運用大數據機器學習方法預測臺灣經濟成長率*

何宗武、葉國俊、牟萬馨、林雅淇**

摘　要

近年來國際間逐漸應用機器學習方法，進行總體經濟變數或其他金融指數、資產和股市崩盤事件的預測。相較於理論驅動的計量模型，資料驅動的機器學習方法不僅有助於減少預測誤差，亦能改善捕捉經濟成長率波動性的精確度。順應此國際趨勢，本文應用機器學習方法來預測我國經濟成長率，目的在找出最具預測績效，能捕捉整體時間序列趨勢、波峰與波谷的模型。我們發現在未添加其他解釋變數下進行單步預測時，機器學習模型未必較佳；當進行多步動態遞迴預測時，機器學習模型可能具備優於傳統時間序列模型的預測績效。而在添加其他解釋變數下進行多步預測時，機器學習模型的預測能力則大幅優於傳統時間序列模型，其中最佳者為自動化機器學習模型 (autoML)。另外，我們也把解釋變數及其落後期，以機器運算找出最佳組合。有趣的是，過多解釋變數不一定能得到較佳的預測績效，也可能產生大而無當的成本。最後，受限於經濟成長率為季資料，觀察值數量較少，對於號稱大數據的學習演算法而言，訓練和交叉驗證的設計都受到極嚴重限制，學習不足使其預測優勢難以被確認。如何使機器學習模型也能在低頻時間序列資料上有所發揮，應是未來可再努力的研究方向。

* 本文係摘錄自中央銀行委託研究計畫報告。所有論點皆屬作者意見，不代表中央銀行及作者服務單位之立場，文中錯誤皆由作者負責。作者特別感謝陳宜廷教授、徐士勛教授、中央銀行林前處長宗耀、余助理研究員軒與計量科同仁對本計畫所提供的寶貴意見、指正與協助。囿於篇幅與印製過程，文中相關細節如需完整說明或原始彩圖，請與作者聯繫取得。

** 何宗武為臺灣師範大學管理學院教授，葉國俊為臺灣大學國家發展研究所教授，牟萬馨為中原大學財務金融學系副教授，林雅淇為逢甲大學經濟學系助理教授。

壹、緒　論

本文透過預測績效評估，探討機器學習模式 (machine learning) 與時間序列模型方法，何者能對臺灣經濟成長率預測做出較佳貢獻。前者直接分析從數據資料所萃取的特徵，透過訓練模式以提升預測表現；後者無訓練機制，主要將預先設定的模型套用於數據資料，並漸進地尋找解決方案。機器學習方法是否在預測能力上優於時間序列模型迄無定論，較早的文獻如Makridakis and Hibon (2000) 和Morlidge (2014) 並不認同，但近年發展則已使學界較為樂觀 (Baek and Kim, 2018; Beyca et al., 2019; Bolhuis and Rayner, 2020; Chakraborty and Joseph, 2017; Chatzis et al., 2018; Kong et al., 2017; Richardson and Mulder, 2018; Tiffin, 2019; Torres et al., 2018; Zhao et al., 2017a)。

近期機器學習模式，諸如支援向量機器 (support vector machine, SVM)、自動化機器學習 (automatic machine learning, autoML) 和深度學習長短期記憶 (long-short term memory, LSTM) 的循環類神經網絡 (recurrent neural network, RNN) 等聲譽鵲起。首先是被視為淺層機器學習的SVM，以最小化度量法求解參數估計值。相較於最小平方法，它不易受離群值偏離整體數據資料趨勢的影響；其次是autoML，是一個自動機器學習和大數據的公開資源解決方案，依照數據資料的時間序列結構產生大量特徵，再將這些特徵作為解釋變數進行演算，可在該環境下建立自動化演算法如類神經網路 (neutral network, NN) 或隨機森林 (random forest, RF)，自動執行機器學習工作流程，利於執行建模相關的任務；第三是LSTM的RNN，該方法擅長趨勢識別 (pattern-recognition)，得以處理長期歷史訊息，將之前的網絡做連結，彌補NN無法克服的問題。

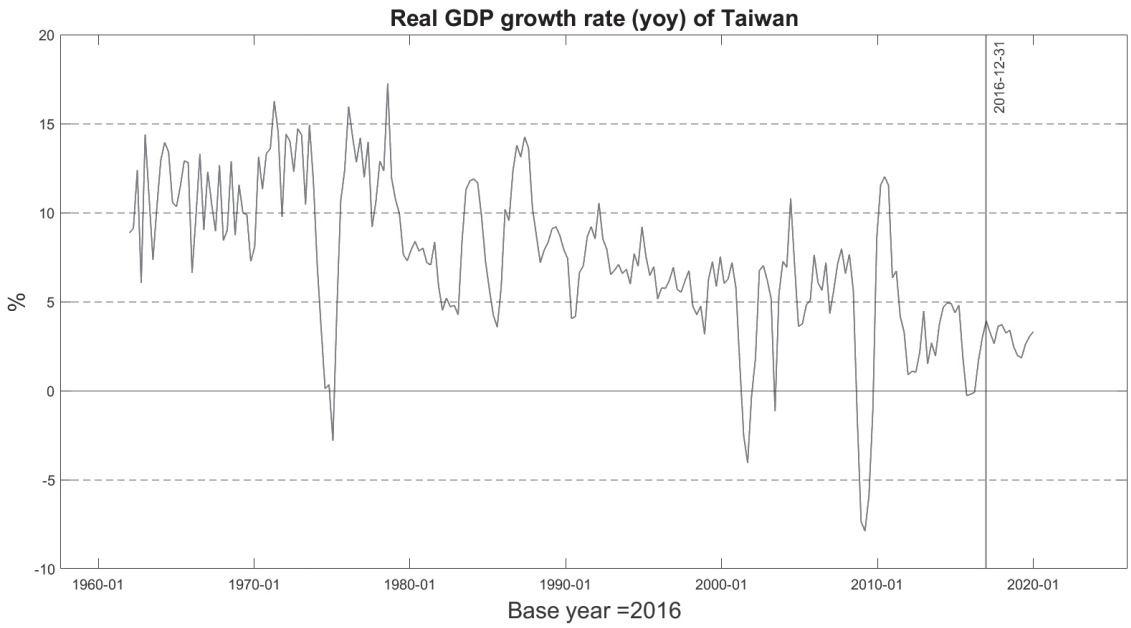
Chakraborty and Joseph (2017) 為英格蘭銀行所做的機器學習研究報告，可說是開風氣之先，在金融機構的資產負債表檢測示警、通貨膨脹率、以及科技創投公司的籌資模式等領域影響深遠，其結論指出RNN與微調的SVM，預測績效超越時間序列模型。SVM方法已在預測總體經濟數據、金融危機事件和能源消費方面有卓越的表現。部份研究則認可深度學習的autoML架構在預測電力消費或經濟成長率的表現，因為該方法預測誤差較線性模型低。也有一些研究將RF應用在預測經濟成長率、美國消費者物價膨脹率、或是金融資產風險溢酬。也有文獻肯定LSTM方法應用在住宅道路、交通及股市指數預測方面的表現。上述成果也是激發我們以臺灣經濟成長率為題進行嘗試的主因，因為過去文獻偏重以計量方法預測臺灣經濟成長率，以機器學習進行者尚付之闕如。

本文欲彌補這部份之不足，因此我們將採用SVM、autoML和LSTM等方法。

至於相應的計量研究方法，本文依據Hyndman et al. (2018) 時間序列預測模型集合挑選以下模型，這些曾受學界肯定的方法包括：(1) Autoregressive Integrated Moving Average (ARIMA. Ediger and Akar, 2007)，(2) Self-Exciting Threshold Autoregressive (SETAR. Feng and Liu, 2003)，(3) logistic STAR (LSTAR. Stock and Watson, 1998)，(4) Generalized Additive Models (GAMs. Serinaldi, 2011)，(5) Bootstrapping aggregation (bagged. Zhao et al., 2017b)，以及 (6) Neural network (NN. 常被運用於匯率預測)。

如圖1所示，本文採用支出面並以2016年為基期的實質GDP成長率，樣本期間為1962年第一季至2019年第四季，並將之分成兩個子樣本期間，一是1962年第一季至2016年第四季，以及2017年第一季至2019年第四季，分別以時間序列模型和機器學習方法進行預測。至於預測方法，本文執行單步 (one-step) 以及多步預測 (multi-step prediction)。雖然多數時間序列模型使用最大概似估計法 (maximum likelihood estimation) 進行單步預測之表現優異，尤其是樣本內數據 (in-sample data，亦即訓練數據 training data)；但實務上樣本外數據 (out-of-sample data，亦即測試數據 test data) 的預測必須是多步預測。由於臺灣經濟成長率為季資料，單步預測僅在低頻樣本內歷史數據有良好績效。不同於單步預測，多步預測使得政策制定者得以展望未來進行評估。如文獻所述 (Hamzaçebi et al., 2009; Kline, 2004)，遞迴方法 (recursive) 是最具直覺性的預測方法，即先以模型預測下一期的值，依據該期的值再預測下下一期的值，再使用解釋變數真實值的落後期，從預測樣本的第一期，計算樣本外預測 (如Saad et al., 1998)。再者，面對季資料樣本數不足的問題，前述Charkraborty and Joseph (2017) 以60季開始，用逐次增1的方法遞迴更新，樣本外只採用8季。因此在執行多步預測方面，本研究亦使用遞迴和直接預測方法。

圖1　臺灣經濟成長率時間序列



資料來源：中華民國統計資訊網。

本文綜整相關資料後，分別以傳統時間序列和機器學習分析，預測我國經濟成長率並評估其績效。我們使用多個模型設定，考量包括趨勢模型、線性模型、非線性模型、包含固定趨勢的線性 (機器學習) 模型，與包含季節性虛擬變數的線性 (機器學習) 模型，執行樣本單步和多步預測與績效評估。雖然部份研究認可機器學習方法應用在時間序列預測的績效，但本文結論則與上述文獻不盡相同，茲將實證結果重點臚列如下：

首先，就單步預測而言，統計時間序列方法，尤其是ARIMA和bagged的模型表現很好，傳統時間序列與機器學習方法在此處不分上下。

其次，採行多步動態遞迴預測評估，以樣本外預測績效而言，autoML表現最佳。以整體時間序列表現來說， SVM的4種設定優於autoML和以RNN深度學習的LSTM模式。因此，在多步動態遞迴預測方面，機器學習方法優於時間序列方法。

其三，在混合落後結構和添加其他變數時，三個機器學習模型均有相當優異的預測績效，且以autoML為最佳。而在整體時間序列結果上， LSTM難以有效捕捉樣本內高峰或低谷，autoML在某些情形下表現良好，SVM則相對有穩定良好表現。

最後，以機器學習方法改善時間序列預測雖屬可行，但仍有很大發展空間。就資訊科學的角度思考，只要硬體演算能力夠強大，用autoML對資料直接進行大規模計算，

找出最佳資料結構，再使用預測平均法，應是現階段改善時間序列預測較為有效的方式。

本文結構如次：第2節說明統計時間序列與機器學習模型的設定，及其績效評估方法；第3節敘述臺灣經濟成長率資料型態與預測方法；第4節將二種模型延伸搭配AR(1)進行預測；第5節簡述其他延伸模型的結果，細節囿於篇幅可參考委託研究報告；第6節綜合結論並提出預測模式的建議。

貳、統計方法與機器學習模型設定

令 y_t 為經濟成長率的時間序列，如式(1)為簡單帶有漂移項 (drift) 的隨機漫步模型 (random walk model)：

$$y_t = \bar{c} + y_{t-1} + \varepsilon_t, \quad (1)$$

其中 ε_t 呈常態分配。季節性的隨機漫步模型如式 (2)：

$$y_t = \bar{c} + y_{t-m} + \varepsilon_t, \quad (2)$$

其中 m 表示季節性的落後項。

h 期預測則為 $y_{t+h} = \bar{c} \cdot h + y_t$。

一、統計方法

(一) 差分整合移動平均自我迴歸模型 (autoregressive integrated moving average model, ARIMA)

ARIMA模型依據自我相關，當前預期價格與該前期價格是線性相關的，ARIMA模型的限制在於選擇AR項數和MA項數的適當值時，需要仔細觀察ACF和PACF圖，也需要更深入理解ARIMA模型在統計學實作的概念。只要滿足了選擇AR項數和MA項數適當值的條件，ARIMA可視為一個良好的模型，詳見Tsay (2010) 第2章的說明。

我們採用AIC的自動選擇程序，以最大落後階次為5，最大差分階次為2的條件，搜尋最適ARIMA結構，標示符號為auto.arima。在這個架構之下，我們也額外添加外生變數：確定趨勢、季節虛擬變數和捕捉頻率無關的週期傅立業 (Fourier)，週期傅立業項取2做傅立業週期函數展開。

另外，我們也添加分量自我迴歸 (Quantile AutoRegression, QAR)，採取Koenker and Xiao (2006) 的作法，標示符號為QAR(p)。

(二) 非線性移轉模型：SETAR和LSTAR

我們使用兩個移轉模型：SETAR和LSTAR，前者是自我激勵門檻自我迴歸模型 (self-exciting threshold autoregression) 的簡稱。SETAR(p) 模型如式 (3)：

$$y_t = \begin{cases} \mu_1 + \rho_{1,1} y_{t-1} + \cdots + \rho_{1,p1} y_{t-p1} + \varepsilon_t & \text{if} \quad x_{t-d} \le \theta_{m-1} \\ \mu_2 + \rho_{2,1} y_{t-1} + \cdots + \rho_{2,p2} y_{t-p2} + \varepsilon_t & \text{if} \quad \theta_{m-1} < x_{t-d} \le \theta_{m-2} \\ \vdots & \\ \mu_m + \rho_{m,1} y_{t-1} + \cdots + \rho_{m,pm} y_{t-pm} + \varepsilon_t & \text{if} \quad \theta_1 < x_{t-d} \end{cases}, \tag{3}$$

其中參數如次：

m : 狀態的數量

$\mu_1 \ldots \mu_m$: 每個狀態的截距

$p_{j,1} \ldots p_{j,m-1}$: 在狀態j的落後期數量

$\theta_1 \ldots \theta_{m-1}$: 門檻值

d : 移轉變數的延遲項數

x_{t-d} : 門檻 (移轉) 變數

該模型稱為SETAR在於 $x_{t-d} = y_{t-d}$, 亦即AR 項本身是門檻變數。我們將理論上無限的狀態數量限制為2或3，如高、中、低等三個狀態，因此該模型可套用如式 (4) 的簡易型式：

$$y_t = \begin{cases} \mu_L + \rho_{L,1} y_{t-1} + \cdots + \rho_{L,pL} y_{t-pL} + \varepsilon_t & \text{if} \quad \theta_L \le x_{t-d} \\ \mu_M + \rho_{M,1} y_{t-1} + \cdots + \rho_{M,pM} y_{t-pM} + \varepsilon_t & \text{if} \quad \theta_L < x_{t-d} \le \theta_H \\ \mu_H + \rho_{H,1} y_{t-1} + \cdots + \rho_{H,pH} y_{t-pH} + \varepsilon_t & \text{if} \quad \theta_H < x_{t-d} \end{cases}, \tag{4}$$

LSTAR是羅吉斯平滑門檻自我迴歸模型 (logistic smoothing threshold autoregression)，納入一個羅吉斯平滑AR(p) 建置

$$\begin{aligned} y_t = {} & (\mu_L + \phi_{L,1} y_{t-1} + \cdots + \phi_{L,p} y_{t-p} + \varepsilon_t) \cdot G(z_t, m, \gamma) \\ & + (\mu_H + \phi_{H,1} y_{t-1} + \cdots + \phi_{H,p} y_{t-p} + \varepsilon_t) \\ & \cdot (1 - G(z_t, m, \gamma)) \end{aligned}$$

其中 $G(\cdot)$ 為羅吉斯函數，而 z_t 為門檻變數 (以 x_t 加上延遲項數 m，即 $z_t = x_{t-m}$)。

(三) GAMs:

一般化的加成模型 (generalized additive models, GAMs) 依據線性平滑模型做修正。依照迴歸設定，GAMs具有以下型式：

$$\begin{aligned} E(y_t \mid x_{1t}, x_{2t}, \ldots . x_{pt}) = {} & \alpha + f_1(x_{1t}) + f_2(x_{2t}) \\ & \cdots + f_p(x_{pt}), \end{aligned} \tag{5}$$

其中 f_j 是未指定的平滑 (非參數性) 函數。如欲展開基底函數建立模型，則結果由最小平方法算出。GAMs則是每個函數由散佈點平滑法配置，如三次樣條函數 (cubic spline) 或核平滑函數 (kernel smoothing function)，通過這種方式，GAMs具有充分彈性包含任何線性和非線性成分的組合。在時間序列應用方面，Wood and Augustin (2002) 延伸懲罰迴歸樣條方法 (penalty regression splines approach) 以便更進一步利用一般化樣條平滑法 (generalized spline smoothness, GSS) 的好處。本文參照Wood and Augustin (2002) 的方法，採用一般化的廣義加成AR(p) 模型預測經濟成長率。

(四) BATS (Box-Cox transform, ARMA, Trend & Seasonality)

Taylor (2003) 延伸Hold-Winters的線性版本，並且包含二個季節性成份的模型如式(6)

$$\begin{aligned} y_t &= L_{t-1} + b_{t-1} + s_t^{(1)} + s_t^{(2)} + e_t, \quad i = 1, 2 \cdots T \\ L_t &= L_{t-1} + b_{t-1} + \alpha \cdot e_t \\ b_t &= b_{t-1} + \beta \cdot e_t \\ s_t^{(1)} &= s_{t-m_1}^{(1)} + \gamma_1 \cdot e_t \\ s_t^{(2)} &= s_{t-m_2}^{(2)} + \gamma_2 \cdot e_t \end{aligned} \quad , (6)$$

其中 m_1 和 m_2 為季節性周期，e_t 是代表干擾項的白噪音變數。L_{t-1} 和 b_{t-1} 分別代表在時間 t 的 y_t 截距和趨勢，s_t 是季節性成分，係數 α，β，γ_1 和 γ_2 則是平滑參數，$\{s_{1-m_1}^{(1)}, \ldots, s_0^{(1)}\}$ 和 $\{s_{1-m_1}^{(2)}, \ldots, s_0^{(2)}\}$ 代表起始狀態變數。

上述模型有幾個缺點，一是不穩定的非線性，另一是尚未被認可的複雜季節性走勢。為了解決這些問題，De Livera et al. (2011) 主張採用BATS方法預測具有複雜季節性走勢的時間序列，該方法由其四個成分命名，分別為**B**ox-Cox 轉置，**A**RMA，趨勢 (**T**rend) 和季節性 (**S**easonality)。Box-Cox 轉置是一個廣泛被接納的轉換法，因為它可以穩定時間序列的變異數。BATS定義如式 (7)

$$y_t^{(\lambda)} = L_{t-1} + \phi b_{t-1} + \sum_{i=1}^{T} s_{t-m_i}^{(i)} + e_t \quad (7)$$

$$\begin{aligned} y_t^{(\lambda)} &= \begin{cases} \dfrac{y_t^{\lambda} - 1}{\lambda}, \lambda \neq 0 \\ \log(y_t), \lambda = 0 \end{cases} \\ L_t &= L_{t-1} + \phi b_{t-1} + \alpha \cdot e_t \\ b_t &= (1-\phi) b_{t-1} + \phi b_{t-1} + \beta \cdot e_t \\ s_t^{(i)} &= s_{t-m_i}^{(i)} + \gamma_i \cdot e_t \end{aligned} \quad ,$$

其中 e_t 符合ARMA(p, q) 型態。

(五) BAGGED (Bootstrap Aggregation)

與BATS相似，Bergmeir et al. (2016) 為指數平滑的bootstrap aggregation提出一個技術方法，因此取部份字母簡稱為bagged，並為預測績效帶來顯著改進。BATS依照STL分解使用Box-Cox 轉換法，將時間序列分成三個部分：趨勢、季節性以及殘餘項。

指數平滑的概念，在於較諸早期的觀察值，以最近的觀察值做預測是更適宜的，這表示最近的觀察值應該賦予更高的權重。在本文實證中，ETS和ARIMA(p,d,q)自動最佳階次演算 (auto.arima) 皆可用bagged方法強化。此案BAGGED.ETS指用bootstrap aggregation強化的ETS，BAGGED.autoArima指用bootstrap aggregation強化的auto.arima。ETS在文獻上有兩種通用名稱：一是Error, Trend, Seasonal；二是ExponenTial Smoothing。有關ETS細節，請參考Hyndman and Athanasopoulos (2018) 第7章。

(六) 簡易人工類神經網絡 (Simple Artificial Neural Network)

類神經網絡方法本身是包括隱藏層 (hidden layers) 和規模節點 (size nodes) 的非線性設置 (以下簡稱NNETAR)。該方法是前饋 (feed-forward) 類神經網絡，且使用時間序列落後值作投入，細節可參考Hyndman and Athanasopoulos (2018) 的第11章。

多層類神經網絡可以想像成一大串的合

成函數 (composite functions)

$$x \to f(x) \to g(f(x)) \to h(g(f(x))) \to \cdots\cdots \to y$$

也就是說，函數 f, g, h 是由演算法依據資料點對點型態，模擬神經系統運作所產生。整個認知網路的產生很複雜，在類神經網絡的演算法中，函數也就由一串一串神經元節點 (neural nodes) 的認知 (感知) 系統所生成，也稱為layer function。類神經網絡方法有三項特色：(i) 在一個多層前饋網絡中，節點的每一層接收前一層的投入；(ii) 結合使用線性組合將投入置於每個節點；(iii) 在輸出前，以非線性函數修改結果。

使用非線性函數傾向減少極端投入值，使得網絡不受離群值影響且仍是穩健的，人工類神經網路特性亦有三項： (i) 權重以隨機值開始，接著隨觀察值更新；(ii) 預測中包含隨機，所以網絡由不同的隨機起始點訓練，結果是經由平均而來；(iii) 必須在事前將隱含層以及每個隱含層的節點數設定好。

此外我們也使用了Franses and Van Dijk (2000) 提出的簡單類神經網絡模型，該模型是以一個隱藏層和線性產出的非線性AR呈現：

$$y_{t+s} = \beta_0 + \sum_{j=1}^{D} \beta_j \cdot h\left(\delta_{0j} + \sum_{i=1}^{m} \delta_{ij} y_{t-(i-1)d}\right), \quad (8)$$

其中D是延遲順序 (delay order) 而m是嵌入維度 (embedding dimension)。為求簡化以下該模型簡稱為NNET。

除了標準的類神經網絡之外，機器學習文獻通常使用循環類神經網絡 (RNN)方法，而屬於RNN的長短期記憶模型 (LSTM) 則廣泛應用在預測時間序列。

二、機器學習方法

(一) SVM

近年來，機器學習方法蓬勃發展，SVM已經被認為是淺層機器學習方法，其角色類似統計方法的線性迴歸，但重要性卻不容忽視。SVM 概念由 Vapnik (2013) 建立，不同於統計學習方法使用的實證風險最小化原則 (empirical risk minimization, ERM)，將訓練資料數據最小化，SVM體現將結構風險最小化 (structural risk minimization, SRM) 的原理，將一般化誤差的上限最小化，此一差異使得SVM方法較不易受到極端值影響，並可以達成非線性的資料配適與預測，因此具備較高的應用潛力。

SVM是將資料序列模型化，具有強大與充分彈性的研究方法。該方法起源於分類問題，類似線性規劃，找到一個能完美切割資料點的超平面以將樣本資料分類，但亦可以應用在迴歸、預測時間序列等課題。支援向量迴歸 (Support Vector Regression, SVR) 隱含的理論建立在SVM分類模型概念上，相較於線性迴歸使用最小平方法將平方誤差總和 (sum of squared errors, SSE) 最小化以尋找迴歸係數估計值，SVR是嘗試在既定的預測誤差水準與對超過誤差水準的離群值的容忍

度下，找到一曲線 (或超平面) 去配適樣本資料。因為SVR可以調整對離群值的容忍度，並且可以達成非線性的配適，使用SVR可以得到較線性迴歸更具彈性的資料配適與預測。

雖然SVM和多數機器學習，看似是對類別變數作分類的計算，但其分類概念皆可直接用於連續資料，類似統計的連續資料所產生的信賴區間，統計依照條件期望值和標準差建構信賴區間，然後取一個size (0.05) 把被解釋變數歸成兩類。我們的SVM架構基本上有AR(*P*)，據此可以產生動態預測，依次添加時間確定趨勢和季節虛擬變數矩陣。

(二) 自動化機器學習autoML

我們採用H2O.ai平臺提供的自動機器學習和大數據公開資源解決方案。H2O.ai是專業雲端運算平臺，平臺上可使用由H2O.ai公司開發的深度學習人工智慧組件 (例如H2O.glm即是在H2O環境建立一個一般化線性模型)，組件中包括尖端機器學習演算法、績效矩陣與附屬函數等，使得機器學習方法強大而不失簡潔。我們在H2O.ai平臺上嘗試以多個機器學習模式進行演算，包括：監督深度學習 (類神經網絡)、隨機森林 (random forest)、一般化線性模型、梯度提高機器 (gradient boosting machine)、素樸貝式 (naïve Bayes)、疊積集成 (stacked ensembles) 等等。以上模式在訓練和驗證兩個子樣本之間經大規模演算後取出各模型的最佳結果，最後以疊積強化 (stacked ensembles) 合併出最適的預測(代表符號為autoML)，在機器學習中此方式也稱為委員會方法 (committee approach)，類似統計的預測平均法 (forecasting average)。也就是說，autoML求得的預測數列並非基於單一種機器學習模型，而是多模型組合模式下的綜合結果。

autoML是目前機器學習的典型架構，它會依照資料的時間序列結構，產生大量的特徵 (features) 作為解釋變數，然後根據這些特徵資料，進行大規模的分類演算。例如前述SVM我們可添加AR(*P*) 以產生動態特徵，但autoML已有時間特徵虛擬變數，我們就不需添加諸如趨勢和季節等額外虛擬變數。

(三) 具LSTM 效果的RNN

深度學習方法的應用場域是型態辨識 (pattern recognition)，不是為了時間序列預測所開發。不過辨識只是從所掃描的圖形萃取特徵，再和資料庫特徵比對，它其實就是資料驅動的預測。因此若應用在時間序列預測，就需要透過歷史資料訓練，建一個型態資料庫，然後比對產生預測。而深度學習的類神經模式，對於層級和節點的函數，是由演算法根據資料的複雜關聯產生。

要理解長短期記憶模型 (LSTM) 需要先理解遞迴類神經網路 (Recursive Neural Networks, RNNs）的概念，簡易類神經網路的預測路徑 (資料輸入與輸出) 純粹是單向的，但對某些深度學習與預測問題而言，資

料的出現可能有高度的前後相關性。以經濟成長率預測為例，若造成經濟成長率變動的因素（例如生產力衝擊或信用危機）具有持續性，則本期經濟成長率的預測值很大一部分受到前期預測值影響，但傳統的前饋型類神經網絡卻無法做到持久性想法。從資料的角度，保留短期記憶是避免局部特徵被全面資料給稀釋 (可理解為平均趨勢稀釋了資料的局部特徵，例如線性迴歸將所有資料以單一斜率配置)，因此對經濟成長預測有幫助。RNN對此議題主張「有著迴圈的網絡使訊息得以持久」，將前期產生的資訊投入到後期的深度學習程序中。

比較傳統機器學習和深度學習，後者其實是一種更複雜的神經網路。RNNs近年在文獻中被大量使用，並應用於如語音辨識、語言模型、翻譯、圖像描述等等，且各種應用仍持續有所進展，可參閱Karpathy (2015) 的詳盡說明。[註1] RNN網路架構特性使其本身具有記憶性，故已成功應用在時序數據的處理上。應用在預測經濟成長率時，投入 x_t (向前一期經濟成長率及其他變數) 並透過類神經網絡，可預測出Y_t (經濟成長率預測)。

理論來說RNN可以處理長期記憶依賴，人們謹慎選取參數，進而以這種形式解係數估計值，但模型可能出現無法學習的現象，稱之為梯度消失問題 (vanishing gradient problem)，這在梯度變得相當大或相當小時出現，造成輸入資料庫結構長期全距依賴性 (long-range dependencies) 模型化的困難。此時最具效率性的方法是使用DL4J支持下RNN的LSTM變異，也被稱為LSTM網絡。

本文SVM，LSTM也將添加AR(P) 以及趨勢和季節等額外虛擬變數，並於模型的合理值範圍內做grid search，由RMSE最小值來判斷最適層級與節點數。

三、評估模型的預測績效

令 y 和 $\hat{y}$ 分別代表真實值和預測值，我們應用四個精確衡量指標，就預測誤差 $e_t=y_t-\hat{y}_t$ 評估預測準確性。衡量指標有多個，Makridakis (1993) 針對預測指標穩定性的研究，指出 Mean Absolute Prediction Error (MAPE) 相對穩定。Hyndman and Koehler (2006) 則建議 Mean Absolute Scaled Errors (MASE)。因本研究為小樣本，scaled的處理沒有差異，因此採取多個指標綜合判斷，敘述如下：

(一) 兩個尺度依存誤差 (two scale-dependent errors)

首先為RMSE=$\sqrt{mean(e_t^2)}$，$mean$ 代表平均值，e_t 為誤差，其次是平均絕對誤差 MAE=$mean(|e_t|)$，

(二) 百分比誤差 (percentage errors)

令 $p_t=\frac{y_t-\hat{y}_t}{y_t}$，平均絕對誤差率 (mean absolute percentage errors) 定義MAPE=$mean(|p_t|)$，

(三) Theil's U

用來評估預測準確性的Theil's U

$$\text{Theil's U}=\sqrt{\frac{\sum(y_t-\hat{y}_t)^2}{\sum y_t^2}},$$

(四) 殘差項在落後一期的自我相關

殘差項在落後一期的自我相關 (autocorrelation of errors at lag 1, ACF1) 衡量預測誤差的持久性，以下列ACF公式中的係數 ϕ 表示：

$$e_t=\phi e_{t-1}+\varepsilon_t,$$

其中係數 ϕ 亦衡量殘差項隨時間衰減的速度，若 ϕ 值近於1則預測誤差較持久，意味著該模型預測力較差，反之則表示預測誤差將迅速衰減至零。

另外，這些預測績效指標在一些研究會展現樣本變異性 (sampling variations)，即隨著抽樣數增加，指標的相對變化 (例如RMSE/MAPE)。此作法在機器學習的 k-fold 交叉驗證 (cross validation, CV) 也十分常見，但因本文案例樣本數不多，我們沒有設計不同的訓練和測試時段，而將重心放在比較前述三個機器學習模式，在各種投入變數組合下的預測表現，藉以探索作為時間序列預測方法的可行性，故僅計算預測誤差及其預測績效。

參、資料與預測方法

行政院主計總處依照國民經濟會計制度 (System of National Accounts, SNA) 規範，分成支出面、生產面以及消費面，按季編製臺灣國內生產毛額 (gross domestic product, GDP) 及其成長率，衡量期間內的生產活動表現。如表1所示，我們採用支出面的實質GDP (以2016年為基期) 成長率，將1962年第一季至2019年第四季，再分成1962年第一季至2016年第四季，以及2017年第一季至2019年第四季等兩個子樣本期間。在預測方面，本文分別以傳統時間序列和機器學習方法，為臺灣實質GDP成長率進行預測。

本預測模型樣本內期間為1962年第一季至2016年第四季，樣本外期間為2017年第一季至2019年第四季。樣本內期間為訓練樣本 (training sample)，由此樣本配適出合理的模型與係數。樣本外期間則是測試樣本 (testing sample)，兩個子樣本期間的時間趨勢圖及敘述統計如前述圖1和表1所示：圖1呈現兩段資料的時間序列走勢圖，可比較其波動幅度。期間臺灣經濟成長率大多為正值，在1975年受到石油危機衝擊，從15%跌至約-2%，隨後反彈並在1970年代末期達到將近17.3%的最高水準，其後逐漸趨緩。2001年和2003年分別因科技類股泡沫和SARS，經濟成長率呈現負值，2009年由於全球金融危機，經濟負成長近-7.9%。2010年升至近12%，之後多維持在 0% 至 5% 之間。由表

1則可發現：(1) 第一段子樣本期間經濟成長率平均值、中位數與標準差皆高於第二段子樣本期間；(2) 在偏態方面，兩段子樣本皆尚稱對稱；在峰態部分，第一段樣本期間為高峻峰，第二段為低闊峰。

表1　臺灣經濟成長率之敘述統計

經濟成長率(%)

	1962Q1-2016Q4	2017Q1-2019Q4
平均數	7.49	3
中位數	7.26	3.23
標準差	4.38	0.65
偏度	-0.55	-0.39
峰度	0.74	-1.19
最小值	-7.88	1.84
最大值	17.26	3.92
樣本數	219	13

資料來源：中華民國統計資訊網；本研究自行整理。

由於使用「訓練、驗證、測試」三期作法，資料量必須夠大，故我們的方法設計如圖2所示：1962Q1-2016Q4為訓練期，2017Q1-2019Q4為測試期，不另外採用驗證期的原因在於演算法在訓練期都已內建驗證。SVM採用10-fold 交叉驗證，autoML和LSTM採用5-fold 交叉驗證，因此訓練期產生的模式就是驗證後的模式。在機器學習設計中，訓練期會learning by iterating直到選出最佳模型。

圖2　本文機器學習訓練及測試期

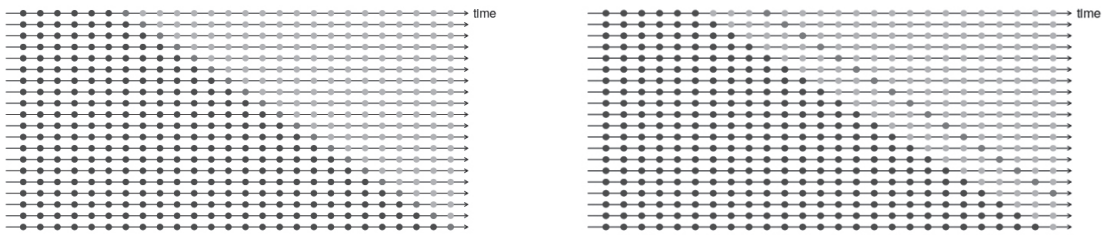
臺灣經濟成長率共232季的觀察值，用 (6:2:2) 標準分割對傳統時間序列模型是足夠的，但對目前冠上學習演算法之名的autoML 或 LSTM，至少需要250個觀察值，因為資料的序列分段，會導致樣本偏少而訓練不足。此處採用預測樣本外12季，也是遷就下的選擇。兩種資料滾動方式如圖3 (a) 與 (b) 所示：Chakraborty and Joseph (2017) 的英格蘭銀行研究樣本外只用8季，以60季開始用逐次增1的方法 recursively update，如圖3 (a) 的rolling。我們沒有使用rolling樣本遞增方式的原因，在於本文沒有要探索精確指

標如MAPE的樣本變異性，就訓練階段所採用的CV，用一筆最長資料 (1962Q1-2016Q4) 訓練出來的結果就足以預測，不需要分段rolling。Chakraborty and Joseph (2017) 因為沒有採用自動化機器學習做大規模強化預測，所以可小規模學習。就一般機器學習而言，小樣本或許只是訓練不足，但是就自動化機器學習而言，往往是訓練不良，容易產生過度配置。因此在達到本研究目的考量下，我們沒有做他種拆分。

本文模型除了相關虛擬變數如季節和趨勢，均會使用至少一階AR。因此除了單步 (one-step) 靜態預測，另增加了多步 (multi-step) 預測。在樣本內期間訓練數據，亦即由數據配適出最合適的模型，再進行單步與多步預測。單步預測亦稱靜態預測，是將t+2的預測，建立在t+1期的資料。多步預測在文獻中有遞迴 (recursive) 和直接 (direct) 兩種方法。

圖3　遞迴預測方法示意圖

(a) 單步滾動　　(b) 多步滾動



單步預測是將t+2的預測，建立在t+1期的預測，步驟如下：(1) 估計樣本內相關參數；(2) 樣本外第1期，使用前1期解釋變數資料，計算樣本外第2期預測；(3) 依序再代入向前2…N期解釋變數真實值，與樣本內資料估計係數，產生被解釋變數的向前2…N期預測值。亦即利用解釋變數真實值的落後期，從預測樣本的第1期，計算樣本外預測。多步預測步驟如下：(1) 估計樣本內相關參數；(2) 樣本外第1期，使用前1期的預測，計算樣本外第2期預測；(3) 用第2期的預測值，計算第3期的預測值；(4) 重複步驟 (2) 及 (3)，依序產生向前2…N期被解釋變數預測值。

以前期的預測值再進行未來期數值的預測，稱為「動態多步預測」。遞迴多步是一步一步遞迴而成多步。直接多步則直接依據預測期估計模型。例如要預測未來4期，可以直接估計一個 $y_t=f(y_{t-4})$，然後帶入(y_{t-3}, y_{t-2}, y_{t-1}, y_t)。因此實際上的模式更為複雜，因為 $y_t=f(y_{t-5})$ 也可以達到同樣的預測目的。所以在直接落後中，落後期是向量，不是純量。如

同 Bontempi et al. (2013) 指出，機器學習的直接多步預測，針對一個預測期h，要解一個模型m_h的多種組合，演算十分龐大。我們後續皆採取直接預測，做大規模運算。

肆、時間序列模型與機器學習模型搭配AR(1) 的預測表現

一、單步靜態預測

本文比較多個時間序列模型與機器學習模型的單步靜態預測績效，以RMSE、MAE、MAPE、Theil's U和ACF(1) 來衡量，結果如表2及圖4。圖4 (a)-(d) 繪出各模型2017年第一季之後的預測能力比較，以RMSE、MAE、MAPE、Theil's U，顯示各模型預測正確性。若綜合考量其結果，時間序列模型優於機器學習方法，其中時間序列的bagged模型和ARIMA(2,1,2)(2,0,1) 相關設定模型最佳，機器學習中考量趨勢項的SVM:AR(1) 與純粹SVM:AR(1) 模型次之，autoML:AR(1) 模型殿後，再來則是虛擬季節性變數和趨勢項的SVM:AR(1)，而LSTM:AR(1) 模型表現較差。然而Theil's U指出autoML:AR(1) 模型最佳， bagged模型次之，接著為LSTM:AR(1) 和ARIMA (2,1,2)(2,0,1) 之家族設定模型居中，SVM:AR(1) 家族模型排名位居後段。

圖5為單步預測誤差的ACF(1) 值，顯示預測誤差的持續性，並指出機器學習中虛擬季節性變數和趨勢項的SVM:AR(1)、虛擬季節性變數的SVM:AR(1)、LSTM:AR(1)，以及時間序列模型中考量趨勢項的ARIMA (2,1,2)(2,0,1) 模型，預測誤差ACF(1) 值最小。表現居後者為機器學習的autoML:AR(1)，加入趨勢項的SVM:AR(1)，時間序列方法ARIMA (2,1,2)(2,0,1) 的設定模型和bagged模型。前述四個指標表現較佳的時間序列模型，如ARIMA(2,1,2)(2,0,1) 模型和bagged模型，在ACF(1) 值的表現較差，而在四個指標表現居於後段的機器學習模型SVM:AR(1)、LSTM:AR(1) 和autoML:AR(1)，則在ACF(1) 表現較佳。

綜合前述，各機器學習模型與時間序列模型的預測表現上，前者以考量趨勢項的SVM:AR(1) 與autoML:AR(1) 模型較佳，後者則是bagged模型和ARIMA(2,1,2)(2,0,1) 家族模型。但就整體而言，時間序列模型預測表現和機器學習模型沒有太大差距，這樣我們就要檢視ACF(1) 係數。根據圖5，前10名都沒有太大的ACF(1) 值。因此在單步預測，機器學習和時間序列模型皆有不錯表現。

最後檢視單步預測的整體時間序列表現。圖6至圖8比較機器學習模型，包括SVM四個模型設定，autoML和LSTM的單步預測值 (含2016年第四季後的樣本外預測值)。圖

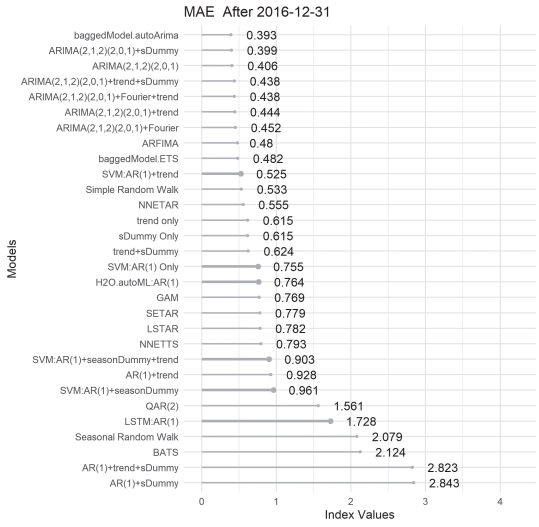
6顯示SVM四個模型中，AR(1) 和含季節性虛擬變數之AR(1) 等模型，不僅在捕捉樣本內序列最好，樣本外預測表現亦最佳。

比較圖7的autoML以及圖8的LSTM，以2000年後的期間而言，本文發現LSTM不論是在樣本內或樣本外預測表現都是較好的。即使受到金融海嘯的衝擊，該機器學習模型在2009年的低谷，與2010年因基期較低而導致的大幅回升，都能夠進行良好的配適。在單步預測方面，SVM在所有機器學習方法中的預測績效最好。綜合上述，如果預測單位只需要單步預測，則傳統時間序列模型就可以有很好的表現，不必浪費時間去執行冗長的計算，此與Makridakis and Hibon (2000) 和Morlidge (2014) 一致。

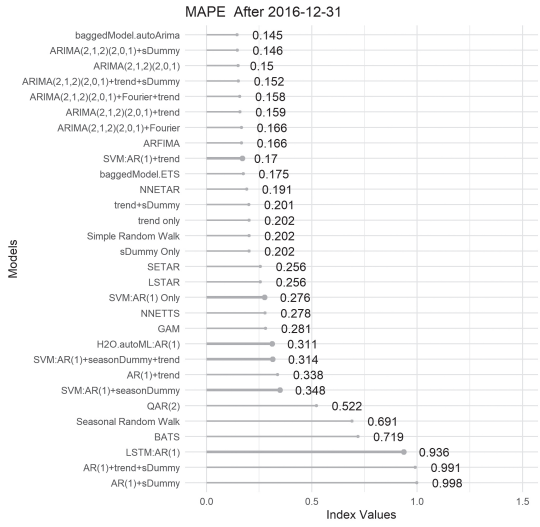
表2　單步預測績效

	RMSE	MAE	MAPE	ACF1	Theil's U
Simple Random Walk	0.624	0.533	0.202	0.511	0.0116
Seasonal Random Walk	2.488	2.079	0.691	0.067	0.0344
trend only	0.696	0.615	0.202	0.432	0.0107
sDummy Only	0.696	0.615	0.202	0.432	0.0107
trend+sDummy	0.709	0.624	0.201	0.405	0.0101
AR(1)+trend	1.104	0.928	0.338	0.159	0.0159
AR(1)+sDummy	3.004	2.843	0.998	0.138	0.0445
AR(1)+trend+sDummy	2.981	2.823	0.991	0.124	0.0444
ARIMA(2,1,2)(2,0,1)	0.472	0.406	0.150	0.024	0.0087
ARIMA(2,1,2)(2,0,1)+trend	0.502	0.444	0.159	-0.147	0.0086
ARIMA(2,1,2)(2,0,1)+sDummy	0.458	0.399	0.146	0.048	0.0086
ARIMA(2,1,2)(2,0,1)+trend+sDummy	0.502	0.438	0.152	$-6\cdot10^{-4}$	0.0086
ARIMA(2,1,2)(2,0,1)+Fourier	0.550	0.452	0.166	0.062	0.01
ARIMA(2,1,2)(2,0,1)+Fourier+trend	0.524	0.438	0.158	0.086	0.0098
baggedModel.ETS	0.592	0.482	0.175	0.293	0.0085
baggedModel.autoArima	0.468	0.393	0.145	0.262	0.0074
NNETAR	0.649	0.555	0.191	0.263	0.0116
BATS	2.684	2.124	0.719	0.111	0.0282
ARFIMA	0.544	0.480	0.166	-0.076	0.0095
LSTAR	0.947	0.782	0.256	0.040	0.0135
SETAR	0.940	0.779	0.256	0.018	0.0135
GAM	0.905	0.769	0.281	0.094	0.0138
NNETTS	0.920	0.793	0.278	0.070	0.0141
QAR(2)	2.582	1.561	0.522	0.400	0.0466
SVM:AR(1)+seasonDummy	1.068	0.961	0.348	-0.204	0.018
SVM:AR(1)+trend	0.622	0.525	0.170	0.037	0.01
SVM:AR(1)+seasonDummy+trend	0.967	0.903	0.314	-0.245	0.0175
SVM:AR(1) Only	0.892	0.755	0.276	0.015	0.0136
autoML:AR(1)	0.937	0.765	0.311	0.018	0.0055
LSTM:AR(1)	2.027	1.729	0.936	-0.095	0.0087

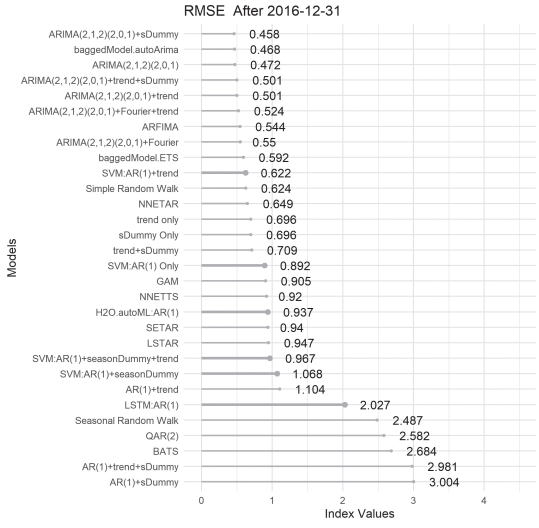
圖4　單步預測正確性：4個指標



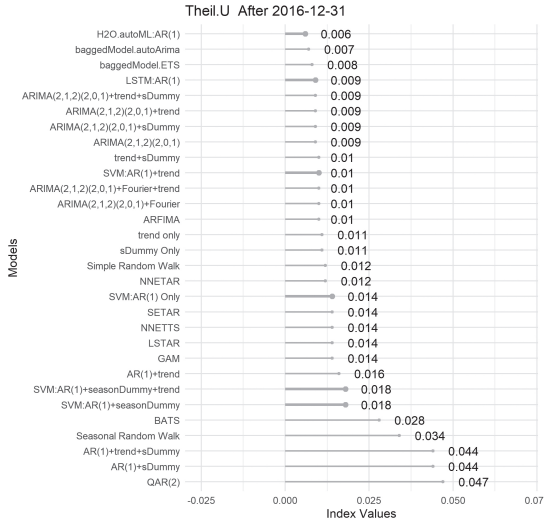
(a)



(b)



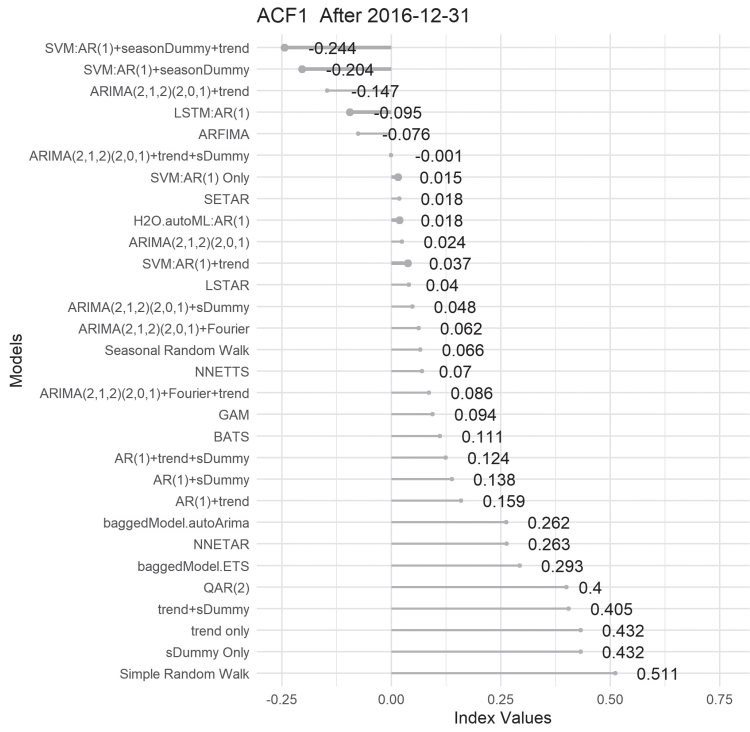
(c)



(d)

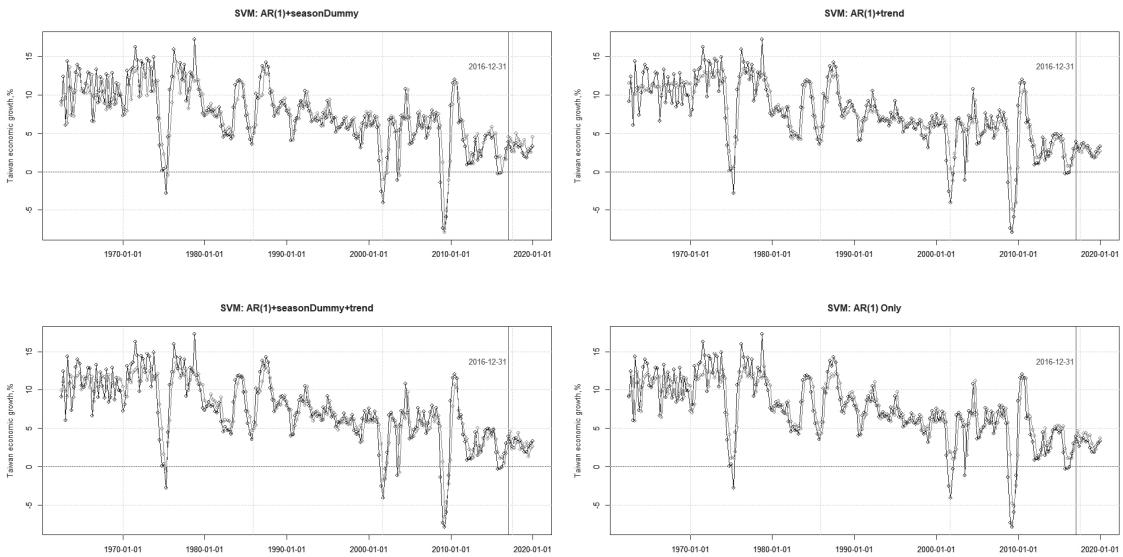
資料來源：本研究自行整理。

圖5 單步預測正確性：ACF(1) 值



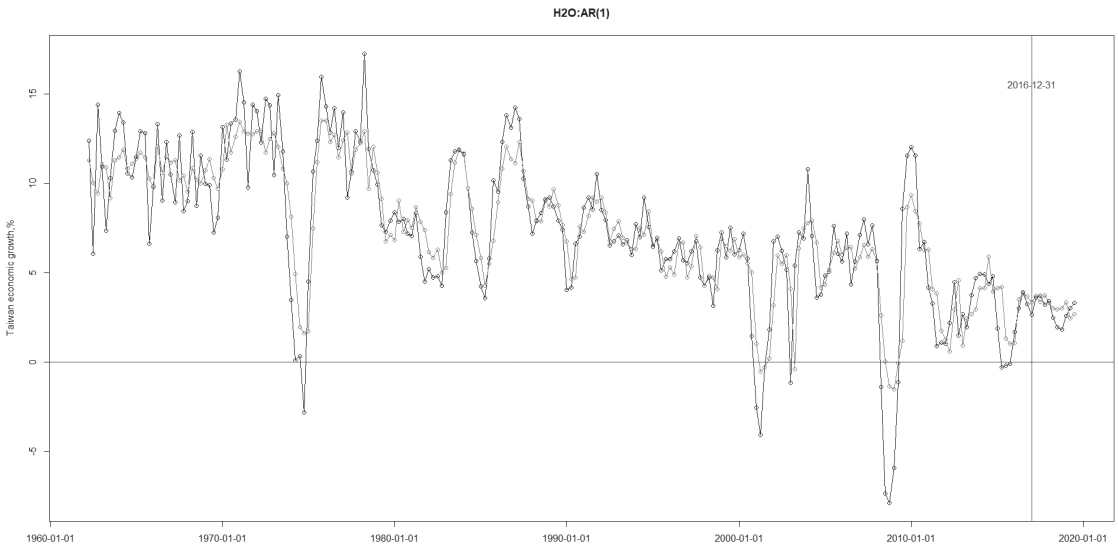
資料來源：本研究自行整理。

圖6 單步預測的整體時間序列表現：SVM 4個設定



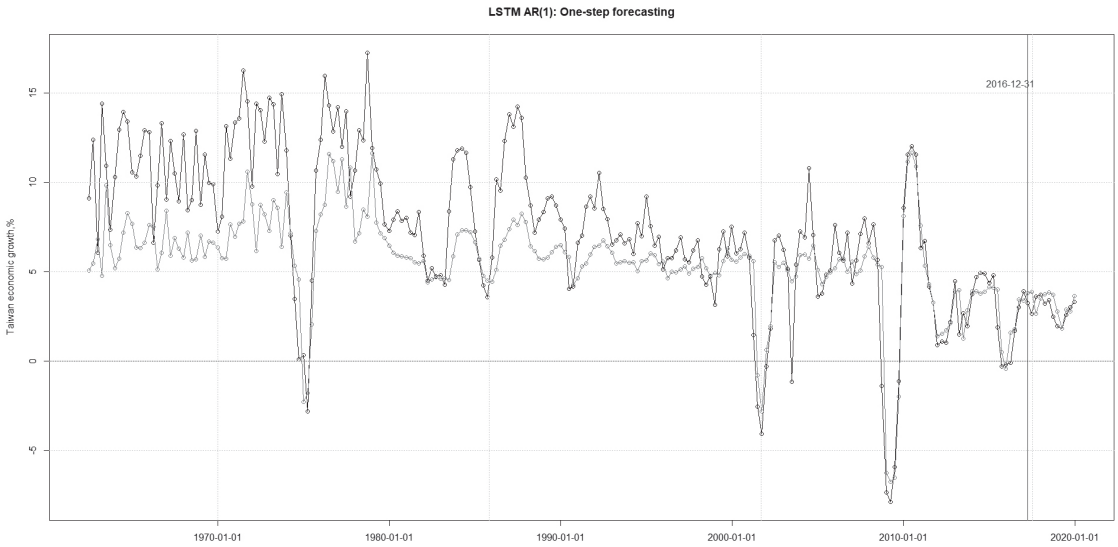
資料來源：本研究自行整理。

圖7　單步預測的整體時間序列表現：autoML



資料來源：本研究自行整理。

圖8　單步預測的整體時間序列表現：LSTM



資料來源：本研究自行整理。

二、多步動態遞迴預測

依據前述架構，本文也進行2016年第四季後多步動態遞迴預測並評估其表現，所有模型預測績效指標，按表現優劣排列如圖9及10。

圖9 (a)-(d) 顯示在預測正確性方面，autoML multistep:Recursive 在RMSE、MAPE、MAE及Theil’s U上均有名列前茅的表現，bagged模型以及 ARIMA (2,1,2)(2,0,1) 相關設定模型則次之，再來是考量趨勢項和季節性虛擬變數的SVM: AR(1)。SVM:AR(1) 家族的其他設定模型排名居中，並且位居LSTM multistep:Recursive之前。但是由數據來看，前10名的預測誤差相距甚小，因此我們繼續檢視圖10的預測誤差持續性。

就圖10的前幾名來看，加入趨勢項和季節性虛擬變數的SVM:AR(1)，autoML multistep:Recursive，以及ARIMA (2,1,2)(2,0,1) 相關設定模型等三個模式的ACF(1) 係數介於0.232-0.389，持續預測錯誤的表現亦佳。然而SVM:AR(1) 其他三個設定的ACF(1) 係數較大，在所有模型位居中間與較後端排名，這表示SVM設置不正確，就難以產生良好的預測模型。至於LSTM multiStep:Recursive排名則與SVM: AR(1) 其他三個設定模型差異不大。再者，在多步動態遞迴預測的LSTM模型，其ACF(1) 係數較大且排名較後段，這與單步預測的ACF(1)呈現結果相反。

最後檢視多步動態遞迴預測的整體時間序列表現，本文將機器學習下的預測值與真實時間序列描繪於圖11至圖13，其中預測值在2016年第四季前為樣本內預測值，2016年第四季後則為樣本外多步預測值。

首先在SVM四個設定中，如圖11所示，可發現以下重要結果：(1) 與單步預測一致，加入趨勢項及季節性虛擬變數的AR(1)模型，在樣本內的預測表現與在樣本外的預測表現比其他模型好；(2) SVM其他三個設定，在捕捉樣本內的1970年代的能源危機和2001年科技類股泡沫化的表現較弱，預測值在樣本外期間偏離真實值較多。

其次，根據圖12與圖13，autoML multistep:Recursive模型在樣本內與樣本外的表現，與LSTM multistep:Recursive 預測值與經濟成長率真實值走勢有些許落差，然而前者對樣本外預測有較好的掌握。相較於單步預測，autoML多步動態遞迴預測在捕捉樣本內2008年金融海嘯期間，及樣本外的2016第四季後表現更佳。LSTM在捕捉樣本內1970年代能源危機，和2008年金融海嘯期間表現比autoML更佳，但是在樣本內其他期間和樣本外期間預測表現皆出現較大的偏離。

綜合上述，如果預測單位需要執行臺灣經濟成長率的多步動態遞迴預測，使用機器學習模型是較好的選擇，此結果呼應既有文

獻對機器學習方法的肯定。三種方法以SVM最佳，autoML次之，最後是LSTM。

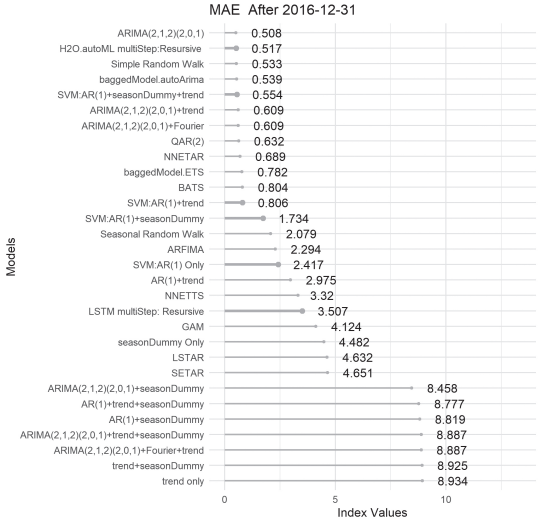
本研究結果亦發現在單步預測下，傳統時間序列模型ARIMA和bagged模型表現很好，但多步預測時機器學習方法即優於時間序列方法。可能的解釋原因，是多步預測運用前期預測值進行未來預測，而機器學習的優勢在於透過大量訓練找出最適參數，訓練過程可減少前期預測誤差，呈現較佳的預測績效。單步預測運用前期真實值進行未來預測，時間序列方法因估計參數較少且自由度高，故能得到較佳的預測績效。

表3　多步動態遞迴預測績效

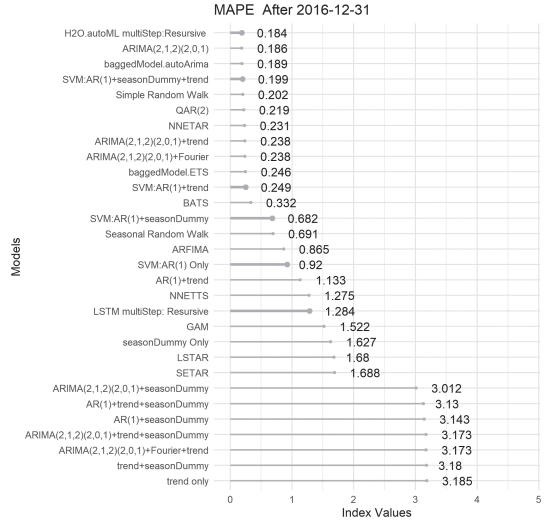
	RMSE	MAE	MAPE	ACF1	Theil's U
Simple Random Walk	0.624	0.533	0.202	0.511	0.012
Seasonal Random Walk	2.488	2.079	0.691	0.067	0.034
trend only	8.952	8.935	3.185	0.430	0.157
seasonDummy Only	4.528	4.482	1.627	0.476	0.082
trend+seasonDummy	8.944	8.925	3.181	0.389	0.157
AR(1)+trend	3.275	2.975	1.133	0.681	0.065
AR(1)+seasonDummy	8.837	8.819	3.143	0.427	0.155
AR(1)+trend+seasonDummy	8.797	8.777	3.130	0.385	0.155
ARIMA(2,1,2)(2,0,1)	0.585	0.508	0.186	0.492	0.011
ARIMA(2,1,2)(2,0,1)+trend	0.783	0.609	0.238	0.473	0.014
ARIMA(2,1,2)(2,0,1)+seasonDummy	8.475	8.458	3.012	0.442	0.148
ARIMA(2,1,2)(2,0,1)+trend+seasonDummy	8.911	8.887	3.173	0.389	0.157
ARIMA(2,1,2)(2,0,1)+Fourier	0.783	0.609	0.238	0.473	0.014
ARIMA(2,1,2)(2,0,1)+Fourier+trend	8.912	8.887	3.173	0.389	0.157
baggedModel.ETS	0.897	0.783	0.246	0.511	0.013
baggedModel.autoArima	0.606	0.539	0.189	0.432	0.010
NNETAR	0.799	0.689	0.231	0.569	0.012
BATS	1.015	0.804	0.332	0.579	0.021
ARFIMA	2.433	2.294	0.865	0.396	0.047
LSTAR	4.740	4.632	1.680	0.215	0.086
SETAR	4.757	4.651	1.689	0.236	0.086
GAM	4.286	4.124	1.522	0.407	0.081
NNETTS	3.725	3.320	1.275	0.732	0.075
QAR(2)	0.723	0.632	0.219	0.524	0.011
SVM:AR(1)+seasonDummy	2.001	1.734	0.683	0.665	0.041
SVM:AR(1)+trend	0.927	0.806	0.249	0.437	0.014
SVM:AR(1)+seasonDummy+trend	0.657	0.554	0.199	0.232	0.011
SVM:AR(1) Only	2.609	2.417	0.920	0.563	0.051
autoML multiStep:Resursive	0.607	0.517	0.185	0.274	0.010
LSTM multiStep: Resursive	3.563	3.507	1.285	0.511	0.066

資料來源：本研究自行整理。

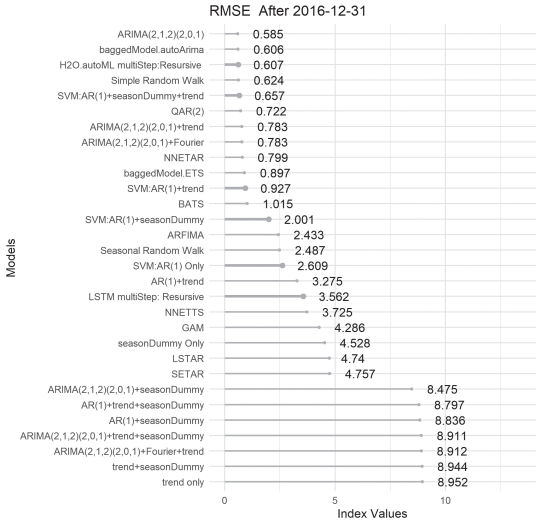
圖9　多步動態遞迴預測的整體時間序列表現：4個指標



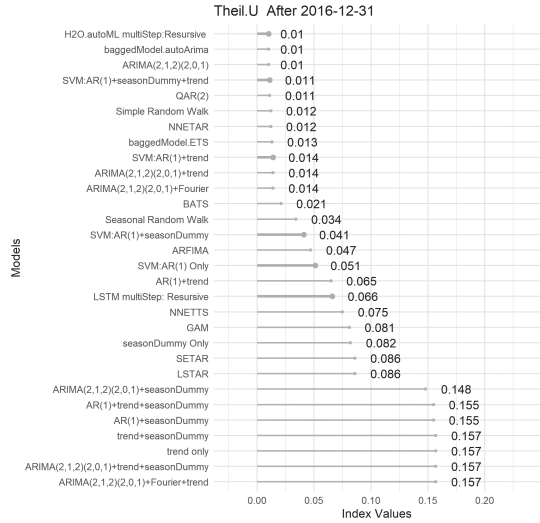
(a)



(b)



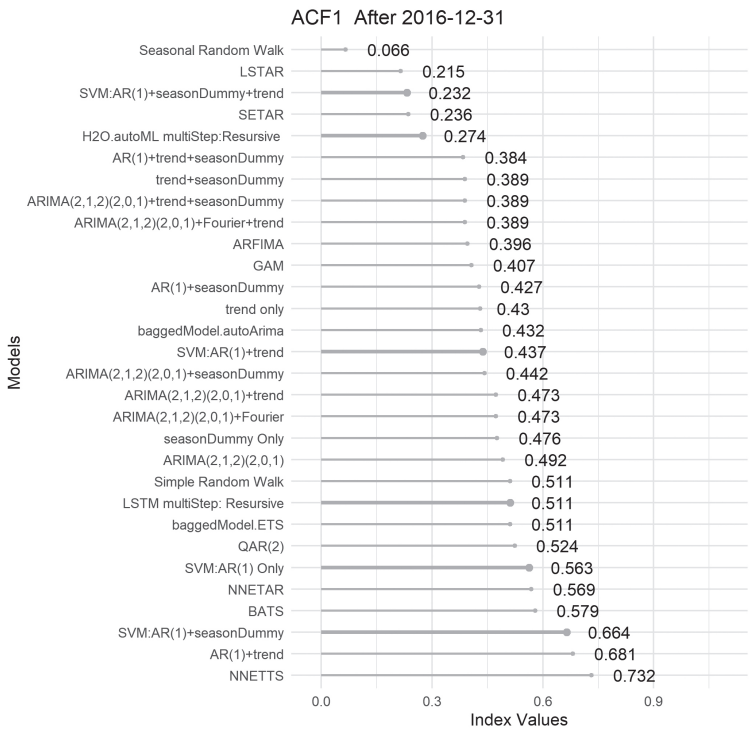
(c)



(d)

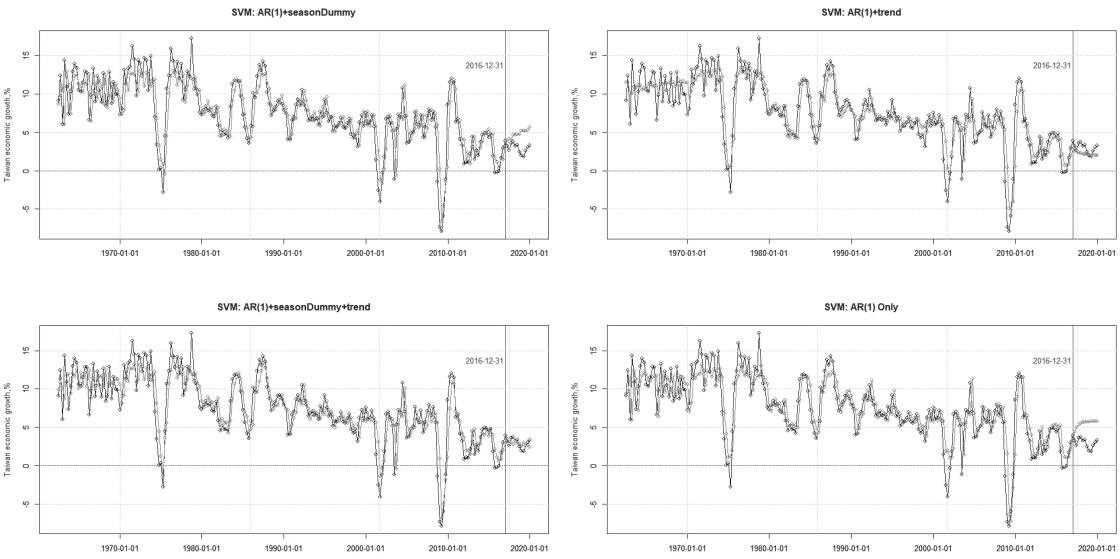
資料來源：本研究自行整理。

圖10　多步動態遞迴預測正確性：誤差的ACF(1)值



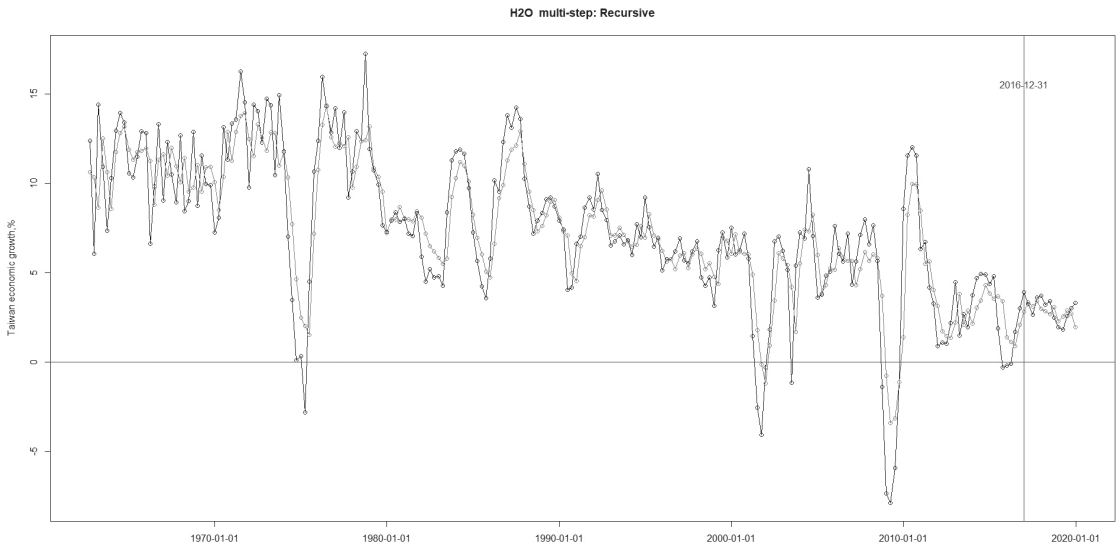
資料來源：本研究自行整理。

圖11　多步動態遞迴預測的整體時間序列表現：SVM 4個設



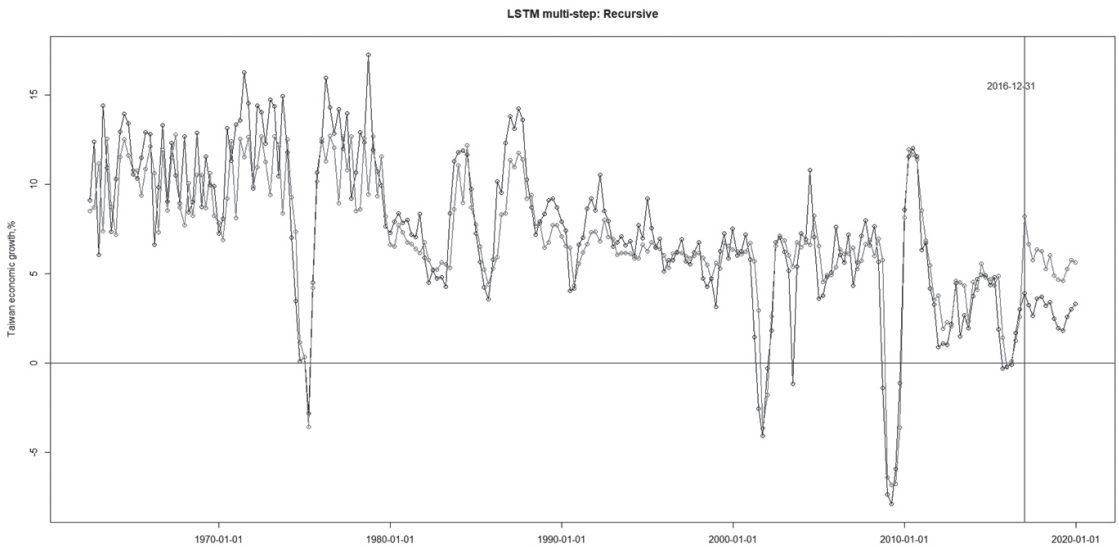
資料來源：本研究自行整理。

圖12　多步動態遞迴預測的整體時間序列表現：autoML



資料來源：本研究自行整理。

圖13　多步動態遞迴預測的整體時間序列表現：LSTM



資料來源：本研究自行整理。

伍、其他模型測試結果簡述

本文囿於篇幅，無法一一詳述所有測試，細節請參考研究報告內容，在此僅報導各項結果。[註2] 我們首先將之前的單變量AR(1) 模式，分別延伸為混合落後結構與添加其他變數等二種，繼之論述所有變數的混合落後結構，預測平均法，及以機器學習診斷時間序列模型預測誤差等議題。

一、混合落後結構

令 F 代表機器學習的演算模式，對被解釋變 (y) 的連續落後，我們沿用AR寫法$\mathbf{L}^p$

$$y_t = F(\mathbf{L}^p(y_t)) = F(y_{t-1}, y_{t-2}, ...y_{t-p}) + e_t ,$$

對解釋變數X的連續落後，我們用$\mathbf{L}^h$

$$y_t = F(\mathbf{L}^h(X_t)) = F(X_{t-1}, X_{t-2}, ...X_{t-h}) + e_t ,$$

對非連續落後，我們用k

$$y_t = F(\mathbf{y}_{t-k_1}; \mathbf{x}_{t-k_2}) + e_t$$
$$k_1, k_2 \in \{(1,2),(1,3),(1,4),(2,3),\cdots(1,2,3,4)\}$$

其他變數X中，採用和經濟成長率期間相同的總經數據。

以自我相關落後模型進行2016年第四季後多步預測的表現評估，整體而言機器學習模型的預測誤差都相當低，各類機器學習模型均有非常優異的預測表現。從 MAE、MAPE、RMSE、Theil's U 指標均發現，autoML均具有最佳表現，SVM其次，LSTM第三。此外，當連續落後期模型改為混合落後期時，預測績效均可再改善，由此可知採用混合落後期，有機會改善預測績效。

在預測誤差持續性方面，整體而言ACF(1) 絕對值與選用的機器學習模型無明顯關連。此外亦發現連續落後期模型 (如k=(1,2,3) 或k=(1,2,3,4)) 在刪去1或2個落後期後，在大部分情況下將使ACF(1) 絕對值增加，預測誤差持續性加大。

檢視預測的整體時間序列表現：首先，autoML預測的連續落後期模型發現，相較於前述圖7的AR(1) 模型，k=(1,2) 在樣本內與樣本外的預測值與真實值接近，顯示增加落後期數可提升預測表現。然而不論何種設定，捕捉樣本內1974-1975年與2009年高峰或低谷的表現均不甚佳，但增加落後期數 (k=(1,2,3) 及k=(1,2,3,4)) 之後，捕捉高峰或低谷表現將可提升。另外，預測值的波峰或波谷發生時點會較晚於真實值。[註3] autoML的非連續落後結果，大多數情況而言，k=(1,2,3) 及k=(1,2,3,4) 模型在刪去1或2個落後期後，不論是樣本內的趨勢預測，或是對高峰或低谷資料的預測能力均變差，與真實值差距變大。唯一特例是，樣本內趨勢及高峰或低谷的預測，在k=(1,3,4) 時的表現優於k=(1,2,3,4)。

其次，LSTM連續落後期模型整體時間序列預測，與圖8的AR(1) 模型相比，加入落後期數越多 (由k=(1,2) 至k=(1,2,3,4))，樣

本內與樣本外的預測表現越良好，捕捉樣本內1974-1975年與2009年高峰或低谷的表現明顯改善。但若資料出現高頻率波動如1970年初，LSTM將難以預測。在非連續落後模型預測方面，當連續落後期模型刪去1或2個落後期，LSTM預測時間序列的特性更明顯。LSTM受到歷史資料記憶影響，預測值的高點及低點往往受限於一個範圍內，如k=(2,3)、k= (2,4)、k= (3,4)。

最後，SVM的連續落後模型整體時間序列預測，不論在整體趨勢或捕捉樣本內高峰或低谷之表現上，均較前二種機器學習模型為佳。尤其在捕捉樣本內的1974-1975年與2009年高峰或低谷之表現，預測值已能逼近真實值。與圖6的AR(1) 模型相比，則發現增加落後期數後 (由k=(1,2) 至k=(1,2,3,4))，樣本外預測值更接近真實時間序列。SVM在不連續落後期下，顯示整體時間趨勢及捕捉樣本內高峰或低谷表現優LSTM模型，且預測值有明顯落後於真實值的情形。

綜合上述，三個機器學習模型均有相當優異的預測績效，且以autoML為最佳。而在整體時間序列結果上， LSTM難以有效捕捉樣本內高峰或低谷，autoML在某些情形下表現良好，SVM則相對有穩定良好表現。當改為不連續落後期模型後，有機會提升預測績效，但對高峰或低谷資料的預測能力均變差，與真實值差距變大，並且預測值明顯有落後於真實值的情形。

非連續落後期模型在時間序列ARMA架構下並不常見，然而機器學習實務上旨在萃取可預測訊息，並不需要遵從AR。本節運用非連續落後期數下的自我相關落後模型，在經濟意義解釋上，因經濟衝擊對經濟成長率的影響有時間落後，故非連續落後期數含有可預測資訊。事實上類似的問題在ARIMA也有，即處理長期記憶時間序列的ARFIMA (F表fractionally)，此情形下ARIMA(p,d,q) 差分項 d 不一定是整數。

二、經濟成長率的AR(P) 和添加其他變數的AR(P)

令$y_t = F(\mathbf{L}^p(y_t, \mathbf{X}_t)), p = 1,2,3,4$，

此處投入變數範例如下

$y_t = F(y_{t-1}, y_{t-2}, X_{t-1}, X_{t-2}), p = 2$，

其他變數X：臺灣經濟變數和國外經濟成長率

臺灣經濟數據因遷就時間，只有以下部份變數起始季和經濟成長率相同，其餘變數則因資料太短無法採用，例如短期利率始自1987Q2，長期利率始自1995Q4，通貨膨脹率始自1982Q1。使用的四筆國內資料與來源如下：(1) 商品及服務輸出：1961Q1-2019Q4，主計總處；(2) 商品及服務輸入：1961Q1-2019Q4，主計總處；(3) 工業生產指數：1961Q1-2019Q4，經濟部工業局；(4) 新臺幣兌美元匯率：1961Q1-2019Q4，中央銀行。

對應經濟成長率，上述工業生產指數與新臺幣兌美元匯率資料係由月頻率轉成季頻率，新臺幣兌美元匯率換頻率時，採用平均 (average) 和期末 (end-of-period) 兩種方式，故以上共有5筆資料。變動率計算時，使用當季對上年同季的年變動率。國外數據則為美國、OECD及G7的經濟成長率。雖然OECD和G7的集團概念都包括美國，但是兩者也不盡然相同。其他變數共8個，包括5個臺灣變數變動率加3個外國經濟成長率。

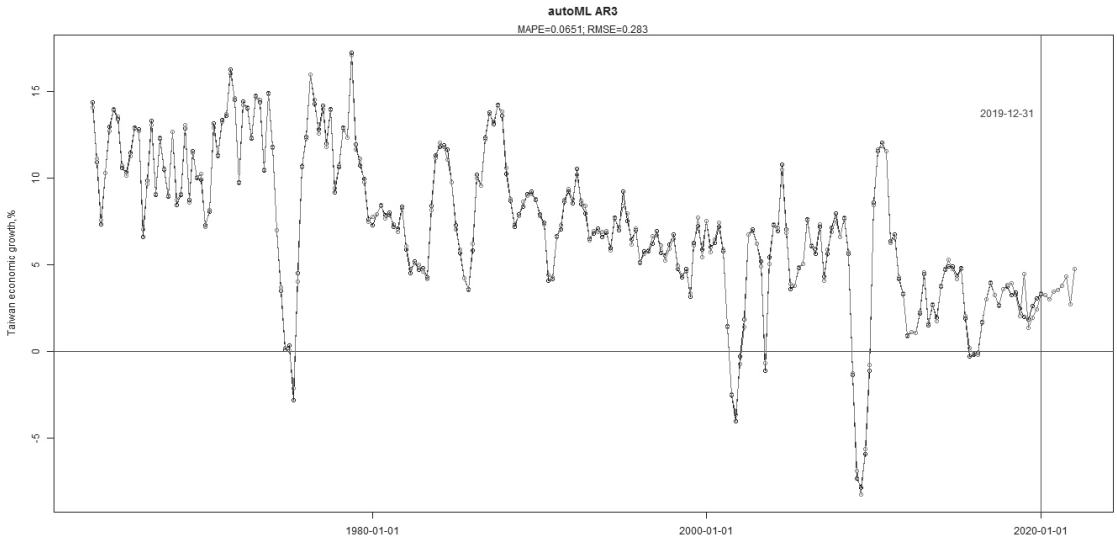
從預測精確度指標看來，LSTM雖然在深度學習享有盛名，但是本文時間序列的樣本太短，導致其訓練不足，無法展現其預測優勢。其次，最佳ARIMAX的結果並不差，和SVM與autoML相比，顯示機器學習方法不盡然佔據絕對優勢。本節各模型可產生大量圖形，我們使用三個模式 (分別是SVM，autoML和傳統時間序列模型) 預測未來8季 (2020年第1季至2021年第4季) 的經濟成長率。附帶說明的是，機器學習模型都是從歷史資料訓練的學習，對於空前的事件衝擊難以有效預測，例如2020前兩季受新冠疫情影響，是前所未有的外生衝擊，因此預測與實際必有相當偏差。本文以autoML 經濟成長率的AR(P) 預測為範例，如表4與圖14所示。

表4　經濟成長率預測：autoML AR(P)

樣本外預測 (%)				
	P=1	P=2	P=3	P=4
2020Q1	2.868	2.974	3.230	3.361
2020Q2	2.718	2.765	2.979	3.192
2020Q3	2.792	2.714	3.465	3.157
2020Q4	3.953	3.092	3.514	3.104
2021Q1	3.728	3.379	3.774	3.455
2021Q2	4.408	3.028	4.308	3.026
2021Q3	4.505	2.818	2.692	3.286
2021Q4	4.446	4.915	4.762	4.795
樣本內訓練績效				
RMSE	1.613	1.653	0.283	2.536
MAE	1.149	1.230	0.186	1.832
MAPE	0.463	0.568	0.065	0.983
ACF1	0.220	0.257	-0.238	0.535
Theil's U	0.0075	0.0100	0.0015	0.0211

資料來源：本研究自行整理。

圖14　autoML 預測2020Q1-2021Q4：以精確指標最好的AR(3)為例



資料來源：本研究自行整理。

三、所有變數的混合落後

當我們遇到很多投入變數時，有兩個解決途徑：其一是逐次清算，計算量大；其二是降維，計算量少。降維度的方法可以採用主成分 (Principal Component Analysis, PCA) 或獨立成分法 (Independent Component Analysis, ICA)。如果資料量很大，降維或許是一個選擇，但本文嘗試PCA和ICA的結果並不理想，因此採用逐次清算作法，理由如下：

1. 可以檢視變數組合的關鍵。如果逐一清算可以發現少量變數，其實就不需要ICA降維。降維度的合理性，也可以由逐一清算的方式得到支持；
2. 本文有明確的預測變數 (經濟成長率)。機器學習的預測，不是單方面對投入變數獨自降維即可一體適用。換言之降維也必須多次計算，適合SVM的不一定適合LSTM。

$y_t = F(\mathbf{L}^{\mathbf{k}_1}(y_t); \mathbf{L}^{\mathbf{k}_2}(X_t))$, $\mathbf{k}$ 集合定義如前，此處投入變數範例如下

$$y_t = F(y_{t-1}, y_{t-4}; X_{t-2}, X_{t-4}).$$

賡續前述，多步預測在演算上更大的負擔，就是如何決定解釋變數的混合落後。為避免演算規模過於龐大，此處解釋變數X只選取3個變數：OECD經濟成長率 (x_1)，OECD領先指標成長率 (x_2)，和美國領先指標成長率 (x_3)，合併臺灣經濟成長率 (y) 自我相關落後期數。[註4] 落後期 $\mathbf{k}_1, \mathbf{k}_2 = 1,2,3,4$，臺灣經濟成長率自我落後結構有15種組合 $\mathbf{k}_1 \in \{(1,2),(1,3),\cdots.(1,2,3,4)\}$。另外，在給定

$\mathbf{k}_1$下，x_1, x_2, x_3 共有12個落後變數。從組合算數，這樣的變數交錯有4,095組

$$\sum_{k=1}^{12} C_k^{12} = 4,095 ,$$

為了簡化我們的運算，12種狀況中，前11種每一抽樣6個落後變數組合，最後一種 C_{12}^{12} 全取，總計66+1=67種模型。為了簡化過多的計算，此處只選擇計算較方便的SVM。15組自我落後組合，每組自我落後結構的X有67種隨機組合，總計算15×67=1,005種投入變數組合。

結果簡單摘要如表5，可知如果數據大而無當，對預測結果未必有所助益。經過這樣的運算，與前面純粹的自我相關模型相比，添加解釋變數的確能夠改善預測表現，只不過要從大量的變數中選出有用的，需要大規模計算。

本文旨在找出預測績效最佳的機器學習模型及變數組合，Makridakis (1993) 針對預測指標穩定性的研究，指出MAPE相對穩定；Hyndman and Koehler (2006) 則建議採scaled error的MAE。我們得到的最佳結果為RMSE=0.51，MAPE=0.157。受限於經濟成長率為季資料，資料長度短，自由度有限，因此計算力弱。倘能改善資料長度問題，應能有更佳預測表現。由此可見，機器學習的原則係資料驅動，與建構一個經濟模型下的因果關係不同。選擇變數前並沒有特別的經濟考量，僅從資料的相關性來判斷，因此不限於經濟理論模型所討論的控制變數。此外，機器學習也透過資訊型態配適出合適的模型及參數，故可有最佳的預測績效。

表5 SVM 混合落後自我相關並增加其他變數模型的多步預測正確性

MAPE	SVM 演算變數組合
0.157	$F(y_{t-1}, y_{t-3}; x_{2t-1})$
0.164	$F(y_{t-1}, y_{t-2}; x_{1t-2}, x_{1t-4}, x_{2t-3})$
0.176	$F(y_{t-1}, y_{t-2}, y_{t-3}, y_{t-4}; x_{1t-3}, x_{1t-4}, x_{2t-1})$
0.180	$F(y_{t-1}, y_{t-3}, y_{t-4}; x_{1t-1}, x_{1t-2}, x_{1t-4}, x_{2t-1}, x_{2t-2})$
0.181	$F(y_{t-1}; x_{1t-3}, x_{1t-4})$
0.190	$F(y_{t-1}, y_{t-2}, y_{t-3}; x_{1t-1}, x_{2t-1}, x_{3t-1}, x_{2t-3}, x_{2t-4}, x_{3t-2})$
0.201	$F(y_{t-1}, y_{t-4}; x_{1t-2}, x_{1t-3}, x_{1t-4}, x_{2t-1}, x_{2t-3}, x_{3t-1})$
0.206	$F(y_{t-1}, y_{t-2}, y_{t-4}; x_{1t-1})$
0.21	$F(y_{t-2}; x_{1t-2}, x_{1t-3}, x_{2t-2}, x_{2t-3}, x_{2t-4})$
0.24	$F(y_{t-2}, y_{t-3}; x_{1t-1}, x_{1t-3}, x_{2t-2})$
RMSE	**SVM 演算變數組合**
0.51	$F(y_{t-1}, y_{t-3}; x_{2t-2})$
0.562	$F(y_{t-2}, y_{t-3}, y_{t-4}; x_{1t-3}, x_{2t-1}, x_{2t-2})$
0.591	$F(y_{t-1}, y_{t-2}; x_{1t-2}, x_{1t-4}, x_{2t-3})$
0.592	$F(y_{t-1}, y_{t-2}, y_{t-3}, y_{t-4}; x_{1t-4}, x_{2t-1}, x_{2t-3})$
0.66	$F(y_{t-1}; x_{1t-1}, x_{1t-4}, x_{2t-1})$
0.67	$F(y_{t-2}; x_{1t-1}, x_{1t-3}, x_{2t-2})$
0.69	$F(y_{t-1}, y_{t-2}, y_{t-3}; x_{1t-1}, x_{2t-1}, x_{2t-2}, x_{2t-2}, x_{3t-1}, x_{3t-2})$
0.73	$F(y_{t-1}, y_{t-2}, y_{t-4}; x_{1t-1})$
0.733	$F(y_{t-1}, y_{t-4}; x_{1t-2}, x_{1t-3}, x_{1t-4}, x_{2t-1}, x_{2t-3}, x_{3t-1})$
0.778	$F(y_{t-2}, y_{t-4}; x_{1t-3}, x_{2t-2}, x_{2t-4})$

資料來源：本研究自行整理。

本文也執行1,005組autoML的混合運算。表6的預測績效指標顯示其預測績效更好，MAPE或RMSE最小值的下降幅度相當明顯。以下我們將說明，計算這1,005組的預測平均值，將可呈現出更好的預測結果。

表6　autoML混合落後自我相關預測正確性

最小指標	autoML演算變數組合
MAPE=0.071	$F(y_{t-2}, y_{t-3}, y_{t-4}; x_{1t-1}, x_{2t-2}, x_{2t-4}, x_{3t-4})$
RMSE=0.29	$F(y_{t-1}, y_{t-2}; x_{1t-1}, x_{1t-3}, x_{1t-4}, x_{2t-1}, x_{2t-2}, x_{3t-1}, x_{3t-3}, x_{3t-4})$

資料來源：本研究自行整理。

四、預測平均法 (Forecasting Average)

機器學習係在訓練過程中，根據驗證方法選擇最佳結果，然而實際上很難知道什麼是最好的預測。計量經濟學有一個模型平均法 (model average)，也就是將多個模式的預測予以平均，如Hansen (2007)、Hansen and Racine (2012) 和Liao and Tsay (2020)。在機器學習文獻中，稱之為委員會方法 (committee approach)，見Bishop (2009) 及Hastie et al. (2009)。多模型預測平均法的構思如下：因為模型產生的預測，相對於真實值不是高估就是低估，因此取其平均或中位數，就很可能會提高精確度。

為簡化起見，此處不從最適多步預測平均 (optimal multistep forecasting averaging) 的文獻方向討論理論、檢定與組合法，我們只計算兩種機器學習的簡單平均。第1部份以台灣經濟成長率自我落後結構的15種組合設定所產生的45個預測結果，計算cross-sectional平均和中位數作為預測值，再跟真實的經濟成長率對照計算精確度。從表7可知道，autoML的15個預測結果的平均 (autoML.mean) 或中位數(autoML.median)，可以大幅增加預測精確度。LSTM和SVM表現較差，但是當45個模型放一起時 (All.mean和All.median)，整體績效還是提升很多。

表7　各種組合的平均預測精確度指標

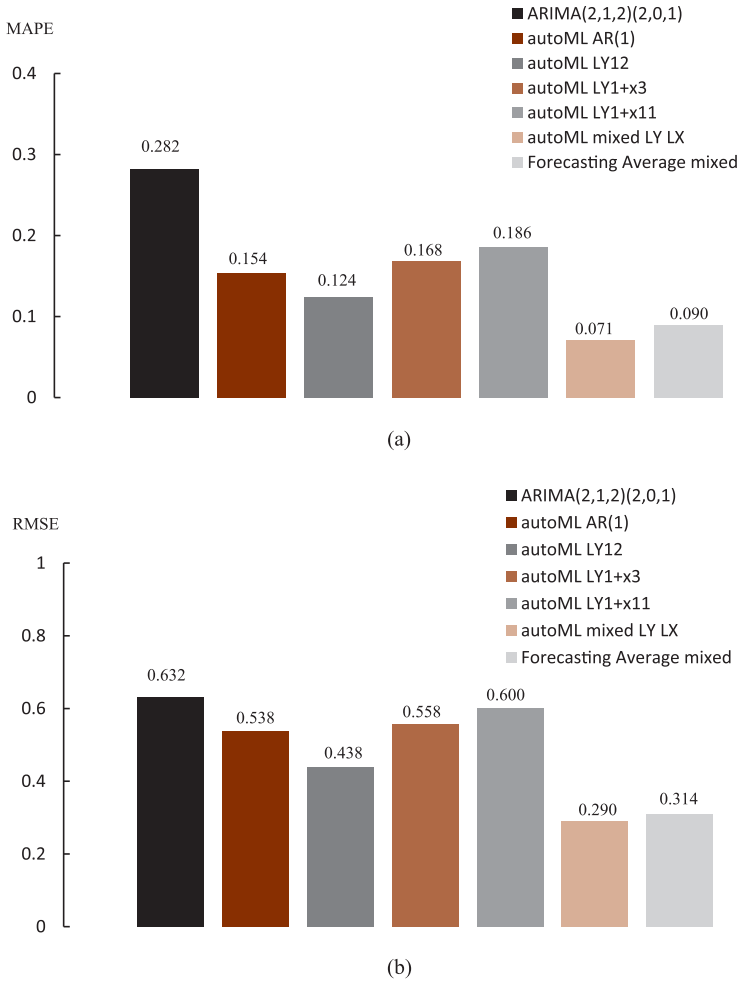
	RMSE	MAE	MAPE	ACF1	Theil's U
autoML.mean	0.4329	0.3626	0.1343	0.3153	0.0075
autoML.median	0.4699	0.3843	0.1414	0.3314	0.0080
LSTM.mean	0.9061	0.7330	0.2630	0.3711	0.0132
LSTM.median	0.9657	0.7973	0.2872	0.3931	0.0149
SVM.mean	1.1591	1.0059	0.3815	-0.0255	0.0189
SVM.median	1.0337	0.8937	0.3374	-0.0710	0.0170
All.mean	0.5393	0.3696	0.1392	0.0551	0.0078
All.median	0.5417	0.4046	0.1541	-0.0752	0.0084

資料來源：本研究自行整理。

綜言之，將多個機器學習模式產生的多步預測，用預測平均法組合成單一預測，實務上應該是很理想的一種作法，文獻上有加權平均的方式，應可以持續發展。再者，以機器學習方法改善時間序列模型預測的可行性，仍需要一個紮實的理論支撐。若就資訊科學的角度思考，只要硬體演算能力夠強大，用autoML對資料直接進行大規模計算，找出最佳資料結構，再使用模型平均法，應是改善時間序列預測更為有效的方式。

最後，我們在圖15列出績效最小值的綜合直方圖。這個綜合結果指出，添加解釋變數不一定對預測有幫助，這再次說明預測平均法或許是一個綜合的好方式。然而，預測平均僅適用於預測狀況比較好的情況，例如SVM的結果就不甚理想。

圖15　預測績效綜合比較



資料來源：本研究自行整理。

陸、結　論

本文探討機器學習演算法，能否改善時間序列預測的表現，並以臺灣經濟成長率為範例，採用SVM，autoM，LSTM三個模式。投入變數 (input variables) 則有三種：*y* (被解釋變數) 從AR(1) 到混合多階，以及添加大量解釋變數的混合多階落後組合。說明如次：

1. 根據既有機器學習的演算經驗，多半需要先進行大量訓練，然後再進行預測。我們發現產生多組預測雖屬可行，但是十分耗時。以SVM的混合計算為例，採用平行運算依然用了30小時計算完1千多組。autoML的狀況更為嚴重，因為下載的元件只支援單集群 (cluster) 演算，在平行處理上受到限制，而LSTM亦同。
2. 本文最佳的方法是自動化機器學習，不論是從自我落後結構的15種組合，或混合落後的1,005組，預測平均法皆可提高其精準度，這也是本研究對於機器學習時間序列預測的建議作法。過多的投入變數恐產生大而無當的成本，且效益也不會比較好。能夠事先訓練好一個小集合，並產生多個預測的平均，應該很有助益。
3. 多種落後期變數的混合，雖然須測試大量的投入變數組合使計算耗時，但利用PCA 或ICA 將資料降維，並未得到較好的結果，因此本文把重心放在原始數據的演算。由預測平均法的表現看來，減少運算量，提高預測表現，應該是一個可行方法。
4. SVM和LSTM容易產生持續性的高估或低估，這或許是因為樣本數太少所致，當樣本數夠多時，這兩個方法應該可以表現更好。
5. 用機器學習修正時間序列模型的預測誤差 (如ARIMA計算過程添加機器學習模式) 雖屬可行，但仍有改善空間，細節可參考研究報告內容。
6. 儘管本文進行上述嘗試，惟經濟成長季資料觀察值僅232個，對於號稱大數據的學習演算法而言，訓練和交叉驗證的設計都受到極嚴重限制，學習不足使其預測優勢難以被確認，這是值得注意的地方。使用間接混頻的方式 (如MIDAS)，結合機器學習對高頻月資料 (如工業生產指數) 進行預測，亦即用機器學習訓練高頻預測模型，再用經濟計量方法預測經濟成長，這對於整合機器學習到計量經濟學或將有正面貢獻，可做為未來研究方向。[註5]

附　註

(註1) http://karpathy.github.io/2015/05/21/rnn-effectiveness/。

(註2) https://www.grb.gov.tw/search/planDetail?id=13214755。

(註3) 不只是機器學習，統計時間序列也不敢說樣本內配適度和樣本外的預測績效之間有一定相關，尤其是未來出現重大經濟衝擊時。然而，機器學習模型可能發生樣本內配適結果良好，但樣本外預測績效太差，遠大於樣本內的預測績效的情形，此即過度配適 (overfitting)。本研究樣本內外的預測績效差異在容許範圍內，過度配適問題並不顯著。

(註4) 在此我們重新擷取國外數據以找出最適合預測臺灣經濟成長率的變數組合。

(註5) 感謝徐士勛教授的建議。

參考文獻

Baek, Y., & Kim, H. Y. (2018). ModAugNet: A new forecasting framework for stock market index value with an overfitting prevention LSTM module and a prediction LSTM module. *Expert Systems with Applications*, 113, 457-480.

Bergmeir, C., Hyndman, R. J., & Benítez, J. M. (2016). Bagging exponential smoothing methods using STL decomposition and Box–Cox transformation. *International Journal of Forecasting*, 32(2), 303-312.

Beyca, O. F., Ervural, B. C., Tatoglu, E., Ozuyar, P. G., & Zaim, S. (2019). Using machine learning tools for forecasting natural gas consumption in the province of Istanbul. *Energy Economics*, 80, 937-949.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

Bolhuis, M., & Rayner, B. (2020). *Deus ex Machina? A Framework for Macro Forecasting with Machine Learning* (No. 20/45). International Monetary Fund.

Bontempi, G., Taieb, S. B., & Le Borgne. Y. A. (2013). Machine learning strategies for time series forecasting. In Marie-Aude Aufaure and Esteban Zimányi (eds.) *Business Intelligence*, pp. 62-77, Spring-Verlag.

Chakraborty, C., & Joseph, A. (2017). Machine learning at central banks. Staff working paper No. 674, Bank of England.

Chatzis, S. P., Siakoulis, V., Petropoulos, A., Stavroulakis, E., & Vlachogiannakis, N. (2018). Forecasting stock market crisis events using deep and statistical machine learning techniques. *Expert Systems with Applications*, *112*, 353-371.

De Livera, A. M., Hyndman, R. J., & Snyder, R. D. (2011). Forecasting time series with complex seasonal patterns using exponential smoothing. *Journal of the American statistical association, 106*(496), 1513-1527.

Ediger, V. Ş., & Akar, S. (2007). ARIMA forecasting of primary energy demand by fuel in Turkey. *Energy Policy*, 35(3), 1701-1708.

Feng, H., & Liu, J. (2003). A SETAR model for Canadian GDP: Non-linearities and forecast comparisons. *Applied Economics*, *35*(18), 1957-1964.

Franses, P. H., & Van Dijk, D. (2000). *Non-linear Time Series Models in Empirical Finance*. Cambridge University Press.

Hamzaçebi, C., Akay, D., & Kutay, F. (2009). Comparison of direct and iterative artificial neural network forecast approaches in multi-periodic time series forecasting. *Expert Systems with Applications*, 36(2), 3839-3844.

Hansen, B. E. (2007). Least squares model averaging. *Econometrica, 75(4)*, 1175-1189.

Hansen, B. E., & Racine, J. S. (2012). Jackknife model averaging. *Journal of Econometrics, 167*(1), 38-46.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Science & Business Media.

Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: Principles and Practice*. 2nd edition, Available at https://otexts.com/

fpp2/.

Hyndman, R. J., & Koehler, A. B. (2006). Another look at measures of forecast accuracy. *International Journal of Forecasting*, *22*(4), 679-688.

Hyndman, R. J., Athanasopoulos, G., Bergmeir, C., Caceres, G., Chhay, L., O'Hara-Wild, M., & Yasmeen, F. (2018). *Forecast: Forecasting Functions for Time Series and Linear Models*. Available at https://rdrr.io/cran/forecast/.

Karpathy, A. (2015). *The Unreasonable Effectiveness of Recurrent Neural Networks*. Available at http://karpathy.github.io/2015/05/21/rnn-effectiveness/.

Kline, D. M. (2004). Methods for multi-step time series forecasting neural networks. In *Neural Networks in Business Forecasting* (pp. 226-250). IGI Global.

Koenker, R. & Xiao, Z. (2006) Quantile autoregression. *Journal of the American Statistical Association*, 101(475), 980-990.

Kong, W., Dong, Z. Y., Jia, Y., Hill, D. J., Xu, Y., & Zhang, Y. (2017). Short-term residential load forecasting based on LSTM recurrent neural network. *IEEE Transactions on Smart Grid*, 10(1), 841-851.

Liao, J. C., & Tsay, W. J. (2020). Optimal multistep VAR forecast averaging. *Econometric Theory*, 1-28.

Makridakis, S. (1993). Accuracy measures: Theoretical and practical concerns. *International Journal of Forecasting*, 9(4), 527-529.

Makridakis, S., & Hibon, M. (2000). The M3-competition: results, conclusions and implications. *International Journal of forecasting, 16*(4), 451-476.

Morlidge, S. (2014). Do forecasting methods reduce avoidable error? Evidence from forecasting competitions. *Foresight: The International Journal of Applied Forecasting*, 32, 34-39.

Richardson, A., & Mulder, T. (2018). Nowcasting New Zealand GDP using machine learning algorithms. CAMA Working Paper No. 47/2018. Available at SSRN: https://ssrn.com/abstract=3256578.

Saad, E. W., Prokhorov, D. V., & Wunsch, D. C. (1998). Comparative study of stock trend prediction using time delay, recurrent and probabilistic neural networks. *IEEE Transactions on Neural Networks*, 9(6), 1456-1470.

Serinaldi, F. (2011). Distributional modeling and short-term forecasting of electricity prices by generalized additive models for location, scale and shape. *Energy Economics, 33*(6), 1216-1226.

Stock, J. H., & Watson, M. W. (1998). *A comparison of linear and nonlinear univariate models for forecasting macroeconomic time series* (No. w6607). National Bureau of Economic Research.

Taylor, J. W. (2003). Short-term electricity demand forecasting using double seasonal exponential smoothing. *Journal of the Operational Research Society, 54*(8), 799-805.

Tiffin, A. (2019). Machine learning and causality: the impact of financial crisis on growth. IMF Working Paper No. 19/228. Available at SSRN: https://ssrn.com/abstract=3496706

Torres, J. F., Galicia, A., Troncoso, A., & Martínez-Álvarez, F. (2018). A scalable approach based on deep learning for big data time series forecasting. *Integrated Computer-Aided Engineering*, 25(4), 335-348.

Tsay, R. S. (2010). *Analysis of Financial Time Series*. John Wiley & Sons Inc.

Vapnik, V. (2013). *The Nature of Statistical Learning Theory*. Springer Science & Business Media.

Wood, S. N., & Augustin, N. H. (2002). GAMs with integrated model selection using penalized regression splines and applications to environmental modelling. *Ecological modelling*, 157(2-3), 157-177.

Zhao, Z., Chen, W., Wu, X., Chen, P. C., & Liu, J. (2017a). LSTM network: A deep learning approach for short-term traffic forecast. *IET Intelligent Transport Systems, 11*(2), 68-75.

Zhao, Y., Li, J., & Yu, L. (2017b). A deep learning ensemble approach for crude oil price forecasting. *Energy Economics, 66*, 9-16.

Machine Learning Methods to Predict Taiwan's Economic Growth Rates

Ho, Tsong-wu Yeh, Kuo-chun Mo, Wan-shin Lin, Ya-chi*

Abstract

This paper aims to use machine learning methods (MLs) to predict Taiwan's economic growth rate, and then to explore the best predictive model that captures time-series trends, peaks, and troughs. We found that MLs would not be better than traditional time series models when a one-step prediction is applied without additional explanatory variables. However, the results would be different when multi-step recursive prediction is applied, and the data-driven MLs have better prediction performance than theory-driven econometric models. Moreover, the multi-step predictability of MLs can be greatly improved a lot after including more explanatory variables, and the automated machine learning model (autoML) is the best one. We also try to find the best combination of the current and lagged variables. Due to the limitation of the quarterly data, the so-called big data learning algorithm with the designs of training and cross-validation is extremely severely restricted. Low-frequency time-series MLs could be a future research direction.

Keywords: machine learning, Taiwan's economic growth rate, econometric forecast, cross-validation.

JEL classification code: C22, C55, E37.

* The views expressed in this paper are those of the authors and do not necessarily reflect the position of the Central Bank of the Republic of China (Taiwan). Any errors or omissions are the responsibility of the authors.