CE6023 Homework1 Report

學號：111502531　姓名：趙啟翔

1. (5%) 具體說明你的資料前處理方式，並說明造成了的差異以及造成差異可能原因。

1.1 資料整合

- 從指定的資料夾中讀取所有.csv 格式的檔案，並將它們合併為單一的資料框架 (DataFrame)
- 除了原始數據，還計算了數據的一些新特徵，包括即期點差和現鈔點差、點差差值以及不同買入和賣出的返回率（Return）

1.2 特徵工程

我共計算了即期點差、現鈔點差、點差差值、以及返回率。點差能有助於捕捉市場的賣出和買入價格之間的差異；點差差值有助於捕捉即期和現鈔之間的價格差異；返回率有助於捕捉市場的價格動向。

- 即期點差公式：

$$即期點差 = \frac{(即期賣出 - 即期買入) \times 100}{即期買入}$$

- 現鈔點差公式：

$$現鈔點差 = \frac{(現鈔賣出 - 現鈔買入) \times 100}{現鈔買入}$$

- 點差差值公式：

$$點差差值 = 即期點差 - 現鈔點差$$

- 返回率公式(Return)：

$$Return = 當前價格 - 前一天價格$$

1.3 資料正規化

了確保模型在不同的資料分布下也能表現良好，我選擇使用 Z-Score 進行資料正規化。

$Z = \frac{X-\mu}{\sigma}$ (X 為原始數據、μ 為平均值、σ 為標準差)

1.4 資料格式轉換

將 DataFrame 資料轉換為 Numpy 數組，再轉換為 PyTorch 的 Tensor 格式，以便進行深度學習。

1.5 訓練/驗證資料分割

我使用固定隨機種子來確保每次分割的一致性，並將資料切成 80%的訓練資料和 20%的驗證資料。

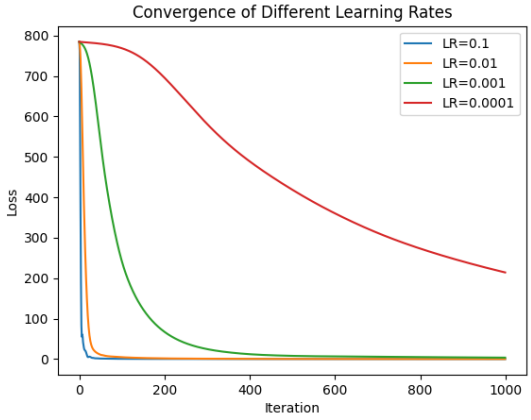
1.6 差異

- 特徵工程:
  通過計算新的特徵，如點差和 Return 值，可以提供更多的資訊給模型，有助於模型學習更深入的模式。
- 資料正規化:
  正規化可以使資料在不同的特徵間具有相似的尺度，有助於模型的收斂和避免梯度消失或梯度爆炸等問題。
- 訓練/驗證分割:
  通過將資料分割為訓練和驗證集，可以確保模型不僅僅是過擬合訓練資料，而是具有一般化的能力。

2. (5%) 使用四種不同的 Learning Rate 進行 Training（方法參數需一致），作圖並討論其收斂過程（橫軸為 Iteration 次數，縱軸為 Loss 的大小，四種 Learning Rate 的收斂線請以不同顏色呈現在一張圖裡做比較）。

2.1 設定

我選擇了四個不同的學習率進行測試。這些學習率分別是：0.1、0.01、0.001、和 0.0001。在所有的實驗中，我都使用了 Adam 優化器，並且都用 LSTM 模型。



2.2 學習率為 0.1

這個學習率使得 LSTM 模型收斂得非常快，但從圖中可以看出，它在初期出現了一些震盪。這可能是因為學習率較大，導致權重更新的步長過大，使得模型在訓練初期不夠穩定。

2.3 學習率為 0.01

這個學習率的收斂速度適中，且整體表現相對平穩。

2.4 學習率為 0.001

此學習率的收斂速度較慢，但它提供了一個非常平滑的收斂曲線，顯示出模型訓練過程中的高度穩定性。

2.5 學習率為 0.0001

這是最慢的學習率，從圖上可以看到它的收斂速度遠遠落後於其他三者。

2.6 結論

適當的學習率對訓練 LSTM 模型的效果會差很多。選擇太高的學習率可能會使模型訓練不穩定，而選擇太低的學習率則可能使訓練過程過於緩慢。在實際操作中，需要透過多次的實驗和調整，來找到最合適的學習率。

3. (5%) 比較取前 2 天和前 4 天的資料的情況下，於 Validation data 上預測的結果，並說明造成的可能原因。

3.1 預測的平穩性

取前 4 天的資料似乎提供了更平穩的預測。這可能是因為使用更長的時間序列可以捕捉到更多的資訊，從而使模型在預測時更加穩定。

3.2 預測的準確性

雖然不能直接從圖中看到實際的損失值，但如果一個模型的預測線更接近實際的目標線，那麼我們可以說該模型在該情境下具有更高的準確性。從您提供的圖像中，取前 4 天的資料似乎提供了更接近目標的預測。

3.3 造成的可能原因

- 更多的資訊

  使用更長的時間窗口意味著模型可以查看更多的歷史數據。這可能有助於捕捉更多的潛在模式，從而提高預測的準確性。

- 減少噪音的影響

  較長的時間窗口可能有助於平均或減少任何短期噪音的影響，從而提供更平穩和更一致的預測。

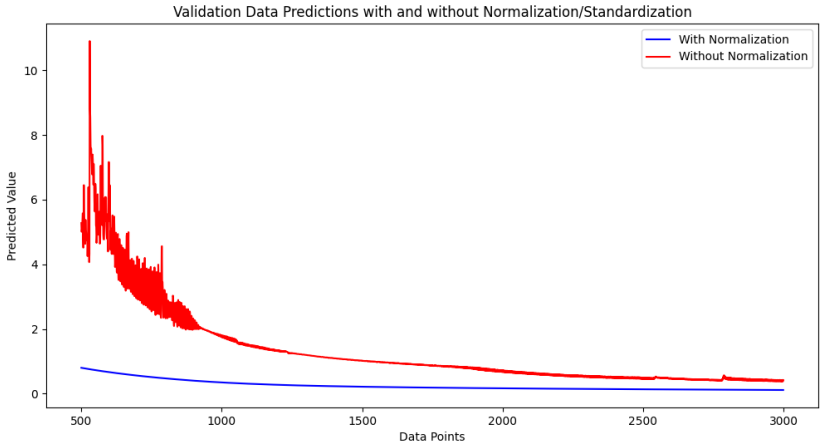
- 模型架構

  LSTM 和其他 RNN 模型是專為處理序列數據設計的。它們的內部結構允許它們記住過去的信息。因此，提供更多的過去信息可能有助於這些模型更好地理解數據中的模式。

3.4 結論

從提供的圖像中，我們可以看到取前 4 天的資料似乎在這種情境下提供了更好的預測結果。這可能是由於提供更多的歷史資訊和減少噪音的影響。

4. (5%) 比較資料在有無 Normalization 或 Standardization 的情況下，於 Validation data 上預測的結果，並說明造成的可能原因。



4.1 分析

- 經過 Normalization 處理的資料，在預測結果上似乎更加平滑，而沒有經過此種處理的資料則在某些地方出現突波。
- 在大多數情況下，經過 Normalization 處理的資料的預測結果似乎更接近實際值。

4.2 可能的原因

- 數據分佈

  Normalization 和 Standardization 調整了資料的分佈，使其有較小的標準差和 0 的均值（在 Standardization 的情況下）。這使得模型在訓練過程中能更容易地捕捉到資料中的模式，因為所有特徵都在相同的尺度上。
- 梯度優化

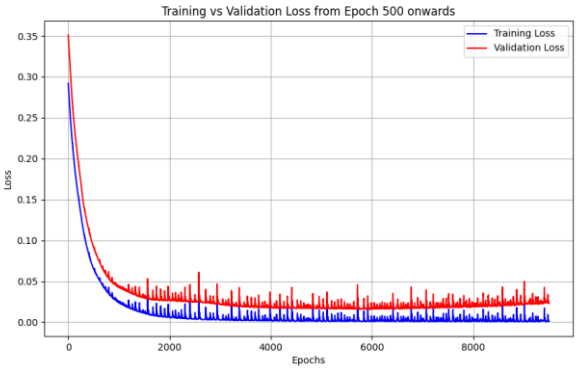
  當特徵在相同的尺度上時，模型在進行參數更新（如梯度下降）時，更新方向更加穩定。這有助於模型更快地收斂。
- 避免數值不穩定性

  未經處理的資料可能包含非常大或非常小的值，這可能會導致數值不穩定性，如梯度消失或梯度爆炸。Normalization 和 Standardization 減少了這種可能性。

4.3 結論

資料的 Normalization 或 Standardization 是模型訓練中的一個關鍵步驟，尤其是當使用深度學習模型時。從結果中可以看到，正確地縮放和中心化資料可以顯著提高模型的預測性能。

5. (5%) 請作圖並說明你的 Training Loss 和 Validation Loss 間的關係，並說明可能造成的原因。



5.1 圖表分析

- 下降趨勢

  在初期階段，Training Loss 和 Validation Loss 都呈現下降趨勢。這表示模型正在學習並適應數據集中的模式。

- Validation Loss 平穩

  在某一點後，Validation Loss 似乎開始變得相對平穩，而 Training Loss 則繼續下降。這可能是過度擬合（overfitting）的跡象，意味著模型在訓練數據上表現得越來越好，但在驗證數據上的表現卻沒有顯著改善。

- 驗證損失的波動

  驗證損失似乎有些許的波動，這可能是由於模型在訓練過程中的某些階段更能適應驗證數據，而在其他時期則不然。

資料的 Normalization 或 Standardization 是模型訓練中的一個關鍵步驟，尤其是當使用深度學習模型時。從結果中可以看到，正確地縮放和中心化資料可以顯著提高模型的預測性能。

5.2 可能的原因

- 過度擬合(Overfitting)

  當 Training Loss 持續下降，但 Validation Loss 停止改善或開始上升時，這通常意味著模型過度擬合。這意味著模型可能過於複雜，或者訓練時間過長，使其過於專注於訓練數據中的特定模式，而忽略了其應具有的泛化能力。

- 數據不均勻(Data Imbalance)

如果驗證數據集的分佈與訓練數據集不同，則可能會看到這種模式。例如，某些在訓練數據中出現的特定模式可能在驗證數據中並不常見。

- 學習率(Learning Rate)
  如果學習率設定不當，則模型可能會在最佳解決方案附近震盪，而不是收斂。這可能會導致驗證損失出現波動。

5.3 改進

- 提早停止 (Early Stopping)
  一種避免過度擬合的策略是提早停止。當您看到驗證損失不再改善時，可以提早終止訓練過程。
- 使用正則化 (Regularization)
  為模型添加 L1 或 L2 正則化可以幫助防止過度擬合。
- 調整學習率
  嘗試使用不同的學習率或使用學習率衰減策略。
- 數據增強 (Data Augmentation)
  這可以幫助模型更好地泛化到驗證數據。

6. (5%) 請說明你超越 Baseline 的 Model（最後選擇在 Kaggle 上提交的）是如何實作的（若你有額外實作其他 Model，也請分享是如何實作的）

6.1 Baseline Model

Baseline 模型是一個簡單的線性回歸模型，它由一個 nn.Linear 層組成，並不包含任何非線性激活函數

6.2 超越 Baseline 的實作方式

- 模型選擇
  我實作了多種模型供選擇，包括：LSTM (LSTM_Model)、雙向 LSTM (BiLSTM_Model)、加上 MLP 的 LSTM (LSTM_MLP_Model) 、加上 MLP 的雙向 LSTM (BiLSTM_MLP_Model)、有時間序的注意力機制模型 (TimeSeriesAttentionModel)、GRU 模型 (GRU_Model)。
  在這次的訓練中，表現成果最好的是單層的 LSTM 模型。
- 資料前處理
  利用前 3 天的資料作為輸入，預測目標值。
- 損失函數
  我選擇使用跟題目一樣地均方誤差 (MSE) 作為損失函數，這是一個常用於回歸問題的損失函數。
- 訓練策略

首先進行了 3000 次的 "Warming Up"，使用 SGD Optimizer，學習率設定為 0.01。再使用 Adam Optimizer 進行訓練，學習率設定為 0.001，並設定了早停策略以避免過度擬合。

- Early Stopping

  我設定了 Early Stopping，當驗證集上的損失不再改善時，模型訓練會提前終止。這可以幫助模型避免過度擬合並節省訓練時間。

6.3 結論

通過使用較 BaseLine Model 複雜的 LSTM 模型，結合適當的資料前處理、訓練策略以及早停策略，我成功地超越了簡單的 Baseline 模型。