東海大學

資訊工程研究所

碩士論文

指導教授：林祝興

雲端共存攻擊多目標回應策略機器學習模式之研究

Machine Learning Modelling for Multi-objective Response Strategy to Co-resident Attacks in Cloud Computing

研究生：呂曉雯

中華民國 108 年 7 月 20 日

東海大學碩士學位論文考試審定書

東海大學資訊工程學系 研究所

研究生 呂 曉 雯 所提之論文

雲端共存攻擊多目標回應策略機器學習模式之研究

經本委員會審查，符合碩士學位論文標準。

學位考試委員會

召 集 人 _____ 簽章

委 員 _____

_____

_____

指 導 教 授 _____ 簽章

中華民國 108 年 7 月 5 日

摘要

雲端計算可以透過虛擬化技術共享軟硬體資源，但是使用者在使用虛擬化平台時可能會面臨額外的的安全威脅。共存攻擊是指攻擊者利用共享基礎設備的特性，來攻擊共存在同一實體機上的其他虛擬機。2017 年 Abazari 等人提出了雲端共存攻擊多目標回應系統，考量到虛擬機共存時間，以最小成本和最小威脅為目標來回應共存攻擊。但我們發現實際上該系統的回應時間過長。因此在本論文中，我們使用機器學習來訓練入侵回應系統，我們使用 Ridge Regression 演算法，並進行一系列實驗證明模型的效果，實驗中也比較了我們的模型和 Abazari 等人的模型。實驗結果顯示，我們的模型具有 2000 倍加速比的效率提升，同時獲得 87.9% 高準確度的解答，在回應策略與時間效能取得平衡。

關鍵字：雲端計算、共存攻擊、機器學習、雲端入侵回應系統

Abstract

With cloud computing, we can share hardware and software resources through virtualization technology, while the users may face additional security threats when using virtualization platforms. Co-resident attack is one security problem when an attacker exploits the characteristics of a shared infrastructure and attacks other virtual machines co-located on the same physical machine. In 2017, Abazari et al. proposed a multi-objective response system to against co-resident attacks in cloud environment, considering the co-resident time of virtual machines, and responding to the attacks with the goal of minimum cost and minimum threat. Howeverwe found that the response time of their proposed system was actually too long for real-time applications.

In this thesis, we proposed to use machine learning to train the intrusion response system. We use the Ridge Regression algorithm and perform a series of experiments to prove the effect of the proposed model. In the experiments, we also compared our model with that of the Abazari's. From the experimental results, we showed that our model can obtain a solution with efficiency improvement of 2000x speedup and 87.9% of high accuracy.

Keywords: Cloud Computing, Co-resident Attack, Machine Learning, Cloud Intrusion Response System.

目錄

圖目錄

表目錄

公式目錄

Chaper 1　簡介

雲端計算(cloud computing)可按照用戶的需求，以低成本提供使用者軟硬體設備，使用者可以方便的使用服務，並且無需了解雲端環境的細節，也不需進行設備維護。然而，共享軟硬體設備也讓使用者的資料暴露於各種安全威脅之下。在 2018 年，Mazhar Ali 等人 [2] 描述了雲端計算中的資料、軟體、服務、虛擬化和虛擬機管理程序相關的安全問題；除了雲端內部的安全問題，攻擊者也可以將雲端計算當作一些攻擊的來源。虛擬化(virtualization)技術可以按照需求共享資源，提高了資源的使用率，如圖 1.1 所示，雲端虛擬化示意圖。雲端環境可以由多個主機(host or physical server)串連而成，而主機當中透過管理程序(hypervisor)可同時運行多個虛擬機(virtual machine，VM)，每個虛擬機當中有不同使用者運行的作業系統(OS)和應用程式(APP)。由於雲端的虛擬化特性，多個虛擬機會在同一主機上同時運行，如果虛擬機之間沒有適當的隔離，攻擊者可以從同一主機上取得其他虛擬機的相關資訊。

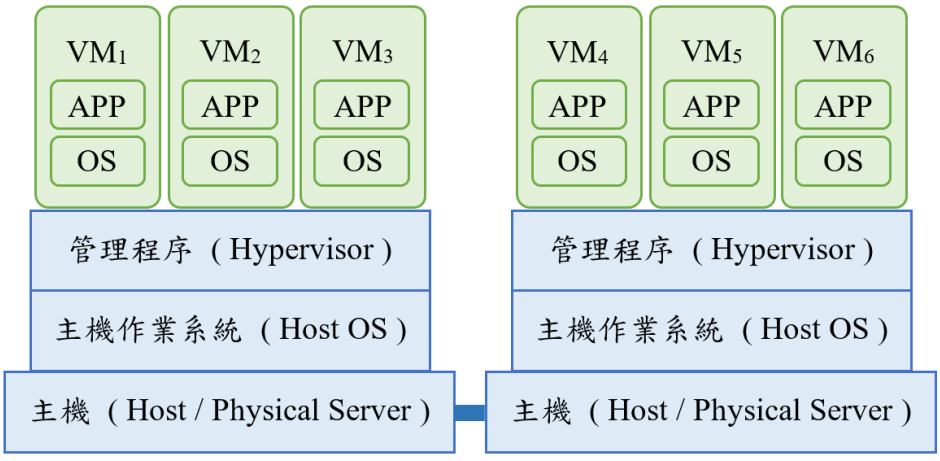

圖 1.1：雲端虛擬化示意圖

在雲端服務的安全中，其中一個挑戰是如何做到回應威脅(response to threat)，Anwar 等人 [3] 根據入侵回應系統 (intrusion response system，IRS) 的反應類型將 IRS 分為三

大類：通知、手動、自動。通知僅以通知的形式產生回應；手動則允許系統管理員根據預定的回應選項集產生適當的回應；自動則是由系統直接根據現有威脅提供即時的回應。自動入侵回應系統的主要缺點是，系統可能會使用不適當的回應，不適當的回應可能會威脅到更多雲端虛擬機，因此需要一個全面的機制來產生針對當前威脅的最佳回應(optimal response)。即時回應威脅(real-time response to threat)和選擇最佳對策(optimal strategy selection)是自動入侵回應系統最重要的兩個目標。

2017 年 Abazari 等人 [1] 提出了雲端共存攻擊多目標回應系統(multi-objective response system)，這篇論文主要在討論雲端共享基礎設備所引起的安全問題。他們整理了虛擬機共存攻擊(co-resident attack)和回應對策的類型，並提出一套模型來計算雲端整體的最佳回應對策。每種類型的共存攻擊都可以透過特定對策(strategy)來進行防禦，每種對策也有不同的執行成本(cost)。他們提出的模型考量到虛擬機之間的共存時間(co-resident time)長短，並在威脅(threat)和成本(cost)這兩個衝突的目標中取得帕雷托最優解(Pareto optimal solutions) 得到最佳回應對策。圖 1.2 為帕雷托最優解示意圖，其中， F_1 代表總威脅(total threat)； F_2 代表總成本(total cost)。若有一個解 x，我們無法在不提升總成本的情況下再降低總威脅，且無法在不提升總威脅的情況下再降低總成本，那解 x 就為帕雷托最優解。在兩個互相衝突的目標的情況下，會有多個帕雷托最優解。

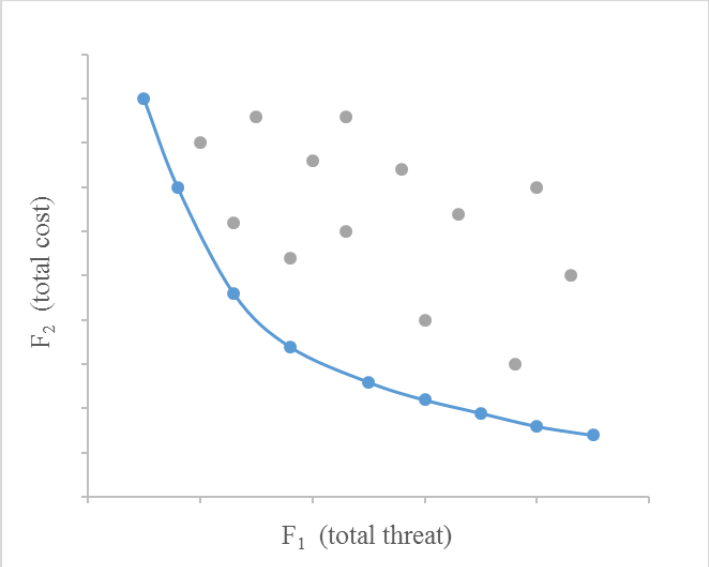

圖 1.2：帕雷托最優解示意圖

Abazari 等人的模型可以針對各種威脅做最佳對策的選擇，然而即時回應威脅也是自動入侵回應系統最重要的目標之一，經過實作後我們發現 Abazari 等人的模型在具有大量虛擬機的雲端環境中，計算回應對策需要耗費相當長的時間。

因此，在這篇論文的啟發下，我們提出了一個自動入侵回應系統的新模型：使用機器學習(machine learning)來產生回應對策。我們進行了一系列實驗來證明模型的效果，從實驗結果可以看出機器學習可以以良好的準確度(accuracy)近似帕雷托最優解，並且在回應速度方面相較於 Abazari 等人的模型具有 2000 倍的提升。

由圖 1.3 簡單介紹我們的研究流程，首先我們研讀 Abazari 等人的論文，並按照論文進行實作，在確定模型的正確性之後，以隨機的方式產生大量模擬雲端環境的虛擬機共存時間關係圖，輸入到 Abazari 等人的模型中產生回應對策的解答。接著，再將雲端環境中所有虛擬機的威脅等級以及解答資料，隨機選取 80% 作為機器學習的訓練資料(training data)，另外 20% 作為測試資料(testing data)。我們針對 Lasso、Elastic Net、Ridge Regression 這三種機器學習演算法進行實驗測試，其中 Ridge Regression 得到的準確度較高，因此，我們就選擇 Ridge Regression 作為機器學習的演算法。為了確認我們的模型的準確度及效率，我們先以實驗 4.2 的 12 個虛擬機的小型雲端環境做實驗，確認機器學習回應的準確度及正確性。再來以實驗 4.3 的 50 個虛擬機的實驗作為較大雲端環境的實驗，再次確認機器學習回應的準確度及正確性，同時也顯現出機器學習在回應時間的優勢。最後再以實驗 4.4 及實驗 4.5 確認我們的機器學習模型的準確度及效能。

圖 1.3：研究流程介紹圖

本論文的組織架構共分為五個章節，首先透過第一章的簡介描述整個論文的內容，接著在第二章背景知識與相關文獻中詳細說明相關的背景知識，第三章中介紹了本論文所使用的模型和運作流程，第四章利用各種實驗證明了本模型的成效，最後在第五章中討論了本模型的優劣和未來的展望。

Chaper 2　背景知識與相關文獻

2.1　雲端概念

隨著網路速度及軟硬體技術的提升，雲端服務(cloud services)也日漸崛起，透過網路將共用的軟硬體資源按照需求提供給終端使用者，雲端計算根據服務類型可以分為 SPI 三類：軟體即服務 (Software as a Service，SaaS)、平台即服務 (Platform as a Service，PaaS)、基礎架構即服務 (Infrastructure as a Service，IaaS)。

SaaS 是讓使用者不須下載軟體到本機上，透過瀏覽器就可以線上使用軟體，常見的有 e-mail、google 文件，也有以企業用戶為導向的人力資源管理、客服管理、流程管理系統等等。SaaS 可以減少使用者安裝和維護軟體的時間與成本，也不必擔心軟體的安裝與更新。

PaaS 主要是針對軟體開發者提供開發平台，開發者無須在本機安裝開發工具，透過服務支援的程式語言和環境，使用者只需要透過瀏覽器、遠端控制等技術就能遠端進行開發，也能快速的進行測試。

IaaS 是提供使用者運算資源的存取，透過虛擬化的技術，使用者可以在提供的基礎架構中建立自己的作業系統和應用程式。雲端服務供應商(cloud service provider，CSP)透過虛擬化的技術在一部大型的實體主機上建立多個虛擬機，根據使用者需求動態分配資源，每個虛擬機就如同一台電腦，擁有獨自的作業系統和運作的應用程式，但是基礎架構比一般電腦更容易因應需求而做調整，使用者可以根據自己的需求租借適量的服務，也可以省下維護硬體的成本。本篇論文討論的雲端共存攻擊就是在 IaaS 架構下隱藏的威脅。

2.2　雲端共存攻擊

IaaS 雲端計算服務中要保護的是在資料中心機器上運行的虛擬機，其中包含使用者的資料、程式等等有價值的訊息。一旦在系統中檢測到可疑行為(例如連接埠掃描，資源

過度消耗，惡意流量的情況)，就應該判斷檢測到的行為是否是惡意的，若是惡意行為就要盡快使用正確對策進行防禦。由於雲端共享基礎設備的特性，攻擊者可以從同一台實體機竊取其他虛擬機的資訊，只要和攻擊者的虛擬機在同一實體機共存，都有可能遭到攻擊。共存攻擊可分為三種類型，第一種包含訊息竊取和側通道攻擊 (side-channel attack，SC)，受害者的隱私遭到窺探而造成不可逆的傷害；第二種類型是攻擊者利用虛擬機之間的通訊來傳播惡意軟體 (malware propagation，MP)；第三種類型攻擊者的目標是得到不公平的共享資源，導致同一實體機上的其他虛擬機受到阻斷服務攻擊 (denial-of-service attack，DoS)。

2.3 側通道攻擊

在這種類型的共存攻擊中，攻擊者透過實體機存取諸如：時間訊息、共享的快取和功耗資訊，可以利用管理程序偷偷地收集受害者的隱私資訊。Bates 等人 [4] 提出了 co-resident watermarking，這是一種流量分析攻擊，攻擊者將 watermark 注入受害虛擬機的網路流中以進行流量分析。而部分虛擬化平台使用的 virtual floppy drive 的漏洞，可以讓攻擊者脫離自身的虛擬機，並獲得對實體機執行存取的權限，導致攻擊者可以存取實體機上所有的虛擬機 [5]。

2.4 惡意軟體傳播

具有網路傳播能力的惡意軟體 (例如蠕蟲) 可以透過網路自動散播，因此虛擬機之間的通訊及其對網路的存取都可能導致雲端基礎設備中惡意軟體的傳播 [6]。Mazhar Ali 等人 [2] 介紹了虛擬機之間兩種類型的通訊，外部：使用者和雲端之間，內部：雲端的虛擬化環境中。內部的虛擬網路存在於所有虛擬機和實體網路之間，虛擬網路負責管理通過虛擬機的通訊 [7]，虛擬交換機為相同實體機上的虛擬機之間提供網路連接。由於實體機上的入侵偵測系統和防火牆等安全機制無法偵測到虛擬網路的流量，因此惡意流量可以通過虛擬網路而不被檢測 [8]。

2.5 阻斷服務攻擊

虛擬機管理程序若缺乏適當的隔離會導致攻擊者能夠利用資源爭奪，影響共存虛擬機上運行的應用程式的執行。Chiang 等人 [9] 說明了虛擬 I/O 工作負載的爭奪導致的安全問題，攻擊者可以通過爭奪共享 I/O 資源 (如硬碟或網路頻寬) 來減緩共存虛擬機中應用程式的執行速度。Varadarajan 等人 [10] 的研究表示，由於來自另一個虛擬機的干擾，虛擬機快取工作負載的性能可能降低 80% 以上，他們稱之為資源釋放攻擊 (resource-freeing attack)，攻擊者更改受害虛擬機的工作負載，使受害者釋放競爭資源。由於在雲端環境中共享大量的資源，尤其是彈性雲 (elastic cloud) 可以提供使用者幾乎無限的資源， DoS 攻擊會更具影響力 [11]。

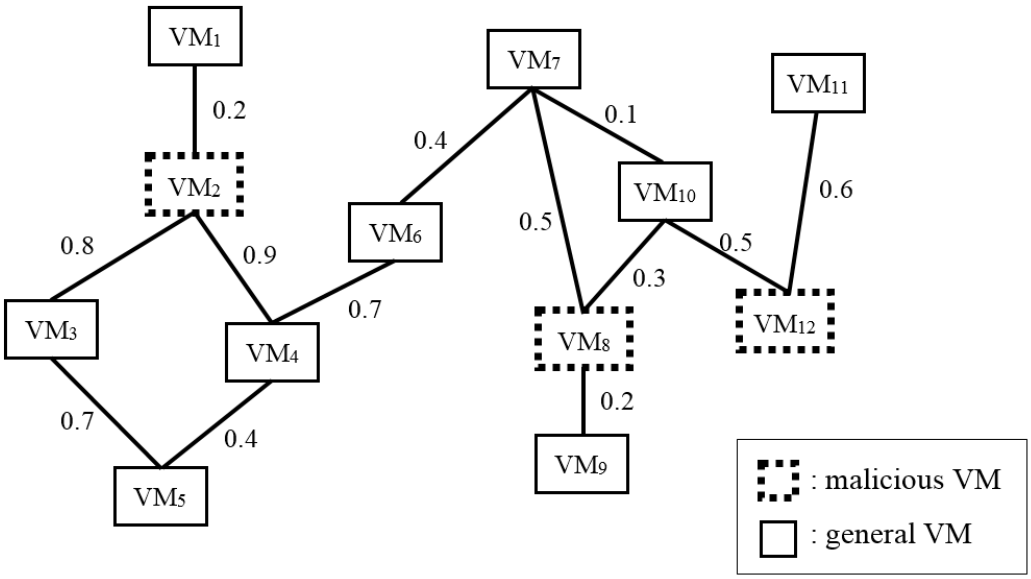

圖 2.1：虛擬機共存時間關係圖示意圖

2.6 虛擬機共存時間關係圖

根據研究文獻，虛擬機受到共存攻擊的機率與共存時間呈正相關(positive correlation) [12] [13]，透過虛擬機共存時間關係圖可以獲得虛擬機之間的共存時間，並用來計算雲

端環境中所有虛擬機的威脅等級 (threat level) [14]。如圖 2.1 所示，虛擬機共存時間關係圖示意圖，圖中的節點 (node) 代表虛擬機 (virtual machine，VM)，其中虛擬機 (VMs) 可分為兩種類型：集合 M 表示惡意 VM (malicious VM)，由虛線方框表示；集合 S 表示一般 VM (general VM) 由實線方框表示；節點之間的連線 (edge) 代表虛擬機曾經或正在同一實體機中共存 [15]。連線上的數值代表共存時間長短，時間會經過正規化 (normalization) 變成 0~1 之間的數值，由 $w_{i,j}$ 表示 VM_i 和 VM_j 的共存時間，舉例來說，在圖 2.1 中 VM_2 和 VM_3 的共存時間$w_{2,3} = 0.8$，VM_8 和 VM_9 的共存時間$w_{8,9} = 0.2$。

2.7 回應對策表

每種攻擊都可以透過特定的對策來防禦。在表 2.1 中，欄位 SC 表示側通道攻擊、MP 表示惡意軟體傳播、DoS 表示阻斷服務攻擊，由 Y 表示該對策可防禦此種攻擊，由 N 表示該對策無法防禦此種攻擊。表 2.1 中列出了每個共存攻擊的對策，總共可分為 5 個對策，對策 6 表示不進行任何動作，以下對每個對策進行詳細說明：

1. 第一種對策是虛擬機遷移 (VM migration)，其能夠防禦所有共存攻擊。
2. 第二種對策能防禦側通道攻擊，該對策根據虛擬機的屬性可分為兩種情況。若是針對攻擊者的虛擬機，就是增加惡意行動的延遲 [16]。如果是受害虛擬機，則可以將記憶體、CPU 和快取等共享資源的存取方式改為一致的模式，就可以防止攻擊者從受害者的行為中獲取訊息。
3. 第三種對策可以防止惡意軟體傳播，利用虛擬交換器 (virtual switch) 的功能重新配置虛擬網路 [17]。
4. 第四種對策可以防禦阻斷服務攻擊，主要是透過更改管理程序的配置以降低最大計算負載或快取容量，來限制攻擊者的資源使用 [18]。
5. 第五種對策是透過限制相關虛擬機允許的最大流量速率，來防禦惡意軟體傳播和阻斷服務攻擊 [18]。
6. 第六種對策是不進行任何動作。

表 2.1：回應對策表

編號	VM 屬性	防禦方法	SC	MP	DoS	成本
1	M/S	虛擬機遷移	Y	Y	Y	增加功耗且性能下降 成本=高
2	M	更改管理程序	Y	N	N	增加耗能 成本=中
	S	更改資源的存取模式	Y	N	N	增加耗能 成本=中
3	M/S	通訊隔離	N	Y	N	網路重新配置 成本=低
4	M/S	限制資源分配	N	N	Y	增加功耗 成本=中
5	M/S	限制網路流量速率	N	Y	Y	網路重新配置 成本=低
6	M/S	不進行任何動作	N	N	N	成本=零

2.8 機器學習

機器學習(machine learning)屬於人工智慧(artificial intelligence)的範疇，是一種讓機器從「學習」到「推理」的過程，機器學習的演算法結合了各種學科，如：機率學、統計學、凸分析、逼近論和計算複雜性理論等等，透過組合這些數學模型設計出讓計算機可以自我學習的演算法。從訓練資料中截取重要特徵，訓練時透過演算法將特徵進行分析獲得規律而得到模型，最後就可以利用模型對新資料進行預測。在多次學習後的輸出結果具有高準確度的分析和決策能力，使機器學習在近幾年被廣泛使用在各種領域，常見的有人臉辨識、車牌辨識、推薦系統、自然語言分析、手寫識別...等等，甚至在棋類遊戲、戰略遊戲中都有戰勝人類選手的成績。

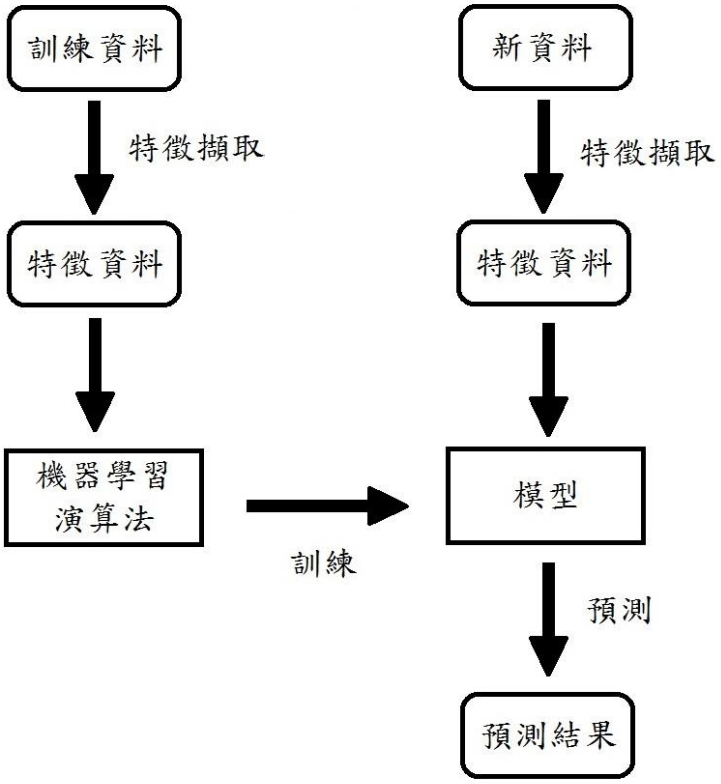

圖 2.2：機器學習概念圖

機器學習的概念如圖 2.2 所示，我們的訓練資料由 Abazari 等人提出的模型產生，Abazari 等人的模型是使用 Matlab 的 fgoalattain 函數來計算多目標優化問題，目標是指總回應成本和總威脅這兩個目標，我們要降低威脅的同時也降低成本。

我們以隨機的方式產生大量模擬雲端環境的虛擬機共存時間關係圖，套用到 Abazari 等人的模型產生大量訓練資料。而在機器學習演算法方面，我們對 Lasso、Elastic Net、Ridge Regression 這三種機器學習演算法進行測試，我們先製造 200 筆資料，並隨機取 80%作為訓練資料，20%為測試資料，接著分別使用三種機器學習演算法，平均準確度如表 2.2 所示。從表 2.2 顯示，其中 Ridge Regression 得到的準確度較高，所以本論文就選擇 Ridge Regression 作為機器學習的演算法。

表 2.2：機器學習演算法準確度比較

機器學習演算法	Lasso	Elastic Net	Ridge Regression
平均準確度	63.7%	65.6%	70.2%

2.9 Scikit-learn

Scikit-learn 簡稱 SKlearn (在 Python 使用時為 import sklearn)，是 Python 的機器學習和資料分析的開源套件，當中包含許多知名的機器學習演算法，也內建許多的知名的資料集 (例如手寫數字資料集、鳶尾花資料集、乳腺癌資料集)。在演算法方面，Scikit-learn 官方網站依照演算法功能分為 6 個類別：Classification、Regression、Clustering、Model selection、Preprocessing、Dimensionality reduction，也提供了簡易流程圖讓使用者根據資料集的形態來挑選適合的演算法，如圖 2.3 所示 [19]。

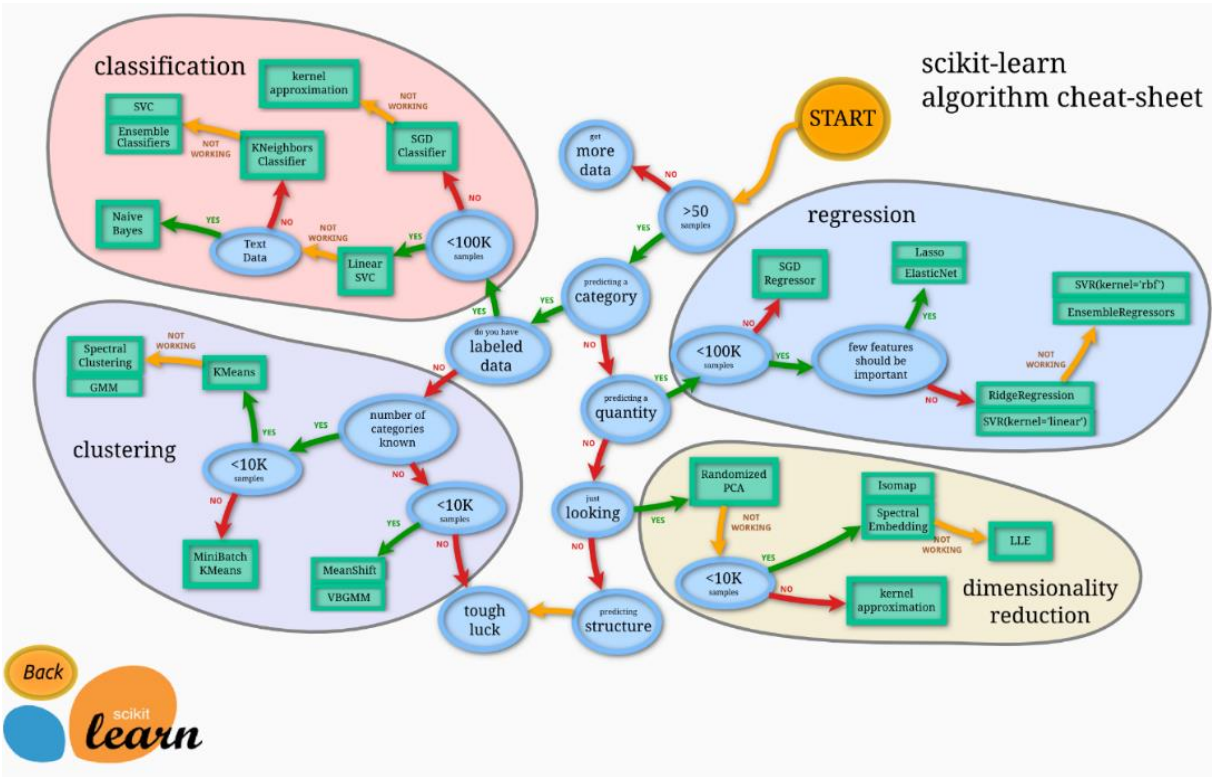

圖 2.3：Scikit-learn 演算法挑選流程圖

(source: https://scikit-learn.org)

2.10 Ridge Regression

在 Scikit-learn 中，脊迴歸(Ridge Regression)可以解決損失函數(loss function)為線性最小平方函數(linear least squares function)且採 L2-norm 方式做正規化的迴歸問題 [19]。Ridge Regression 也可稱為嶺迴歸或吉洪諾夫正規化(Tikhonov regularization)。Ridge Regression 支援多變量迴歸(multi-variate regression)，例如二維矩陣。

Chaper 3　研究方法

3.1 計算威脅等級

虛擬機受到共存攻擊的機率與共存時間呈正相關，因此，為了計算雲端環境中所有虛擬機的威脅等級，首先必須擁有虛擬機共存時間關係圖，還有檢測到的攻擊者所操控的惡意虛擬機集合 M。攻擊者要進行攻擊時，其虛擬機通常會存在資源大量使用、系統呼叫和 cache miss 等異常現象，一般可透過持續的雲端監控來發現惡意虛擬機 [20]。

如第 2 章所述，雲端共存攻擊可分為三類型：側通道攻擊、惡意軟體傳播和阻斷服務攻擊，因此我們設定虛擬機威脅等級為三元組 (triplet) $t_i = \left[t_i^{SC},\ t_i^{MP}, t_i^{DOS}\right]$，分別表示三種攻擊的機率，如公式 1。

$$t_i = \left[t_i^{SC}, t_i^{MP}, t_i^{DOS}\right] \tag{1}$$

t_i 是代表 VM_i 被攻擊的機率(威脅等級)，一個受害 VM 會因為和多個不同惡意 VM 共存，造成被攻擊的機率提升，威脅等級必須包括任一威脅發生的機率與多個威脅同時發生的機率，因此我們針對 Abazari 等人 [1] 的威脅等級 (threat level) 計算方法進行修改，實際的計算方式如公式 2 所示。

$$\begin{cases} t_i = [1\ \ 1\ \ 1] - \prod_{j=1}^{n}\left([1\ \ 1\ \ 1] - t_j \times w_{i,j}\right) \\ subj.to \\ VM_j \in M \end{cases} \tag{2}$$

首先，VM_j 要屬於惡意虛擬機集合 M，才需要計算其可能對 VM_i 造成的威脅，n 為總虛擬機個數，$w_{i,j}$ 表示 VM_i 和 VM_j 的共存時間。$t_j \times w_{i,j}$ 表示 VM_i 被 VM_j 攻擊的機率，$[1\ \ 1\ \ 1] - t_j \times w_{i,j}$ 就是不被 VM_j 攻擊的機率。多個不被攻擊的機率求積就等於完

全不被攻擊的積率，最後由[1 1 1]減去完全不被攻擊的機率就是包括任一威脅發生的機率與多個威脅同時發生的機率。

以圖 2.1 為例，我們給予惡意虛擬機隨機的威脅等級，並根據公式 2 計算一般虛擬機的威脅等級，如圖 3.1 所示。假設要計算 VM_4 的威脅等級，VM_4 和 VM_2、VM_5、VM_6 曾經共存或正在共存，當中只有 VM_2 是惡意虛擬機，所以 $t_4 = [1\ 1\ 1] - ([1\ 1\ 1] - t_2 * 0.9) = t_2 * 0.9$。另外假設要計算 VM_{10} 的威脅等級，VM_{10} 和 VM_7、VM_8、VM_{12} 曾經共存或正在共存，當中 VM_8 和 VM_{12} 是惡意虛擬機，所以 $t_{10} = [1\ 1\ 1] - ([1\ 1\ 1] - t_8 * 0.3) * ([1\ 1\ 1] - t_{12} * 0.5)$。

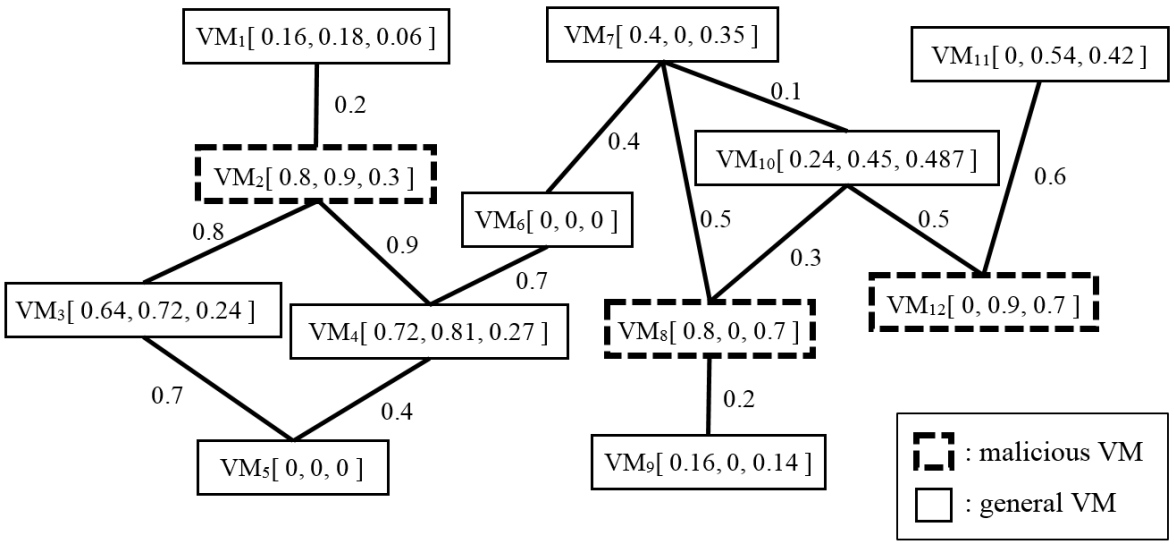

圖 3.1：虛擬機威脅等級示意圖

雲端環境的總威脅等級 (total threat) T，就是將所有虛擬機的威脅等級做加總，計算方式如公式 3。

$$T = \sum_{i=1}^{n} t_i \tag{3}$$

3.2 回應對策矩陣和成本向量

首先，我們定義矩陣 $\mathrm{C} = [c_1, c_2, \cdots, c_q]^T$ 表示回應對策，其中 $c_i = [c_i^{sc},\ c_i^{MP}, c_i^{DoS}]$，$q$代表對策總數量，我們的$q = 6$，如公式 4 所示。公式 5 的$\mathrm{RC} = [rc_1, rc_2, \cdots, rc_q]^T$ 是回應成本向量，rc_i 代表對策 c_i 耗費的成本。

$$C = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_q \end{bmatrix} = \begin{bmatrix} c_1^{sc} c_1^{MP} c_1^{DoS} \\ c_2^{sc} c_2^{MP} c_2^{DoS} \\ \vdots \quad \vdots \quad \vdots \\ c_q^{sc} c_q^{MP} c_q^{DoS} \end{bmatrix} \tag{4}$$

$$RC = \begin{bmatrix} rc_1 \\ rc_2 \\ \vdots \\ rc_q \end{bmatrix} \tag{5}$$

每種對策能回應的攻擊類型不同，所耗費的成本也不同。根據上述定義的公式，我們由表 2.1 的 SC、MP 和 DoS 三個欄位產生回應對策矩陣 C，如公式 6。舉例來說，C(3, 2)=1 代表對策 3 可以應對惡意軟體傳播攻擊，C(4, 3)=1 代表對策 4 可以應對阻斷服務攻擊。由表 2.1 的成本欄位產生成本向量RC，如公式 7。$RC(3)$=0.2 代表對策 3 擁有低的耗費成本。

$$C = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix} \tag{6}$$

$$RC = \begin{bmatrix} 1 \\ 0.5 \\ 0.2 \\ 0.4 \\ 0.3 \\ 0 \end{bmatrix} \tag{7}$$

3.3 資料生成

Abazari 等人 [1] 提出的雲端共存攻擊多目標回應系統，已經能針對雲端整體環境選擇出最佳回應對策，但是卻存在回應時間過久的問題。因此，我們的訓練資料就由 Abazari 等人的模型來產生，步驟如下：

1. 產生一個隨機的 VM 共存時間關係圖
2. 隨機挑選約 30% 的 VM 作為惡意 VM
3. 隨機產生惡意 VM 威脅等級
4. 計算每個 VM 的威脅等級
5. 使用 Matlab 的 fgoalattain 函數產生解答

資料生成的部分皆使用 Matlab 完成。資料集產生之後，隨機選取 80% 作為機器學習的訓練資料，再將資料隨機選取 20% 作為測試資料。機器學習使用 Scikit-learn 的 Ridge regression。

Matlab 的 fgoalattain 函數是用來計算多目標優化的問題，包括線性和非線性，完整方程式定義如公式 8 所示。weight、goal、b 和 beq 是向量。A 和 Aeq 是矩陣。F(x)、c(x)和 ceq(x)是回傳向量的函數，可以為非線性函數，x、lb、和 ub 可以是向量或矩陣。

$$minimize_{x,\gamma}\ \gamma\ \ such\ \ that \begin{cases} F(x) - weight \cdot \gamma \leq goal \\ c(x) \leq 0 \\ ceq(x) = 0 \\ A \cdot x \leq b \\ Aeq \cdot x = beq \\ lb \leq x \leq ub \end{cases} \tag{8}$$

其中，F(x)是要進行多目標優化的函數，我們要進行優化的目標是雲端總威脅和總

成本，表示如公式 9。

$$\begin{aligned} & min_{x,\gamma}\ \gamma\ \ subj.to \\ & F_1(X) - W_1\gamma \le F_1^* \\ & F_2(X) - W_2\gamma \le F_2^* \end{aligned} \tag{9}$$

$F_1(X)$是總威脅方程式，F_1^*是其目標值；$F_2(X)$是總成本方程式，F_2^*是其目標值，我們希望雲端的總威脅和總成本越低越好，因此$(F_1^*, F_2^*) = (0,0)$。總威脅和總成本的方程式如公式 10 所示。

$$\begin{cases} Total\ \ Threat = \sum_{i=1}^{n}\left(\left([1\ \ 1\ \ 1] - \sum_{j=1}^{q} x_{i,j} \times C_j\right) \times t_i\right) \\ Total\ \ Cost = X \times RC \end{cases} \tag{10}$$

權重W_1和W_2分別代表威脅和成本的重要性(權重)，用來衡量要較低的威脅或是較低的成本，其中$W = (W_1,\ W_2)$，$W_1 + W_2 = 1$，舉例來說，當設定W_1接近 0 的時候，意味著我們得到的對策會有高的成本搭配極低威脅的成效。當設定的權重不同，就會得到不同的解X，X的定義如公式 11，$x = \{0,1\}$，其中q為對策數量，n為虛擬機數量，解X會顯示每個虛擬機該使用哪一個對策。

$$X = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,q} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,q} \\ & \vdots & & \\ x_{n,1} & x_{n,2} & \cdots & x_{n,q} \end{bmatrix} \tag{11}$$

最後，將權重W、所有虛擬機的威脅等級t、解X，都轉為一維陣列整合成一筆資料，資料欄位內容如表 3.1 所示。創建多筆資料後輸出成 .csv 檔，方便後續使用 python 讀取進行機器學習。

表 3.1：資料集欄位內容

W_1	W_2	t_1^{SC}	t_1^{MP}	t_1^{DOS}	…	t_n^{SC}	t_n^{MP}	t_n^{DOS}	$x_{1,1}$	…	$x_{n,q}$

Chaper 4　實驗分析與討論

4.1　實驗介紹

本論文提出一個以機器學習回應雲端共存攻擊的系統，為了驗證所提出的模型具有實用性，我們設計以下 4 種實驗並和 Abazari 等人 [1] 的模型進行比較。第一個實驗是 12 個虛擬機的簡易雲端模擬環境，第二個實驗是 50 個虛擬機的雲端模擬環境，第三個實驗是 50 個虛擬機各種權重設定下的雲端總威脅總成本比較實驗，最後是固定 50 個虛擬機，資料量由 100 到 1000 來檢視我們的模型的準確度及效能。實驗環境如表 4.1：

表 4.1：實驗環境

CPU	Intel i7-6700
記憶體	8GB
作業系統	Windows10
工具	Matlab version R2017b、Python3.6

在準確度計算方面，我們設定 Matlab 產生的資料是正確答案，計算方式如公式 12。

$$準確度 = \frac{所選擇的對策和\ Matlab\ 相同的虛擬機個數}{虛擬機總個數} \tag{12}$$

舉例來說，VM_1 的威脅等級 t_1=(0.00, 0.5, 0.3)，Matlab 選擇的對策為對策 3 (僅能防禦 MP，耗費成本 0.2)，機器學習選擇對策 5 (能防禦 MP 和 DoS，耗費成本 0.3)，在威脅和成本方面 Matlab 和機器學習各有好壞，但由於我們假設 Matlab 產生的資料是正確答案，所以仍會判斷機器學習在此虛擬機的判斷是錯誤的。若在 50 個虛擬機的環境中，有 40 個虛擬機選擇的對策和 Matlab 相同，則準確度為 80%。

4.2 12 個虛擬機的簡易雲端模擬環境

在 12 個虛擬機的簡易雲端模擬環境中，我們產生 200 筆資料，其中 160 筆作為訓練資料，40 筆作為測試資料。Matlab 在 40 筆測試資料的平均回應時間為 3.12 秒，尚且還算是立即的回應。而機器學習在 160 筆訓練資料的情況下，準確度平均為 70%，測試時所需要的回應時間平均為 0.017 秒，如表 4.2 所示。

表 4.2：12 個虛擬機 40 筆測試資料的平均回應時間比較表

使用方法	平均回應時間(s)
Matlab	3.12
Machine Learning	0.017

我們從其中一筆權重 W = (0.5, 0.5)的測試資料中，列出 Matlab 和機器學習提供的回應對策所產生的雲端總威脅和總成本，顯示在表 4.3。

表 4.3：12 個虛擬機之總威脅和總成本比較

W = (0.5, 0.5)	Total Threat	Total Cost
Matlab	3.10	2.20
Machine Learning	3.31	1.92

在這麼少量虛擬機的雲端模擬環境中，機器學習在回應時間就比 Matlab 快了 183 倍，但準確度 70%略有不足，有可能是訓練資料不足，或是虛擬機個數太少所導致。若使用不適當的對策可能會白耗費成本，因此我們從隨機一筆測試資料中，隨機挑選 3 個虛擬機，在表 4.4 列出其威脅等級 t 以及使用的對策 x，來檢視機器學習的回應是否不適當。

表 4.4：12 個虛擬機威脅等級及使用對策比較表

VM_i	t_i (SC, MP, DoS)	Matlab x(i)	Machine Learning x(i)
VM_1	t_1(0.04, 0.56, 0.00)	x_1=(0,0,1,0,0,0)	x_1=(0,0,1,0,0,0)
VM_3	t_3(0.00, 0.42, 0.24)	x_3=(0,0,0,0,1,0)	x_3=(0,0,1,0,0,0)
VM_{11}	t_{11}(0.38, 0.00, 0.00)	x_{11}=(0,0,0,0,0,1)	x_{11}=(0,1,0,0,0,0)

在表 4.4 中，VM_1 的 MP 威脅等級很高，其餘很低，Matlab 推薦使用對策 3，機器學習也推薦相同對策。VM_3 的 MP 威脅等級較高，DOS 威脅等級較低，Matlab 推薦使用對策 5 (可同時防禦 MP 和 Dos，成本 0.3)，機器學習推薦對策 3 (只能防禦 MP，成本 0.2)，兩者都可以正確回應威脅等級較高的 MP 威脅，但機器學習使用的對策產生的成本較低、威脅較高。VM_{11} 的 SC 威脅等級偏高，其餘為 0，Matlab 推薦使用對策 6 (不進行任何動作)，機器學習推薦對策 2 (防禦 SC，成本 0.5)，SC 攻擊僅有對策 1 和對策 2 可以防禦，但這兩個對策成本都偏高，因此推測這是 Matlab 推薦不進行任何動作的原因。從上面三個例子中，可以看出機器學習雖然在 VM_3 和 VM_{11} 推薦了和 Matlab 不同的回應對策，但並非不適當的對策，機器學習推薦的對策也可以適當的回應攻擊，只是花費成本略有不同。

4.3 50 個虛擬機的雲端模擬環境

在 50 個虛擬機的雲端模擬環境中，我們產生 500 筆資料，其中 400 筆作為訓練資料，100 筆作為測試資料。Matlab 在 100 筆測試資料的平均回應時間為 48.23 秒，這已經不算是立即的回應，在實際的雲端環境中虛擬機個數只會更多，因此此模型的實用性有待改善。機器學習在 400 筆訓練資料的情況下，準確度平均為 85%，測試時所需要的回應時間平均為 0.021 秒，如表 4.5 所示。

我們一樣挑選一筆測試資料，列出 Matlab 和機器學習提供的回應對策所產生的雲端總威脅和總成本，顯示在表 4.6，該筆資料權重為 W = (0.6, 0.4)。

表 4.5：50 個虛擬機 100 筆測試資料的平均回應時間比較表

使用方法	平均回應時間(s)
Matlab	48.23
Machine Learning	0.021

表 4.6：50 個虛擬機之總威脅和總成本比較

W = (0.6, 0.4)	**Total Threat**	**Total Cost**
Matlab	25.7	8.0
Machine Learning	31.2	7.8

在 50 個虛擬機的雲端模擬環境中，機器學習在回應時間方面比 MATLAB 快 2296 倍，準確度為 85%，可見訓練資料和虛擬機個數的增加可以提升機器學習的準確度。我們一樣從隨機一筆測試資料中，挑選 3 個虛擬機，在表 4.7 中列出其威脅等級 t 以及使用的對策 x，來檢視機器學習的選擇的對策是否不適當。

表 4.7：50 個虛擬機威脅等級及使用對策比較表

VM_i	**t_i (SC, MP, DoS)**	**Matlab x(i)**	**Machine Learning x(i)**
VM_{12}	t_{12} (0.24, 0.56, 0.80)	x_{12}=(0,0,0,0,1,0)	x_{12}=(0,0,0,0,1,0)
VM_{36}	t_{36} (0.50, 0.83, 0.61)	x_{36}=(1,0,0,0,0,0)	x_{36}=(0,0,0,0,1,0)
VM_{43}	t_{43} (0.73, 0.42, 0.17)	x_{43}=(1,0,0,0,0,0)	x_{43}=(0,1,0,0,0,0)

在表 4.7 中，VM_{12} 的 MP 和 DOS 威脅等級偏高，Matlab 和機器學習都推薦對策 5。VM_{36} 的三個威脅等級都偏高， Matlab 推薦使用對策 1 (可同時防禦所有攻擊，成本 1)，機器學習推薦對策 5 (只能防禦 MP 和 DOS，成本 0.3)，兩者都可以正確回應威脅等級較高的 MP 和 DoS 威脅，但機器學習使用的對策無法防禦 SC，機器學習在此虛擬機產

生的成本較低、威脅較高。VM_{43} 的 SC 威脅等級較高、MP 稍微高、DoS 低，Matlab 推薦使用對策 1 (防禦所有威脅，成本 1)，機器學習推薦對策 2 (防禦 SC，成本 0.5)，仍可以適當地防禦最大的威脅。

和 12 個虛擬機的實驗結論相同，從在表 4.7 的三個虛擬機可以看出機器學習雖然在部分虛擬機推薦了和 Matlab 不同的回應對策，但並非推薦不適當的對策，機器學習推薦的對策也可以適當的回應威脅等級中最高的攻擊種類，只是總威脅和總成本略有差距。

4.4 雲端總威脅、總成本比較實驗

為了比較 Matlab 和機器學習，我們做了這個實驗，在 50 個虛擬機的環境中，我們使用一組固定的威脅等級，並使用不同的權重 W＝(0.1, 0.9) 到 W＝(0.9, 0.1) 每次增減 0.1。分別使用 Matlab 和機器學習產生回應對策，然後計算使用對策後的雲端總威脅和總成本，結果如圖 4.1：不同權重下的威脅與成本比較圖所示。

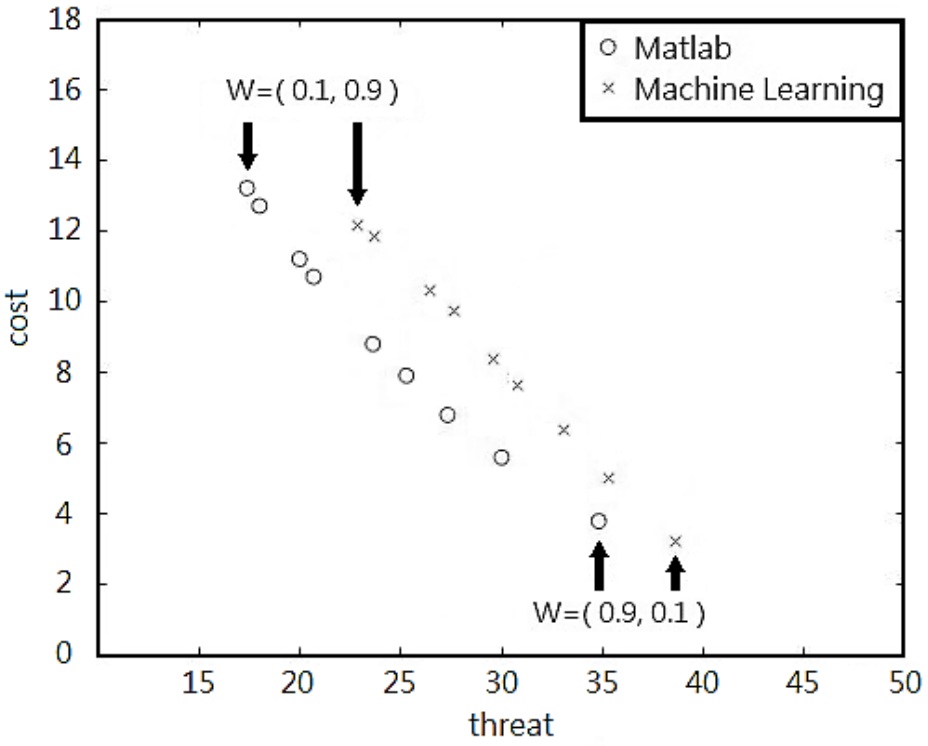

圖 4.1：不同權重下的威脅與成本比較圖

圖 4.1 中 Matlab 的點可視為帕雷托最優解 (Pareto optimal solutions) ，而機器學習的曲線也近似於帕雷托最優解，並不會偏離太多。在相同權重的設定下，機器學習的解有較高的威脅，但有較低的成本。

4.5 準確度及效能實驗

實驗四是為了確認我們的機器學習模型在資料數量的差異下對準確度和效能的影響。實驗是固定 50 個虛擬機的模擬雲端環境，資料量從 100 筆每次增加 100，直到 1000 筆資料。每次實驗皆取 20% 資料作為測試資料，由圖 4.2 顯示測試的準確度，由圖 4.3 顯示測試的平均回應時間。

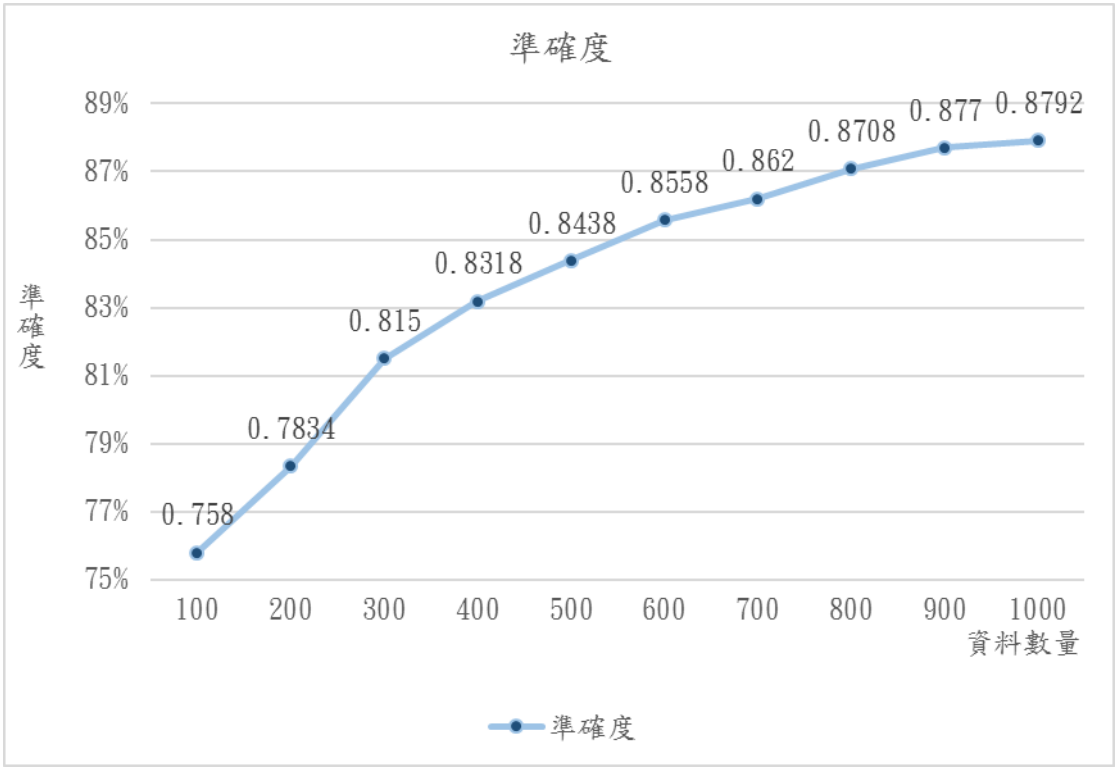

圖 4.2：準確度圖

由圖 4.2 可以看出資料量的增加讓準確度有明顯的提升，然而到 900~1000 筆資料時成長趨於緩和，因此可以推測 1000 筆資料後即使再增加資料，對準確度的提升也有限。若要在提升準確度，得要修改機器學習使用的演算法。

圖 4.2 的 200 筆資料準確度為 78.3%，與實驗 4.1 相比(12 個虛擬機 200 筆資料)，準確度提升約 8%，因此可推測虛擬機個數的增加也對準確度提升有幫助。

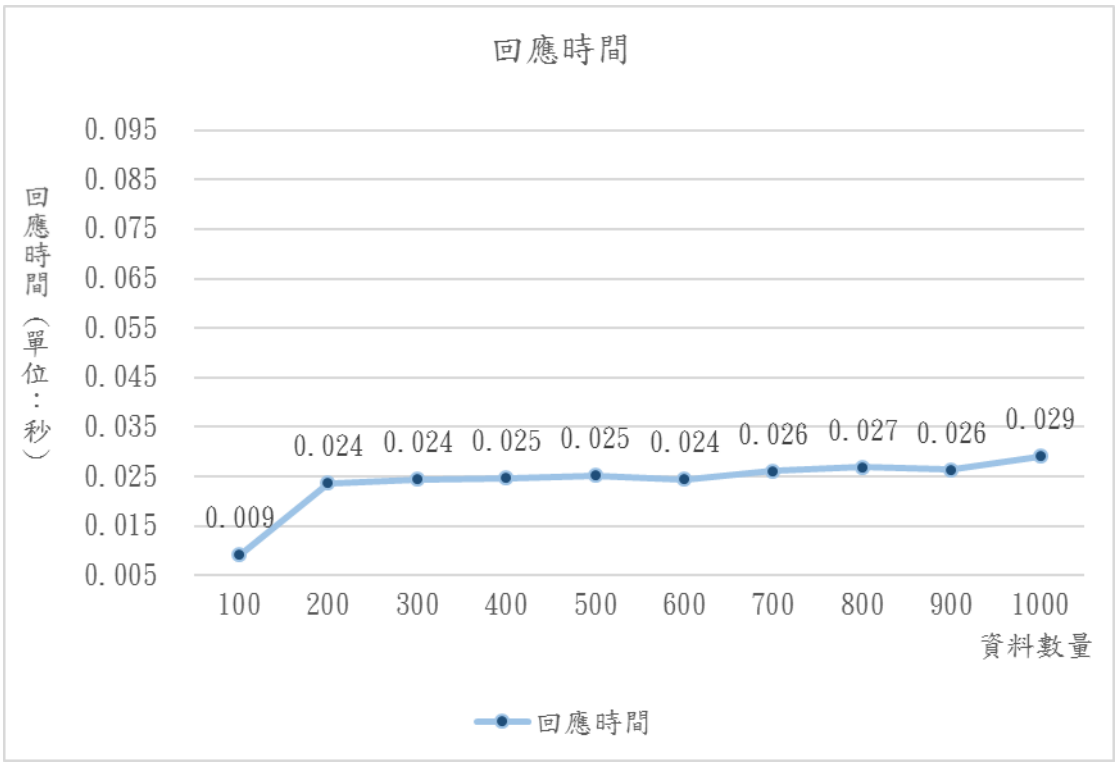

圖 4.3：回應時間圖

圖 4.3 顯示了回應時間，資料量的的增加並沒有讓回應時間有明顯增加，均在 0.03 秒內。

訓練資料量的增加可以提升準確度並且不影響測試時的回應時間，所以若使用在雲端入侵回應系統，可以事先由大量資料訓練好模型，實際使用時就可以快速的得到高準確度近似帕雷托最優解的解答。

Chaper 5　結論

雲端計算提供使用者方便、簡單、大量的軟硬體服務，是現在以及未來的趨勢，但雲端的虛擬化環境伴隨而來的風險，需要透過額外的雲端入侵回應系統來防禦。過去的研究已經將許多重要的因素加入考量，例如回應成本、威脅和虛擬機共存時間，已經是理論上完好的模型，但僅有計算所需的時間不足以應付實際環境，一但在雲端檢測到潛在的攻擊，立即回應是減少損害的重要關鍵，傳統的方法不足以應付實際狀況。

如今機器學習正火紅，被廣泛的應用在各種辨識、分析、歸類的應用中，而且都擁有優秀的成績。因此，我們想到將這兩項技術結合，使用機器學習來產生雲端入侵回應系統的回應對策。從實驗結果也可以看出，傳統方法和機器學習都可以得到針對當前環境適當的回應對策，傳統方法在總威脅和總成本的取捨下得到帕雷托最優解，而機器學習可以以高準確度得到近似的解。在回應時間方面，從實驗 4.2 的 12 個虛擬機到實驗 4.3 的 50 個虛擬機的雲端環境，傳統方法需要的回應時間增長了約 12 倍，機器學習僅增加 1.23 倍，資料筆數的增加和虛擬機量的增加也讓機器學習的準確度有所提升，可見在更大量虛擬機的雲端環境中，機器學習有更好的前途。

然而，機器學習是一門很深的學問，現有的演算法已五花八門，甚至還有深度學習，一定有比我們現在用的更適合應用在雲端入侵回應系統的模型，未來將朝此方向做研究，變換機器學習的模型或修改參數，以優化整體的準確率，並應用到實際的雲端環境中，來證明本研究的價值。

References

1. F. Abazari, M. Analoui, H. Takab: Multi-objective response to co-resident attacks in cloud environment. Int. J. Inf. Commun. Technol. Res. 9(3), 25–36 (2018)
2. M. Ali, S. U. Khan, and A. V. Vasilakos, "Security in cloud computing: Opportunities and challenges," Information Sciences, vol. 305, pp. 357–383, 2015.
3. Shahid Anwar et al., "Response option for attacks detected by intrusion detection system," in Software Engineering and Computer Systems (ICSECS), 2015 4th International Conference on, 2015, pp. 195-200.
4. Adam Bates et al., "On detecting co-resident cloud instances using network flow watermarking techniques," International Journal of Information Security, vol. 13, pp. 171-189, 2014
5. "CVE-2015-3456," Technical Report 2015
6. Farzaneh Abazari, Morteza Analoui, and Hassan Takabi, "Effect of anti-malware software on infectious nodes in cloud environment," Computers & Security, 2016.
7. Candid Wueest, "Security for Virtualization: Finding the Right Balance," Kaspersky Lab, 2012.
8. Candid Wueest, "Threats to virtual environments," Symantec, 2014.
9. Ron C Chiang, Sundaresan Rajasekaran, Nan Zhang, and H Howie Huang, "Swiper: Exploiting virtual machine vulnerability in third-party clouds with competition for I/O resources," Parallel and Distributed Systems, IEEE Transactions on, vol. 26, pp. 1732-1742, 2015
10. Venkatanathan Varadarajan, Thawan Kooburat, Benjamin Farley, Thomas Ristenpart, and Michael M Swift, "Resource-freeing attacks: improve your cloud performance (at your neighbor's expense," in Proceedings of the 2012 ACM conference on Computer and communications security, 2012, pp. 281-292.
11. Wanchun Dou, Qi Chen, and Jinjun Chen, "A confidence-based filtering method for DdoS attack defense in cloud environment," Future Generation Computer Systems, vol. 29, pp. 1838-1850, 2013.
12. A. O. F. Atya, Z. Qian, S.V. Krishnamurthy, T. L. Porta, P. McDaniel, L. Marvel: Malicious co-residency on the cloud: attacks and defense. In: IEEE INFOCOM 2017 – IEEE Conference on Computer Communications, Atlanta, GA, pp. 1–9 (2017)
13. W. Zhang, X. Jia, C. Wang, S. Zhang, Q. Huang, M. Wang, P. Liu: A comprehensive study of co-residence threat in multi-tenant public PaaS clouds. In: Lam, K.Y., Chi, C.H., Qing, S. (eds.) Information and Communications Security, ICICS, Lecture Notes in Computer Science, vol. 9977. Springer, Cham (2016)
14. M. Altunay, S. Leyffer, J. T. Linderoth, Z. Xie: Optimal response to attacks on the open science grid. Comput. Netw. 55(1), 61–73 (2011)

15. F. Abazari, M. Analoui: Exploring the effects of virtual machine placement on the transmission of infections in cloud. In: 7th International Symposium on Telecommunications, Tehran, pp. 278–282 (2014)
16. Jingzheng Wu, Liping Ding, Yuqi Lin, Nasro Min-Allah, and Yongji Wang, "Xenpump: a new method to mitigate timing channel in cloud computing," in Cloud Computing (CLOUD), 2012 IEEE 5th International Conference on, 2012, pp. 678--685.
17. Chun-Jen Chung, Pankaj Khatkar, Tiany Xing, Jeongkeun Lee, and Dijiang Huang, "NICE: Network intrusion detection and countermeasure selection in virtual network systems," Dependable and Secure Computing, IEEE Transactions on, 2013.
18. Swaminathan Balasubramanian, Matthew M Lobbes, Brian M O'connell, and Brian J Snitzer, "Automated Response to Detection of Threat to Cloud Virtual Machine," US Patent 20,160,094,568, March 2016.
19. Scikit-learn https://scikit-learn.org/stable/tutorial/machine_learning_map/index.html
20. Smitha and Squcciarini, Anna C Sundareswaran, "Detecting malicious co-resident virtual machines indulging in load-based attacks," Information and Communications Security, pp. 113--124, 2013.