

R + Keras + Shiny on Azure/AWS

Case : MNIST

Kristen Chan

2018.12.15



Hello !

I am Kristen Chan

Data Scientist

E-Commerce / Telecom

R-Ladies Taipei

Co-Organizer



Agenda

Shiny Application

1. Web Service -- Virtual Machine
 - a. Azure
 - i. Ubuntu 16.04 LTS
 - ii. RStudio Server / Shiny Server
 - b. AWS
 - i. Ubuntu
 - ii. RStudio Server / Shiny Server
2. Keras -- MNIST
3. Shiny
 - a. Introduction to Shiny
 - b. Shiny Application



Web Service



What is Azure ?

Microsoft Azure

聯絡銷售員: 0809090343 搜尋 我的帳戶 入口網站 登入

概觀 解決方案 產品 文件 定價 訓練 Marketplace 合作夥伴 支援 博客 部落格 其他

免費帳戶 >

首頁 / 概觀 / 何謂 Azure ?

何謂 Azure ?

Microsoft Azure 是一款不斷擴大的雲端服務組合，可協助組織解決商業挑戰。它可讓您自由地在大規模的全球網路上使用自己慣用的工具和架構來建置、管理及部署應用程式。

開始免費使用 >

已回答的熱門問題

所有人都能使用的雲端

我們相信，所有企業和組織，不論大小新舊，都應該能夠均享雲端帶來的成功機會。



Note <https://azure.microsoft.com/zh-tw/overview/what-is-azure/>

Kristen Chan 6



建立免費 Azure 帳戶

Microsoft Azure

聯絡銷售員: 080909343 搜尋 我的帳戶 入口網站 登入

概觀 解決方案 產品 文件 定價 訓練 Marketplace 合作夥伴 支援 部落格 其他

立即建立免費的 Azure 帳戶

提供 12 個月免費服務協助使用者入門

[開始免費使用 >](#)

[或立即購買 >](#)

The dashboard displays several key performance indicators (KPIs) and monitoring charts across different service categories:

- Resource Group:** Shows a bar chart for "My first Web App" with a value of 5.2x.
- Web Front End:** Includes a chart for "My first web app" showing CPU Percentage and Memory Percentage over the past week, with values around 2.88% and 1.75% respectively.
- Database:** Shows a chart for "My first database" with a value of 32 ms.
- Processes:** Shows a chart for "My first process" with a value of 23.2x.
- Cognitive Services:** Shows a chart for "My first cognitive service" with a value of 55000 ms.

On the left sidebar, the navigation menu includes:

- New
- All resources
- Resource groups
- App Services
- SQL Databases
- SQL Data warehouses
- MySQL (Preview)
- Virtual machines
- Load balancers
- Storage accounts
- Virtual networks
- ARM templates
- Active Directory
- Monitor
- Access Advisor
- Security Center
- Billing
- Help + support
- More services >

Note

<https://azure.microsoft.com/zh-tw/free/>

KristenChen 7



申請 Azure Pass

1

Microsoft Azure

MY ACCOUNT SIGN IN

Ready to get started?

Try Microsoft Azure Pass

We're offering an Azure Pass, so for a limited time period, you can try Azure for free

*No credit card required

Start >

Use the links below to learn more

[Redemption Process Guide](#)

[Azure Documentation](#)

[Explore Azure](#)

Note

<https://www.microsoftazurepass.com/>

Krigen Chan 8



申請 Azure Pass

2

Microsoft Azure

MY ACCOUNT SIGN OUT

The following Microsoft Account will be used for Azure Pass:

Given name:

Surname:

Microsoft Email:

If the above email address is incorrect, please [sign out](#) and redeem using the correct Microsoft Account

[Confirm Microsoft Account >](#)

© 2018 Microsoft

[Support](#) | [Privacy & Cookies](#) | [Terms of Use](#) | [Personal Data](#)



申請 Azure Pass

3

Microsoft Azure

MY ACCOUNT

SIGN OUT

The following Microsoft Account will be used for Azure Pass:

Given name:

Surname:

Microsoft Email: s

If the above email address is incorrect, please [sign out](#) and redeem using the correct Microsoft Account

Enter Promo code:

Enter your Azure Pass



Claim Promo Code



Azure VM -- Ubuntu Server 16.04 LTS

1 Microsoft Azure

儀表板

所有資源

No 資源 to display

開始使用

- 虛擬機器 在幾分鐘內佈建 Windows 及 Linux 虛擬機器
- App Service 建立適用於任何平台與裝置的 Web 與行動裝置應用程式
- SQL Database 受管理的關聯式資料庫即服務
- 儲存體 持久、高可用性及可大規模調整的儲存體
- Azure 入口網站 了解如何使用 Azure 入口網站

服務健康狀況

我的資源

克里斯·陳 11



Azure VM -- Ubuntu Server 16.04 LTS

The screenshot shows the Microsoft Azure portal interface. The left sidebar contains a navigation menu with items like '建立資源', '儀表板' (selected), '所有服務', '我的最愛', '所有資源', '資源群組', '應用程式服務', 'SQL 資料庫', 'SQL 資料倉儲', 'Azure Cosmos DB', '虛擬機器' (selected), '負載平衡器', '儲存體帳戶', '虛擬網路', 'Azure Active Directory', '監視', 'Advisor', '資訊安全中心', '成本管理 + 計費', and '說明 + 支援'. The main content area is titled '虛擬機器' and shows a list of resources. At the top of this list, there is a message: '訂用帳戶: 選擇了 2 個 (共 3 個) - 看不到任何訂閱嗎? 開啟 [目錄 + 訂用帳戶] 設定'. Below this message, there are four columns: '名稱' (Name), '類型' (Type), '狀態' (Status), '資源群組' (Resource Group), '位置' (Location), '維護狀態' (Maintenance Status), and '訂用帳戶' (Subscription). There are no resources listed under these columns.



Azure VM -- Ubuntu Server 16.04 LTS

3 Microsoft Azure

儀表板 > 虛擬機器 > 建立虛擬機器

搜尋資源、服務及文件

建立虛擬機器

預設目錄

5@hotmail.com 預設目錄

+ 新增 保留 ... 更多

依名稱篩選...

名稱 ↑

建立虛擬機器

基本 磁碟 網路 管理 客體設定 標籤 檢閱 + 建立

請建立執行 Linux 或 Windows 的虛擬機器。請從 Azure Marketplace 選取映像，或使用您自己的自訂映像。

完成「基本」索引標籤，然後檢閱並建立，以使用預設參數建虛擬機器，或檢閱每個索引標籤進行完全的自訂。

在尋找傳統 VM 嗎？ 從 Azure Marketplace 建立 VM

專家詳細資料

選取用以管理部署資源及成本的訂用帳戶。使用像資料夾這樣的資源群組來安排及管理您的所有資源。

* 訂用帳戶 ①

Azure Pass - 贊助

* 資源群組 ①

選取現有的...
新建

執行個體詳細資料

* 虛擬機器名稱 ①

* 區域 ①

美國西部 2

可用性選項 ①

不需要基礎結構備援

* 影像 ①

Ubuntu Server 16.04 LTS

瀏覽所有映像及磁碟

* 大小 ①

選取大小

ADMINISTRATOR 帳戶

驗證類型 ①

密碼 SSH 公開金鑰

檢閱 + 建立 上一個 下一步：磁碟 >

The screenshot shows the 'Create Virtual Machine' wizard in the Azure portal. On the left is a dark sidebar with various service icons. The main area has a light background. It's on the 'Basic' step of the wizard. Key fields shown include the region 'United States West 2' and the image 'Ubuntu Server 16.04 LTS'. Other visible fields include the VM name, resource group, and administrator account type (password selected). Buttons at the bottom include 'Review + Create' and 'Next: Disk'.



Azure VM -- Ubuntu Server 16.04 LTS

4 Microsoft Azure

儀表板 > 虛擬機器 > 建立虛擬機器

搜尋資源、服務及文件

建立虛擬機器

+ 新增 保留 ... 更多

依名稱篩選...

名稱

建立虛擬機器

基本 磁碟 網路 管理 客體設定 標籤 檢閱 + 建立

Azure VM 搭載一部作業系統磁碟及一部磁碟供短期儲存之用。您可以連結其他資料磁碟。VM 的大小決定您可以使用的儲存體類型，以及所能連結的資料磁碟數量。 [深入了解](#)

磁碟選項

* OS 磁碟類型 [標準 SSD](#)

資料磁碟

您可以另外為虛擬機器新增和設定資料磁碟，或連結現有磁碟。此 VM 也隨附暫存磁碟。

LUN	名稱	大小 (GiB)	磁碟類型	主機快取

建立並連結新的磁碟 連結現有磁碟

進階

檢閱 + 建立 上一個 下一步：網路 >



Azure VM -- Ubuntu Server 16.04 LTS

5 Microsoft Azure

儀表板 > 虛擬機器 > 建立虛擬機器

搜尋資源、服務及文件

建立虛擬機器

+ 新增 ⏪ 保留 ⏪ 更多

依名稱篩選...

名稱

建立虛擬機器

基本 磁碟 網路 管理 客體設定 標籤 檢閱 + 建立

請為您的 VM 設定全新或現有虛擬網路，以及在虛擬網路上存取您 VM 的方式。 [深入了解](#)

網路介面

在建立虛擬機器時，會為您建立網路介面。

設定虛擬網路

* 虛擬網路 [篩選虛擬機器](#) [建立新的](#)

公用 IP [無](#) [建立新的項目](#)

網路安全性群組 None 基本 進階

* 公用輸入連接埠 無 允許選取的連接埠

選取輸入連接埠 [選取一或多個連接埠](#)

根據預設，將會封鎖所有來自網際網路的流量。您可在 [VM] > [網路功能] 頁面中變更輸入連接埠規則。

加速的網路 開啟 關閉

選取的 VM 大小不支援高速網路。

LOAD BALANCING

您可以將此虛擬機器放在現有 Azure 負載平衡解決方案的後端集區中。 [深入了解](#)

檢閱 + 建立 上一個 下一步：管理 >



Azure VM -- Ubuntu Server 16.04 LTS

6 Microsoft Azure

儀表板 > 虛擬機器 > 建立虛擬機器

搜尋資源、服務及文件

建立虛擬機器

+ 新增 保留 更多

依名稱篩選...

名稱

請為您的 VM 設定全新的名稱。

網路介面

在建立虛擬機器時，會自動為您建立一個新的子網路。

設定虛擬網路

* 虛擬網路

公用 IP

網路安全性群組

* 公用輸入連接埠

選取輸入連接埠

加速的網路

LOAD BALANCING

您可以將此虛擬機器放置在一個或多個負載均衡器中。

建立虛擬網路

Microsoft Azure 虛擬網路服務可讓 Azure 資源在虛擬網路中安全地彼此通訊，虛擬網路運用邏輯，隔離您訂用帳戶專屬的 Azure 雲端。虛擬網路可以連線到其他虛擬網路或您的內部部署網路。[深入了解](#)

* 名稱

位址空間

虛擬網路的位址空間，使用 CIDR 標記法 (例如 192.168.1.0/24) 指定為一或多個位置首碼。

位址範圍 地址 重疊

10.0.0.0/16 10.0.0.0 - 10.0.255.255 (65536 位址) 無 [更多](#)

(0 位址)

子網路名稱 位址範圍 地址

default 10.0.0.0/16 10.0.0.0 - 10.0.255.255 (65536 位址) [更多](#)

(0 位址)

檢閱 + 建立 確定 捨棄

Kristen Chan 16



Azure VM -- Ubuntu Server 16.04 LTS

7 Microsoft Azure

儀表板 > 虛擬機器 > 建立虛擬機器

虛擬機器 優先選項

+ 新增 保留 ... 更多

依名稱篩選...

名稱

建立虛擬機器

基本 磁碟 網路 管理 客體設定 標籤 檢閱 + 建立

為您的 VM 設定監視及管理選項。

監視

開機診斷 開啟 關閉

OS 客體診斷 開啟 關閉

* 診斷儲存體帳戶 建立新的項目

識別

系統指派的受控識別 開啟 關閉

自動關機

啟用自動關機 開啟 關閉

備份

啟用備份 開啟 關閉

檢閱 + 建立 上一個 下一步：客體設定 >

KristenChen 17



Azure VM -- Ubuntu Server 16.04 LTS

8 Microsoft Azure

儀表板 > 虛擬機器 > 建立虛擬機器

搜尋資源、服務及文件

建立虛擬機器

預設目錄

+ 新增 保留 ... 更多

依名稱篩選...

名稱 ↑↓

標籤為成對的名稱和數值，可讓您透過將相同標籤套用至多個資源與資源群組，進而分類資源並檢視合併的帳單。 深入了解

請注意，若您在建立標籤後變更其他索引標籤上的資源設定，您的標籤將會自動更新。

金鑰 值 資源類型

要建立的所有資源

檢閱 + 建立 上一個 下一步：檢閱 + 建立 >



Azure VM -- Ubuntu Server 16.04 LTS

9 Microsoft Azure

儀表板 > 虛擬機器 > 建立虛擬機器

建立虛擬機器

驗證成功

依名稱篩選...

名稱 ↑↓

產品詳細資料

Ubuntu Server 16.04 LTS
由 Canonical
使用規定 | 隱私權原則

標準 H8
由 Microsoft
使用規定 | 隱私權原則

適用訂用帳戶點數 ⓘ
23.9231 TWD//小時
其他 VM 大小的價格

此供應項目未提供定價
如需詳細資訊，請檢視 定價詳細資料。

條款

按一下 [建立]，即表示我 (a) 同意上述 Marketplace 供應項目的相關法律條款及隱私權聲明；(b) 授權 Microsoft 向我目前的付款方式收取供應項目的相關費用，帳單週期與我的 Azure 訂用帳戶相同；並 (c) 同意 Microsoft 將我的連絡資料、使用方式及交易資訊提供給供應項目的提供者，以用於支援、帳單及其他交易活動。Microsoft 不提供第三方供應項目的權利。如需其他詳細資料，請參閱 Azure Marketplace 條款。

基本

訂用帳戶	Azure Pass - 贊助
資源群組	(新) ShinyServerbig
虛擬機器名稱	ShinyServerbig
區域	美國西部 2
可用性選項	不需要基礎結構備援
驗證類型	密碼
使用者名稱	kr.....5

建立 上一個 下一個 下載自動化的範本



Azure VM -- Ubuntu Server 16.04 LTS

10

✓ 您的部署已完成

前往資源

部署名稱: CreateVm-Canonical.UbuntuServer-16.04-LTS-2
訂用帳戶: Azure Pass - 贊助
資源群組: ShinyServerbig

部署詳細資料 (下載)
開始時間: 2018/12/14 下午7:07:57
期間: 3 分鐘 6 秒
相互關聯識別碼: 1cb389e4-3a3d-450a-91ce-ca3943269cb2

Microsoft Azure 儀表板 > CreateVm-Canonical.UbuntuServer-16.04-LTS-20181214182602 - 概觀 > ShinyServerbig

虛擬機器

概觀

活動記錄

存取控制 (IAM)

標籤

診斷並解決問題

設定

網路

磁碟

大小

安全性

延伸模組

持續傳遞 (預覽)

可用性設定組

組態

身分識別

屬性

鎖定

自動化指令碼

作業

搜尋 (Ctrl + F)

重新整理

電腦名稱: ShinyServerbig
狀態: 正在執行
位置: 美國西部 2
訂用帳戶 (變更): Azure Pass - 贊助
訂用帳戶識別碼: 9375e76e-0057-4962-bcae-6249984f9559

標籤 (變更)
按一下這裡即可新增標籤

顯示下列時間內的資料: 1 小時 | 6 小時 | 12 小時 | 1 天 | 7 天 | 30 天

CPU (平均)

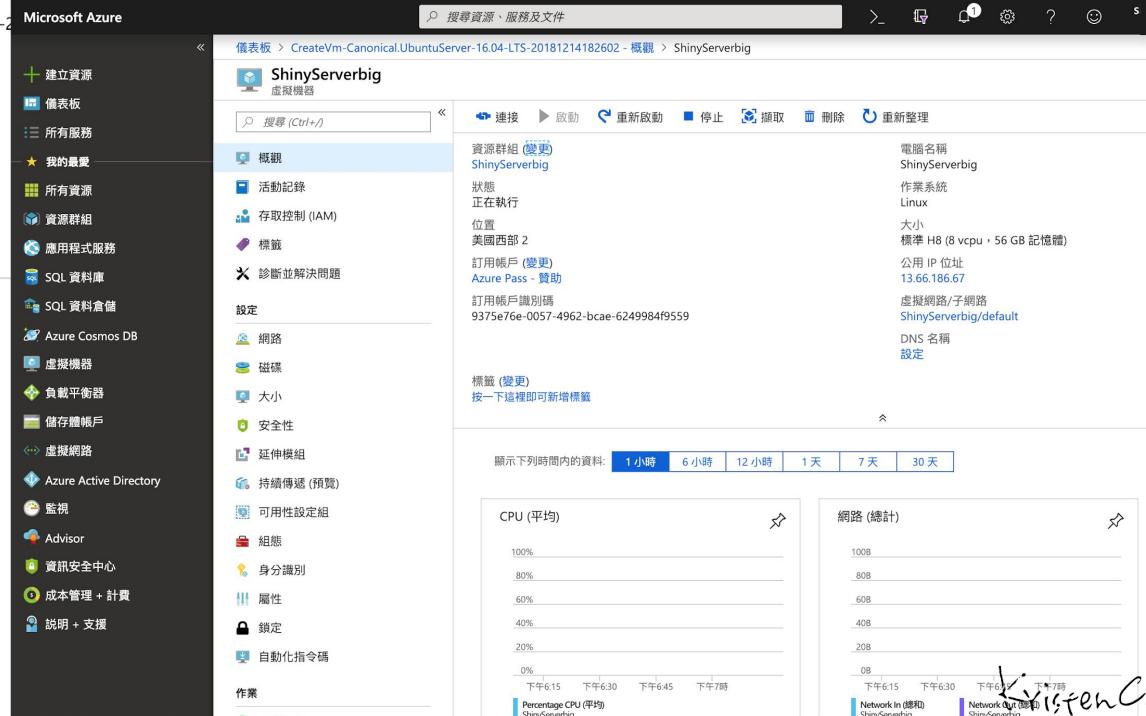
網路 (總計)

Percentage CPU (平均) ShinyServerbig

Network In (總和) ShinyServerbig

Network Out (總和) ShinyServerbig

Vivien Chen 20





Azure VM -- Ubuntu Server 16.04 LTS

11

固定 IP

儀表板 > CreateVm-Canonical.UbuntuServer-16.04-LTS-20181214182602 - 概觀 > ShinyServerbig

ShinyServerbig 虛擬機器

搜尋 (Ctrl+ /)

連接 啟動 重新啟動 停止 擷取 刪除 重新整理

資源群組 (變更)
ShinyServerbig
狀態
正在執行
位置
美國西部 2
訂用帳戶 (變更)
Azure Pass - 幫助
訂用帳戶識別碼
9375e76e-0057-4962-bcae-6249984f9559

電腦名稱
ShinyServerbig
作業系統
Linux
大小
標準 H8 (8 vcpu, 56 GB 記憶體)
公用 IP 位址
13.66.186.67

虛擬網路/子網路
ShinyServerbig/default
DNS 名稱
設定

標籤 (變更)
按一下這裡即可新增標籤

顯示下列時間內的資料: 1 小時 6 小時 12 小時 1 天 7 天 30 天

CPU (平均) 網路 (總計)

Percentage CPU (平均)
ShinyServerbig

下午 6:30 下午 6:45 下午 7:00 下午 7:15

下午 6:30 下午 6:45 下午 7:00 下午 7:15

Network In (總和)
ShinyServerbig Network Out (總和)
ShinyServerbig

網路

磁碟

大小

安全性

延伸模組

持續傳遞 (預覽)

可用性設定組

組態

身分識別

屬性

鎖定

自動化指令碼

作業

儀表板 > CreateVm-Canonical.UbuntuServer-16.04-LTS-20181214182602

ShinyServerbig-ip - 組態 公用 IP 位址

儲存 捨棄

已建立關聯的虛擬!

指派
 動態 靜態

IP 位址 13.66.186.67

閒置逾時 (分鐘) 0

DNS 名稱標籤 (選用)

別名記錄集
要密切追蹤此公用 IP 位址嗎? 訂閱帳戶
+ 建立別名記錄

沒有結果。



Azure VM -- Ubuntu Server 16.04 LTS

12

port: 8787 / 3838

儀表板 > CreateVm-Canonical.UbuntuServer-16.04-LTS-20181214182602 - 概觀 > ShinyServerbig - 網路

ShinyServerbig - 網路

虛擬機器

搜尋 (Ctrl+ /)

連接網路介面 將網路介面中斷連結

網路介面: shinyserverbig674 **有效安全性規則** 拓撲

虛擬網路/子網路: ShinyServerbig/default 公用 IP: 13.66.186.67 私人 IP: 10.0.0.4 加速的網路: 已停用

應用程式安全性群組

設定應用程式安全性群組

輸入連接埠規則

網路安全性群組 ShinyServerbig-nsg (連結到網路介面: shinyserverbig674)

新增輸入連接埠規則

優先順序	名稱	連接埠	通訊協定	來源	目的地	動作
65000	AllowVnetInBound	任何	任何	VirtualNetwork	VirtualNetwork	允許
65001	AllowAzureLoadBalancerInBound	任何	任何	AzureLoadBalanc...	任何	允許
65500	DenyAllInBound	任何	任何	任何	任何	拒絕

輸出連接埠規則

網路安全性群組 ShinyServerbig-nsg (連結到網路介面: shinyserverbig674)

新增輸出連接埠規則

優先順序	名稱	連接埠	通訊協定	來源	目的地	動作
65000	AllowVnetOutBound	任何	任何	VirtualNetwork	VirtualNetwork	允許
65001	AllowInternetOutBound	任何	任何	任何	Internet	允許

網路

- 磁碟
- 大小
- 安全性
- 延伸模組
- 持續傳遞 (預覽)
- 可用性設定組
- 組態
- 身分識別
- 屬性
- 鎖定
- 自動化指令碼
- 作業

新增輸入安全規則

ShinyServerbig-nsg

基本

* 來源 Any

* 來源連接埠範圍 *

* 目的地 Any

* 目的地連接埠範圍 8787

* 通訊協定 Any TCP UDP

* 動作 允許 拒絕

* 優先順序 100

* 名稱 Port_8787

描述

新增

新增輸入安全規則

ShinyServerbig-nsg

基本

* 來源 Any

* 來源連接埠範圍 *

* 目的地 Any

* 目的地連接埠範圍 3838

* 通訊協定 Any TCP UDP

* 動作 允許 拒絕

* 優先順序 110

* 名稱 Port_3838

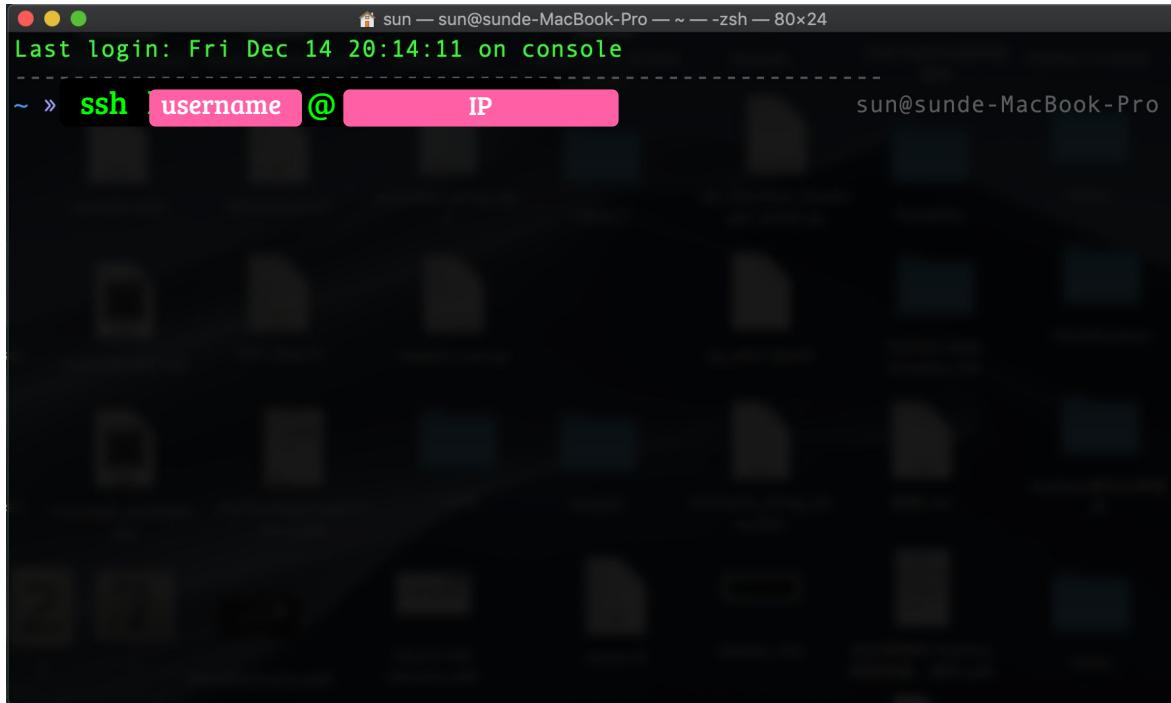
描述

新增

Kristen Chan



SSH



A terminal window on a Mac OS X desktop. The title bar says "sun — sun@sunde-MacBook-Pro — ~ — zsh — 80x24". The command line shows "Last login: Fri Dec 14 20:14:11 on console" followed by a dashed line. Below it, the command "ssh username @ IP" is entered, with "username" and "IP" highlighted in pink. To the right, the prompt "sun@sunde-MacBook-Pro" is visible.



新增輸入安全性規則
ShinyServerbig-nsg

基本

* 來源: Any

* 來源連接埠範圍: *

* 目的地: Any

* 目的地連接埠範圍: 22

* 通訊協定: TCP

* 動作: 允許

* 優先順序: 120

* 名稱: Port_22

描述:

What is AWS ?

The screenshot shows the AWS homepage with a blue gradient background. At the top, there's a dark navigation bar with the AWS logo, the text "re:Invent 2018" and various menu items like "產品", "解決方案", "定價", etc., followed by a search bar and account-related links.

In the center, there's a large white cloud icon above the text "雲端運算與 Amazon Web Services". Below this, a paragraph explains what AWS is and how it can be used. At the bottom, there's a prominent yellow "免費試用" (Free Trial) button with the text "建立免費帳戶" (Create Free Account) below it.

Note <https://aws.amazon.com/tw/what-is-aws/>

申請 AWS Credits



AWS EC2

聯絡銷售人員 支援 中文(繁體) 我的帳戶 建立 AWS 帳戶

re:Invent 2018 產品 解決方案 定價 了解 合作夥伴網路 AWS Marketplace 探索更多

產品與服務

- Amazon EC2 >
- 產品詳細資訊 >
- 入門 >
- 開發人員資源 >
- 常見問答集 >
- Amazon EC2 Run Command >
- 定價 >
- 相關連結
- Amazon EC2 Spot 執行個體
- Amazon EC2 預留執行個體
- Amazon EC2 專用主機
- Amazon EC2 專用執行個體
- Amazon EC2 Elastic GPU
- Windows 執行個體
- VMware Cloud on AWS
- Systems Manager
- Server Migration Services

Amazon EC2 定價

使用隨需執行個體時，您只需要支付所使用的 EC2 執行個體的費用。使用隨需執行個體可讓您不用考慮計劃、採購和維護硬體的成本和複雜性，並可將常見高額固定成本轉換為較小的可變成本。

隨需定價

隨需執行個體可讓您依小時或秒數 (最低限制 60 秒) 支付運算容量，無須簽訂長期合約。因此，您可以不用考慮計劃、採購和維護硬體的成本和複雜性，並可將常見高額固定成本轉換為較小的可變成本。

以下定價包括在指定作業系統上執行私有和公用 AMI 的費用 (「Windows 使用量」價格適用於 Windows Server 2003 R2、2008、2008 R2、2012、2012 R2 和 2016)。Amazon 還為您提供適用於執行於 Microsoft Windows 搭配 SQL Server 的 Amazon EC2、執行於 SUSE Linux Enterprise Server 的 Amazon EC2、執行於 Red Hat Enterprise Linux 的 Amazon EC2 及執行於 IBM 的 Amazon EC2 的其他執行個體，它們的定價不同。

開始免費使用 AWS

[建立免費帳戶](#)

AWS 免費方案包括為期一年每月 750 小時的 Linux 和 Windows t2.micro 執行個體使用量。若要繼續符合免費方案資格，您只能使用 EC2 微型執行個體。

[查看 AWS 免費方案詳細資訊 »](#)

Linux	RHEL	SLES	Windows	採用 SQL Standard 費率的 Windows	採用 SQL Web 費率的 Windows
採用 SQL Enterprise 費率的 Windows	採用 SQL Standard 費率的 Linux	採用 SQL Web 費率的 Linux			
採用 SQL Enterprise 費率的 Linux					

區域: 美國東部 (俄亥俄) ▾

Note <https://aws.amazon.com/tw/ec2/pricing/on-demand/>

RStudio Server

1 Install R

DigitalOcean Products Customers Community Pricing

Q Docs Support Log In Sign Up

Subscribe Share

Contents Prerequisites Step 1 — Installing R Step 2 — Installing R Packages from CRAN

Posted September 19, 2016 287.9k APPLICATIONS R UBUNTU UBUNTU 16.04

Melissa Anderson

Mark as Complete Not using Ubuntu 16.04? Choose a different version:

Introduction

R is a popular open source programming language that specializes in statistical computing and graphics. It is widely used for developing statistical software and performing data analysis. R is easily extensible, and the community is known for continuously adding user-generated packages for specific areas of study, which makes it applicable to many fields.

Note

<https://www.digitalocean.com/community/tutorials/how-to-install-r-on-ubuntu-16-04-2>

Krisjen Chan 27

RStudio Server

2 Install R



A terminal window titled "sun — kristen625@ShinyServerbig: ~ — ssh k" with a session ID "5@1" and a size of "7 — 80x24". The window shows a series of terminal commands for installing R:

- 1 sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys E298A3A825C0D65DFD57CBB651716619E084DAB9
- 2 sudo add-apt-repository 'deb [arch=amd64,i386] https://cran.rstudio.com/bin/linux/ubuntu xenial/'
- 3 sudo apt-get update
- 4 sudo apt-get install r-base

RStudio Server

3 Open R

```
sun — kristen625@ShinyServerbig: ~ — ssh k      5@1      :7 — 80x24
kristen@ShinyServerbig:~$ sudo -iR
R version 3.4.4 (2018-03-15) -- "Someone to Lean On"
Copyright (C) 2018 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

>
```



RStudio Server



Install RStudio Server

Studio

Products Resources Pricing About Us Blogs

Download RStudio Server v1.1.463

RStudio Server enables you to provide a browser based interface to a version of R running on a remote Linux server, bringing the power and productivity of the RStudio IDE to server-based deployments of R.

Do you need support or a commercial license? Compare our commercial and open source products.

[LEARN MORE](#)

Learn Why Enterprise Customers Use RStudio Server Pro

Need Enterprise Grade Security, Support, & More?

Choose your Linux Platform

Debian/Ubuntu RedHat/CentOS openSUSE/SLES Other Platforms

Prerequisites

RStudio Server v1.1 requires Debian version 8 (or higher) or Ubuntu version 12.04 (or higher).

Installing R

RStudio requires a previous installation of R version 3.0.1 or higher. To install the latest version of R you should first add the CRAN repository to your system as described here:

- Debian Packages for R

Note <https://www.rstudio.com/products/rstudio/download-server>

Try RStudio Server Pro for free

RStudio Server

5

Install RStudio Server

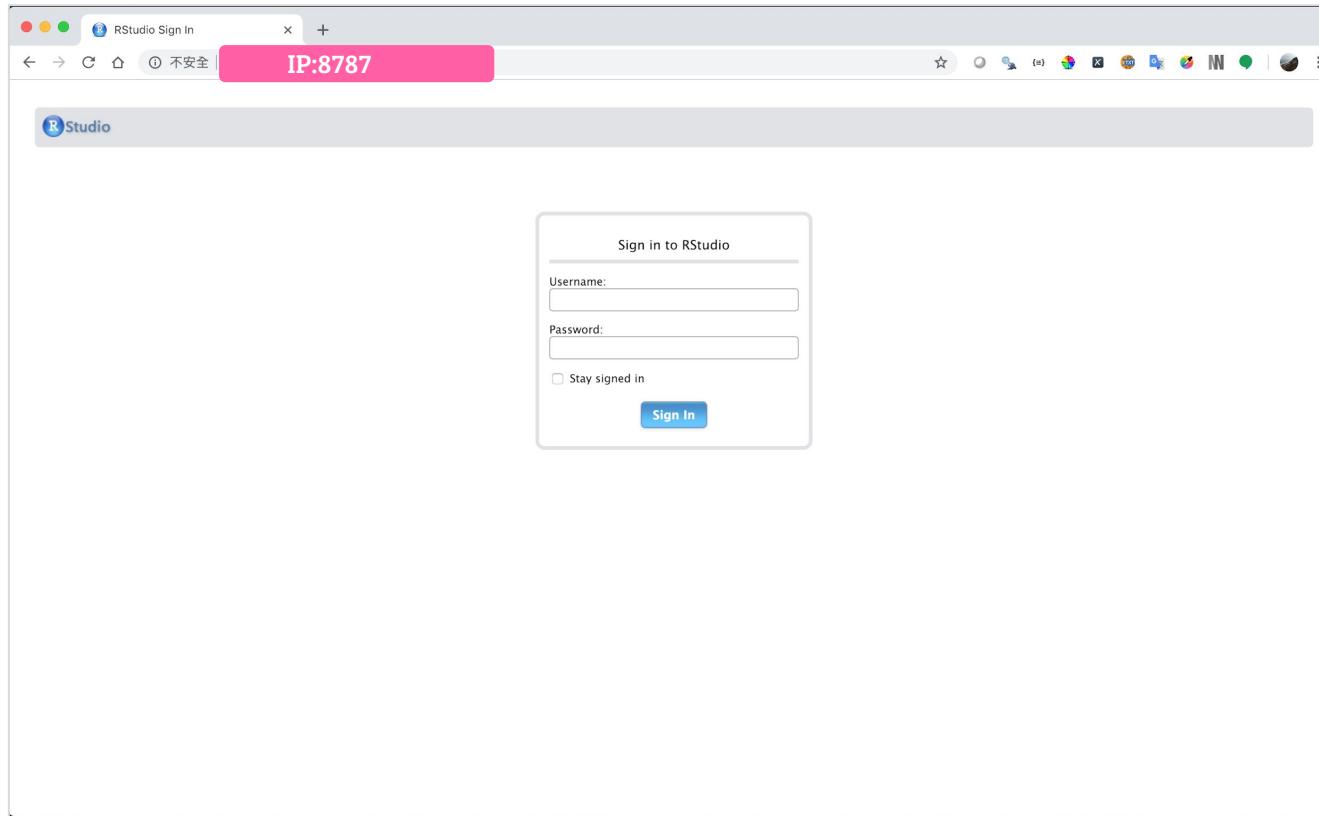


A terminal window titled "sun — kristen625@ShinyServerbig: ~ — ssh k" with a session ID "5@1" and a size of "7 — 80x24". The window shows three numbered steps for installing RStudio Server:

- 1 sudo apt-get install gdebi-core
- 2 wget https://download2.rstudio.org/rstudio-server-1.1.463-amd64.deb
- 3 sudo gdebi rstudio-server-1.1.463-amd64.deb

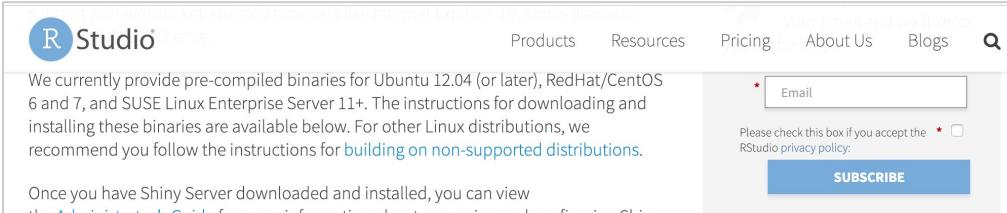
RStudio Server

6 Open RStudio Server



Shiny Server

1 Install Shiny Server



We currently provide pre-compiled binaries for Ubuntu 12.04 (or later), RedHat/CentOS 6 and 7, and SUSE Linux Enterprise Server 11+. The instructions for downloading and installing these binaries are available below. For other Linux distributions, we recommend you follow the instructions for [building on non-supported distributions](#).

Once you have Shiny Server downloaded and installed, you can view the [Administrator's Guide](#) for more information about managing and configuring Shiny Server, or use the [mailing list](#) to get support.

Download RStudio Shiny Server v1.5.9.923

Ubuntu 14.04 or later	RedHat/CentOS 6 & 7	SLES 12	Other Platforms
-----------------------	---------------------	---------	-----------------

Ubuntu 14.04 or later

We currently only provide a pre-built binary for the 64-bit architecture. Running on other architectures will require [building from source](#).

Before installing Shiny Server, you'll need to install R and the Shiny package. To install the latest version of R you should first add the CRAN repository to your system as described here:

- [Ubuntu Packages for R](#)

You can then install R using the following command:

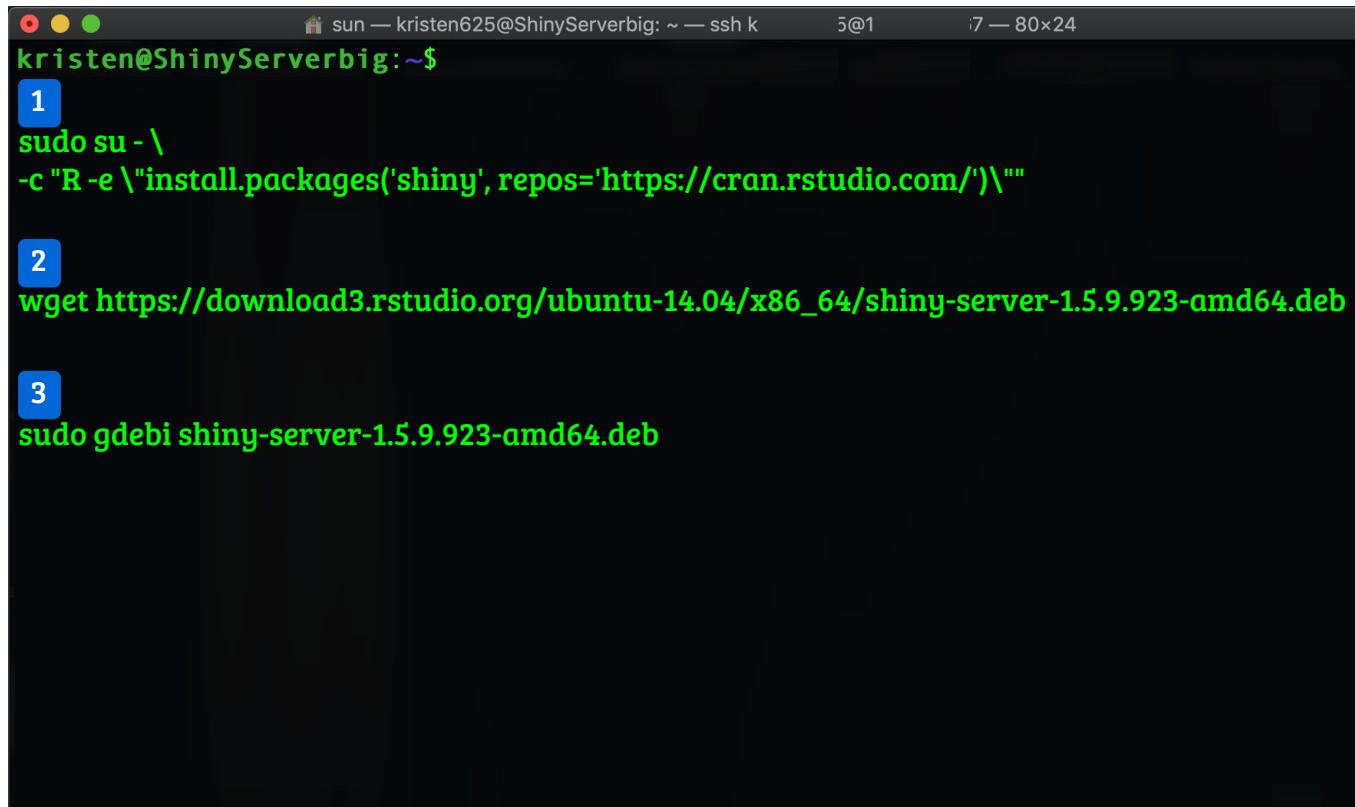
```
$ sudo apt-get install r-base
```

Note

<https://www.rstudio.com/products/shiny/download-server/>

Shiny Server

2 Install Shiny Server



A terminal window titled "sun — kristen625@ShinyServerbig: ~ — ssh k" with a session ID "5@1" and a size of "7 — 80x24". The window shows three numbered steps:

- 1 `sudo su - \ -c "R -e \"install.packages('shiny', repos='https://cran.rstudio.com/')\""`
- 2 `wget https://download3.rstudio.org/ubuntu-14.04/x86_64/shiny-server-1.5.9.923-amd64.deb`
- 3 `sudo gdebi shiny-server-1.5.9.923-amd64.deb`

Shiny Server

3 Open Shiny Server

Welcome to Shiny Server!

If you're seeing this page, that means Shiny Server is installed and running. **Congratulations!**

What's Next?

Now you're ready to setup Shiny — if you haven't already — and start deploying your Shiny applications.

If you see a Shiny application running on the right side of this page, then Shiny is configured properly on your server and already running an example. Bravo! You can see this application on your server at </sample-apps/hello/>.

If you see a gray box or an error message, then there's a bit more work to do to get Shiny running fully. You can continue with the installation instructions or use the [Admin Guide](#) for more information. If you're seeing an error message in the panel to the right, you can use it to help diagnose what may be wrong. If you think Shiny is installed and setup properly and things still aren't working, you can look in the Shiny Server log which may have more information about what's wrong. By default, the log is stored in `/var/log/shiny-server.log`.

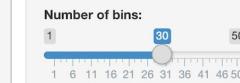
If you're really stuck and you've read the relevant sections in the [Admin Guide](#) then please ask for help on our [RStudio Community forum](#).

rmarkdown

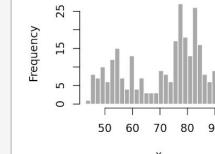
Once you have Shiny working properly (the top application on the right sidebar), you can optionally proceed to setup rmarkdown to enable your server to host Shiny docs using the `rmarkdown` package.

Once you have `rmarkdown` installed, the lower example to the right should

It's Alive!



Histogram of x

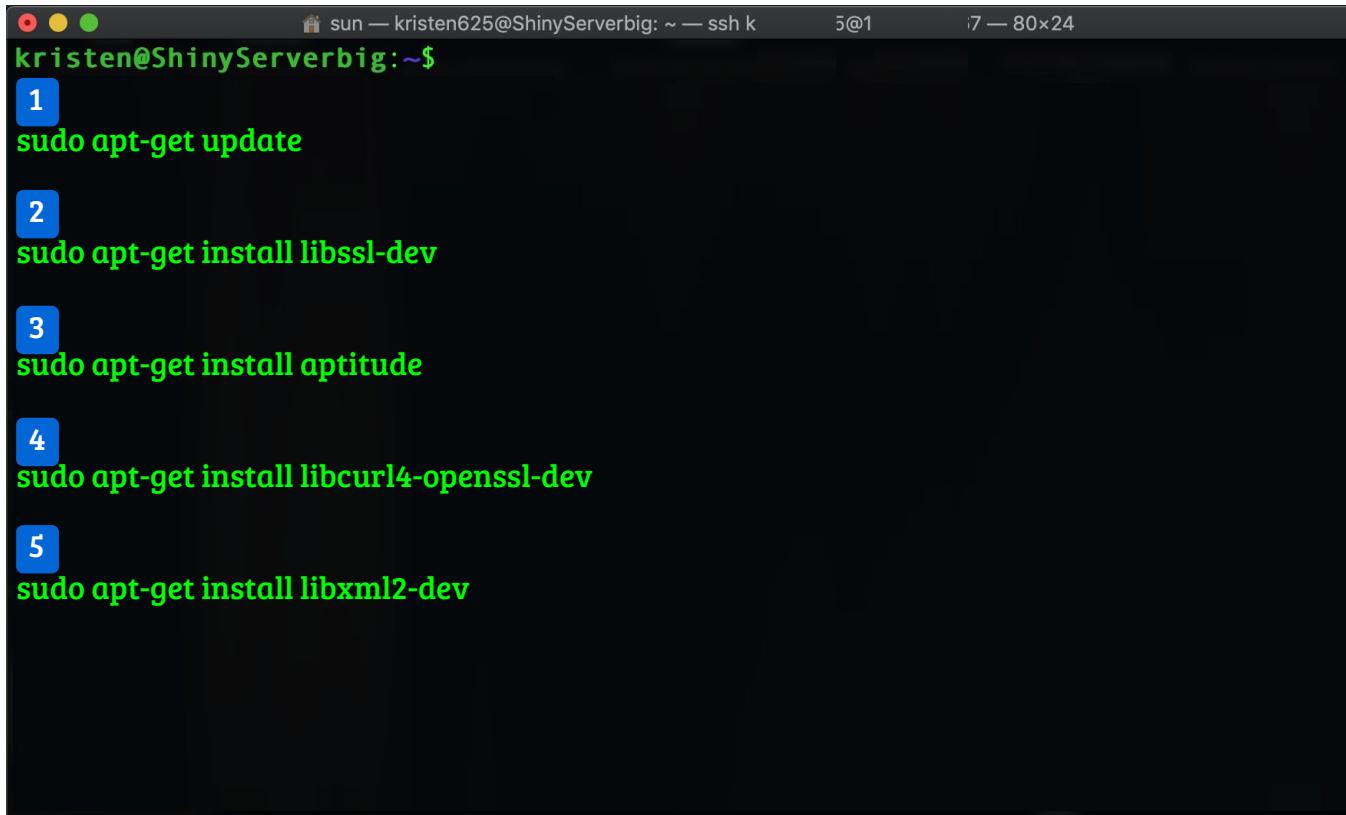


When Shiny is properly configured on your server, you'll see a Shiny app above.

An error has

安裝套件的各種雷

1 不斷的安裝

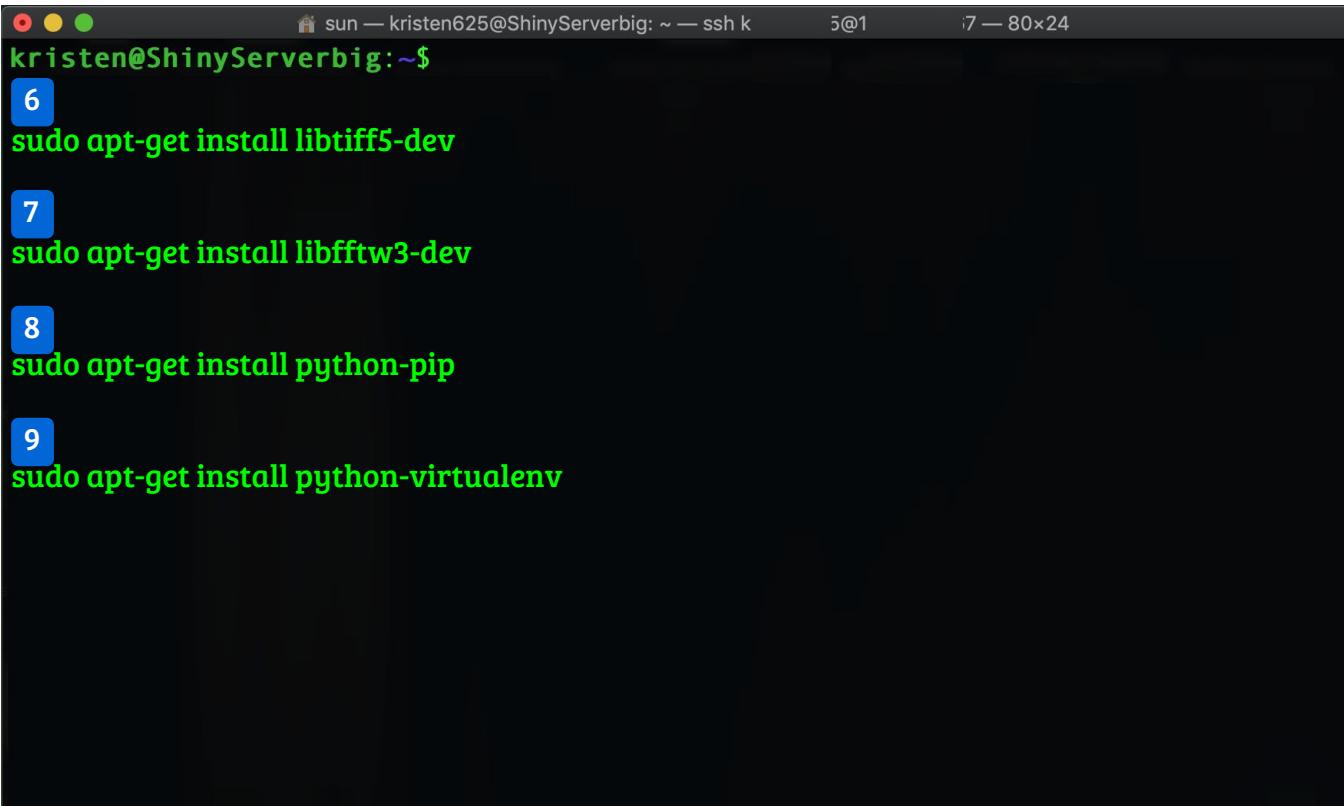


A screenshot of a terminal window titled "sun — kristen625@ShinyServerbig: ~ — ssh k". The window has three tabs open, with the current tab being "1" at index 1. The terminal shows the following sequence of commands:

- 1 sudo apt-get update
- 2 sudo apt-get install libssl-dev
- 3 sudo apt-get install aptitude
- 4 sudo apt-get install libcurl4-openssl-dev
- 5 sudo apt-get install libxml2-dev

安裝套件的各種雷

1 不斷的安裝



A screenshot of a terminal window titled "sun — kristen625@ShinyServerbig: ~ — ssh k". The window has three tabs labeled "5@1", "7", and "80x24". The terminal shows the following sequence of commands:

- 6 sudo apt-get install libtiff5-dev
- 7 sudo apt-get install libfftw3-dev
- 8 sudo apt-get install python-pip
- 9 sudo apt-get install python-virtualenv

安裝套件的各種雷

2 進去 R 繼續裝

```
sun — kristen625@ShinyServerbig: ~ — ssh k      5@1      :7 — 80x24
kristen@ShinyServerbig:~$ sudo -i R
1
> install.packages("devtools")

2
> install.packages('shiny')

3
> install.packages('shinydashboard')

4
> install.packages('stringi')

5
> source('http://bioconductor.org/biocLite.R')
> biocLite('EBImage')
```

安裝套件的各種雷

2 進去 R 繼續裝

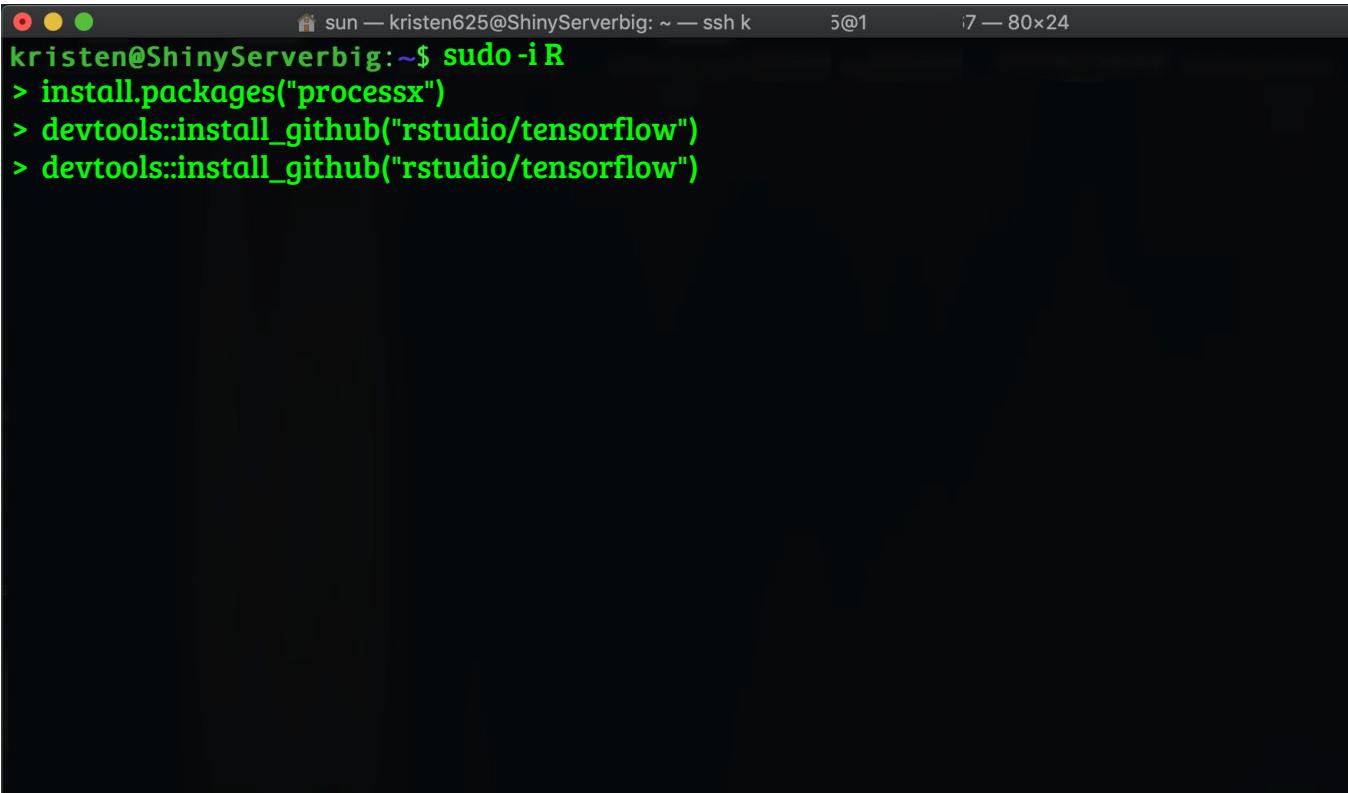
```
sun — kristen625@ShinyServerbig: ~ — ssh k 5@1 :7 — 80x24
kristen@ShinyServerbig:~$ sudo -i R
6
> devtools::install_github("rstudio/tensorflow")
> library(tensorflow)
> install_tensorflow()

7
> devtools::install_github("rstudio/keras")
> library(keras)
> install_keras()
```

安裝套件的各種雷

Note 安裝 tensorflow 可能遇到的錯誤

```
1 ERROR: dependency 'processx' is not available for package 'tensorflow'  
2 * removing 'C:/Users/Administrator/Documents/R/win-library/3.2/tensorflow'  
3 Error: Command failed (1)
```



A screenshot of a terminal window titled 'sun — kristen625@ShinyServerbig: ~ — ssh k'. The window shows the following R session:

```
kristen@ShinyServerbig:~$ sudo -i R  
> install.packages("processx")  
> devtools::install_github("rstudio/tensorflow")  
> devtools::install_github("rstudio/tensorflow")
```

安裝套件的各種雷

Note 測試一下你的 tensorflow 可以用

```
sun — kristen625@ShinyServerbig: ~ — ssh k      5@1      :7 — 80x24
kristen@ShinyServerbig:~$ sudo -i R
> library(tensorflow)
> sess = tf$Session()
> hello <- tf$constant("Hello, TensorFlow!")
> sess$run(hello)
> a <- tf$constant(10L)
> b <- tf$constant(32L)

> sess$run(a + b)
> sess$close()
```

Keras -- MNIST

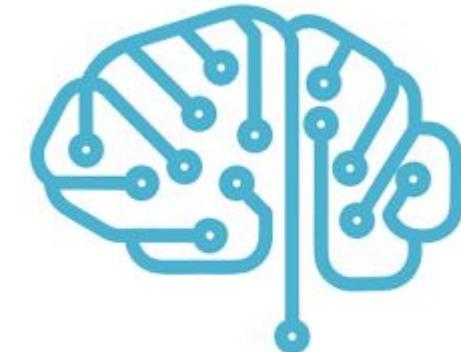
Why Keras ?!

PYTORCH

K Keras

theano

Caffe



Why Keras ?!

The screenshot shows a web browser displaying an article from R-bloggers. The URL in the address bar is <https://www.r-bloggers.com/deep-learning-in-satellite-imagery/>. The page title is "Deep learning in Satellite imagery". The author is Damian Rodziewicz, and the date is December 4, 2018. The article discusses the use of satellite imagery in deep learning. Below the main content, there is a sidebar with social sharing buttons for Facebook and Twitter, and sections for "RECENT POPULAR POSTS" and "MOST VISITED ARTICLES OF THE WEEK". The sidebar also includes links for "SEARCH R-BLOGGERS" and "RSS". The overall layout is clean and professional.

- ◊ R + keras + shiny
分析衛星遙測照片
- ◊ 文章中有提到實際
R 程式的網址

Why Keras ?!

- 開放原始碼
- Python的深度學習函式庫
- 內建非常多常用的深度學習的類神經網路元件
- 少少的程式碼就可建構出複雜的深度學習網路
- 比 **TensorFlow** 和 **Theano** 更簡潔、更具可讀性



MNIST

◦ 手寫數字的資料集

- Training data : 60,000 筆 (mnist.train)
- Test data : 10,000 筆 (mnist.test)
- X(自變數): images
- Y(目標變數): labels
- 圖片 : 28 pixel x 28 pixel
 $28 \times 28 = 784$ 個數字

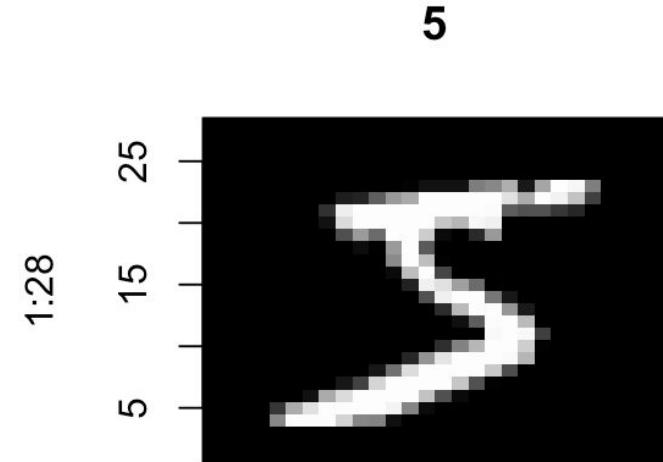


MNIST : Plot in R

```
#===== Import package =====
library(tensorflow)
library(keras)

#===== Import Images =====
mnist <- dataset_mnist()
c(train_images, train_labels) %<-% mnist$train
c(test_images, test_labels) %<-% mnist$test

#---- Plot in R : MNIST Digits----
idx = 1
im <- train_images[idx,,]
im <- t(apply(im, 2, rev))
image(1:28, 1:28, im, col=gray((0:255)/255) ,
      xaxt="n", main=paste(train_labels[idx]))
```



1:28

MNIST : Data Prepare

```
#===== Images Prepare =====
# reshape
train_images <- array_reshape(train_images, c(nrow(train_images), 784))
test_images <- array_reshape(test_images, c(nrow(test_images), 784))

# rescale
train_images <- train_images / 255
test_images <- test_images / 255

# one-hot encode
train_labels <- to_categorical(train_labels, 10)
test_labels <- to_categorical(test_labels, 10)
```

MNIST : Data Prepare

```
#===== Images Prepare =====  
# reshape  
train_images <- array_reshape(train_images, c(nrow(train_images), 784))  
test_images <- array_reshape(test_images, c(nrow(test_images), 784))  
  
# rescale  
train_images <- train_images / 255  
test_images <- test_images / 255  
  
# one-hot encode  
train_labels <- to_categorical(train_labels, 10)  
test_labels <- to_categorical(test_labels, 10)
```

```
> dim(train_images)  
[1] 60000      28      28
```



```
> dim(train_images)  
[1] 60000      784
```

MNIST : Data Prepare

```
#===== Images Prepare =====  
# reshape  
train_images <- array_reshape(train_images, c(nrow(train_images), 784))  
test_images <- array_reshape(test_images, c(nrow(test_images), 784))
```

```
# rescale  
train_images <- train_images / 255  
test_images <- test_images / 255
```

```
# one-hot encode  
train_labels <- to_categorical(train_labels, 10)  
test_labels <- to_categorical(test_labels, 10)
```



- ◊ 標準化
 - 除以 255
 - 讓值介在 0-1 之間

MNIST : Data Prepare

```
#===== Images Prepare =====
# reshape
train_images <- array_reshape(train_images, c(nrow(train_images), 784))
test_images <- array_reshape(test_images, c(nrow(test_images), 784))

# rescale
train_images <- train_images / 255
test_images <- test_images / 255

# one-hot encode
train_labels <- to_categorical(train_labels, 10)
test_labels <- to_categorical(test_labels, 10)
```

```
> dim(train_labels)
[1] 60000
> train_labels[1]
[1] 5
```



```
> dim(train_labels)
[1] 60000      10
> train_labels[1,]
[1] 0 0 0 0 0 1 0 0 0 0
```

Keras -- Activation Functions

◊ 常用的函數

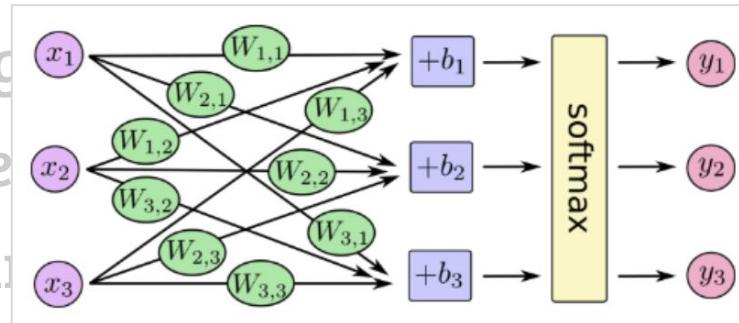
- softmax
- sigmoid
- Relu (Rectified Linear Units)
- tanh

Keras -- Activation Functions

◇ 常用的函數

- softmax

- sig
- Re
- tan



$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \text{softmax} \begin{bmatrix} W_{1,1}x_1 + W_{1,2}x_2 + W_{1,3}x_3 + b_1 \\ W_{2,1}x_1 + W_{2,2}x_2 + W_{2,3}x_3 + b_2 \\ W_{3,1}x_1 + W_{3,2}x_2 + W_{3,3}x_3 + b_3 \end{bmatrix}$$

MNIST : Model

```
#===== Model =====
model <- keras_model_sequential()
model %>%
layer_dense(units = 256, activation = 'relu', input_shape = c(784)) %>%
layer_dropout(rate = 0.4) %>%
layer_dense(units = 128, activation = 'relu')%>%
layer_dropout(rate = 0.3) %>%
layer_dense(units = 10, activation = 'softmax')
```

```
summary(model)
```

Layer (type)	Output Shape	Param #
dense_2 (Dense)	(None, 256)	200960
dropout_2 (Dropout)	(None, 256)	0
dense_3 (Dense)	(None, 128)	32896
dropout_3 (Dropout)	(None, 128)	0
dense_4 (Dense)	(None, 10)	1290
<hr/>		
Total params:	235,146	
Trainable params:	235,146	
Non-trainable params:	0	

MNIST : Model

```
#===== Training and Evaluation =====
model %>% compile(
  loss = 'categorical_crossentropy',
  optimizer = optimizer_rmsprop(),
  metrics = c('accuracy')
)

history <- model %>% fit(
  train_images, train_labels,
  epochs = 30, batch_size = 128,
  validation_split = 0.2)
plot(history)

model %>% evaluate(test_images, test_labels)
model %>% predict_classes(test_images)

#---- Setting ----
Path = '/Users/sun/Desktop'
save_model_hdf5(model, file.path(Path, "my_keras_mnist_resape.h5"))
```

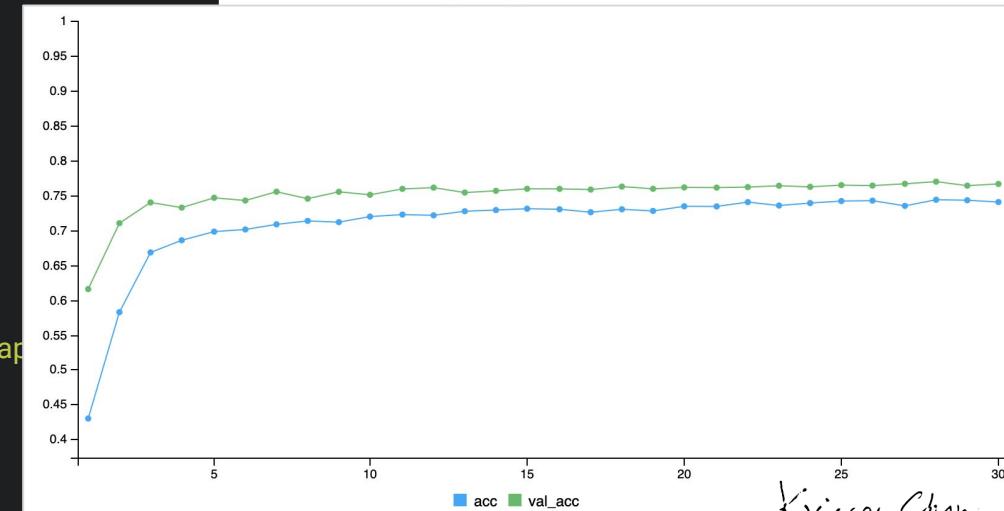
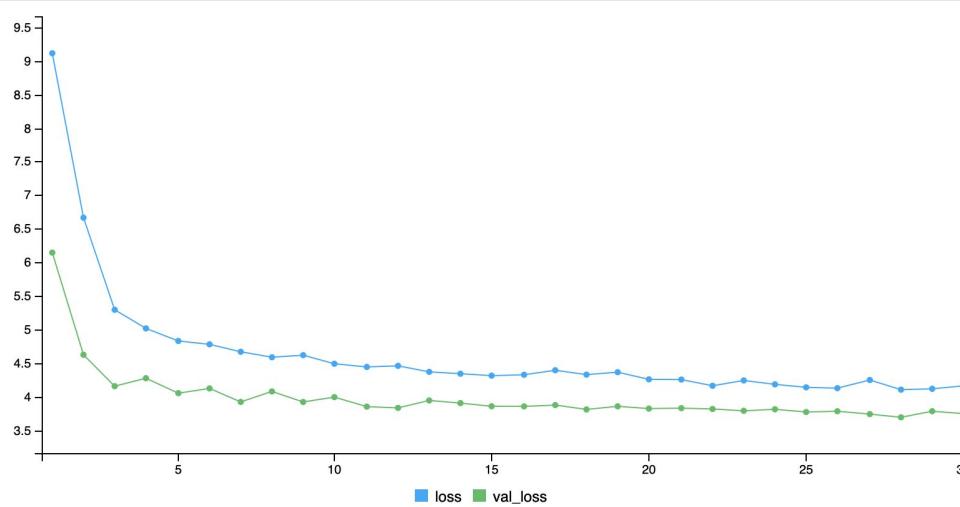
MNIST : Model

```
#-----  
model %>% compile(  
  loss = 'categorical_crossentropy',  
  optimizer = optimizer_rmsprop(),  
  metrics = c('accuracy'))  
)
```

```
history <- model %>% fit(  
  train_images, train_labels,  
  epochs = 30, batch_size = 128,  
  validation_split = 0.2)  
plot(history)
```

```
model %>% evaluate(test_images, test_labels)  
model %>% predict_classes(test_images)
```

```
#----- Setting -----  
Path = '/Users/sun/Desktop'  
save_model_hdf5(model, file.path(Path, "my_keras_mnist_resap"))
```



Kristen Chia

MNIST : Model

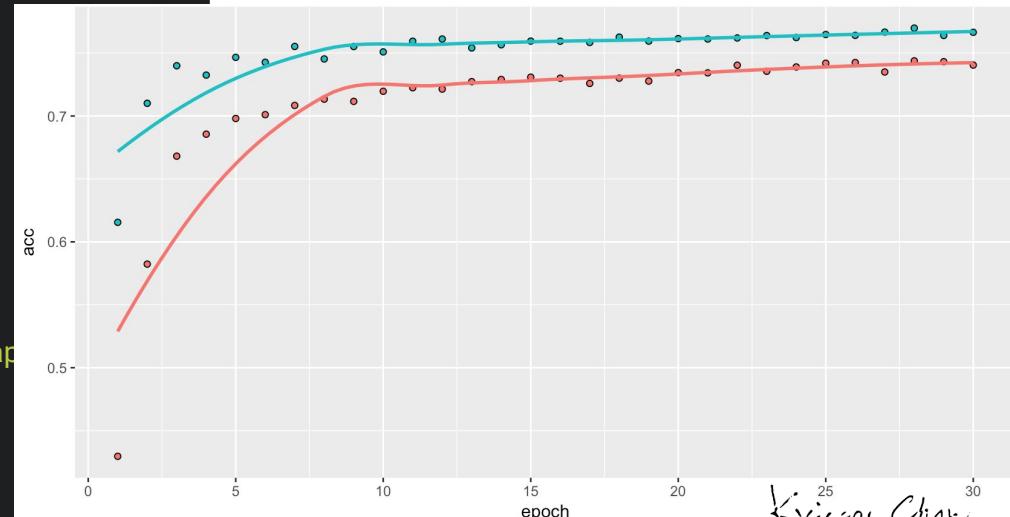
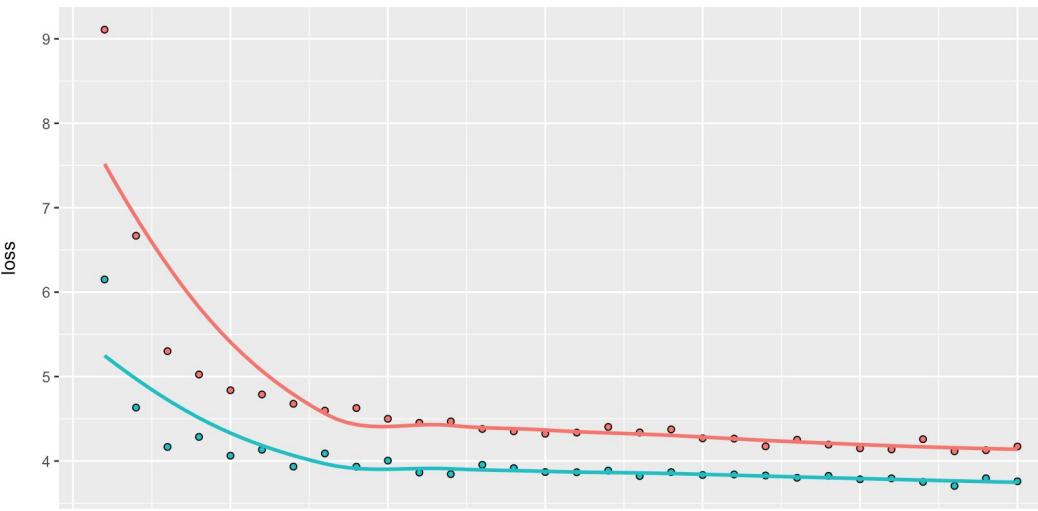
```
#-----  
model %>% compile(  
  loss = 'categorical_crossentropy',  
  optimizer = optimizer_rmsprop,  
  metrics = c('accuracy'))  
)
```

```
history <- model %>% fit(  
  train_images, train_labels,  
  epochs = 30, batch_size = 128,  
  validation_split = 0.2)  
plot(history)
```

```
model %>% evaluate(test_images, test_labels)  
model %>% predict_classes(test_images)
```

```
#----- Setting -----
```

```
Path = '/Users/sun/Desktop'  
save_model_hdf5(model, file.path(Path, "my_keras_mnist_resnet.h5"))
```



MNIST : Model

```
#===== Training and Evaluation =====
model %>% compile(
  loss = 'categorical_crossentropy',
  optimizer = optimizer_rmsprop(),
  metrics = c('accuracy')
)
history <- model %>% fit(
  train_images, train_labels,
  epochs = 30, batch_size = 128,
  validation_split = 0.2)
plot(history)

model %>% evaluate(test_images, test_labels)
model %>% predict_classes(test_images)

#---- Setting ----
Path = '/Users/sun/Desktop'
save_model_hdf5(model, file.path(Path, "my_keras_mnist_resape.h5"))
```

> model %>% evaluate(test_images, test_labels)
10000/10000 [=====] - 0s 34us/step
\$loss
[1] 2.277729

\$acc
[1] 0.1486



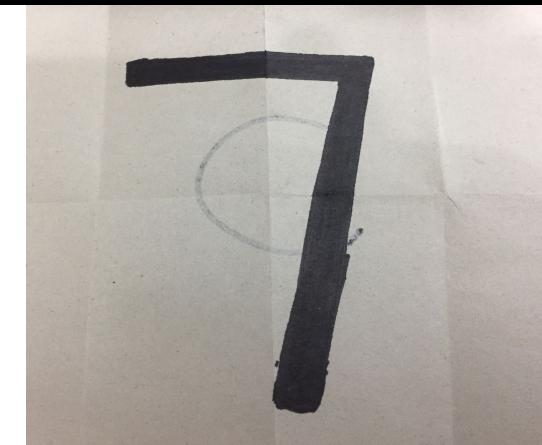
Your Digits Image

```
#===== Your Digits =====
library(EIImage)

#---- Import Image ----
import_color.image <- readImage( file.path(Path,"7.png") )
import_gray.image <- channel(import_color.image,"gray")
resize_gray.image <- resize(import_gray.image, 28, 28)
trans_gray.image <- transpose(resize_gray.image)

#---- Predict Result ----
your_im <- array( dim=c(1,28,28), data=trans_gray.image )
your_im <- array_reshape(your_im, c(nrow(your_im), 784))
model %>% predict_classes(your_im)
```

```
> model %>% predict_classes(your_im)
[1] 5
```



Shiny

今天我們要做...

Keras with MNIST

Import Image

Input Keras

1:28

Upload New File
Please click [Upload New File] before you upload

Choose PNG/JPEG File
Browse... 2.png Upload complete

Check Image
File OK & Go Predict

PREDICT RESULT

The application interface consists of several sections. On the left, there's a sidebar with a dark background containing buttons for 'Upload New File' (disabled), 'Choose PNG/JPEG File' (with 'Browse...' and 'Upload complete' buttons), and 'Check Image' (with a 'File OK & Go Predict' button). The main area has two large windows. The left window, titled 'Import Image', shows a handwritten digit '2' on a piece of paper. The right window, titled 'Input Keras', shows a 28x28 pixel grayscale image of the same digit, labeled '1:28'. Below these windows is a blue button with a white hand icon and the text 'PREDICT RESULT'.

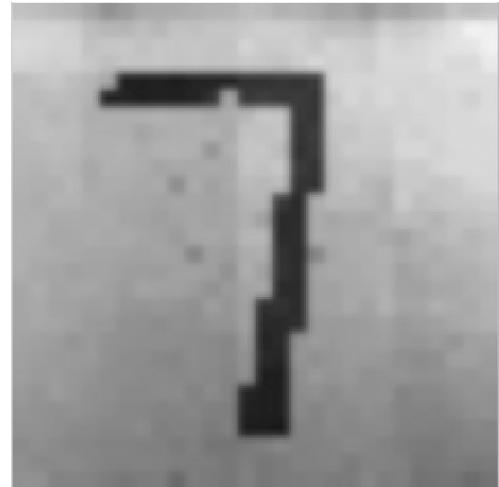
Your Digits Image

```
#===== Your Digits =====
library(EIImage)

#---- Import Image ----
import_color.image <- readImage( file.path(Path,"7.png") )
import_gray.image <- channel(import_color.image, "gray")
resize_gray.image <- resize(import_gray.image, 28, 28)
trans_gray.image <- transpose(resize_gray.image)

#---- Predict Result ----
your_im <- array( dim=c(1,28,28), data=trans_gray.image )
your_im <- array_reshape(your_im, c(nrow(your_im), 784))

plot(resize_gray.image)
```



Introduction to Shiny

- ✓ Shiny is a package from Rstudio
- ✓ It's a web development framework in R.
- ✓ It can be easy to build interactive web applications with R.

You don't need to know following tools:

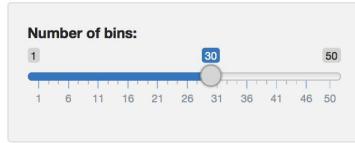
- ✗ Html
- ✗ Javascript
- ✗ CSS

Hello Shiny

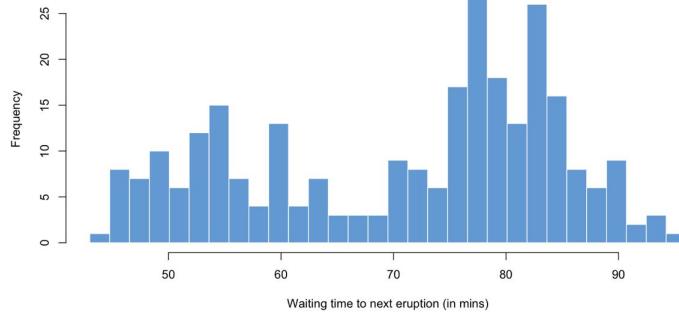


```
library(shiny)
runExample("01_hello")
```

Hello Shiny!



Histogram of waiting times



Hello Shiny!

by RStudio, Inc.

This small Shiny application demonstrates Shiny's automatic UI updates.

Move the *Number of bins* slider and notice how the `renderPlot` expression is automatically re-evaluated when its dependant, `input$bins`, changes, causing a histogram with a new number of bins to be rendered.

app.R

↓ show with app

```
library(shiny)
```

```
# Define UI for app that draws a histogram ----
ui <- fluidPage(
```

```
# App title ----
titlePanel("Hello Shiny!"),
```

```
# Sidebar layout with input and output definitions ----
sidebarLayout(
```

```
# Sidebar panel for inputs ----
sidebarPanel(
```

```
# Input: Slider for the number of bins ----
sliderInput(inputId = "bins",
            label = "Number of bins:",
            min = 1,
            max = 50,
            value = 30)
```

A Shiny Project

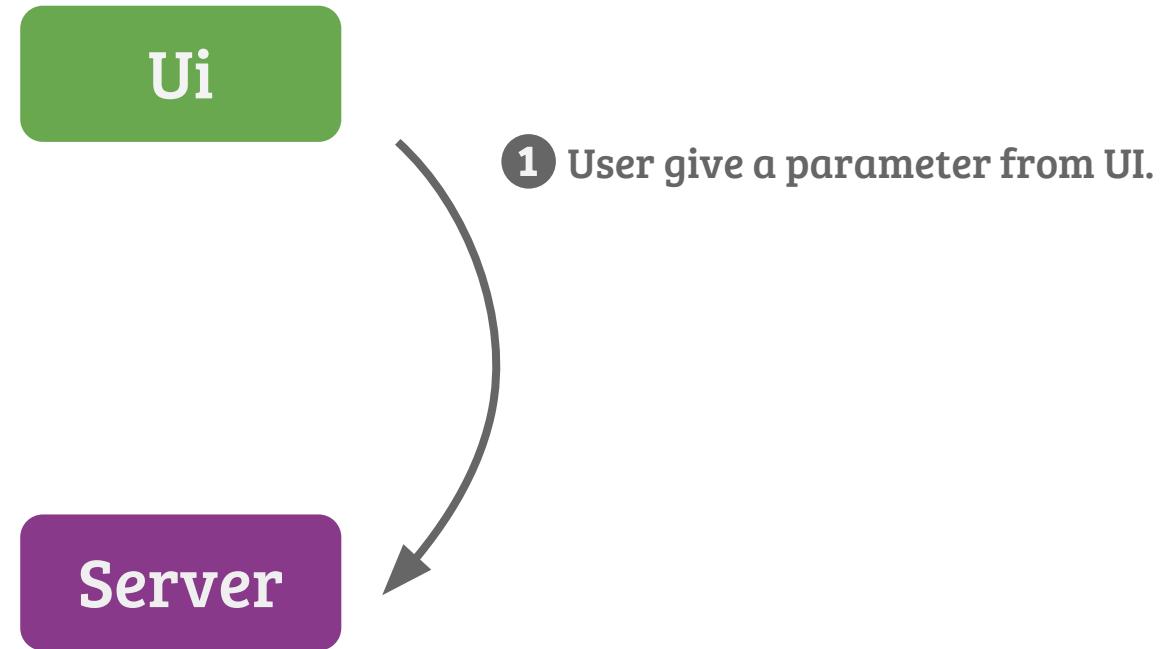
✓ **Ui**

- User interface
- Determine how your app looks

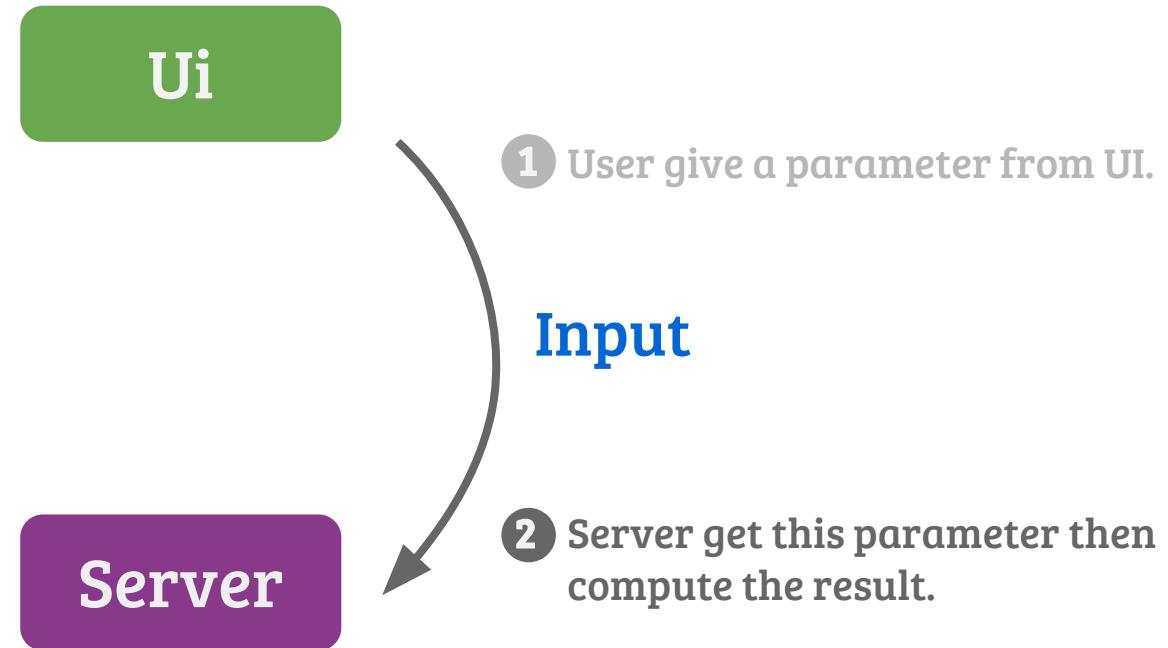
✓ **Server**

- Backend or engine of the application
- It's where the data is processed
- Control what your app does

How does Shiny work

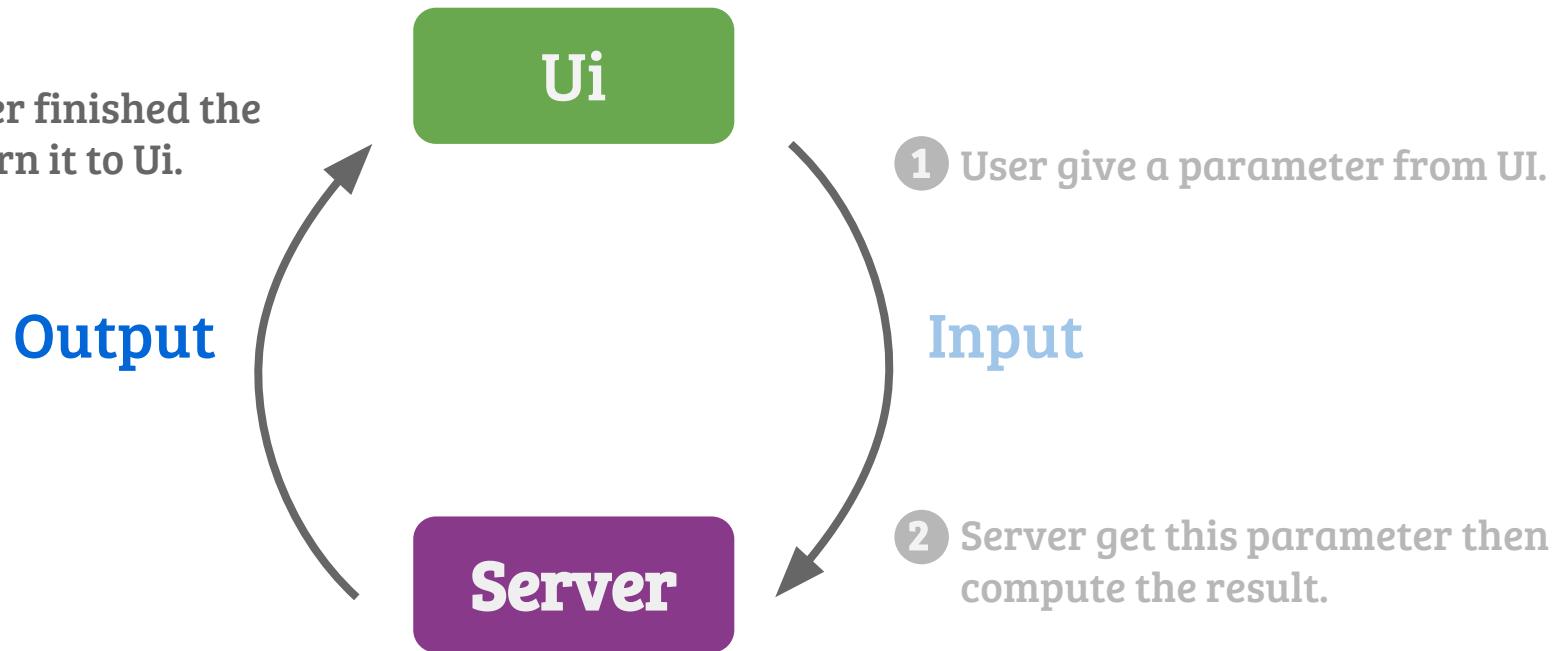


How does Shiny work



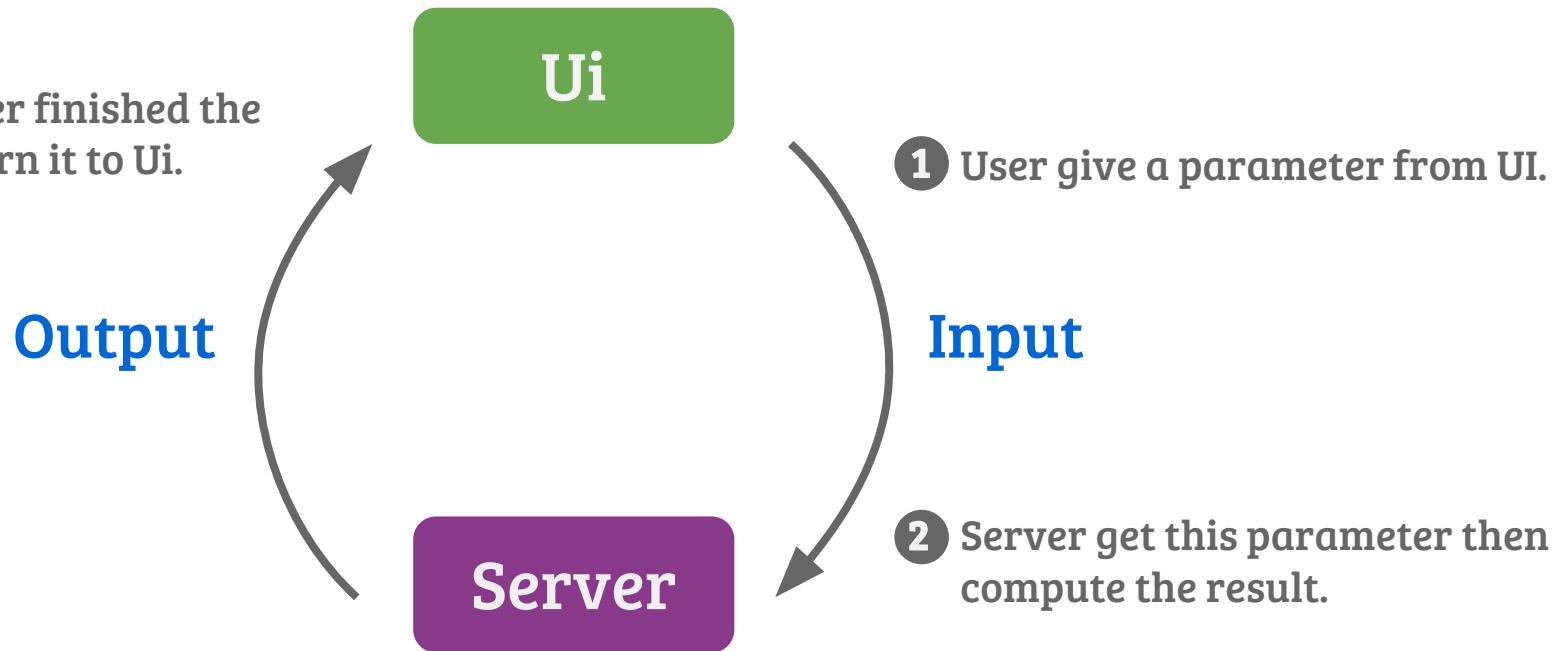
How does Shiny work

- 3 When Server finished the result, return it to Ui.



How does Shiny work

- 3 When Server finished the result, return it to Ui.



Let's Do it !!

Keras with MNIST

Import Image

Input Keras

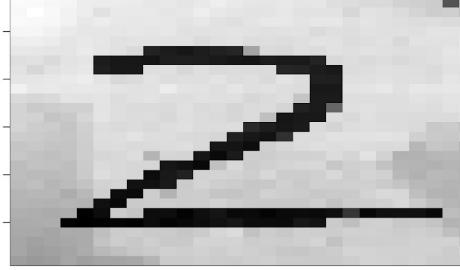
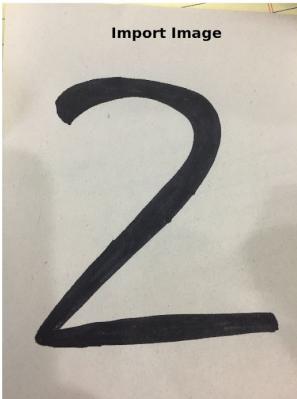
1:28

Upload New File
Please click [Upload New File] before you upload

Choose PNG/JPEG File
Browse... 2.png Upload complete

Check Image
File OK & Go Predict

PREDICT RESULT  2



Thanks !!

Q & A





Home > My Projects > GlobalAI_AutomatedML

GlobalAI_AutomatedML

Clone 0

Star 0

Cloned from NileshA/GlobalAI ↗

Status: Running on Free Compute

Project Settings

Download Project

Share ▾

Run on Free Co... ▾

Search files, notebooks.

Show hidden items

▼	Name	File Type	Modified On	Created On
	auto-ml-Energy Demand Forecasting.ipynb	Notebook	Dec 15, 2018	Dec 15, 2018
	README.md	Markdown	Dec 15, 2018	Dec 15, 2018

Showing 2 search results (1 hidden)

Product

What's new

Status

Help

Communication

FAQ

Contact us

Jupyter.org

Legal

Privacy and cookies

Terms of use

Trademarks



Powered by jupyter

GlobalAI_AutomatedML

Files Running Nbextensions

Select items to perform actions on them.

Upload New

<input type="checkbox"/>	0		/	Name	Last Modified	File size
<input type="checkbox"/>		auto-ml-Energy Demand Forecasting.ipynb		Running	12 分鐘前	180 kB
<input type="checkbox"/>		README.md			12 分鐘前	0 B

<< 首頁 > 新增

新增



搜尋 Marketplace

Azure Marketplace

查看全部

精選

查看全部

開始使用

Machine Learning 服務工作區

深入了解

最近建立

Data Science Virtual Machine - Windows 2016

快速入門教學課程

計算



網路

Storage

Web App Bot

Web

快速入門教學課程

Mobile

電腦視覺

容器

快速入門教學課程

Databases

臉部

Analytics

快速入門教學課程

AI + Machine Learning

文字分析

快速入門教學課程

物聯網

語言理解

快速入門教學課程

Integration

Developer tools

快速入門教學課程

Security

Management Tools

快速入門教學課程

Identity

Bing 搜尋 v7

快速入門教學課程

Developer tools

Management Tools

快速入門教學課程

Management Tools

KrisfenChen

Kristen Chan

