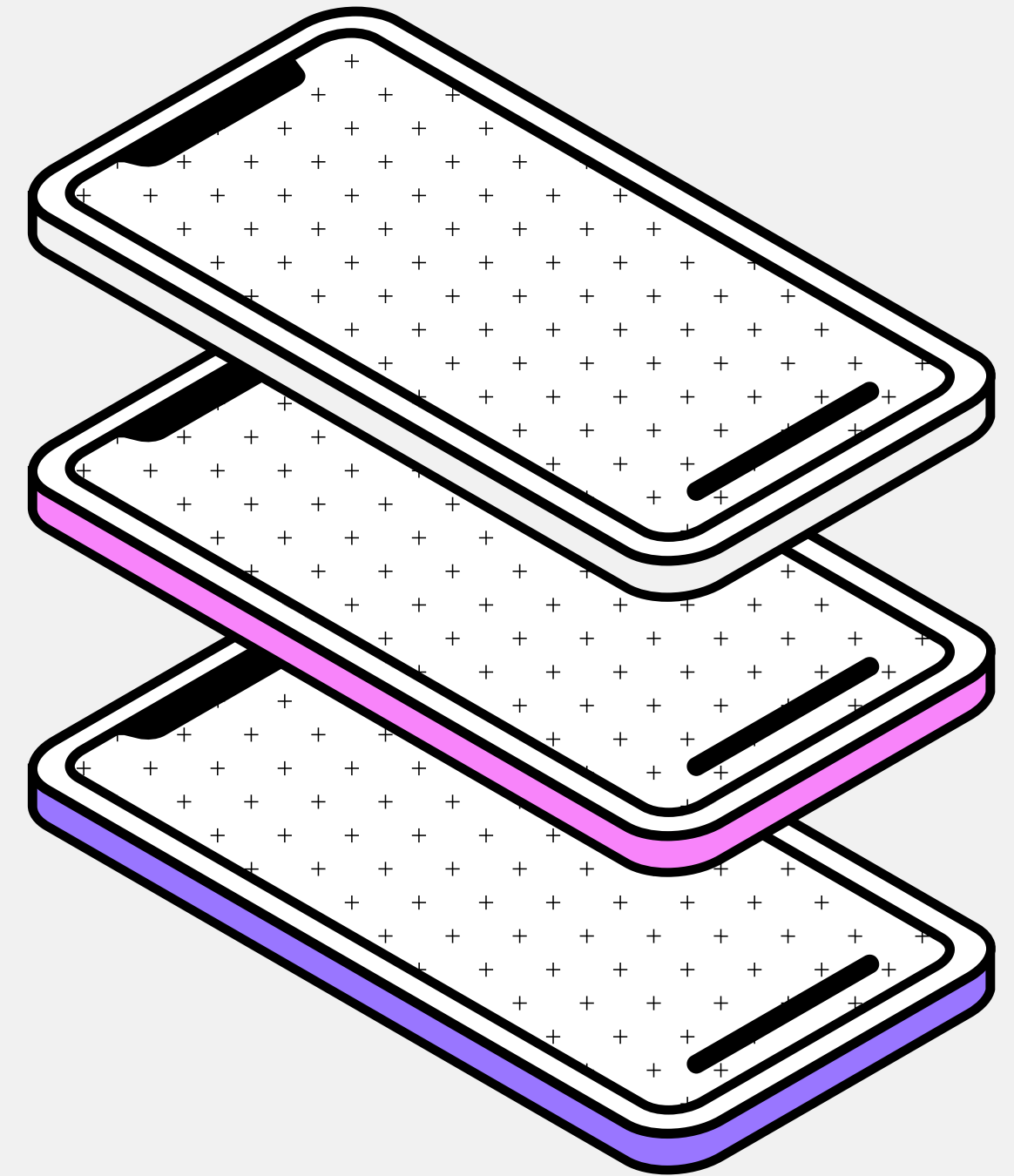
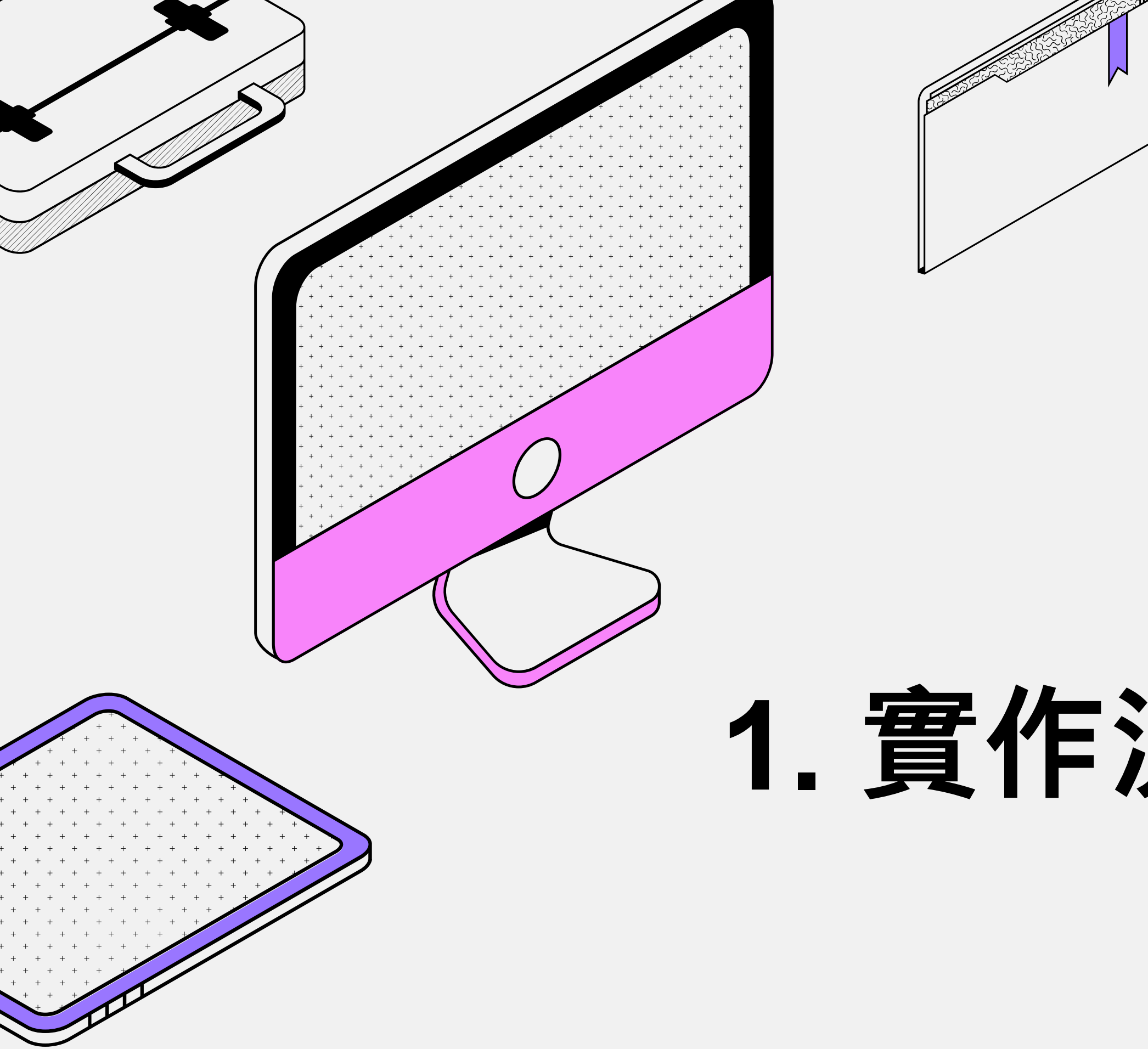


深度學習 HW3

共享腳踏車需求預測

313707045 方碩





1. 實作流程與思路說明

實作流程概覽

資料準備 與前處理

- 讀取資料集
- 蕭榮實驗後，移除 instant、dteday、mnth、holiday、atemp 欄位
- 使用 StandardScaler 對變數進行標準化
- sliding window 轉換 time series 資料
- 分割 training 和 validation dataset
- 建立 dataLoader

模型建立與訓練

- 定義 LSTM Model，包含 Bi-LSTM、dropout 和 fully connected 的序列模型
- 設定 MSE Loss、optimizer AdamW、用 CosineAnnealingLR 調整 learning rate。
- 加入 early stopping 防止過度訓練
- 訓練過程中算 val 的 MSE、儲存最佳模型

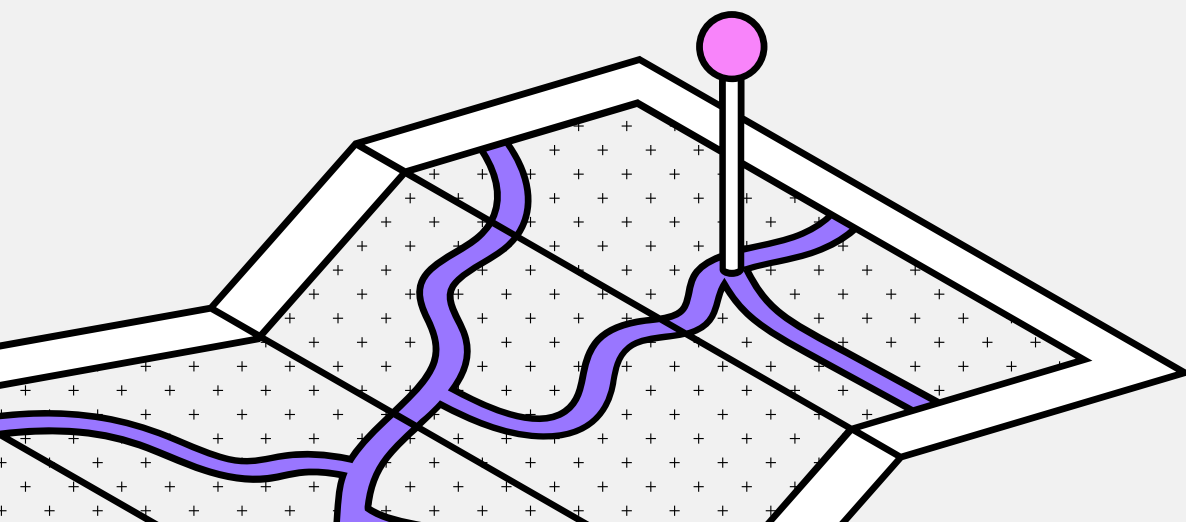
預測與結果輸出

- test 前處理，第一日用複製開頭資料 24 次，使用訓練集的 StandardScaler 標準化
- sliding window 轉換 time series 資料
- 將預測結果反標準化，算實際的 cnt 預測值
- 將每小時的預測結果彙總為每日的總量
- 除存 submission.csv

最佳模型參數

- batch : 128
- epoch : 500 (有設 early stopping)
- laerning rate : 0.0001
- window size : 24

- hidden size : 128
- layer : 3
- criterion : MSELoss()
- optimizer : AdamW(weight_decay=1e-4)
- scheduler : CosineAnnealingLR(T_max=10)



模型架構

使用：三層的 Bi-LSTM

Input Sequence ($x_{\{t-\text{WINDOW_SIZE}+1\}} \dots x_t$) ->

Forward LSTM (hidden=128) >>>
Backward LSTM (hidden=128) <<<
Concatenate Outputs (size 256) >>> Dropout(0.5) >>>

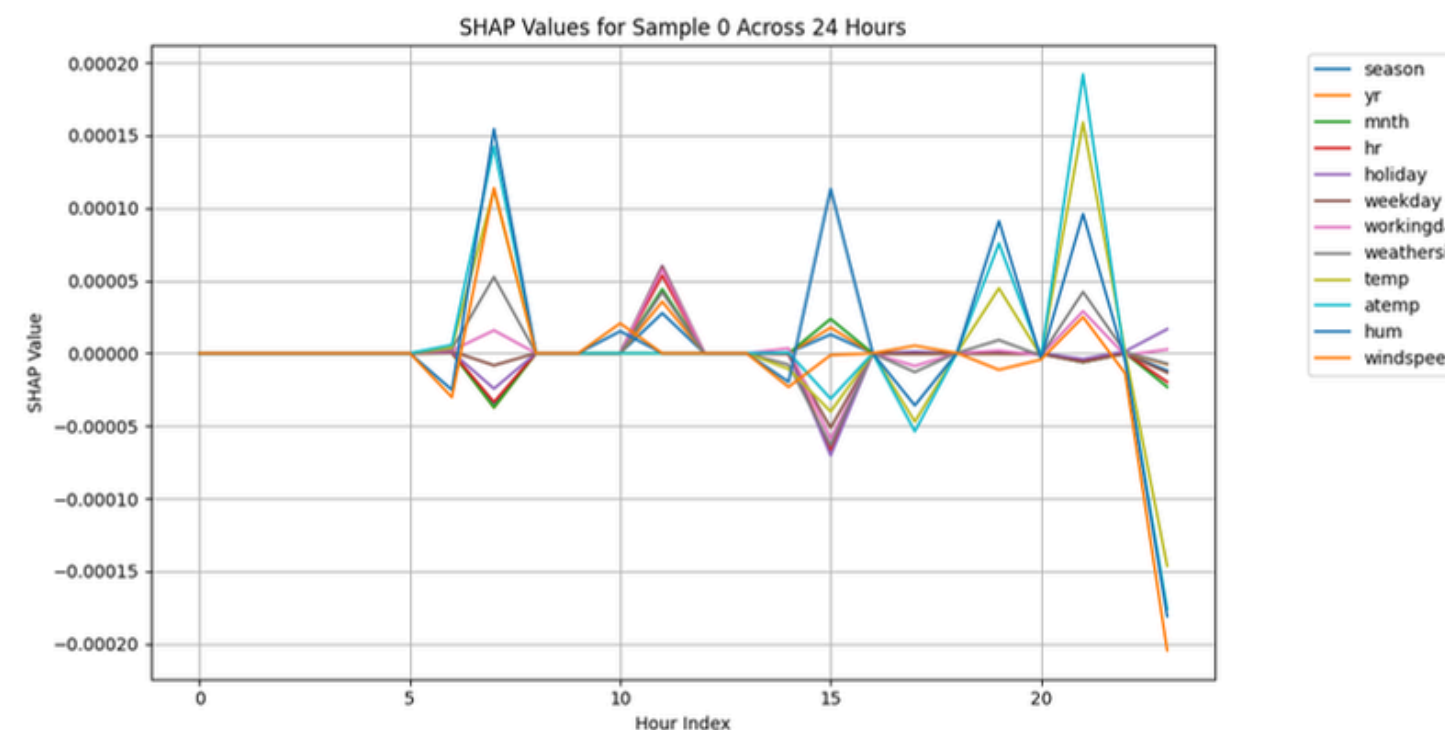
Forward LSTM (hidden=128) >>>
Backward LSTM (hidden=128) <<<
Concatenate Outputs (size 256) >>> Dropout(0.5) >>>

Forward LSTM (hidden=128) >>>
Backward LSTM (hidden=128) <<<
Concatenate Outputs (size 256) >>> Dropout(0.5) >>>

>>> Take Output at Last Time Step (size 256) >>>
Fully Connected Layer (Linear 256 > 1) >>>
Output (y_t)

輸入特徵

先做 SHAP 在座參數消融，
由於所有參數幾乎都有正有負，
因此下面多比較拿掉個別參數的 val MSE



Ablation by MSE Increase

Feature	Val MSE	ΔMSE
workingday	2.5579	-0.9052
windspeed	2.9956	-0.4675
mnth	3.0873	-0.3758
atemp	3.1467	-0.3164
temp	3.1719	-0.2912
hum	3.3650	-0.0981
all_features	3.4631	+0.0000

Feature	Val MSE	ΔMSE
all_features	3.4631	+0.0000
holiday	3.5259	+0.0628
season	3.9891	+0.5260
weekday	4.1540	+0.6909
weathersit	4.7551	+1.2920
hr	6.0174	+2.5542
yr	16.0623	+12.5992

最終參數使用 (灰底劃掉的為刪除的參數)

instant	dteday	season	yr	mnth	hr	holiday
weekday	workingday	weathersit	temp	atemp	hum	windspeed

Sliding Window 設計

以小時為單位預測一日，因此需轉換 24 小時的資料為 1 日的資料

- Time Stamp 逐一移動滑動窗口的起始位置
- 每個 Time Stamp i ，從標準化的 X_scaled 中提取長度為 24 (代表 24 小時 = 1 天) $X_scaled[i:i+WINDOW_SIZE]$

預測流程說明

1. 資料處理：

- a. 為了讓在預測第 1 筆測試資料時有足夠欄位，第一筆資料複製 24 次 (24小時的資料)
- b. `test_df` 去除不必要欄位並與 `head_pad` 合併為 `X_padded`

2. 標準化

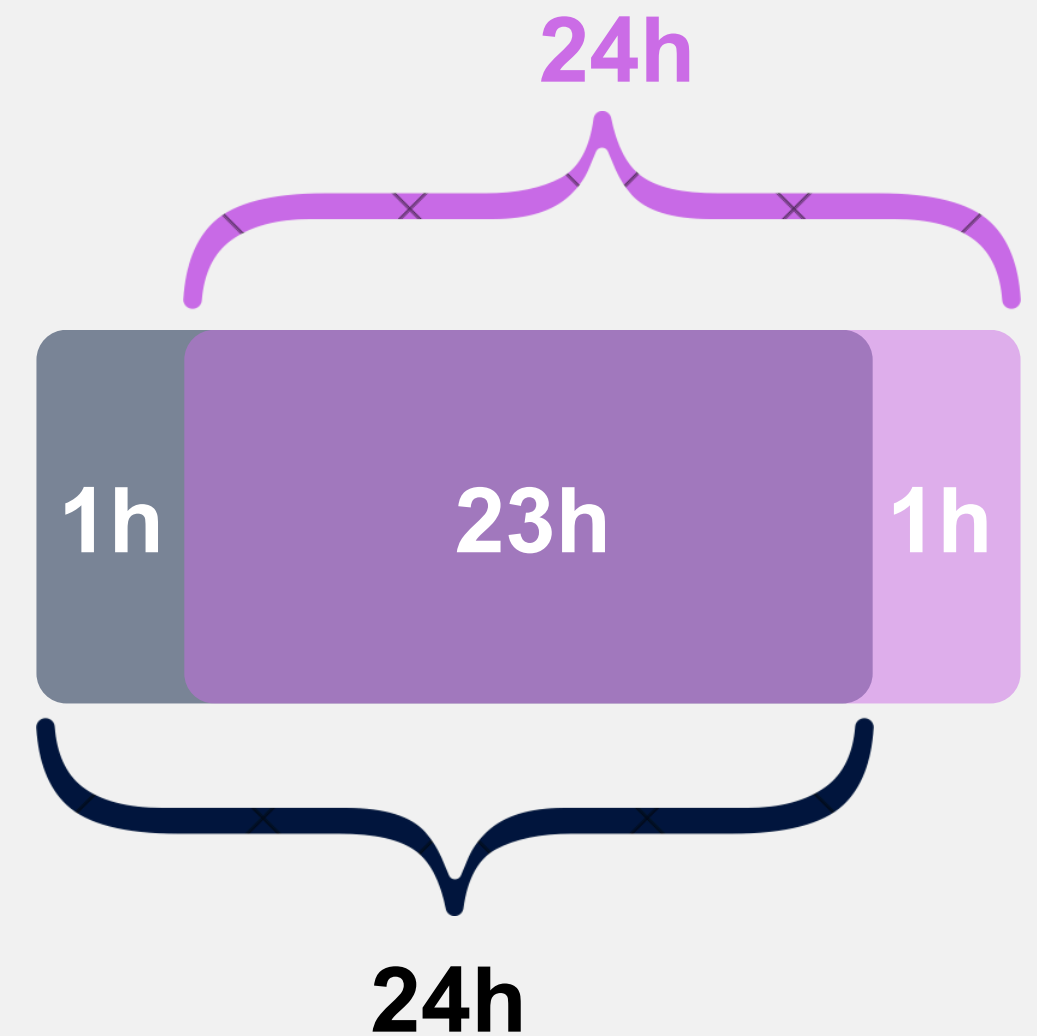
- a. 使用事先訓練好的 `scaler_x`，進行正確的標準化處理

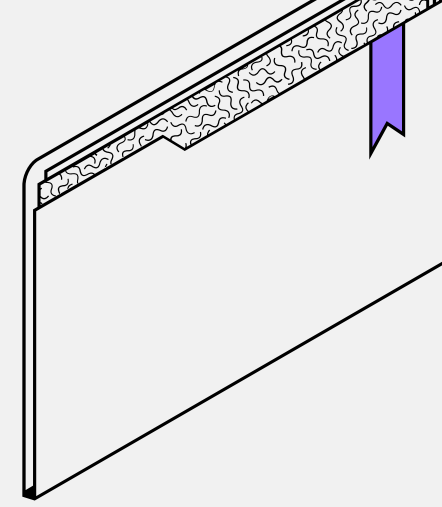
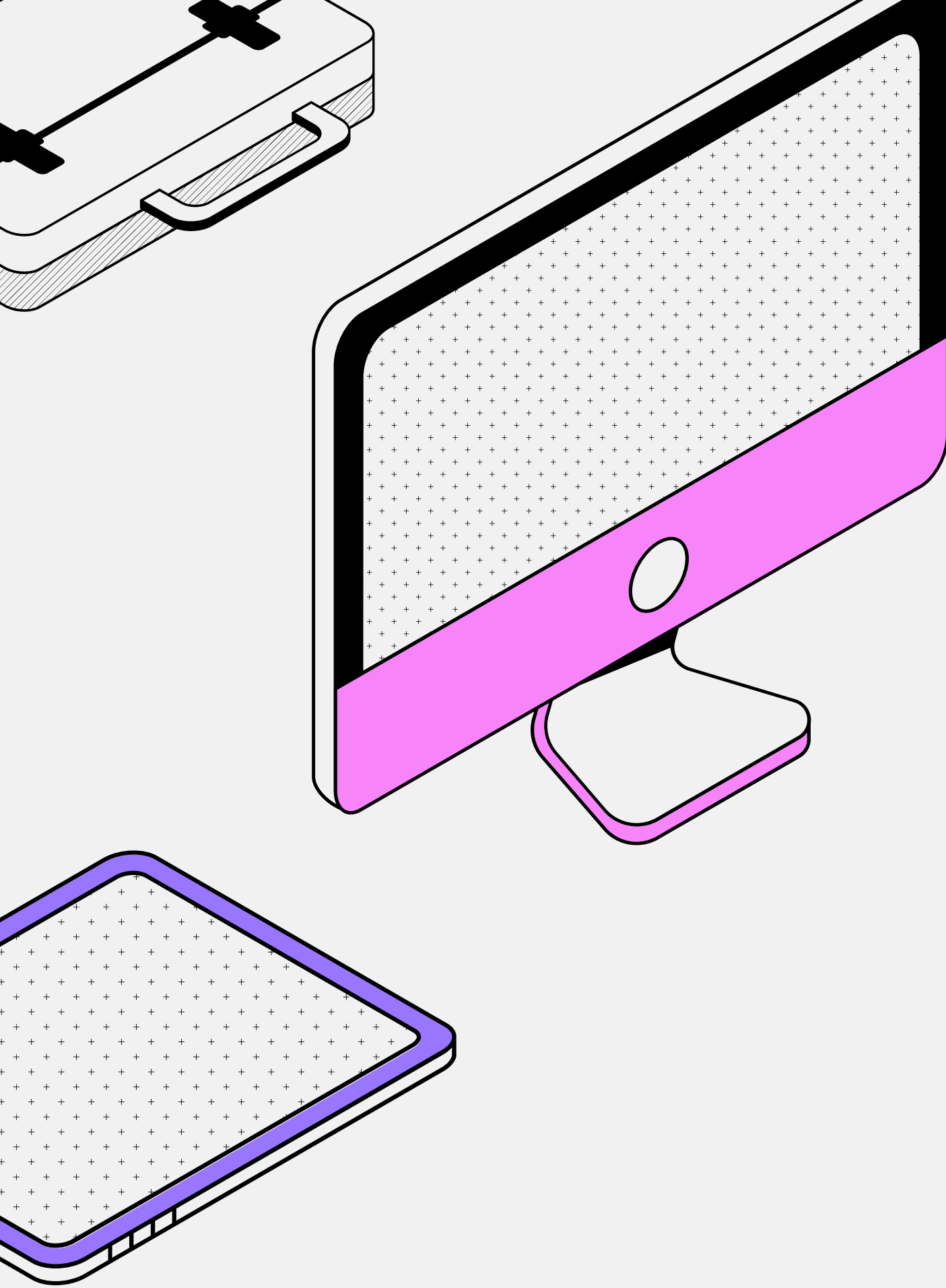
3. 模型建構與 inference 初始化

- a. 模型架構與訓的一致，載入 `best_model.pth`
- b. 使用 sliding window 每次預測 1 小時，重疊部分為 23 小時

4. 反標準化與輸出

- a. 使用 `scaler_y.inverse_transform()` 將預測還原成原始的 `cnt` 數值
- b. 使用 `np.sum()` 將每 24 筆合併為每日總量 (共 20 筆)
- c. 輸出為 `submission.csv`



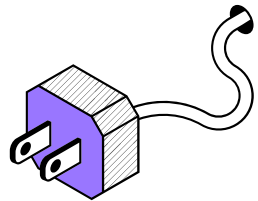


2. 參數比較與討論

參數比較與討論

由結果可觀察，適度減少不穩定特徵可提升模型泛化性，
但若刪過多 (windspeed、temp) 反而使誤差增加。

STEP 1 選擇使用的參數



根據前面 Ablation by MSE Increase 的結果，先將所有參數組合都拿去跑本地端測試，
選擇訓練完後 Val MSE 最低的版本，比較其 test 預測結果 (比如SHAP中windspeed 表現不好，但拿掉後 MSE 變差)。

拿掉的參數	訓練後的 Val MSE
(baseline) "all_para"	0.129443923687757
"mnth"	0.124058141688288
"holiday"	0.117807472321359
"atemp"	0.122544493784026
"windspeed"	0.158334591312393
"temp"	0.143261240053022
"mnth, holiday"	0.123437495878521
"mnth, atemp"	0.145291294658305
"holiday, atemp"	0.133763966292915
"mnth, holiday, atemp"	0.11547858718523
"mnth, holiday, atemp, windspeed"	0.127680483698867
"mnth, holiday, atemp, windspeed, temp"	0.228883742321524

test MSE 結果：

所有參數

726977.706

刪掉 "mnth, holiday, atemp"

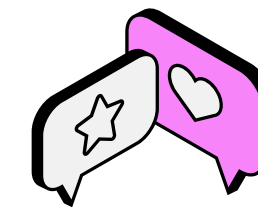
394783.880

此處使用的 LSTM Model 設定：

- 非 Bi-LSTM
- layer : 1
- hidden size : 64
- optimizer : Adam
- scheduler : ReduceLROnPlateau

參數比較與討論

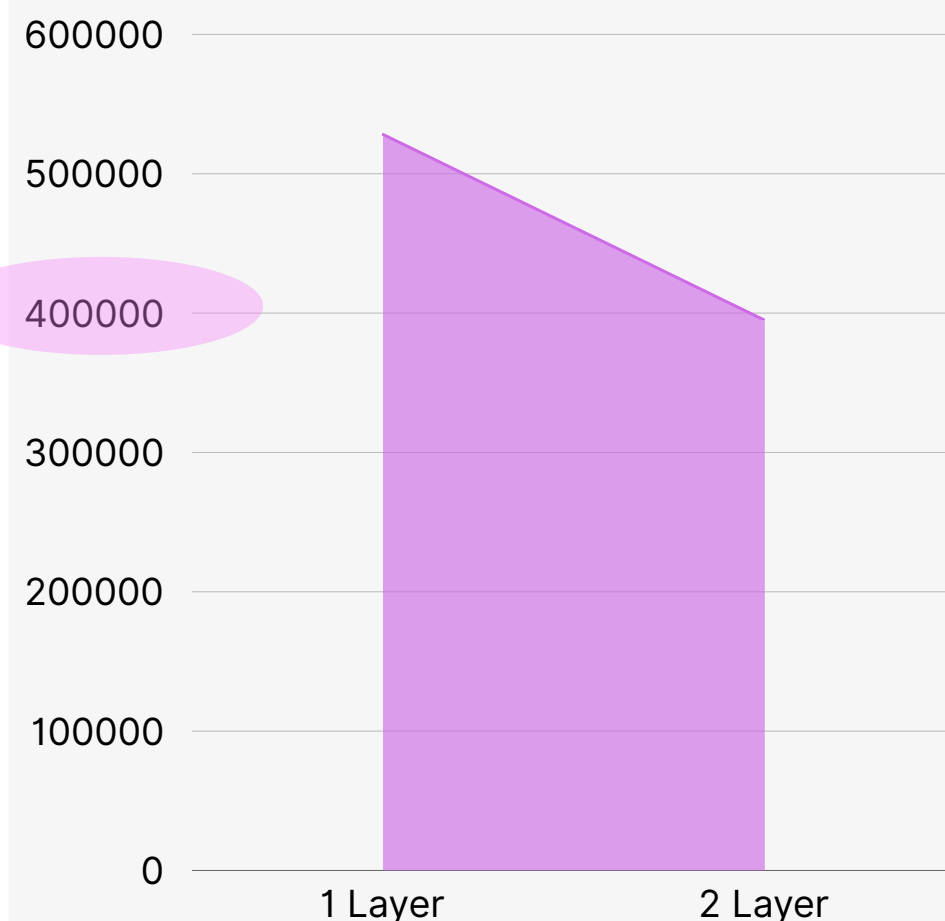
STEP 2 模型架構選用



先嘗試調整模型 Layer 數量，再使用 Bi-LSTM Model，並嘗試不同 Layer 數量的結果。

1. 測試 Layer 數調整

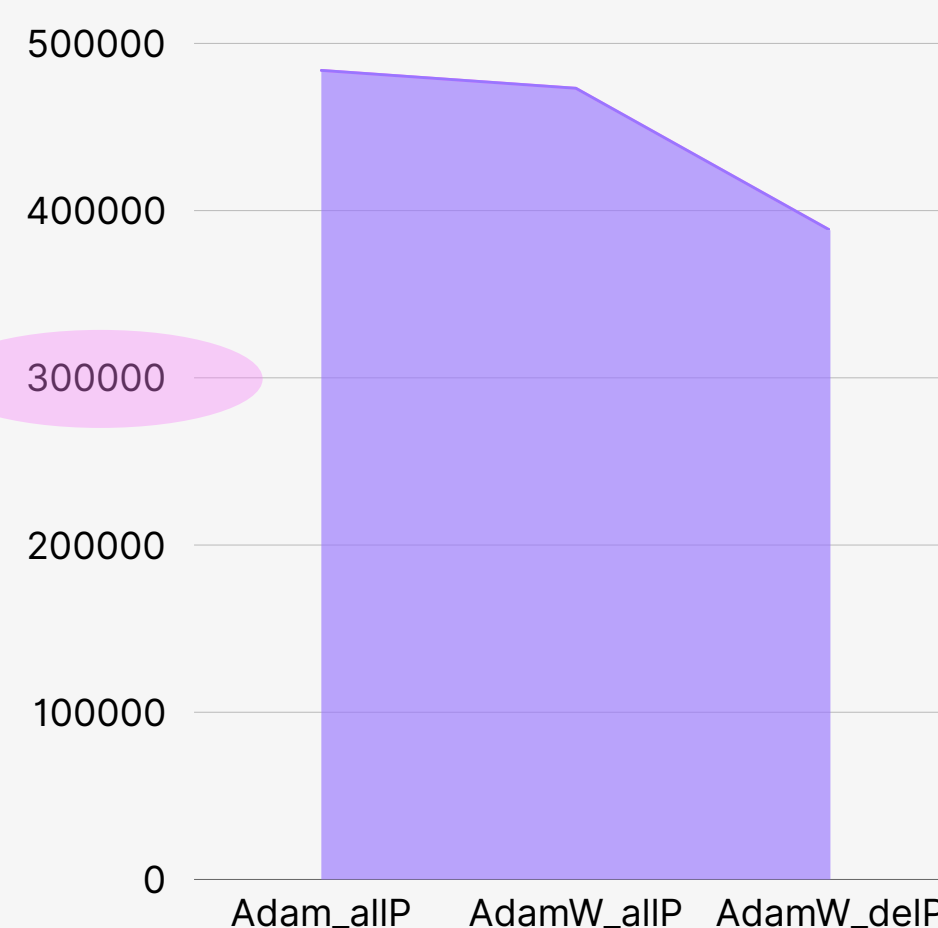
- 非 Bi-LSTM (layer 1~2)
- hidden size : 64
- batch size : 32
- optimizer : Adam
- scheduler : CosineAnnealingLR



確定多層數有表現較好

2. 改為 Bi-LSTM Model

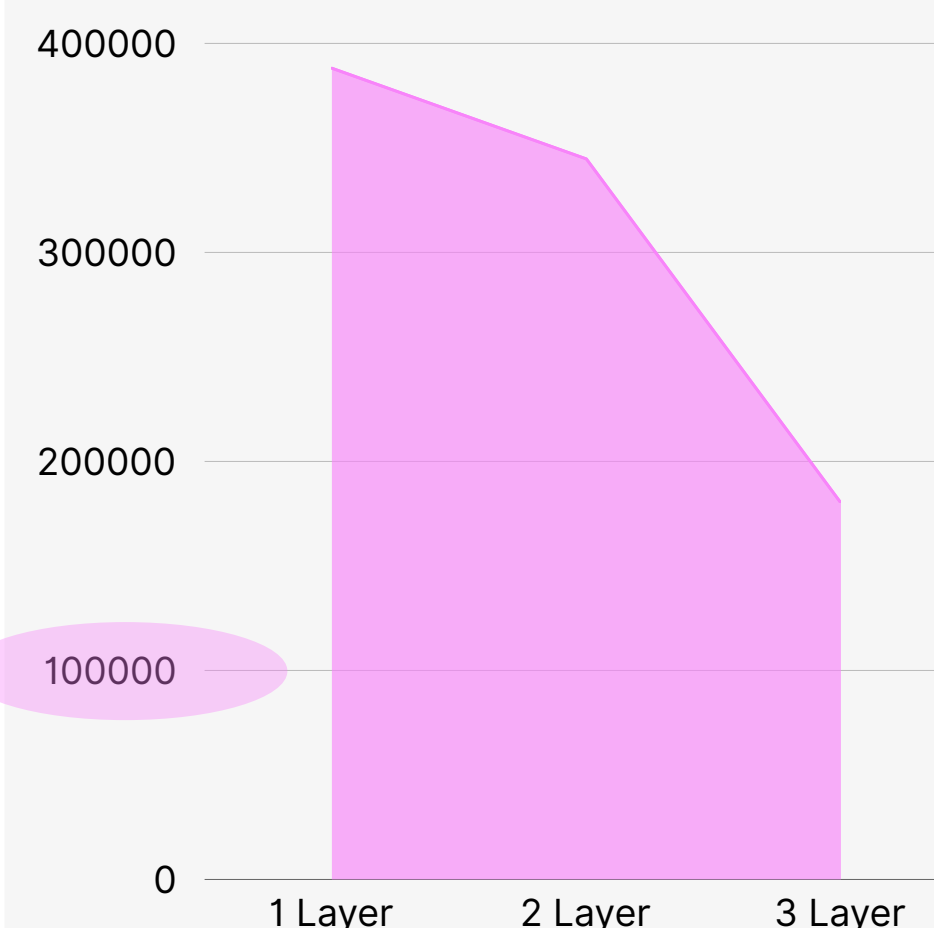
- Bi-LSTM (layer : 1)
- 嘗試全部跟刪參數後的結果
- hidden size : 64
- batch size : 32
- optimizer : Adam、AdamW



確定 Bi-LSTM、AdamW

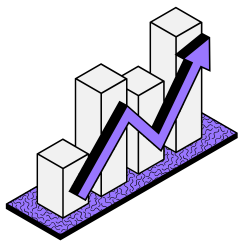
3. 調整 Bi-LSTM Model 層數

- Bi-LSTM (固定為刪參數版本)
- layer : 1~3
- hidden size : 64
- batch size : 32
- optimizer : AdamW



確定 Bi-LSTM 用 3 layers

參數比較與討論



大致確定模型架構、optimizer 和 scheduler 後，嘗試各種不同超參數組合。

1. hidden size 調整

- layer : 3
- batch size : 128
- dropout : 0.5
- val size : 0.1
- T_max : 10

2. Val size 設為 0.15

Val size 設 0.1 表現較好

3. AdamW 的 weight_decay 調整

- val size : 0.1

3. CosineAnnealingLR 的 T_max 調整

- weight_decay=1e-4

使用的參數設定	Test MSE
64 hidden size	180104.628
128 hidden size	104102.757
256 hidden size	270230.197

64 hidden size	236657.240
128 hidden size	211792.468
256 hidden size	271270.014

weight_decay=1e-4	136090.146
weight_decay=1e-5	233436.044

T_max=500	136090.146
T_max=10	104102.757

Bi-LSTM 最佳超參數組合：

- 移除參數 : mnth, holiday, atemp
- layer : 3
- batch size : 128
- dropout : 0.5
- val size : 0.1
- optimizer : AdamW、weight_decay=1e-4
- scheduler : CosineAnnealingLR、T_max : 10

