Test MSE**
1	Open, High ,Low, Close	29.8438	0.6244	98.742035

2	Open, High ,Low, Close, Volume	878.6817	998.4980	1357.9923
3	Open, High ,Low, Close, OpenInt	31.7135	114.4678	147.7994
4	Open, High ,Low, Close, Volumn,OpenInt	971.1161	568.3157	1364.4885
5	Open, Close	36.4826	103.3180	148.2158

結論:經過不同特徵組合的實際測試後,發現 Open, High ,Low, Close 組合的 MSE 最小。

方法二：使用 Lasso Regression 篩選特徵

特徵	Open	High	Low	Close	Volume	OpenInt
權重	13.1637	9.1652	0.27705	7.9740	0	0

結論：使用 Lasso Regression 進行特徵篩選後，發現 Open, High ,Low, Close 的特徵最為重要，與實際測試結果相符。

3. Analyze the performance of the model with and without normalized inputs.

Feature: Open, High ,Low, Close / Window size:5 / Step:10

(i) With Normalize

Train MSE	**Val MSE**	**Test MSE**
29.8438	0.6244	98.742035

(ii) Without Normalize

Train MSE	**Val MSE**	**Test MSE**
31.5301	0.8806	218.0893

In this experiment, the model trained with normalized inputs achieved better performance than the model trained without normalization. Normalization reduced the test MSE from 218.09 to 98.74, demonstrating improved generalization ability. This is because normalization ensures that all input features have similar scales, preventing any single feature from dominating the learning process (Goodfellow et al., 2016). Without normalization, larger magnitude features can bias the optimization, making the model harder to train and prone to poor generalization.

4. **Why should the window size be less than the step size in Lab 4? Do you think**

this is correct?

I agree with it. If the window size is greater than the step size (e.g., window = 10, step = 1), the extracted sequences will overlap heavily, resulting in highly redundant training data. This increases the risk of overfitting, slows down training, and prevents the model from learning new information.
In contrast, if the window size is smaller than the step size, it ensures that each sequence is relatively independent. This increases the diversity of the training data and helps the model learn meaningful temporal patterns instead of simply memorizing details.
Reference: Goodfellow et al., 2016, *Deep Learning*.

5. **Describe one method for data augmentation specifically applicable to time-series data.**

One method for data augmentation specifically applicable to time-series data is **Time Warping**.
Time warping involves stretching or compressing portions of the time-series signal along the time axis.
This technique simulates variations in the speed of underlying processes without changing the overall pattern, thereby generating realistic new samples.
It helps improve model robustness by exposing it to a wider variety of temporal distortions commonly found in real-world time-series data.

Reference: Li, H. (2015). On-line and dynamic time warping for time series data mining. *International Journal of Machine Learning and Cybernetics*

6. **Discuss how to handle window size during inference in different model architectures:**

(i) Convolution-based models

During inference, convolutional models typically use a fixed window size determined during training. Because convolutions are local operations, the model expects inputs of a consistent size. If larger inputs are provided, global pooling layers or sliding windows can be used to aggregate features for prediction.

(ii)Recurrent-based models

Recurrent models, such as LSTM and GRU, are naturally flexible with sequence lengths. At inference, they can process sequences longer than the training window size by sequentially updating hidden states. However, longer sequences may require careful handling of hidden state initialization and memory.

(iii)Transformer-based models

Transformer models are capable of handling variable-length inputs. However, during inference, attention mechanisms scale quadratically with sequence length. Therefore, it is common to restrict the input window size to a reasonable limit or apply techniques like sliding windows or sparse attention to manage computational costs.