

## 1. 選題與競賽題目介紹

台灣地處亞熱帶地區，氣候優越而多樣，非常適合各種花卉的種植和育種。再者，可以通過海拔差異來進行花卉生產的季節性調整，從而使台灣花卉產業蓬勃發展。根據行政院農業委員會（農業合作社）的統計，台灣的花卉生產面積已從 1983 年的 2996 公頃增長到 2015 年的 13864 公頃，產值也從 16 億元增長到 18 億元。超過 167 億元。這表明台灣花卉業的產值在過去 30 年中增長了十倍以上。台灣的國內花卉產業主要集中在出口銷售上。由於其地理優勢，它靠近世界第三大花卉消費市場--日本，並且國內運輸和銷售技術得到了改善，因此國內花卉出口的產值逐年增加，從 2005 年的總出口額 7167 萬美元增長到 2015 年的 19362 萬美元。在國內市場，隨著國內經濟的發展和國民收入的增加，除了傳統的需求外，公眾也逐漸關注改善環境和家庭生活質量。因此，儘管花不是人們的必需品，但它們也是重要的國內農產品之一。

根據農業和食品管理局的 2017 年統計數據，批發市場約佔總產量的 90%，這表明批發市場對花卉的重要性。批發市場的目的是為生產者和銷售者提供可靠的交易服務平台，並建立運輸和銷售訂單。通過批發市場建立拍賣系統和公開透明的交易流程，我們可以提高運輸和銷售效率，保護買賣雙方的權利，並在市場供求之間保持平衡。目前國內近九成花卉交易以電腦化拍賣進行，拍賣成交價格直接影響供應商（花農）、承銷商、批發市場之經營決策，因此花卉成交價格之預測自然為各方決策者所重視。

## 2. 資料與評分標準

此競賽預測的標的為，台北花市特定期間，十種香水百合的「上價、中價、平均價、交易量」

預測期間：

Public Leaderboard：2020/01/10～2020/01/24（農曆十二月十六～十二月三十，2020/1/12 與 2020/1/15 休市除外，計 13 日）

Private Leaderboard：2021/01/28～2021/2/11（農曆十二月十六～十二月三十，2021/1/31 與 2021/2/3 休市除外，計 13 日）

最後成績以 Private Leaderboard 為主。

十種香水百合：

FS443 香水百合 馬可波羅粉三朵

FS479 水晶香水 多朵

FS592 薇薇安娜 雙朵

FS609 香水百合 康卡多多朵  
 FS639 OT 曼尼薩 多朵  
 FS779 OT 百合 節奏舞者多朵  
 FS859 百合 瑪丹娜多朵  
 FS879 OT 大連 多朵  
 FS899 OT 帕雷諾 多朵  
 FS929 OT 帕芳多 多朵

本競賽採用 MAPE(Mean absolute percentage error)進行計分，計分的項目共有四個，其對應之權重為：

項目	權重
上價 ( price_high )	0.1
中價 ( price_mid )	0.15
平均價 ( price_avg )	0.25
交易量 ( volume )	0.5

$$MAPE = \frac{100\%}{n} \sum \left| \frac{y - \hat{y}}{y} \right|$$

Multiplying by 100% converts to percentage  
 The residual  
 Each residual is scaled against the actual value

此競賽以行政院農業委員會農糧署的公開資料為主，並允許使用外部資料幫助解題。花卉交易歷史資料皆從「農產品批發市場交易行情站」進行下載。

資料欄位說明：

最高價（元/把）：當日該農產品成交之每把最高價。

上價（元/把）：以當日該農產品總交易量中最高價格之 20%，加權平均計算得之。

下價（元/把）：以當日該農產品總交易量中最低價格之 20%，加權平均計算得之。

中價（元/把）：以當日該農產品總交易量中扣除最高最低價格各 20% 剩餘之 60%，加權平均計算得之。

平均價（元/把）：當日該農產品總成交之每把平均價。

交易量（把）：當日該農產品總成交量。

殘貨量（把）：係指花卉經線上拍賣二次未能於底價（50 元）成交，即為殘貨（流標），該產品即直接下架，並於拍賣後統一銷毀，不得再轉以其他交易形式銷售。

在訓練過程中，使用從 1998-12-01 至 2020-12-01 的資料，由於這次比賽要進行未來兩週的價格預測，因此將資料分別切分成 6630 天的訓練資料以及 14 天的驗證資料以進行模型好壞的評估。

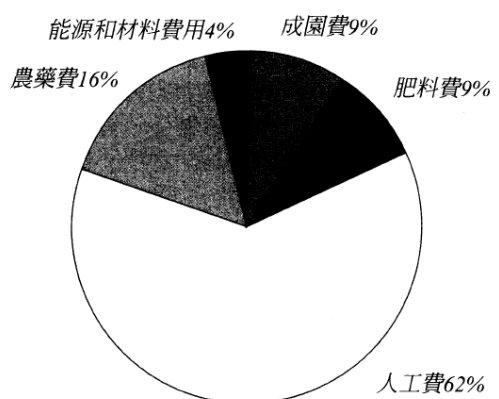
### 3. 文獻參考

在開始建立模型前，先行整理國內外對百合花價格預測相關研究以利後續建模時會有更清楚的方向。簡榮樞(2011)建構影響台灣花卉批發市場決價因素之多元線性迴歸分析模型，其結果顯示在市場內部中以花卉進貨量及殘貨率影響花卉價格最為顯著，其次相對數量；林彥甫(2017)以台北花市的拍賣價格資料中，分類多種影響價格之因素為變數進行元迴歸析實驗結果得到花卉在全部品種上皆會受到前月殘貨量及節慶因素而對成交價格產生顯著影響。綜觀各方研究結果指出最直接花卉價格變化的因素共有：進貨量、前日殘貨量、前日均價及節慶影響。在與花價預測直接有關的研究中，總結來說會有幾個重要的特徵可能會對價格產生影響，進貨量、殘貨率、該市場之全部殘貨率、相對數量、粗結婚率、選舉、節日、生長時程等。

此外，我也搜尋到進行高麗菜價格預測的相關研究，其中提到民眾在網路上的搜尋行為與高麗菜的平均批發價格兩者具有相關性，兩者的皮爾森相關係數呈負相關。從歷史走勢圖中也可以發現搜尋量與批發價格呈現反向關係，當搜尋量上漲時，批發價格則是下跌，代表說網路搜尋量有可能是影響農產品價格的其中一個因素。另外，從建模所使用的變數來看，天氣資訊、油價等也會是值得嘗試用來建模的選擇。

表 3-10 輸入變數資料

輸入變數	當日氣候	最高氣溫
		最低氣溫
		平均氣溫
		測站氣壓
		相對濕度
		風速
		年份 (104 年~107 年)
		月份 (1 月~12 月)
		日期
		降雨量
		油價
		有無颱風 (有侵台 1，無則 0)
		侵台路線 (依中央氣象局圖示)
		嚴重程度 (輕颱 1，中颱 2，強颱 3)
	歷史資料	前一日交易價
	60 天前氣象資料	氣溫
		風速
		降雨量
輸出變數		每日甘藍價格



圖三：八十四年期每公頃玫瑰花生產費用組成結構

表九：八十四年期縣別玫瑰花每公頃材料和能源使用費

單位：元

項目	平均a	台中縣	彰化縣	南投縣	屏東縣	台東縣	花蓮縣
材料費	47,749	36,798	48,860	48,501	17,739	8,778	59,234
稻草	2,687	2,537	2,434	2,398	4,826	-	3,793
紙漿及紙	1,098	1,207	1,428	1,374	-	356	-
紙製品	35,941	28,251	41,349	30,316	2,652	1,956	49,114
塑膠製品	7,813	4,803	3,649	14,086	10,261	6,467	5,293
其它	210	-	-	128	-	-	1,034
能源費	7,588	3,512	8,039	6,531	5,661	7,227	11,379
汽油費	1,854	1,330	3,050	306	443	3,004	2,759
柴油費	676	246	-	204	-	1,911	3,448
電費	5,057	1,936	4,989	6,020	5,217	2,311	5,172

由於花的成本的浮動有可能也是影響拍賣價格的原因，因此針對花的成本分析，搜尋到了玫瑰花的成本分布，雖然百合花在某些層面上成本來源可能不同於玫瑰花，但裡面提到的成本來源值得進行深入的探究。

#### 4. 模型建構與分析

本次的競賽，針對 10 種不同種類的花需要分別對上價、中價、平均價、交易量做出 13 步預測。在模型架構設計與選擇上，需要能以歷史資料對未來產生 13 步的結果，因此我認為最容易進行的模型架構為 **Multivariate Multi-step Multi-target Sequence to Sequence Model**，也就利用所蒐集到的歷史資 **Sequence**(包含花的各種價格資料、天氣、成本相關資料)直接對上價、中價、平均價、交易量這四個預測目標產生長度為 13 的 **Sequence**。

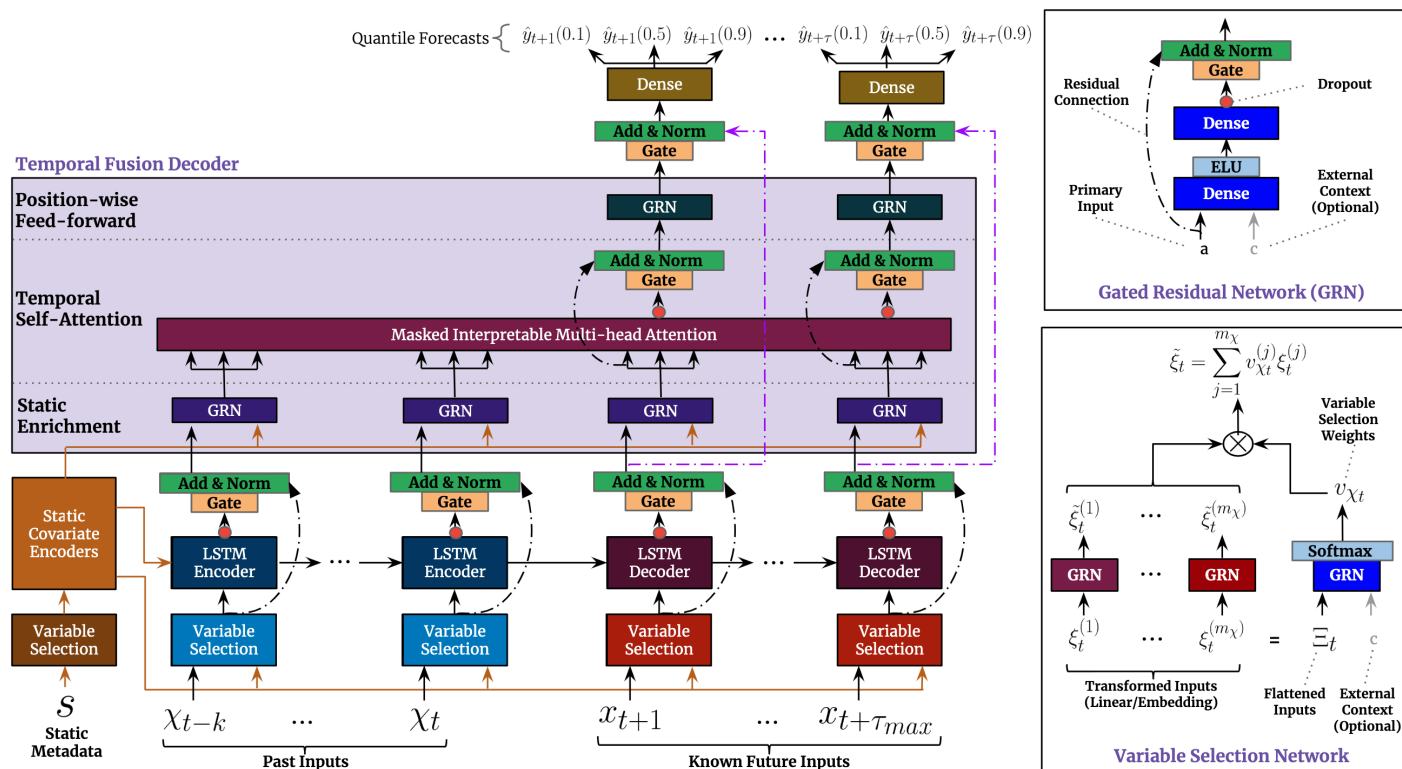
雖然上述是最容易進行的，但不見得會是最準確的。因此我也不侷限於只使用深度學習的架構進行預測，同時列出了幾種嘗試的方向：

1. 傳統時間序列預測法：整合移動平均自回歸模型 **ARIMA**
2. 基於機器學習的線性、**Tree-based** 等回歸模型：
  - **BayesianRidge**
  - **ElasticNet**
  - **Lasso**
  - **Support Vector Regression**
  - **K-nearest neighbors Regression**
  - **Tweedie distribution Regression**
  - **Gradient Boosting Regression**
  - **ExtraTreesRegressor**
  - **XGBoost Regression**
  - **LightGBM Regression**

### 3. 基於深度學習等模型：

- DeepAR
- Transformer
- CanonicalRNN
- DeepFactor
- DeepState
- Temporal Fusion Transformer
- Encoder-Decoder LSTM
- CNN-LSTM Encoder-Decoder
- ConvLSTM Encoder-Decoder

#### Temporal Fusion Transformer



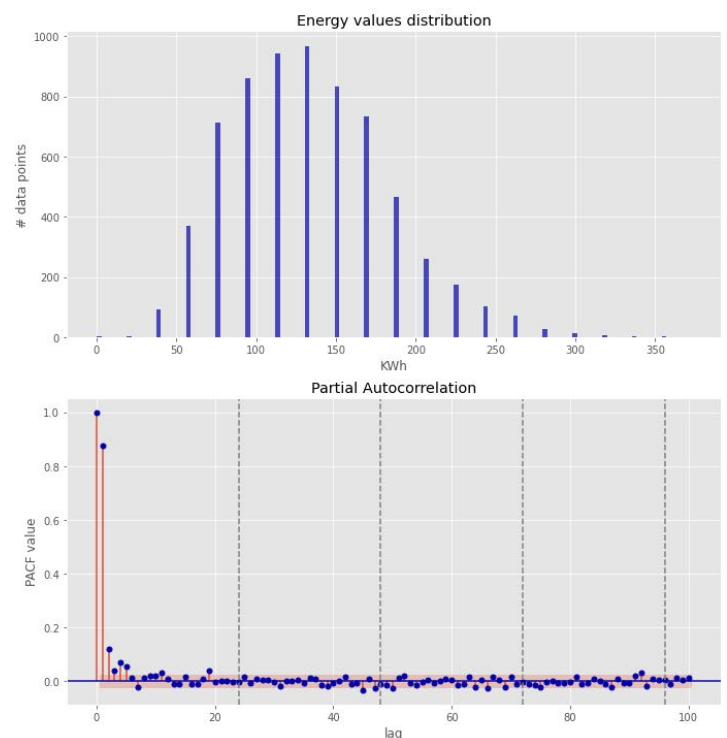
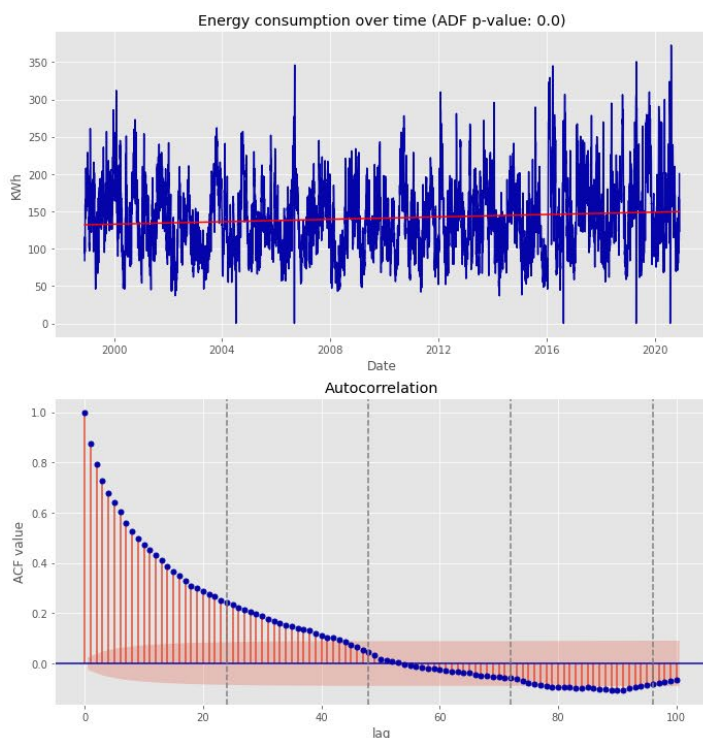
### 4. 其他模型：Facebook Prophet

上述的模型中，ARIMA、Prophet 與深度學習為基礎的模型都可以直接對未來產生多步預測，而機器學習為基礎的模型則需要經過額外處理才可以產生多步預測。

在接下來呈現的實驗結果中，由於評分項目平均價所佔的比重最高，因此在評估時先以「FS443 香水百合 馬可波羅粉三朵」此種百合的平均價作為模型好壞的評估，用以了解不同的處理方式能達到較準確的結果。

由於最終評分標準為評估 2021/01/28 ~ 2021/2/11 共 13 天這段時間的預測準確度，而上傳結果時間為 2021/01/27，因為前一天的歷史資料對未來結果預測的結果有相對大的影響，若需要對未來進行 13 步預測的話，代表需要等 2021/01/27 日拍賣結束後並有辦法獲取當日拍賣資訊，並且要立即進行預測才有辦法獲得 2021/01/28 起 13 天的結果。在經過考慮後，決定採用對未來進行 14 步預測，並在繳交期限前一天的 2021/01/26 獲取當日拍賣資訊後對未來進行 14 步預測並擷取後面的 13 天繳交。(P.S. 後來發現拍賣資料的更新並不是那麼即時，有可能最早只能獲取 4 天前的資訊，因此對未來的預測可能需要拉長到 18 步甚至 20 步預測)

資料擷取上，先單純以價格相關歷史資料(最高價、上價、中價、平均價、下價、交易量)進行單變量或多變量的建模，如單變量之模型的輸入為平均價格歷史資料輸出為未來的平均價格預測；而多變量模型的輸入則為最高價、上價、中價、平均價、下價、交易量等歷史資料並輸出未來的平均價格預測。目前擷取從 1998-12-01 至 2020-12-01 的資料拍賣資料，並將 2020-11-16 設為截斷點，2020-11-16 起至 2020-12-01 切為測試資料。至於相關研究有提到殘貨量可能會是影響價格的因素之一，不過在經過確認以後發現所蒐集的殘貨皆為 0，因此並沒有考慮將此項變數放入模型中。



## ARIMA & Prophet :

在配適 ARIMA 模型前，需要先檢查時間序列是否呈現平穩以及季節性。經



過上圖的繪製以及 ADF、KPSS 等皆顯示序列為平穩。接著利用 Auto ARIMA 工具去配適並獲得 AIC 最低之模型，以結果來看會是 ARIMA(3, 0, 3) 表現最好，而經過測試後 ARIMA(3, 1, 3) 增加一階差分的結果會比 ARIMA(3, 0, 3) 還要好，利用 14 筆測試資料所計算出來的 RMSE 為 29.3，此結果可以當作後續與其他模型進行比較的 baseline。

```

ARIMA(1,0,1)(0,0,0)[0] : AIC=60997.487, Time=0.77 sec
ARIMA(0,0,0)(0,0,0)[0] : AIC=85195.810, Time=0.08 sec
ARIMA(1,0,0)(0,0,0)[0] : AIC=61223.542, Time=0.24 sec
ARIMA(0,0,1)(0,0,0)[0] : AIC=77669.923, Time=0.45 sec
ARIMA(2,0,1)(0,0,0)[0] : AIC=60820.994, Time=1.45 sec
ARIMA(2,0,0)(0,0,0)[0] : AIC=inf, Time=0.26 sec
ARIMA(3,0,1)(0,0,0)[0] : AIC=inf, Time=4.16 sec
ARIMA(2,0,2)(0,0,0)[0] : AIC=inf, Time=4.51 sec
ARIMA(1,0,2)(0,0,0)[0] : AIC=60936.374, Time=0.81 sec
ARIMA(3,0,0)(0,0,0)[0] : AIC=inf, Time=0.25 sec
ARIMA(3,0,2)(0,0,0)[0] : AIC=inf, Time=2.98 sec
ARIMA(2,0,1)(0,0,0)[0] intercept : AIC=60706.719, Time=6.63 sec
ARIMA(1,0,1)(0,0,0)[0] intercept : AIC=60744.192, Time=2.72 sec
ARIMA(2,0,0)(0,0,0)[0] intercept : AIC=60754.355, Time=1.02 sec
ARIMA(3,0,1)(0,0,0)[0] intercept : AIC=60707.033, Time=9.60 sec
ARIMA(2,0,2)(0,0,0)[0] intercept : AIC=60705.482, Time=9.56 sec
ARIMA(1,0,2)(0,0,0)[0] intercept : AIC=60731.038, Time=3.61 sec
ARIMA(3,0,2)(0,0,0)[0] intercept : AIC=60711.008, Time=10.56 sec
ARIMA(2,0,3)(0,0,0)[0] intercept : AIC=60698.940, Time=9.85 sec
ARIMA(1,0,3)(0,0,0)[0] intercept : AIC=60701.247, Time=4.40 sec
ARIMA(3,0,3)(0,0,0)[0] intercept : AIC=60695.269, Time=12.07 sec
ARIMA(4,0,3)(0,0,0)[0] intercept : AIC=60698.360, Time=12.28 sec
ARIMA(3,0,4)(0,0,0)[0] intercept : AIC=60698.001, Time=6.66 sec
ARIMA(2,0,4)(0,0,0)[0] intercept : AIC=60698.763, Time=10.43 sec
ARIMA(4,0,2)(0,0,0)[0] intercept : AIC=60700.349, Time=11.79 sec
ARIMA(4,0,4)(0,0,0)[0] intercept : AIC=60698.813, Time=8.11 sec
ARIMA(3,0,3)(0,0,0)[0] : AIC=inf, Time=7.38 sec

```

$$y(t) = g(t) + s(t) + h(t) + \epsilon t$$

- $g(t)$ : piecewise linear or logistic growth curve for modeling non-periodic changes in time series
- $s(t)$ : periodic changes (e.g. weekly/yearly seasonality)
- $h(t)$ : effects of holidays (user provided) with irregular schedules
- $\epsilon t$ : error term accounts for any unusual changes not accommodated by the model

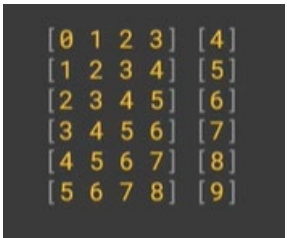
Prophet 是 Facebook 所開發的時間序列預測 API，模型架構如上。主要想法為將時間當作迴歸函數，同時配適多個時間線性和非線性函數。其中需要指定季節效應的時間頻率，並且經過實驗使用不同長度的訓練資料也會影響到膜性效能。而季節效應的時間長度是我認為較難給定的，主要是每個月、每周、每年的序列並不是等長，原因是每周休市的日期並非固定、天數也非固定值。



	RMSE
全部訓練資料	42.69
後 1000 筆訓練資料	51.5
後 365 筆訓練資料	39.82

由於現階段的目標並非將每個模型經過參數調整達到最佳狀態，而是先大致比較不同的模型在採用預設參數的條件下效能表現如何。從結果來看若沒有參數優化，Prophet 的模型皆比使用傳統方法 ARIMA 還差。

機器學習方法：



在開始進行機器學習的建模時，我忽略掉由於目標是需要作出未來的 14 步預測，因此若使用上圖的切割訓練資料方法(如用前 4 步的歷史資料預測第 5 步)這樣的方法將會沒辦法符合我的需求。假設需要預測未來 14 步，採用上述資料切割方法只能預測到未來 1 步的結果，因為要預測未來第 2 步的結果，將會需要有未來第 1 步的資料才可以進行預測。因為上述原因，我在此階段花費很多時間進行嘗試，但因為疏忽這些結果都無法用在實際預測上。

至於如何解決資料切分的方法以利使機器學習的模型達到預期的目標，為了便於描述這裡假設一個時間序列

$$[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, X, Y, Z]$$

假設要做的是預測未來的 3 個時間點，X，Y，Z 的序列的值，並且為了方



便描述，這裡先使用 1 次 lag 項進行舉例

- 第一種：直接多步預測：

直接多步預測的本質還是單步預測，多步轉單步，比如上面若要預測 3 個時間點的序列的值，則，我們就構建 3 個模型如下

Model1 Train: [1, 2, 3, 4, 5, 6, 7, 8, 9] Test: [X]

Model2 Train: [1, 2, 3, 4, 5, 6, 7, 8, 9] Test: [Y]

Model3 Train: [1, 2, 3, 4, 5, 6, 7, 8, 9] Test: [Z]

這種做法的問題是如果我們要預測未來 N 步，則複雜度將會變很高，如預測未來 100 天，則代表要分別訓練 100 個模型；

- 第二種：遞迴多步預測：

遞迴多步預測，遞迴多步預測的本質還是簡單的單步預測，但是和第一種情況不同，遞迴多步預測不需要訓練 N 個模型已產生未來的 N 步預測，僅僅一個模型就夠了。舉例來說，假設我們構建了一個 3 步 lag 模型，資料切分如下：

Train: [lag(t-3), lag(t-2), lag(t-1)] Test: [t]

Train: [1, 2, 3] Test: [4]

Train: [2, 3, 4] Test: [5]

預測時還是以上面的例子為例：

[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, X, Y, Z]

先預測 X，用 model 去 predict([8, 9, 10]) 得到 prediction(X)，假設 predict 出來的結果為 11.24，然後再把預測值當作特徵，得到[9, 10, 11.24]，然後用 model 去 predict ([9, 10, 11.24])得到 prediction(y)，以此類推即可完成 N 步預測。

	Window=2	Window=3	Window=4	Window=5	Average
RandomForest	25.15	36.75	34.5	36.51	33.23
BayesianRidge	26.85	27.35	28.52	29.39	28.03
Lasso	26.85	27.35	28.52	29.39	28.03
XGBoost	40.27	29.93	24.18	42.37	34.19
LightGBM	36.65	41.09	36.76	37.79	38.07
SVR	34.03	43.36	32.44	34.07	35.98
KNN	41.82	58.83	31.6	36.19	42.11
Tweedie	26.85	27.35	28.52	29.39	28.03
GradientBoosting	40.65	40.98	37.3	37.7	39.16
ExtraTrees	32.67	27.2	40.13	39.35	34.84
Average	33.18	36.02	32.25	35.22	

採用遞迴預測，從實驗結果來看，不同的 window size(lag 項)將會影響不同模型的效能表現，而線性模型在整體表現上結果也都較為優良。相較 ARIMA 與 Prophet，平均來看也較為準確。由於此階段只做初淺的比較，因此每個模型皆沒有經過參數調整，相信在經過調參後結果都會有所提升。

深度學習方法：

相較上述所提的模型，基於深度學習的模型設定上相較彈性，經過簡單的調整後即可完成 Multivariate Multi-step Multi-target 的預測，在這此階段為了與前述 ARIMA 與機器學習模型進行效能比較，所以先進行 Multivariate/Univariate Multi-step Single Target 的預測，也就是只輸出未來 14 步的平均價預測。

Model	Uni/Multi	RMSE
DeepAR	Univariate	55.23
DeepAR	Multivariate	41.69
Transformer	Univariate	30.43
CanonicalRNN	Univariate	43.9
DeepFactor	Univariate	58.04
DeepState	Multivariate	41.8
Encoder Decoder LSTM	Multivariate	31.33
Temporal Fusion Transformer	Multivariate	28.29

在進行實驗時，基本上是先遵照各論文的預設參數進行訓練，並無額外做調整參數最佳化的工作。而從結果來看，Seq2Seq 類的模型表現都較為優良，而採用多變量(最高價、上價、中價、平均價、下價、交易量歷史資料)進行建模

又會比單單使用平均價一種歷史資料來建模還來得準確。再後續的嘗試中，將會將重點放在 Seq2Seq 類的模型尤其是 Transformer，期待進行深入的參數調整後能有更好的表現。

## 5. 結論與改進方向

在此階段，主要著重在模型的建構與效能的比較，以利提供後續在選擇候選模型時的方向。從實驗結果來看，部分模型在經過調整以後都有不錯的潛力。

之後改善的方向，除了改良機器學習的架構使其能做到 Multivariate 的多步預測，並同時嘗試加入如天氣、油價等非價格之歷史資料，並且比較不同特徵組合對 Multivariate 模型的影響。此外，也會將參數調整視為接下來的階段性目標，希望在將候選模型透過調參達到單一模型最佳效能，最後能用 Ensemble 的方式來綜合各個模型的優點進行預測。