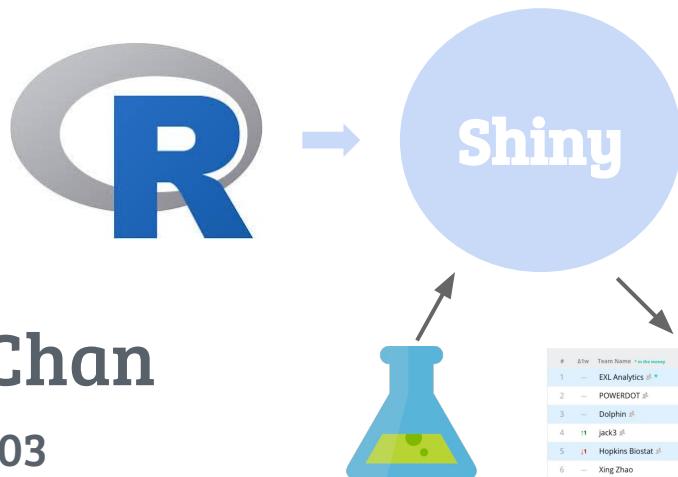


Shiny Application

Cases sharing

Kristen Chan

2018.09.03



Hello !

I am Kristen Chan

Data Scientist

E-Commerce / Telecom

R-Ladies Taipei

Co-Organizer



Shiny Apps

1. **Introduction to Shiny**
2. **Azure ML via Shiny**
 - a. What is Azure Machine Learning Studio ?
 - b. How to connect with Shiny ?
3. **Get Kaggle Leaderboard with Shiny**
 - a. 2018 R-Ladies Kaggle
 - b. Kaggle Leaderboard with Automatic downloading
4. **Let's Play Shiny**
 - a. **Using Open Data -- 水質檢測資料**

(Source: <http://data.taipei/opendata/datalist/datasetMeta?oid=961ca397-4a59-45e8-b312-697f26b059dc>)



Introduction to Shiny

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:

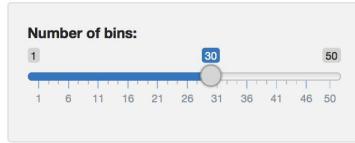
- ✗ Html
- ✗ Javascript
- ✗ CSS

Hello Shiny

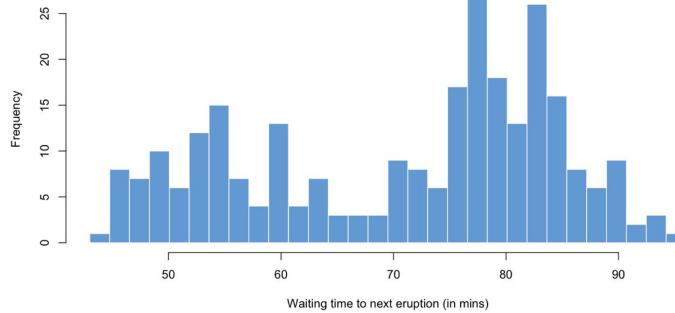


```
library(shiny)
runExample("01_hello")
```

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

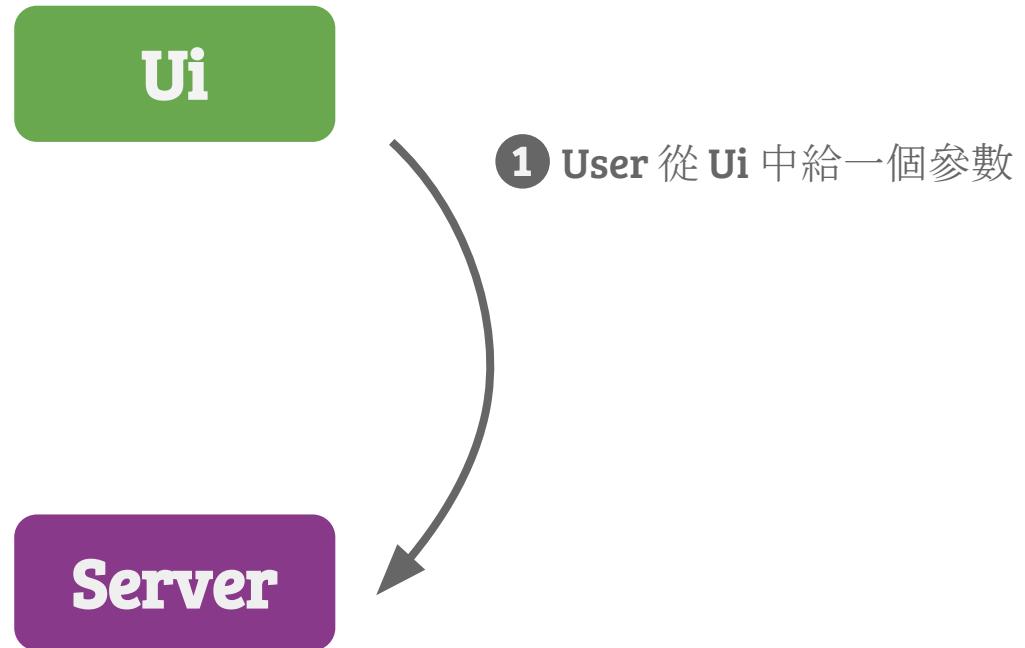
✓ **Ui**

- User interface
- Determine how your app looks

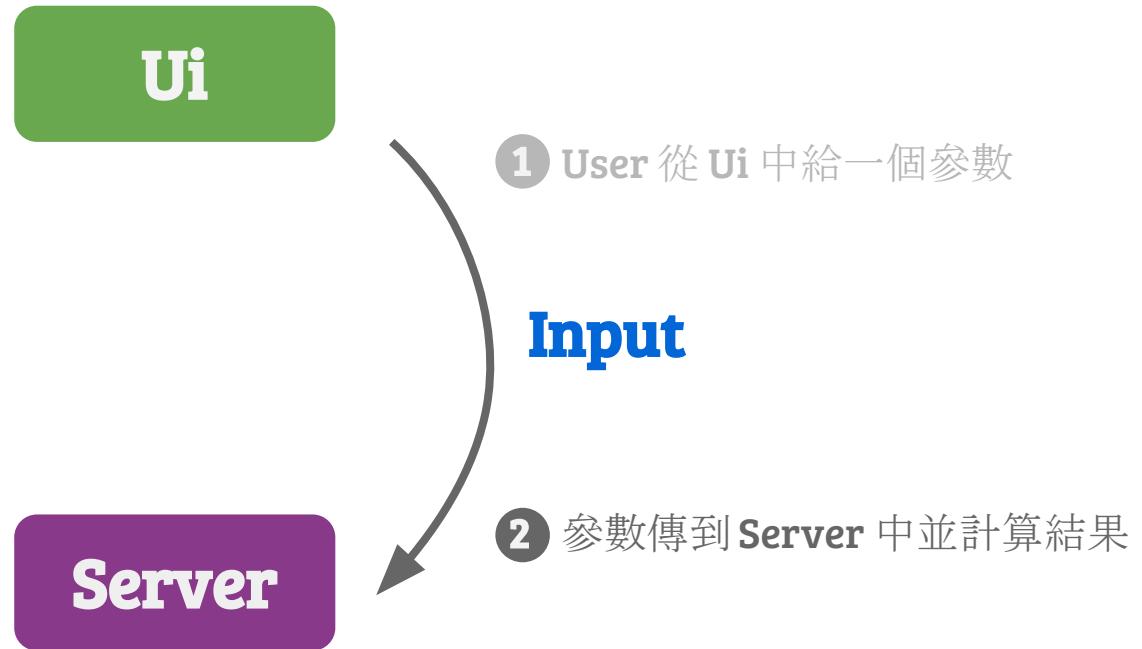
✓ **Server**

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

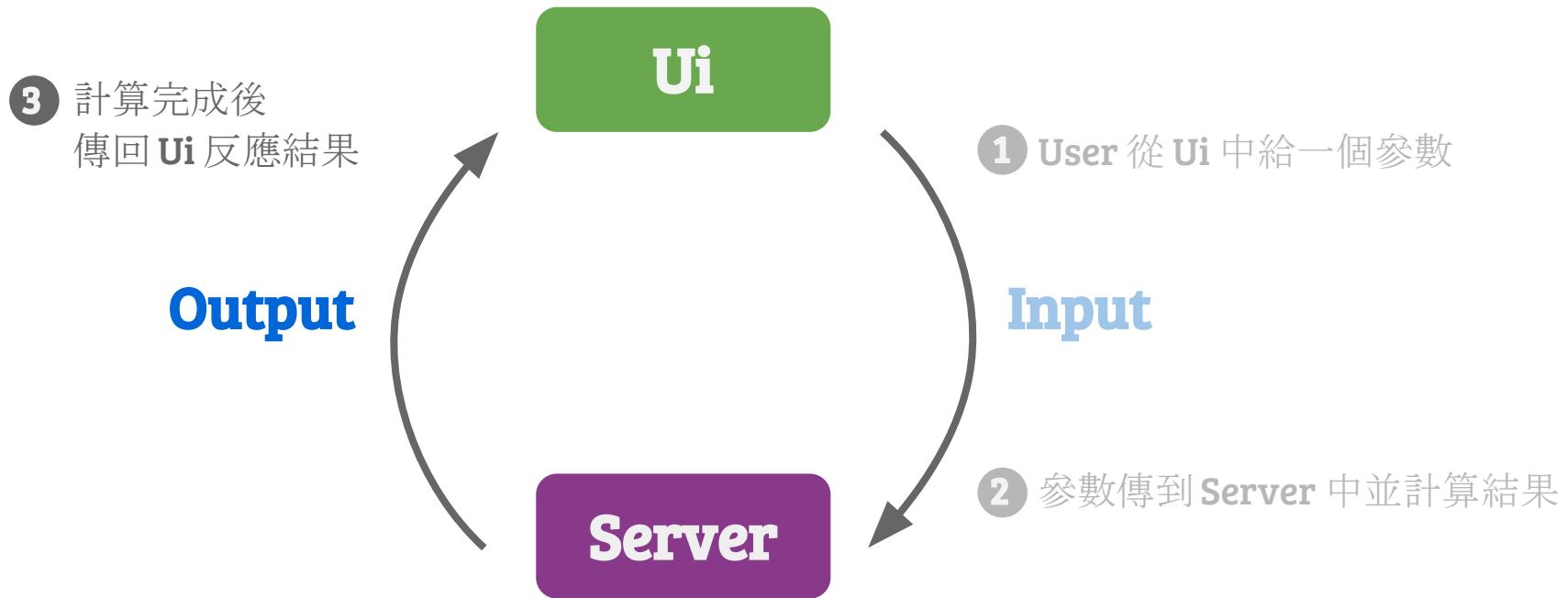
How does Shiny work



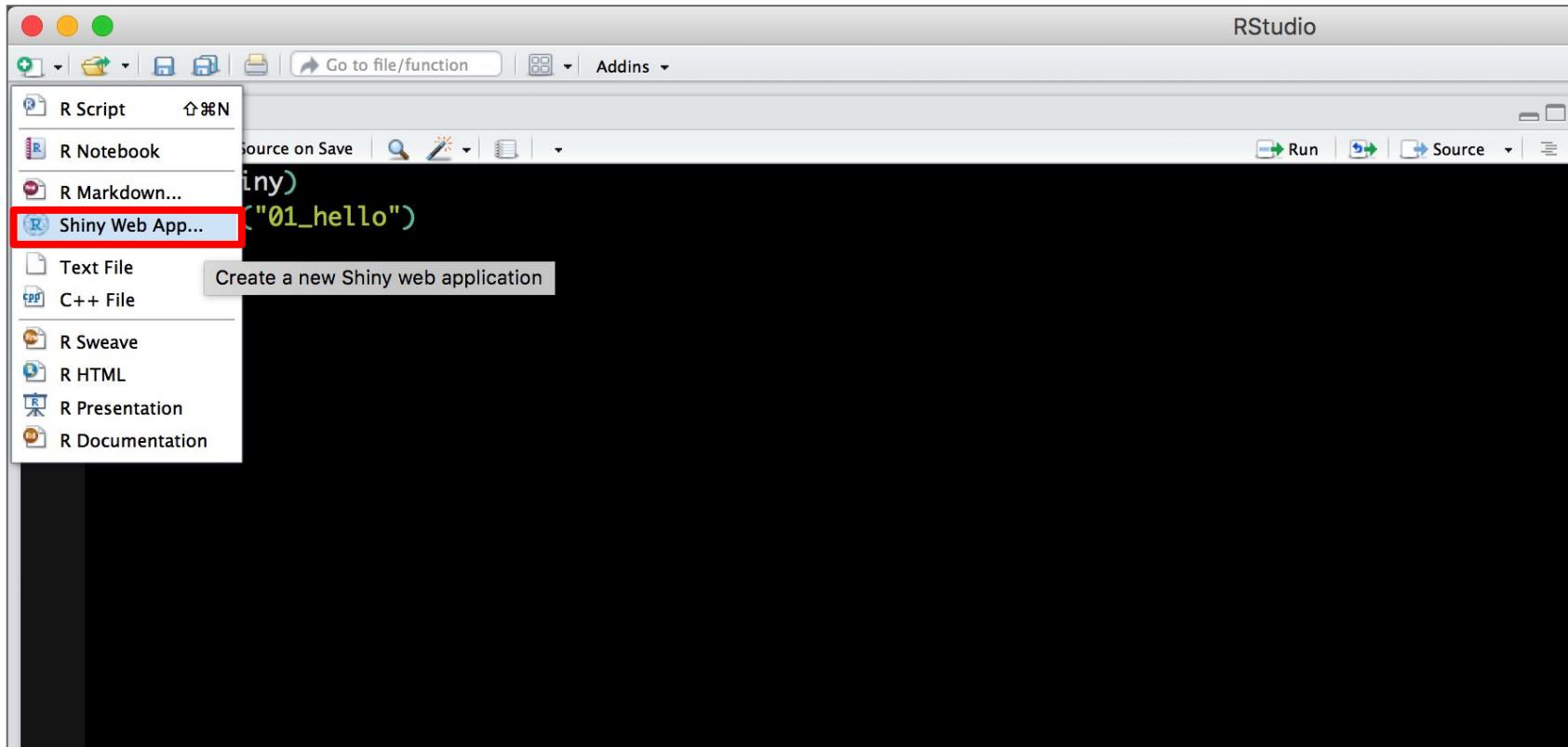
How does Shiny work



How does Shiny work



Your First Shiny App



Your First Shiny App – Case 1 (Ui.R / Server.R)

New Shiny Web Application



Application name: **Enter your app's name**

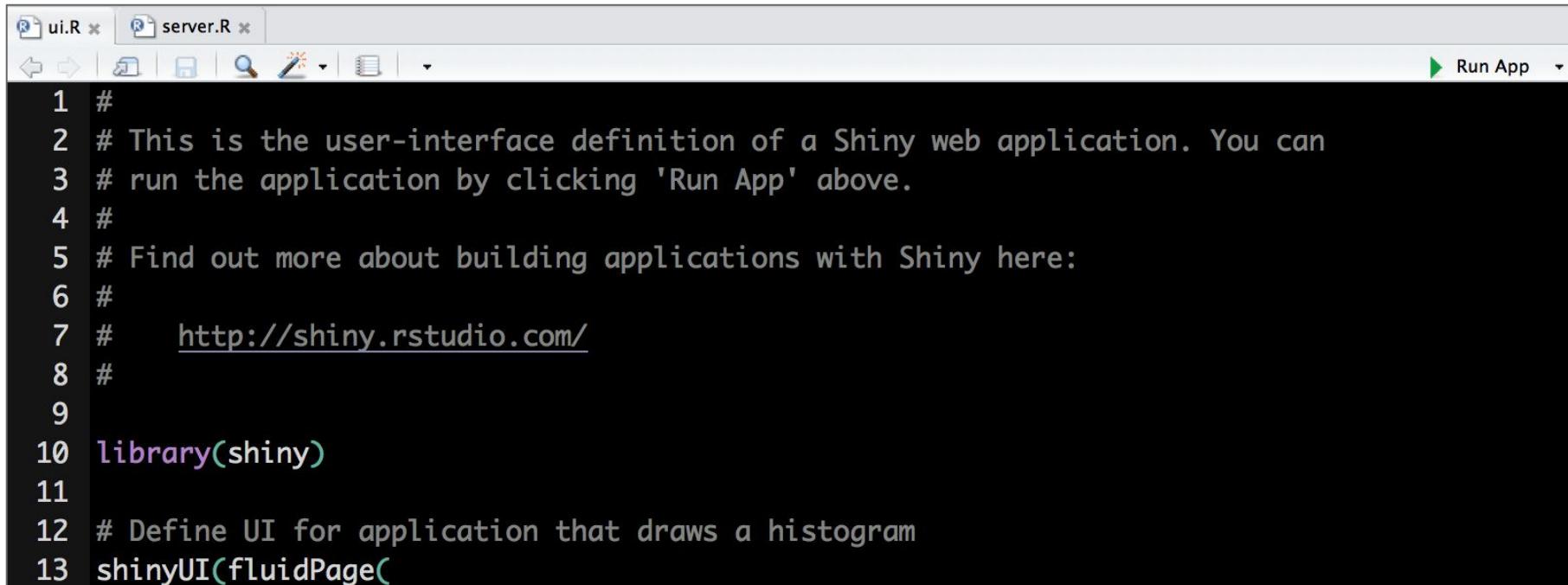
Application type:

Single File (app.R)
 Multiple File (ui.R/server.R)

Create within directory:

[?](#) Shiny Web Applications

Your First Shiny App -- Case 1 (Ui.R / Server.R)



The screenshot shows the RStudio interface with two files open: ui.R and server.R. The ui.R file contains the user-interface definition for a Shiny application, including comments about running the app and links to the shiny.rstudio.com documentation. The server.R file is partially visible at the top. The RStudio toolbar includes standard icons for file operations, search, and run.

```
ui.R x server.R x
Run App

1 #
2 # This is the user-interface definition of a Shiny web application. You can
3 # run the application by clicking 'Run App' above.
4 #
5 # Find out more about building applications with Shiny here:
6 #
7 #     http://shiny.rstudio.com/
8 #
9
10 library(shiny)
11
12 # Define UI for application that draws a histogram
13 shinyUI(fluidPage(
```

Your First Shiny App -- Case 2 (app.R)

New Shiny Web Application



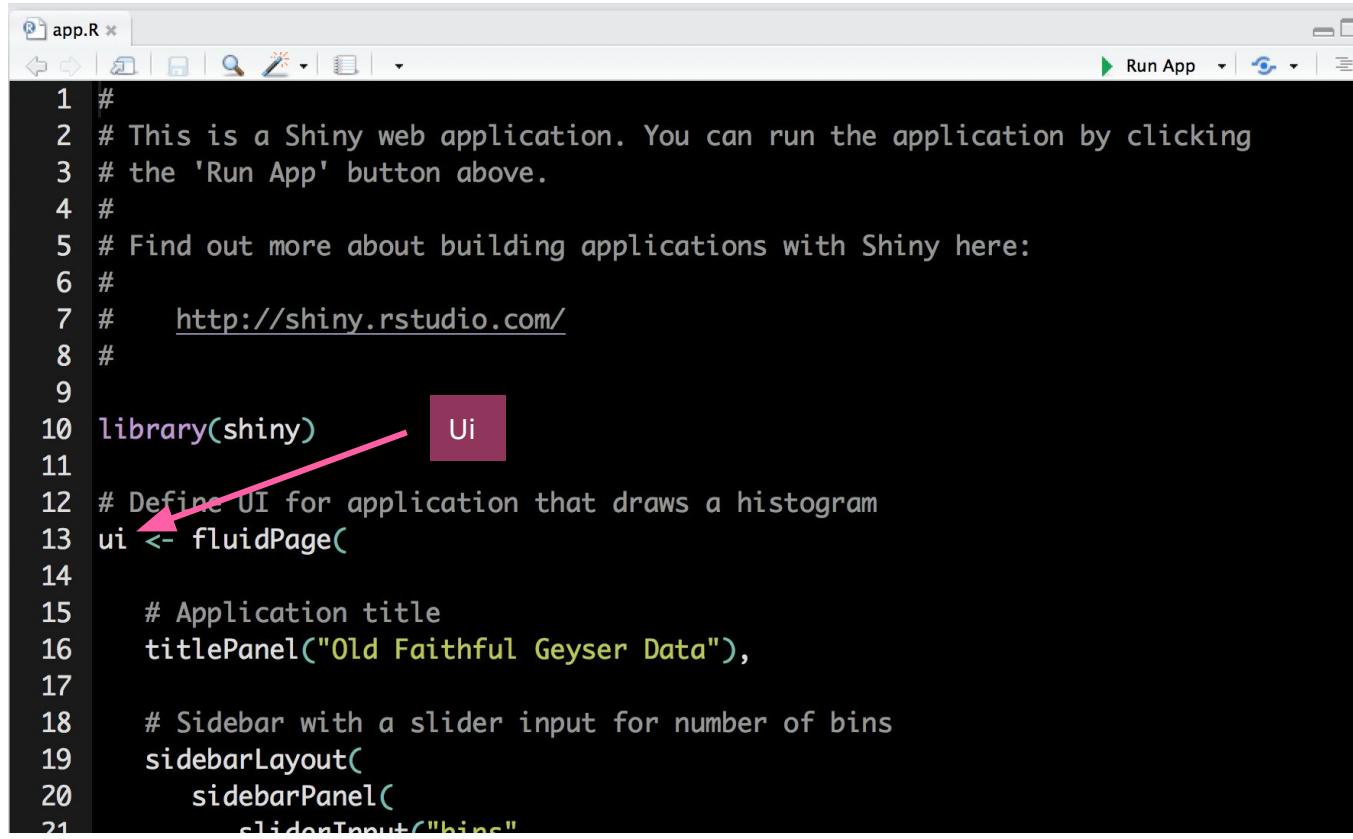
Application name: **Enter your app's name**

Application type: Single File (app.R) Multiple File (ui.R/server.R)

Create within directory:

[?](#) Shiny Web Applications

Your First Shiny App -- Case 2 (app.R)

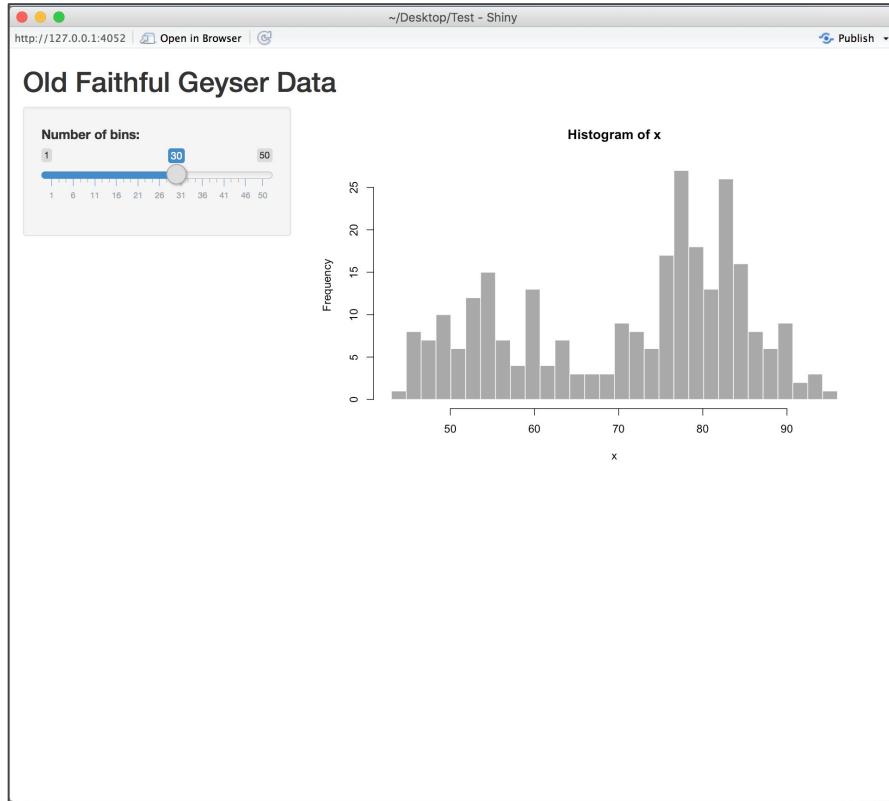


```
app.R x Run App
1 #
2 # This is a Shiny web application. You can run the application by clicking
3 # the 'Run App' button above.
4 #
5 # Find out more about building applications with Shiny here:
6 #
7 #   http://shiny.rstudio.com/
8 #
9
10 library(shiny)    Ui
11
12 # Define UI for application that draws a histogram
13 ui <- fluidPage(
14
15   # Application title
16   titlePanel("Old Faithful Geyser Data"),
17
18   # Sidebar with a slider input for number of bins
19   sidebarLayout(
20     sidebarPanel(
21       sliderInput("bins",
```

Your First Shiny App -- Case 2 (app.R)

```
app.R * | Run App | <| Server |>
50
31
32
33
34
35 # Define server logic required to draw a histogram
36 server <- function(input, output) {
37
38   output$distPlot <- renderPlot({
39     # generate bins based on input$bins from ui.R
40     x      <- faithful[, 2]
41     bins <- seq(min(x), max(x), length.out = input$bins + 1)
42
43     # draw the histogram with the specified number of bins
44     hist(x, breaks = bins, col = 'darkgray', border = 'white')
45   })
46 }
47
48 # Run the application
49 shinyApp(ui = ui, server = server)
50
51
```

Your First Shiny App



Azure ML via Shiny

~/Desktop/Shiny/Shiny_Titanic - Shiny

http://127.0.0.1:3857 | Open in Browser | Publish ▾

Titanic Survival Prediction

Passenger Class :

1

Gender :

female

Port Embarkation :

C

Age :

80

Sibling Spouse :

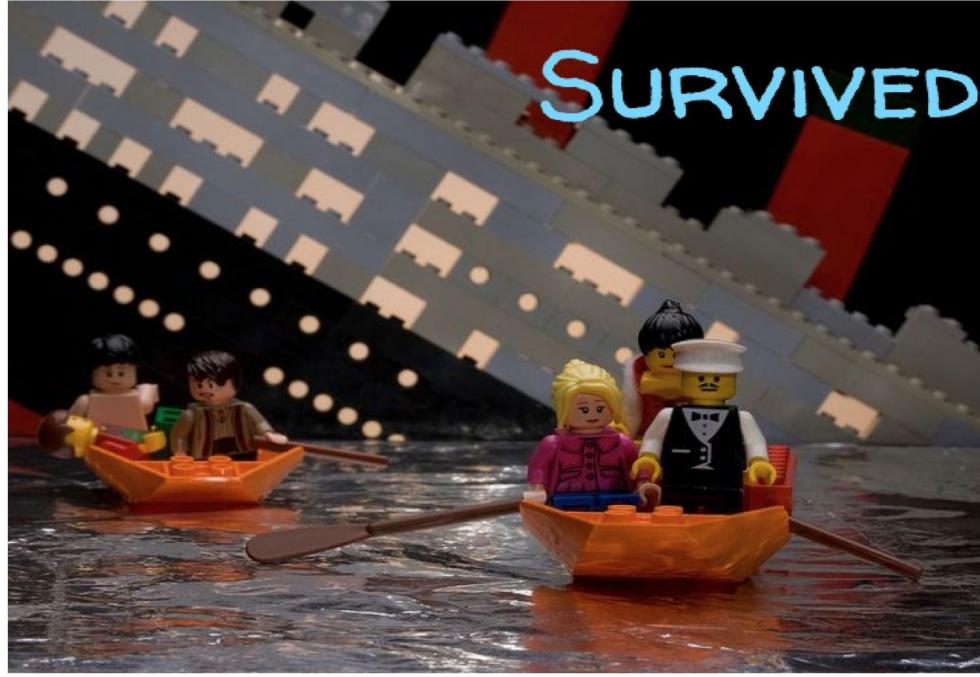
8

Parent Child :

9

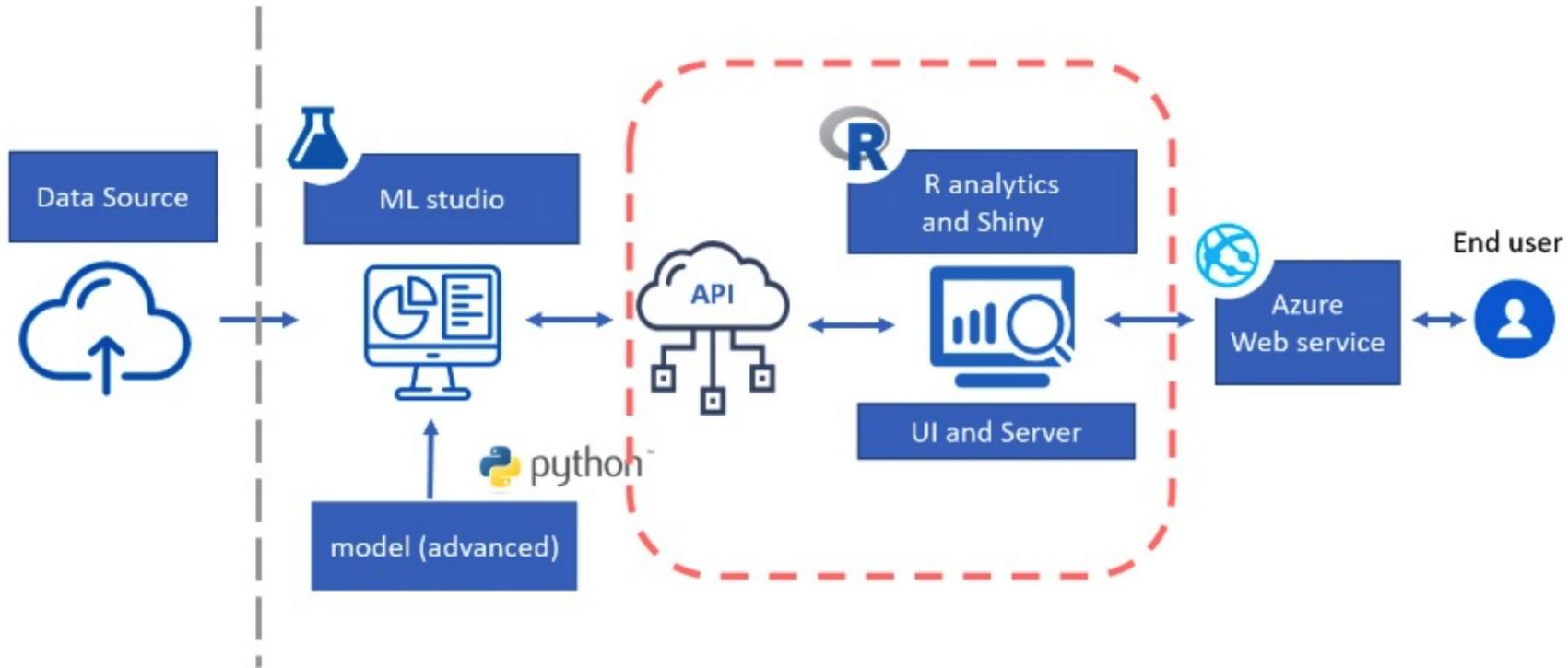
Fare Price :

512.3292



Kristen Chan 19

Data



On April 15, 1912, during her maiden voyage, the Titanic sank after colliding with an iceberg, killing 1502 out of 2224 passengers and crew.

[Note] About Titanic Data

Variable	Definition	Key
survival	Survial	0=No , 1=Yes
pclass	Ticket Class	1=1st , 2=2nd , 3=3rd
embarked	Port of embarkation	C = Cherbourg , Q = Queenstown , S = Southampton
age	Age in years	
sibsp	#of siblings / spouses aboard the Titanic	
parch	#of parents / children aboard the Titanic	
ticket	Ticket number	
fare	Passenger fare	
cabin	Cabin number	
sex	Sex	



[補] Azure ML Step

Azure Machine Learning Studio

Note https://studio.azureml.net/

≡ Microsoft Azure Machine Learning Studio

?

AA

Smile

Sign In



Welcome to Azure
Machine Learning

Try it for free

No Azure subscription? No credit card? No problem! Choose anonymous Guest Access, or sign in with your work or school account, or a Microsoft account.

Sign In

Not an Azure ML user?
[Sign up here](#)

Pricing & FAQ
By using this free version, you agree to be bound by the Microsoft Azure Website Terms of Use.

Announcements

Mining Campaign Funds

Aired on August 02, 2017

Play with 2016 Presidential Campaign finance data while learning how to prepare a large dataset for machine learning by processing and engineering features. This sample experiment works on a 2.5 GB dataset and will take about 20 minutes to run in its entirety.

[Learn More](#)

Inside the Data Science VM

Aired on June 21, 2016

Dsvm is a custom Azure Virtual Machine image that is published on the Azure marketplace and available on both Windows and Linux. It contains several popular data science and development tools both from Microsoft and from the open source community all pre-installed and pre-configured and ready to use. We will cover best practices that would show how you can use the Dsvm effectively to run your next data science or analytics project.

[Learn More](#)

Predictive Maintenance Modeling

Aired on July 05, 2016

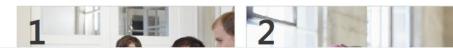
Predictive maintenance is one of the top demanded applications of predictive modelling and is seen as a life-saver in asset-heavy industries such as manufacturing and aerospace due to its potential to provide significant cost savings by reducing downtime due to mechanical problems. We will share what data science techniques are best equipped for this sort of analytics.

[Learn More](#)

More

What's New

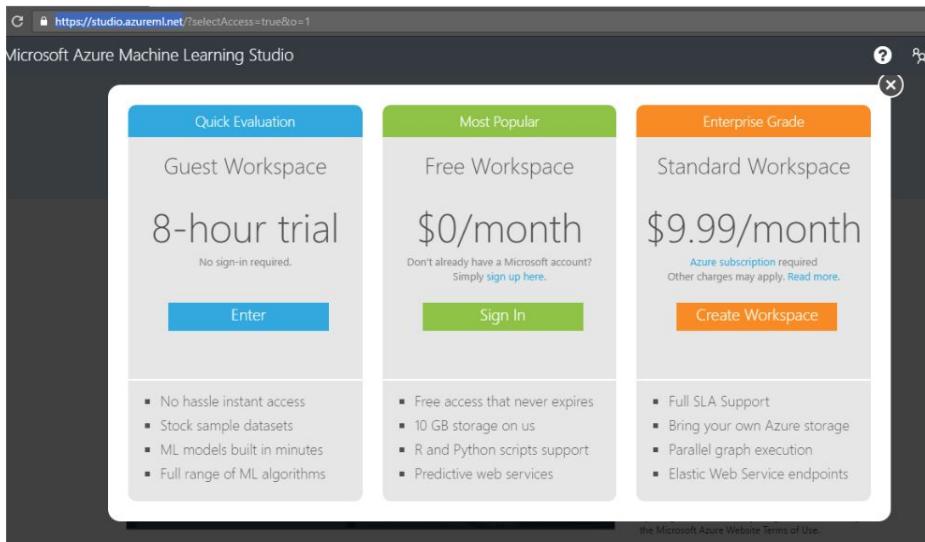
[New text analytics modules](#)



Azure Machine Learning Studio

Note <https://studio.azureml.net/>

- 使用 Microsoft Account 登入，選擇 Free Workspace
- 不登入使用 Guest Workspace



Create an Experiment

Microsoft Azure Machine Learning Studio

agogomei-Free-Workspace ? 🤖 😊 🚙

PROJECTS

EXPERIMENTS

WEB SERVICES

NOTEBOOKS

DATASETS

TRAINED MODELS

SETTINGS

1

NEW

experiments

MY EXPERIMENTS SAMPLES

NAME	AUTHOR	STATUS	LAST EDITED	PROJECT
Ning_044_UsedCarLinear - C...	agogomei	Draft	1/31/2018 11:50:46 AM	None
3inputExperiment created on ...	agogomei	Finished	12/16/2017 7:08:25 AM	None
3inputExperiment created on ...	agogomei	Finished	12/16/2017 7:07:51 AM	None
102 Linear Regression Stock ...	agogomei	Draft	12/4/2017 7:29:44 PM	None
UsedCarLinear	ningchencontact	Draft	12/3/2017 4:53:32 PM	None
SinputExperiment	agogomei	Finished	11/26/2017 4:00:26 PM	None
UsedCarLinear_DecisionRegr...	agogomei	Finished	11/26/2017 2:19:35 PM	None
Experiment created on 2017/...	agogomei	Finished	11/26/2017 11:47:55 AM	None
UsedCarLinear_SVM - Remov...	agogomei	Draft	11/26/2017 11:39:42 AM	None
Experiment created on 2017/...	agogomei	Failed	11/26/2017 10:48:09 AM	None
Experiment created on 2017/...	agogomei	Finished	11/25/2017 6:06:23 PM	None
Experiment created on 2017/...	agogomei	Finished	11/24/2017 9:27:22 PM	None
4input_titanic [Predictive Exp.]	sinue625	Draft	11/21/2017 6:17:39 PM	None
SinputExperiment [Predictive ...	agogomei	Finished	11/21/2017 10:14:13 AM	None
4input_titanic	agogomei	Finished	11/21/2017 10:05:42 AM	None
CheckhereAzureMLPredictMo...	agogomei	Finished	10/27/2017 8:43:49 AM	None

DELETE **ADD TO PROJECT**

```

graph TD
    A[usedcar1209 - Copy.csv] --> B[Select Columns in Dataset]
    B --> C[Linear Regression]
    C --> D[Split Data]
    D --> E[Train Model]
    E --> F[Score Model]
    F --> G[Evaluate Model]
  
```

Create an Experiment

Microsoft Azure Machine Learning Studio

agogomei-Free-Workspace ? ☺ 🚧

PROJECTS experiments MY EXPERIMENTS SAMPLES EXPERIMENTS NEW

2 Blank Experiment Microsoft Samples

VIEW MORE IN GALLERY

Blank Experiment

Experiment Tutorial

Sample 1: Download dataset from UCI: Adult 2 class dataset

Sample 2: Dataset Processing and Analysis: Auto Imports Regression

Sample 3: Cross Validation for Binary Classification: Adult

Sample 4: Cross Validation for Regression: Auto Imports

Sample 5: Train, Test, Evaluate for Binary Classification: Adult

Sample 6: Train, Test, Evaluate for Regression: Auto Imports Dataset

Sample 7: Train, Test, Evaluate for Multiclass Classification: Letter

Sample 8: Apply SQL transformation

Sample 9: Split, partition and sample system

Anomaly Detection: Credit Risk

Binary Classification: Binary Classification: Binary Classification: Binary Classification: Binary Classification: Binary Classification:

Kristen Chan 26

介面介紹

A

可以改標題名稱

The screenshot shows the Microsoft Azure Machine Learning Studio interface. On the left is a sidebar with various icons and categories: Search experiment items, Saved Datasets, Trained Models, Transforms, Data Format Conversions, Data Input and Output, Data Transformation, Feature Selection, Machine Learning, OpenCV Library Modules, Python Language Modules, R Language Modules, Statistical Functions, Text Analytics, Time Series, Web Service, and Deprecated. At the bottom of the sidebar is a 'NEW' button.

The main area is titled 'Experiment created on 2018/9/3'. It contains a 'Canvas' section with the text '編輯實驗並放上你需要的 Module' (Edit experiment and add the modules you need). Below the canvas is a 'Mini Map' window. At the bottom of the main area are several buttons: RUN HISTORY, SAVE, SAVE AS, DISCARD CHANGES, RUN, SET UP WEB SERVICE, and PUBLISH TO GALLERY.

The right side of the interface shows the 'Properties' panel for the experiment. It includes sections for Experiment Properties (Status Code: InDraft), Summary (with a text input field for describing the experiment), and Description (with a text input field for entering detailed experiment descriptions). There is also a 'Quick Help' section at the bottom right.

介面介紹

Microsoft Azure Machine Learning Studio

Experiment created on 2018/9/3 | In draft

Properties Project Experiment Properties STATUS CODE InDraft Summary Enter a few sentences describing your experiment (up to 140 characters).

Description Enter the detailed description for your experiment.

Quick Help

Module

執行各種操作

a. 可從上方搜尋欄找 Module
b. 可直接拖曳要使用到 Module 到 Canvas 中

Search experiment items

Saved Datasets Trained Models Transforms Data Format Conversions Data Input and Output Data Transformation Feature Selection Machine Learning OpenCV Library Modules Python Language Modules R Language Modules Statistical Functions Text Analytics Time Series Web Service Deprecated

NEW

RUN HISTORY SAVE AS DISCARD CHANGES RUN SET UP WEB SERVICE PUBLISH TO GALLERY

The screenshot shows the Microsoft Azure Machine Learning Studio interface. On the left is a sidebar with various module categories like Saved Datasets, Trained Models, and Machine Learning. The main area is a canvas where several rectangular modules are connected by dashed lines, representing a workflow. A green callout box highlights the search bar at the top left and the text 'a. 可從上方搜尋欄找 Module' (You can find modules in the search bar). Another green callout box highlights the text 'b. 可直接拖曳要使用到 Module 到 Canvas 中' (You can drag and drop modules directly onto the canvas) and points to one of the modules on the canvas. The right side of the screen shows experiment properties, summary, and description sections.

介面介紹

The screenshot shows the Microsoft Azure Machine Learning Studio interface. On the left is a sidebar with various project items like Saved Datasets, Trained Models, and Machine Learning modules. The main area is titled "Experiment created on 2018/9/3" and contains a canvas where data flows between modules. A green callout bubble labeled "C" points to the top right corner of the canvas area. To the right of the canvas is a "Properties" panel with sections for Experiment Properties (Status Code: In Draft) and Summary (with a text input field). Below it is a "Description" section (with another text input field) and a "Quick Help" section (which is currently empty). A large text overlay on the canvas area reads: "當點選 Canvas 上的 Module 時, Properties 會出現對應的參數供調整 [Note] Quick Help 可以幫助你更深入了解個參數意義".

Experiment created on 2018/9/3

To create your experiment, drag and drop datasets and modules here

Drag Items Here

C

Properties

當點選 Canvas 上的 Module 時, Properties 會出現對應的參數供調整

[Note] Quick Help
可以幫助你更深入了解個參數意義

STATUS CODE In Draft

Summary

Description

Quick Help

NEW

RUN HISTORY

SAVE

SAVE AS

DISCARD CHANGES

RUN

SET UP WEB SERVICE

PUBLISH TO GALLERY

介面介紹

The screenshot shows the Microsoft Azure Machine Learning Studio interface. On the left, there's a sidebar with icons for Saved Datasets, Trained Models, Transforms, Data Format Conversions, Data Input and Output, Data Transformation, Feature Selection, Machine Learning, OpenCV Library Modules, Python Language Modules, R Language Modules, Statistical Functions, Text Analytics, Time Series, Web Service, and Deprecated items. The main workspace is titled "Experiment created on 2018/9/3" and contains a dashed-line canvas for dragging and dropping datasets and modules. A green callout box labeled "D" points to the "RUN HISTORY" button at the bottom of the interface. Another green callout box labeled "E" points to the "SAVE AS" button. A third green callout box labeled "F" points to the "PUBLISH TO GALLERY" button. The bottom navigation bar includes buttons for RUN HISTORY, SAVE, SAVE AS, DISCARD CHANGES, RUN, SET UP WEB SERVICE, and PUBLISH TO GALLERY.

Access Data

Titanic 資料

Variavle Name	Description
Survived	Survived (1) or died (0)
Pclass	Passenger's class
Name	Passenger's name
Sex	Passenger's sex
Age	Passenger's age
SibSp	Number of siblings/spouses aboard
Parch	Number of parents/children aboard
Ticket	Ticket number
Fare	Fare
Cabin	Cabin
Embark	Port of embarkation

Access Data

Microsoft Azure Machine Learning Studio
Experiment created on 2018/9/3
In draft Properties Project Import Data

匯入資料(當作 Training / Testing)
→ 從 Module 選擇, Data Input and Output 中的 [Import Data]

The screenshot shows the Azure Machine Learning Studio interface. On the left, a sidebar lists various modules under 'Data Input and Output', with 'Import Data' highlighted with a blue box and a large blue arrow pointing to it. In the center workspace, a blue rounded rectangle highlights the 'Import Data' module icon. To the right, a detailed configuration pane for the 'Import Data' module is visible, showing fields for 'Account key', 'Path to container, director...', 'Blob file format' set to 'CSV', and checkboxes for 'File has header row' and 'Use cached results'. At the bottom, a 'Quick Help' section provides a brief description of the module's function.

Import Data

Account key

Path to container, director...

Blob file format

CSV

File has header row

Use cached results

Quick Help

Load data from sources such as the Web, Azure SQL database, Azure table, Hive table, Windows Azure BLOB, or Azure DocumentDB storage. Formerly known as Reader.
(more help...)

RUN HISTORY SAVE AS DISCARD CHANGES RUN SETUP WEB SERVICE PUBLISH TO GALLERY

Access Data

The screenshot shows the Microsoft Azure Machine Learning Studio interface. On the left, there's a sidebar with various icons and a list of experiment items. The main area displays an experiment created on 2018/9/3. In the center, there's a step labeled "Import Data". A blue arrow points from the text below to the "Import Data" step. To the right, the "Properties" panel is open, showing the "Import Data" configuration. It includes fields for "Data source" (set to "Azure Blob Storage"), "Authentication type" (set to "Storage Account"), and "Account name", "Account key", and "Path to container, director..." (all redacted). The "Blob file format" is set to "CSV". There are also checkboxes for "File has header row" and "Use cached results". At the bottom, there are standard Azure ML Studio navigation buttons: NEW, RUN HISTORY, SAVE, SAVE AS, DISCARD CHANGES, RUN, SET UP WEB SERVICE, and PUBLISH TO GALLERY.

Properties 設定

1. Data Source 選擇 : [Web URL via Http]
2. Data Source URL 輸入
3. CSV or TSV has header : 若資料含有 Header 就要打勾
4. Use cached results : 打勾表示把資料 cached 起來就不用每次執行實驗都重抓資料

Access Data

The screenshot illustrates the process of connecting to a data source via HTTP in Microsoft Azure Machine Learning Studio.

- Step 1: Launch Import Data Wizard** (Top right): A pink callout points to the "Launch Import Data Wizard" button in the "Import Data" section of the Properties panel.
- Step 2: Choose data source** (Middle): A pink callout points to the "Web URL via HTTP" option in the "SOURCES" list. The URL `https://raw.githubusercontent.com/radiestaipei/Azureml-shiny` is entered in the "Data source URL" field.
- Step 3: Connect to Web URL via HTTP** (Left): A pink callout points to the "CSV or TSV has header row" checkbox being checked. A note below says "資料有 Header 就要打勾".

Properties Panel (Bottom right):

- Import Data:**
 - Data source: Web URL via HTTP
 - Data source URL: `https://raw.githubusercontent.com/radiestaipei/Azureml-shiny`
 - Data format: CSV
 - CSV or TSV has header row (checked)
 - Use cached results (checked)

Access Data

資料設定完成後，檢查資料輸入狀況

1. [Run]
2. Results dataset 選擇 [Visualize]

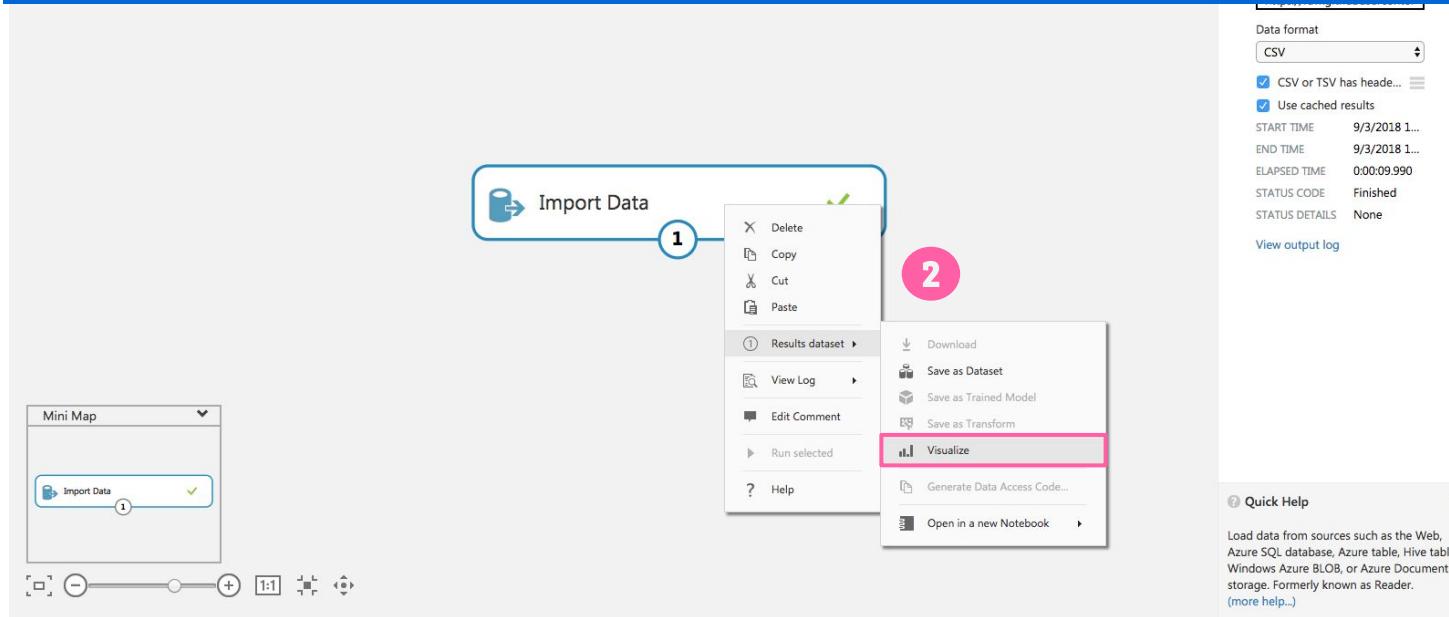
The screenshot shows the Azure Machine Learning Studio interface with the following details:

- Left Sidebar:** A navigation tree with categories like Data Format Conversions, Data Input and Output (selected), Data Transformation, Feature Selection, Machine Learning, etc.
- Middle Area:** A workspace containing a single step labeled "Import Data".
- Bottom Bar:** Buttons for NEW, RUN HISTORY, SAVE, SAVE AS, DISCARD CHANGES, RUN (highlighted with a pink circle and the number 1), SET UP WEB SERVICE, PUBLISH TO GALLERY, and a URL bar showing the workspace ID.
- Right Panel:** A configuration pane for the "Import Data" step, showing:
 - Data source: https://raw.githubusercontent.com
 - Data format: CSV (selected)
 - Checkboxes: CSV or TSV has header... (checked), Use cached results (checked)
- Quick Help:** A tooltip explaining the Import Data step.

Access Data

資料設定完成後，檢查資料輸入狀況

1. [Run]
2. Import Data 右鍵 → Results dataset 選擇 [Visualize]



Access Data

Microsoft Azure Machine Learning Studio

Experiment created on 2018/9/3

experiment created on 2018/9/3 > Import Data > Results dataset

rows 891 columns 8

Survived PassengerClass Gender Age SiblingSpouse ParentChild FarePrice PortEmbarkation

Note 總共有 891 筆資料

檢查 Import 進來的資料有沒有問題

To view, select a column in the table.

RUN HISTORY SAVE AS DISCARD CHANGES RUN SET UP WEB SERVICE PUBLISH TO GALLERY

Data Select

Microsoft Azure Machine Learning Studio
Experiment created on 2018/9/3
In draft Properties Project

篩選欄位

→ 從 Module 選擇, Data Transformation 中 Manipulation 的 [Select Columns in Dataset]

The screenshot shows the Microsoft Azure Machine Learning Studio interface. On the left, there is a sidebar with various icons and a list of modules under 'Data Transformation'. The 'Manipulation' section is expanded, showing several options like 'Add Columns', 'Add Rows', and 'Select Columns in Dataset...'. The 'Select Columns in Dataset...' option is highlighted with a blue box and has a blue arrow pointing to it from the main canvas. The main canvas displays a workflow: 'Import Data' followed by 'Select Columns in Dataset'. The 'Select Columns in Dataset' module has a circled '1' next to it. Below the canvas is a 'Mini Map' showing the same flow. At the bottom, there are buttons for 'RUN HISTORY', 'SAVE', 'SAVE AS', 'DISCARD CHANGES', 'RUN', 'SET UP WEB SERVICE', and 'PUBLISH TO GALLERY'. A 'Quick Help' panel is open on the right, providing information about the 'Select Columns in Dataset' module.

Import Data ✓

Select Columns in Dataset 1

Mini Map

Import Data ✓

Select Columns in Dataset 1

RUN HISTORY SAVE SAVE AS DISCARD CHANGES RUN SET UP WEB SERVICE PUBLISH TO GALLERY

Quick Help

Selects columns to include or exclude from a dataset in an operation. Formerly known as Project Columns.
(more help...)

NEW

enChian 38

Data Select

Microsoft Azure Machine Learning Studio

Experiment created on 2018/9/3

In draft
Draft saved at 午夜 12:48:03

Properties Project

Select Columns in Dataset

Select columns
Selected columns:
Launch the selector tool to make a selection
Launch column selector

Import Data → Select Columns in Dataset

按著連接點拖到連接處

Note 因為還沒設定 Properties

Mini Map

Import Data → Select Columns in Dataset

Run History Save Save As Discard Changes Run Set Up Web Service Publish To Gallery

Data Select

Microsoft Azure Machine Learning Studio

Experiment created on 2018/9/3

In draft
Draft saved at 午後12:48:03

Properties Project

Select Columns in Dataset

Select columns
Selected columns:
Launch the selector tool to make a selection
Launch column selector

Import Data

Select Columns in Dataset

Mini Map

Note 因為還沒設定 Properties

RUN HISTORY SAVE SAVE AS DISCARD CHANGES RUN SET UP WEB SERVICE PUBLISH TO GALLERY

Data Select

Microsoft Azure Machine Learning Studio

Experiment created on 2018/9/3

In draft Draft saved at: 下午12:48:03

Properties Project

1 Selected columns: Launch the selector tool to make a selection

2 Available Columns: PassengerClass, Gender, Age, SiblingSpouse, ParentChild, FarePrice, PortEmbarkation

3 Selected Column: Survived

要的選過來

選完記得打勾

Mini Map

Import Data

Select Columns in Dataset

RUN HISTORY SAVE AS DISCARD CHANGES RUN SET UP WEB SERVICE PUBLISH TO GALLERY

NEW

41

Selects columns to include or exclude from a dataset in an operation. Formerly known as Project Columns.
(more help...)

Data Clean

Microsoft Azure Machine Learning Studio
Experiment created on 2018/9/3
In draft Properties Project

Search experiment items

Saved Datasets
Trained Models
Transforms
Data Format Conversions
Data Input and Output

Data Transformation

- Filter
- Learning with Counts
- Manipulation**

- Add Columns
- Add Rows
- Apply SQL Transformati...
- Clean Missing Data**
- Convert to Indicator Val...
- Edit Metadata
- Group Categorical Values
- Join Data
- Remove Duplicate Rows
- Select Columns in Datas...
- Select Columns Transfor...
- SMOTE

- Sample and Split
- Scale and Reduce
- Feature Selection

篩選欄位
→ 從 Module 選擇, Data Transformation 中 Manipulation 的 [Clean Missing Data]

```
graph TD; Import[Import Data] --> Select[Select Columns in Dataset]; Select --> Clean[Clean Missing Data];
```

1
Cleaning mode
Custom substitution value
Replacement value
0
Generate missing val...
2

Quick Help
Specifies how to handle the values missing from a dataset
(more help...)

NEW RUN HISTORY SAVE SAVE AS DISCARD CHANGES RUN SET UP WEB SERVICE PUBLISH TO GALLERY

Data Clean

Microsoft Azure Machine Learning Studio

In draft agogomei-Free-Workspace ➔

Experiment created on 2018/9/3

In draft Properties Project

Clean Missing Data

Columns to be cleaned
Selected columns: All columns

Launch column selector

Minimum missing value ra...
0

Replace using MICE
Custom substitution value
Replace with mean
Replace with median
Replace with mode
 Remove entire row
Remove entire column
Replace using Probabilistic PCA

處理 Missing 方法

[Remove entire row] : 當其中一個欄位出現遺失值時，整筆(row)刪掉

```

graph TD
    A[Import Data] --> B[Select Columns in Dataset]
    B --> C[Clean Missing Data]
    
```

Quick Help

Specifies how to handle the values missing from a dataset
(more help...)

43

Data Clean

Microsoft Azure Machine Learning Studio

Experiment created on 2018/9/3

Experiment created on 2018/9/3 > Clean Missing Data > Cleaned dataset

712 Note 剩下有 712 筆資料

Gender Age SiblingSpouse ParentChild FarePrice PortEmbarkation

0 3 male 22 1 0 7.25 S

Statistics Visualizations

檢查資料處理狀況

1. Import Data 右鍵 → Results dataset 選擇 [Visualize]

	Age	Sex	Survived	Pclass	SibSp	Parch	Fare	Embarked
Apply	1	3	female	27	0	2	11.1333	S
Clean	1	2	female	14	1	0	30.0708	C
Conv	1	3	female	4	1	1	16.7	S
Edit N	1	1	female	58	0	0	26.55	S
Group	0	3	male	20	0	0	8.05	S
Join D	0	3	male	39	1	5	31.275	S
Remo	0	3	female	14	0	0	7.8542	S
Select	1	2	female	55	0	0	16	S
Select	0	3	male	2	4	1	29.125	Q
Select	0	3	female	31	1	0	18	S
SMO	0	2	male	35	0	0	26	S
Sampling	1	2	male	34	0	0	13	S
Scale and Reduce	1	2	male	34	0	0	13	S
Feature Selection	1	2	male	34	0	0	13	S

RUN HISTORY SAVE SAVE AS DISCARD CHANGES RUN SET UP WEB SERVICE PUBLISH TO GALLERY

enChian 44

Select Features

Microsoft Azure Machine Learning Studio

Experiment created on 2018/9/3

In draft Draft saved at 下午2:05:50

Properties Project

Experiment Properties

- START TIME 9/3/2018 2:05:50
- END TIME 9/3/2018 2:05:50
- STATUS CODE InDraft
- STATUS DETAILS None

Prior Run

Summary

Enter a few sentences describing your experiment (up to 140 characters).

Description

Enter the detailed description for your experiment.

Import Data → Select Columns in Dataset → Clean Missing Data → Filter Based Feature Selection

選擇對 Y 較有影響力欄位
→ 從 Module 選擇, Feature Selection 中的 [Filter Based Feature Selection]

Note 可以使 Training Data 更適合一點

Quick Help

```

graph TD
    Import[Import Data] --> Select[Select Columns in Dataset]
    Select --> Clean[Clean Missing Data]
    Clean --> FeatureSelection[Filter Based Feature Selection]
  
```

Select Features

Microsoft Azure Machine Learning Studio
Experiment created on 2018/9/3
agogomei-Free-Workspace Properties Project
Draft saved at: 下午 1 Properties Project
Properties Project
Filter Based Feature Selection
Feature scoring method
Pearson Correlation (checked)
Mutual Information
Kendall Correlation
Spearman Correlation
Chi Squared
Fisher Score
Count Based
Launch column selector
Number of desired features
5
選擇選取 Feature 的方法
Text Analytics
Time Series
Web Service
Deprecated

1

2 選擇 Target Column
Select a single column
BY NAME WITH RULES
AVAILABLE COLUMNS: All Types search columns
PassengerClass, Gender, Age, SiblingSpouse, ParentChild, FarePrice, PortEmbarkation
7 columns available
SELECTED COLUMNS: All Types search columns
Survived
1 columns selected
✓

3
Selected columns:
Column names: Survived
Launch column selector
Number of desired features
5
欲挑選 Column 數
Quick Help
Identifies the features in a dataset with the greatest predictive power
(more help...)

RUN HISTORY SAVE AS DISCARD CHANGES RUN SET UP WEB SERVICE PUBLISH TO GALLERY
+ NEW

Select Features

看哪些對 Y 較有影響力欄位
→ Filter Based Feature Selection 右鍵 → Feature 選擇 [Visualize]

Target column:
Selected columns:
Column names: Survived
Launch column selector

Number of desired features:
5

START TIME: 9/3/2018 2:...
END TIME: 9/3/2018 2:...
ELAPSED TIME: 0:00:02.629
STATUS CODE: Finished
STATUS DETAILS: None
[View output log](#)

Quick Help:
Identifies the features in a dataset with the greatest predictive power
(more help...)

Workflow steps:
 1. Select Columns in Dataset
 2. Clean Missing Data
 3. Filter Based Feature Selection (highlighted)

Context menu for 'Filter Based Feature Selection':
 Delete, Copy, Cut, Paste, Filtered dataset, Features, View Log, Edit Comment, Run selected, Help, Download, Save as Dataset, Save as Trained Model, Save as Transform, Visualize, Generate Data Access Code..., Open in a new Notebook...

Select Features

Microsoft Azure Machine Learning Studio

Experiment created on 2018/9/3

Experiment created on 2018/9/3 > Filter Based Feature Selection > Features

rows: 1 columns: 8

	Survived	PassengerClass	FarePrice	ParentChild	Age	SiblingSpouse	Gender	PortEmbarkation
1	1	0.356462	0.2661	0.095265	0.082446	0.015523	0	0

view as:

Statistics

Visualizations

To view, select a column in the table.

Chia 48

NEW RUN HISTORY SAVE AS DISCARD CHANGES RUN SET UP WEB SERVICE PUBLISH TO GALLERY

Split Data

Microsoft Azure Machine Learning Studio

Experiment created on 2018/9/3

In draft Draft saved at 午后2:26:38

Properties Project

Split Data

Splitting mode: Split Rows

Fraction of rows in the first set: 0.5

Randomized split

Random seed: 0

Stratified split: False

```

graph TD
    A[Import Data] --> B[Select Columns in Dataset]
    B --> C[Clean Missing Data]
    C --> D[Filter Based Feature Selection]
    D --> E[Split Data]
    E -- 1 --> F(( ))
    E -- 2 --> G(( ))
  
```

切割 Training / Teating
 → 從 Module 選擇, Data Transformation 中 Sample and Split 的 [Split Data]

Split the rows of a dataset into two distinct sets
 (more help..)

NEW RUN HISTORY SAVE DISCARD CHANGES SET UP WEB SERVICE PUBLISH TO GALLERY

Split Data

Properties Project >

Split Data

Splitting mode: Split Rows

Fraction of rows in the first dataset: 0.7

Randomized split

Random seed: 1

Stratified split: True

Stratification key column: Selected columns: Column names: Survived

Launch column selector

1

選擇 Split Rows 的模式, 依照給定的比例切割資料

2

Training Data:70% Testing Data:30%

3

Random Seed : 為了讓每次執行切的 Dataset 一致

4

Stratified Split : 指對某個 Column 的值平均分配

Note

這裡使用的原因是避免 Training Data 中的 Survived 都是 1 或 0 , 盡量使其平均分配在 Training / Testing 中

Model

Microsoft Azure Machine Learning Studio

Experiment created on 2018/9/3

In draft

Properties Project

Two-Class Decision Jungle

- Resampling method: Bagging
- Create trainer mode: Single Parameter
- Number of decision DAGs: 8
- Maximum depth of the de...: 32
- Maximum width of the de...: 128
- Number of optimization st...: 2048
- Allow unknown values...

```

graph TD
    A[Import Data] --> B[Select Columns in Dataset]
    B --> C[Clean Missing Data]
    C --> D[Filter Based Feature Selection]
    D --> E[Split Data]
    
```

建立 Model

→ 從 Module 選擇, Machine Learning 中 Initialize Model 的 [Classification]
 → [Two-Class Decision Forest]

NEW

RUN HISTORY SAVE SAVE AS DISCARD CHANGES RUN SET UP WEB SERVICE PUBLISH TO GALLERY

Chia 51

Model

Microsoft Azure Machine Learning Studio
Experiment created on 2018/9/3

Draft saved at: 2018/9/3 2:55:01

Properties Project

Experiment Properties

- START TIME: 2018/9/3 2:55:01
- END TIME: 2018/9/3 2:55:01
- STATUS CODE: InDraft
- STATUS DETAILS: None

Prior Run

Summary

Enter a few sentences describing your experiment (up to 140 characters).

Description

Enter the detailed description for your experiment.

Import Data → Select Columns in Dataset → Clean Missing Data → Filter Based Feature Selection → Two-Class Decision Jungle → Train Model → Split Data

1

2

Properties Project

Train Model

Label column

Selected columns:
Column names: Survived

Launch column selector

Select a single column

BY NAME WITH RULES

AVAILABLE COLUMNS

All Types: search columns

PassengerClass
Fare
Parch
Pclass
Age
SibSp
Survived

SELECTED COLUMNS

All Types: search columns

Survived

選 Target 變數

5 columns available 1 columns selected

NEW

RUN HISTORY SAVE SAVE AS DISCARD CHANGES RUN SET UP WEB SERVICE PUBLISH TO GALLERY

Chia 52

Model

Microsoft Azure Machine Learning Studio
Experiment created on 2018/9/3

In draft Draft saved at: 2018/9/3 2:55:01

Properties Project

Experiment Properties

- START TIME 9/3/2018 2:55:01
- END TIME 9/3/2018 2:55:01
- STATUS CODE InDraft
- STATUS DETAILS None

Prior Run

Summary
Enter a few sentences describing your experiment (up to 140 characters).

Description
Enter the detailed description for your experiment.

Import Data ✓
Select Columns in Dataset ✓
Clean Missing Data ✓
Filter Based Feature Selection ✓
Two-Class Decision Jungle
Split Data
Train Model

1

2

Properties Project

Train Model

Label column
Selected columns:
Column names: Survived
Launch column selector

Select a single column

BY NAME WITH RULES
All Types search columns
PassengerClass
Fare
Parch
Age
SibSp

SELECTED COLUMNS
All Types search columns
Survived

選 Target 變數

5 columns available 1 columns selected

NEW RUN HISTORY SAVE DISCARD CHANGES RUN SET UP WEB SERVICE PUBLISH TO GALLERY

Chian 53

Score Model

Microsoft Azure Machine Learning Studio

Experiment created on 2018/9/3

In draft

Draft saved at 下午3:01:11

Properties Project

Score Model

Append score column...

Score

Import Data

Select Columns in Dataset

Clean Missing Data

Filter Based Feature Selection

Two-Class Decision Jungle

Split Data

Train Model

Score Model

Quick Help

Score a trained classification or regression model
(more help...)

Run History Save Save As Discard Changes Run Set Up Web Service Publish To Gallery

```
graph TD; Import[Import Data] --> Select[Select Columns in Dataset]; Select --> Clean[Clean Missing Data]; Clean --> FeatureSel[Filter Based Feature Selection]; FeatureSel --> TwoClass[Two-Class Decision Jungle]; FeatureSel --> Split[Split Data]; TwoClass --> Train[Train Model]; Split --> Train; Train --> ScoreModel[Score Model]; ScoreModel -- feedback --> FeatureSel;
```

Evaluate Model

Microsoft Azure Machine Learning Studio

Experiment created on 2018/9/3

In draft Draft saved at 下午3:02:49

Properties Project
Evaluate Model No parameters

The screenshot shows a workflow titled "Eval" in the Microsoft Azure Machine Learning Studio. The workflow consists of the following steps:

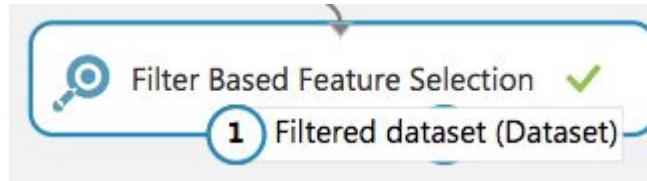
- Import Data
- Select Columns in Dataset
- Clean Missing Data
- Filter Based Feature Selection
- Two-Class Decision Jungle
- Split Data
- Train Model
- Score Model
- Evaluate Model

The "Evaluate Model" step is highlighted with a blue border, indicating it is the current step being viewed. The "Score Model" step is also highlighted with a blue border, suggesting it is the previous step in the process.

Quick Help: Evaluates a scored classification or regression model with standard metrics (more help...)

Navigation icons at the bottom include: NEW, RUN HISTORY, SAVE, SAVE AS, DISCARD CHANGES, RUN, SET UP WEB SERVICE, and PUBLISH TO GALLERY.

Note 為什麼有兩個節點



Note

滑鼠移到連接點上可以查詢

Azure Machine Learning Studio

Microsoft Azure Machine Learning Studio

Titanic

Finished running ✓ Draft saved at 午后3:08:01

Properties Project

Experiment Properties

- START TIME 9/3/2018 3:...
- END TIME 9/3/2018 3:...
- STATUS CODE Finished
- STATUS DETAILS None

Prior Run

Summary

Enter a few sentences describing your experiment (up to 140 characters).

Description

Enter the detailed description for your experiment.

Quick Help

Import Data → Select Columns in Dataset → Clean Missing Data → Filter Based Feature Selection → Two-Class Decision Jungle → Split Data → Train Model → Score Model → Evaluate Model

Search experiment items

Saved Datasets

- My Datasets
 - errors
 - failures
 - machines
 - maint
 - Real Estate Sales Price
 - SECOM_Dataset_AllFeat...
 - table (2).csv
 - telemetry
 - test.csv
 - Titanic
 - Titanic Dataset
 - usedcar1209 - Copy.csv
 - usedcar1209.csv
- Samples
 - Adult Census Income Bi...
 - Airport Codes Dataset
 - Automobile price data (...)
 - Bike Rental UCI dataset
 - Bill Gates RGB Image
 - Blood donation data
 - Book Reviews from Am...

+

NEW

RUN HISTORY SAVE AS DISCARD CHANGES RUN SET UP WEB SERVICE PUBLISH TO GALLERY

```
graph TD; A[Import Data] --> B[Select Columns in Dataset]; B --> C[Clean Missing Data]; C --> D[Filter Based Feature Selection]; D --> E[Two-Class Decision Jungle]; D --> F[Split Data]; E --> G[Train Model]; F --> G; G --> H[Score Model]; H --> I[Evaluate Model]; I --> D;
```

How to connect with Shiny -- Setup Web Service

Microsoft Azure Machine Learning Studio

Titanic

Import Data → Select Columns in Dataset → Clean Missing Data → Filter Based Feature Selection → Two-Class Decision Jungle → Split Data → Train Model → Score Model → Evaluate Model

Properties Project

Experiment Properties

START TIME 9/3/2018 3:...
END TIME 9/3/2018 3:...
STATUS CODE Finished
STATUS DETAILS None

Prior Run

Summary

Enter a few sentences describing your experiment (up to 140 characters).

Description

Enter the detailed description for your experiment.

Quick Help

1 Predictive Web Service [Recommended]
Retraining Web Service

Run History Save Save As Discard Changes Run Set Up Web Service Publish To Gallery

https://studio.azureml.net/Home/ViewWorkspaceCached/852a506a05ab41868939caa8f97d3a57#

How to connect with Shiny -- Setup Web Service

Microsoft Azure Machine Learning Studio

Titanic

STEP 1

INPUT & OUTPUT NODES

You now have the ability to add, change or remove inputs and outputs in your experiment.

NEXT

Filter Based Feature Selection

Two-Class Decision Jungle

Train Model

Score Model

Evaluate Model

Web service input

Web service output

Web service output

Properties Project

Experiment Properties

START TIME 9/3/2018 3:13:51 PM

END TIME 9/3/2018 3:13:51 PM

STATUS CODE InDraft

STATUS DETAILS None

Prior Run

Summary

Description

Quick Help

2

Chia 59

How to connect with Shiny – Deploy Web Service

Microsoft Azure Machine Learning Studio

Titanic

Finished running ✓

Properties Project

Experiment Properties

- START TIME 9/3/2018 3:...
- END TIME 9/3/2018 3:...
- STATUS CODE Finished
- STATUS DETAILS None

Prior Run

Summary

Enter a few sentences describing your experiment (up to 140 characters).

Description

Enter the detailed description for your experiment.

Quick Help

3 Predictive Web Service [Recommended]

Deploy Web Service

Web service input

Import Data

Select Columns in Dataset

Clean Missing Data

Filter Based Feature Selection

Two-Class Decision Jungle

Split Data

Train Model

Score Model

Evaluate Model

Web service output

Input

Output

Deprecated

+

-

1:1

1:n

Run History

Save

Save As

Discard Changes

Run

Set Up Web Service

Publish to Gallery

Go to Azure ML Studio Web Service

Microsoft Azure Machine Learning Studio

titanic

DASHBOARD CONFIGURATION

New Web Services Experience [preview](#)

General Published experiment:

View snapshot View latest

Description

No description provided for this web service.

API key

Cf+IT8XsiGAFZD/ `CUoRZ2RF3A9d+Hlw==`

Default Endpoint

API HELP PAGE TEST LAST UPDATED

REQUEST/RESPONSE Test [Test preview](#) 9/3/2018 3:18:28 PM

BATCH EXECUTION [Test preview](#) 9/3/2018 3:18:28 PM

+

NEW

DELETE

agogomei-Free-Workspace ▾ ? 🔍 😊 🚫

Chia 61

REQUEST/RESPONSE	TEST	LAST UPDATED
	Test Test preview	9/3/2018 3:18:28 PM
	Test preview	9/3/2018 3:18:28 PM

Go to Azure ML Studio Web Service

Microsoft Azure Machine Learning Studio

PROJECTS EXPERIMENTS WEB SERVICES NOTEBOOKS DATASETS TRAINED MODELS SETTINGS

web services

NAME	CREATED ON
<input type="checkbox"/> UsedCarWebService	11/21/2017 11:00:42 AM
<input type="checkbox"/> 5inputExperiment [Predictive Exp.]	11/21/2017 10:13:24 AM
<input type="checkbox"/> 4input_titanic [Predictive Exp.]	11/21/2017 10:12:45 AM
<input type="checkbox"/> 3inputtitanic	11/20/2017 7:54:46 PM
<input checked="" type="checkbox"/> TitanicWS	9/22/2017 9:01:51 PM

Note If You already have web services check here

Azure ML Studio Web Services

≡ Microsoft Azure Machine Learning Web Services

Quickstart

Dashboard

Batch Request Log

Configure

Consume

Test

Swagger API

← TitanicWS

default

View in Studio ↗

Web service consumption options



Excel 2013 or later



Excel 2010 or earlier



Request-Response
Web App Template

Basic consumption info

Want to see how to consume this information? [Check out this easy tutorial.](#)

Primary Key

/D04m7PMuV

rw==



Secondary Key

alcP/Tw3ZJEj

lg==



Azure ML Studio Web Services

≡ Microsoft Azure Machine Learning Web Services

Quickstart

Dashboard

Batch Request Log

Configure

Consume

Test

Swagger API

Sample Code

Request–Response

1

Select R Code

C# Python Python 3+

R

2

Copy



```
library("RCurl")
library("rjson")

# Accept SSL certificates issued by public Certificate Authorities
options(RCurlOptions = list(cainfo = system.file("CurlSSL", "cacert.pem", package = "RCurl")))

h = basicTextGatherer()
hdr = basicHeaderGatherer()

req = list(
  Inputs = list(
    "input1" = list(
      list(
        'PassengerClass' = "3",
        'Gender' = "male",
        'Age' = "22",
        'SiblingSpouse' = "1",
        'ParentChild' = "0",
        'FarePrice' = "7.25",
        'PortEmbarkation' = "S"
      )
    )
  )
)
```

Open R Studio



3 Paste the R Code here

The screenshot shows the R Studio interface. On the left is the 'Script Editor' window with a dark background and a light blue header bar containing the number '1'. A large blue circle highlights the area where R code is pasted. On the right is the 'Console' window, which is currently empty. At the bottom of the screen is the 'Environment' tab, which also has a dark background and displays the message 'Environment is empty'.

R Script 1

Console

Environment is empty

Files Plots Packages Help Viewer

R: Split Data into Test and Train Set [Find in Topic...]

sample.split (caTools)

R Documentation

Split Data into Test and Train Set

Description

Split data from vector Y into two sets in predefined ratio while preserving relative ratios of different labels in Y. Used to split the data used during classification into train and test subsets.

Usage

```
sample.split( Y, SplitRatio = 2/3, group = NULL )
```

Arguments

Y Vector of data labels. If there are only a few labels (as is expected) than relative ratio of data in both subsets will be the same.

SplitRatio Splitting ratio:

- if (0<=splitRatio<1) then SplitRatio fraction of points from Y will be set to TRUE;
- if (SplitRatio==1) then one random point from Y will be set to TRUE
- if (SplitRatio>1) then SplitRatio number of points from Y will be set to TRUE

group Optional vector/list used when multiple copies of each sample are present. In such a case group contains unique sample labels, marking all copies of the same sample with the same label, and the function tries to place all copies in either train or test subset. If provided than it has to have the same length as Y.

Open R Studio



R Studio[®]

The screenshot shows the R Studio interface with the following components:

- Code Editor:** The left pane displays R code for interacting with an API. The code includes imports for RCurl and rjson, sets SSL options, creates basic text and header gatherers, defines a request list with passenger details, and encodes it to JSON. It also includes an API key, constructs an authentication header, and performs a reset.
- Console:** The top right pane is the R console, which is currently empty.
- File Browser:** The bottom right pane shows a file tree with the following contents:
 - ..
 - .History (16.7 KB, modified Nov 21, 2017, 3:12 PM)
 - AnacondaProjects
 - Desktop
 - Documents
 - Downloads
 - Library
 - Movies
 - Music
 - pgadmin.log (1.1 KB, modified Nov 21, 2017, 10:50 AM)
 - Pictures
 - Public
 - Shiny_Titanic
 - VirtualBox VMs
- Environment:** The bottom left pane shows the global environment, which is currently empty.

Azure ML Studio Web Services -- Get API Key

≡ Microsoft Azure Machine Learning Web Services

Quickstart

Dashboard

Batch Request Log

Configure

Consume

Test

Swagger API

← TitanicWS

default

View in Studio ↗

Web service consumption options



Excel 2013 or later



Excel 2010 or earlier



Request-Response
Web App Template

Basic consumption info

Want to see how to consume this information? [Check out this easy tutorial.](#)

Primary Key

/D04m7PMuV

Copy Your API Key

2W==



Secondary Key

alcP/Tw3ZJEj

lg==



Open R Studio



The screenshot shows the R Studio interface with the following code in the script pane:

```
library("RCurl")
library("rjson")
# Accept SSL certificates issued by public Certificate Authorities
options(RCurlOptions = list(cainfo = system.file("CurlSSL", "cacert.pem", package = "RCurl")))
h = basicTextGatherer()
hdr = basicHeaderGatherer()
req = list(
  Inputs = list(
    "input1" = list(
      list(
        'PassengerClass' = "3",
        'Gender' = "male",
        'Age' = "22",
        'SiblingSpouse' = "1",
        'ParentChild' = "0",
        'FarePrice' = "7.25",
        'PortEmbarkation' = "S"
      )
    )
  ),
  GlobalParameters = setNames(fromJSON('{}'), character(0))
)
body = enc2utf8 toJSON(req)
api_key = "abc123" # Replace this with the API key for the web service
authz_header = paste0("Bearer ", api_key, sep=" ")
h$reset()
# curl -X POST "http://mlservice.azuredatabricks.net/api/ml/v1/predictions" -H "Content-Type: application/json" -H "Authorization: Bearer abc123" -d @- <--> body
```

A blue callout box with the number 4 contains the text "Modify Your API Key". A blue arrow points from this box to the line of code where the API key is used.

The RStudio environment pane shows the global environment:

- Environment
- History
- Import Dataset
- Global Environment

The status bar at the bottom indicates "E:\\empty" and "E:\\empty is empty".

A blue callout box at the bottom contains the text "Like: **api_key = "Your API Key"**".

Open R Studio



The screenshot shows the R Studio interface with the following components:

- Code Editor:** Displays the R code for the Azure ML Shiny application. A blue circle labeled **5** is positioned next to the "Run all" button.
- Console:** Shows the R command and its output. A blue circle labeled **6** is positioned next to the output line `$Results$output2[[1]]$PredictedSurvived`.
- Environment:** Shows the global environment variables and their values.
- Note:** A pink note box contains the text: "Just make sure your azure ml can run in R."
- Remember the result format:** A large blue arrow points from the note box to the console output line.
- Result:** The output of the R command is shown in a blue box: `$Results$output2[[1]]$PredictedSurvived`.

```
22
23
24 GlobalParameters = setNames(fromJSON(`[]`), character(0))
25
26
27 body = enc2utf8(toJSON(req))
28 api_key = "/D04m7PMUWw+mhrKLhXnLII9tBQinTmqB8jwfjKoLMiq9FRglwau6rHqqMgov9j+hL/vBFYXcJnQNQGcCr9zPw=="
29 authz_hdr = paste0("Bearer", api_key, sep=" ")
30
31 h$reset()
32 curlPerform(url = "https://ussouthcentral.services.azureml.net/workspaces/852a506a05cb41868939caa8f97d3a57/services/c052c781636540b4a5
33   httpheader=c("Content-Type" = "application/json", "Authorization" = authz_hdr),
34   postfields=body,
35   writefunction = h$update,
36   headerfunction = h$readUpdate,
37   verbose = TRUE
38 )
39
40 headers = h$value()
41 httpStatus = headers[["status"]]
42 if (httpStatus >= 400)
43 {
44   print(paste("The request failed with status code:", httpStatus, sep=" "))
45
46   # Print the headers - they include the request ID and the timestamp, which are useful for debugging the failure
47   print(headers)
48 }
49
50 print("Result:")
51 result = h$value()
52 print(fromJSON(result))
```

Note

Just make sure your azure ml can run in R.

Titanic Survival Prediction

~/Desktop/Shiny/Shiny_Titanic - Shiny

http://127.0.0.1:3857 | Open in Browser | Publish

Titanic Survival Prediction

Passenger Class :

1

Gender :

female

Port Embarkation :

C

Age :

80

Sibling Spouse :

8

Parent Child :

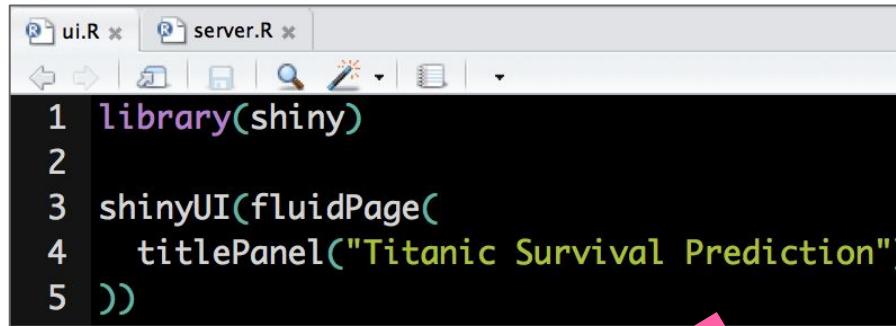
9

Fare Price :

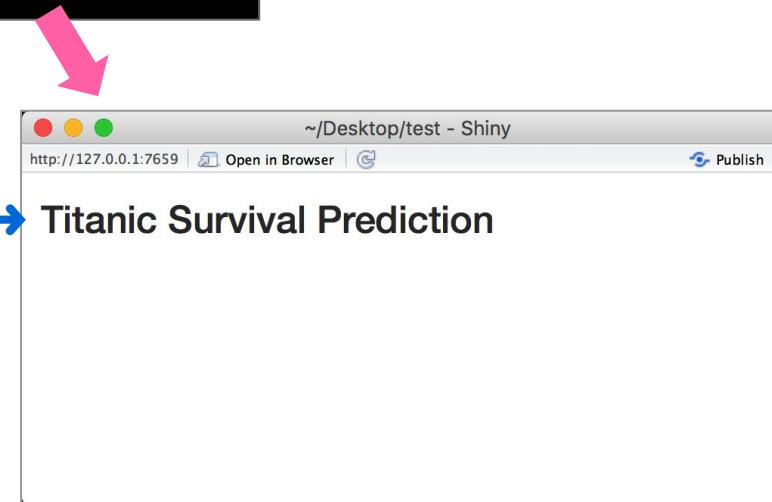
0 51.23 802.466 204.932 307.398 409.864 512.3292



[Code] Titanic Survival Prediction



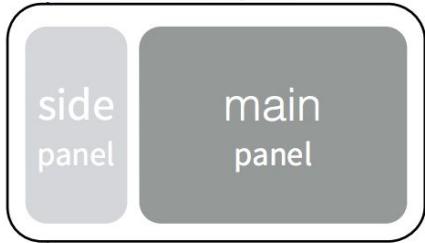
```
ui.R x server.R x
library(shiny)
shinyUI(fluidPage(
  titlePanel("Titanic Survival Prediction")
))
```



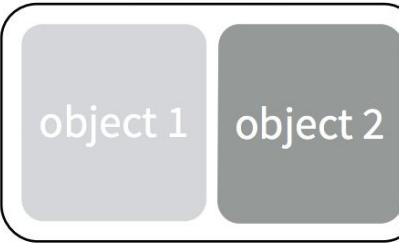
titlePanel → Titanic Survival Prediction

Layout

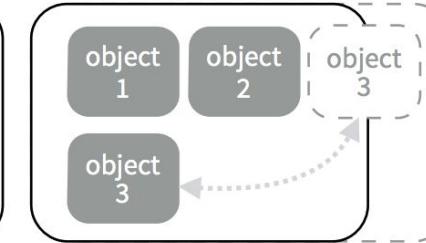
sidebarLayout



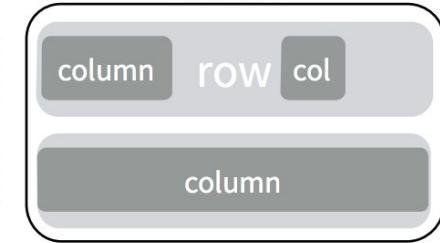
splitLayout



flowLayout/inputPanel



fluidRow



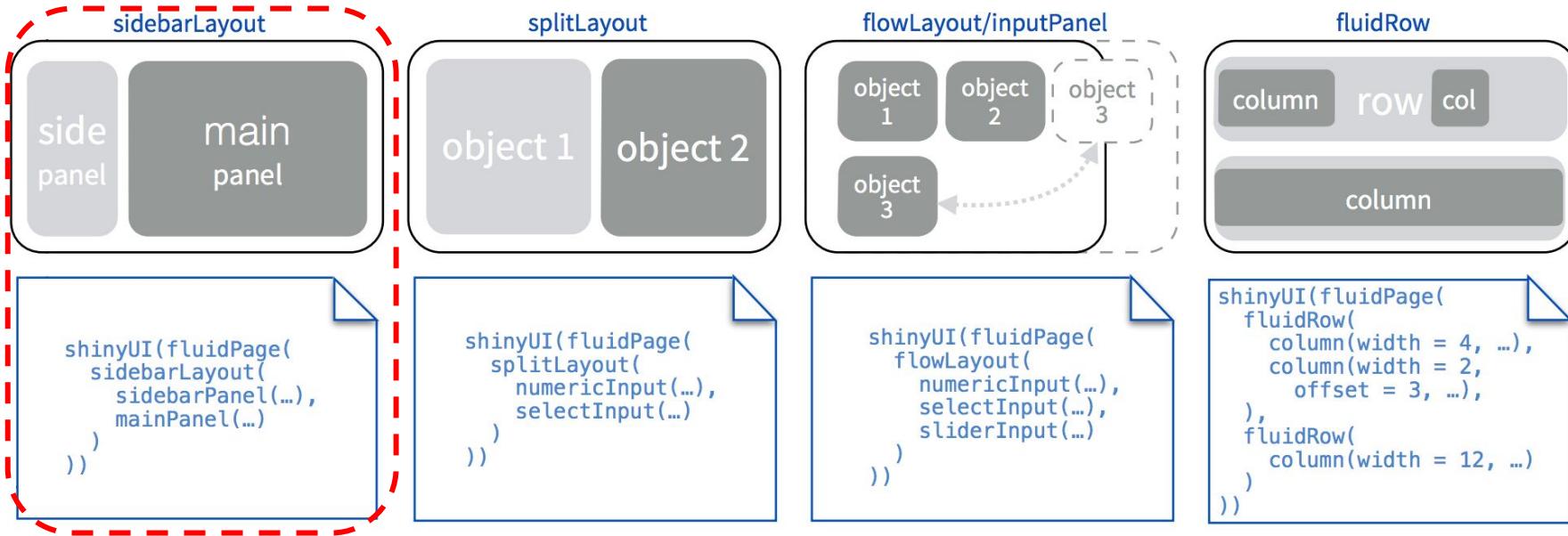
```
shinyUI(fluidPage(  
  sidebarLayout(  
    sidebarPanel(...),  
    mainPanel(...)  
  ))
```

```
shinyUI(fluidPage(  
  splitLayout(  
    numericInput(...),  
    selectInput(...)  
  ))
```

```
shinyUI(fluidPage(  
  flowLayout(  
    numericInput(...),  
    selectInput(...),  
    sliderInput(...)  
  ))
```

```
shinyUI(fluidPage(  
  fluidRow(  
    column(width = 4, ...),  
    column(width = 2,  
      offset = 3, ...),  
  ),  
  fluidRow(  
    column(width = 12, ...)  
  ))
```

Layout



Titanic Survival Prediction

~/Desktop/Shiny/Shiny_Titanic - Shiny

http://127.0.0.1:3857 | Open in Browser | Publish

Titanic Survival Prediction sidebarLayout

Passenger Class : 1

Gender : female

Port Embarkation : C

Age : 80

Sibling Spouse : 8

Parent Child : 9

Fare Price : 0 to 512.3292

```
ui.R x server.R x
library(shiny)
shinyUI(fluidPage(
  titlePanel("Titanic Survival Prediction"),
  sidebarLayout(
```

```
1 library(shiny)
2
3 shinyUI(fluidPage(
4   titlePanel("Titanic Survival Prediction"),
5
6   sidebarLayout(
7
8   )
9
10 ))
```

[Code] Titanic Survival Prediction

The screenshot shows the RStudio interface with the ui.R file open. The code defines a Shiny application with a title panel and a sidebar layout.

```
1 library(shiny)
2
3 shinyUI(fluidPage(
4   titlePanel("Titanic Survival Prediction") ,
5
6   sidebarLayout(
7
8   )
9
10 ))
```

[Code] Titanic Survival Prediction

The screenshot shows the RStudio interface with the ui.R file open. The code defines a sidebar layout with a sidebar panel and a main panel, which are highlighted with colored boxes (pink for sidebarPanel and yellow for mainPanel). The code is as follows:

```
1 library(shiny)
2
3 shinyUI(fluidPage(
4   titlePanel("Titanic Survival Prediction") ,
5
6   sidebarLayout(
7     sidebarPanel(
8
9     ),
10    mainPanel( )
11  )
12
13 ))
```

The Run App button is visible in the top right corner of the RStudio window.

Titanic Survival Prediction

http://127.0.0.1:3857 | Open in Browser | 

~/Desktop/Shiny/Shiny_Titanic - Shiny



Title sidebarPanel **ion**

Passenger Class :

Gender :

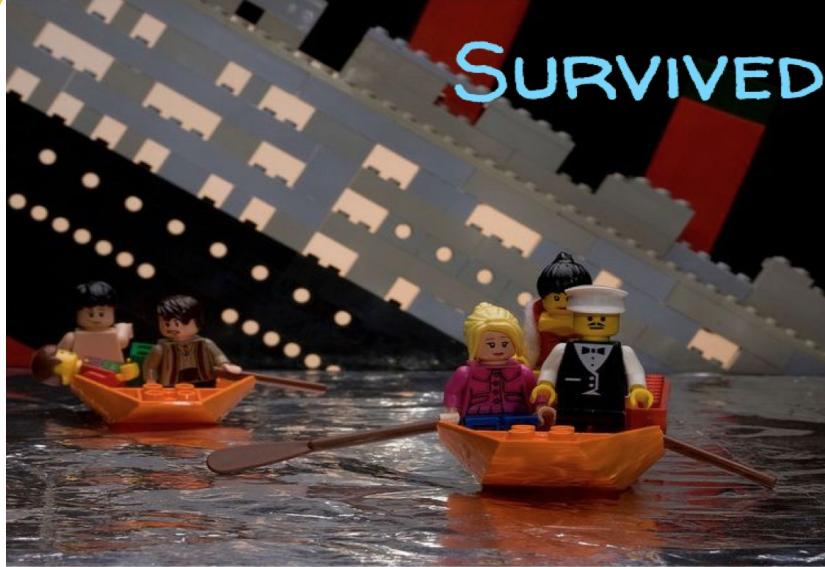
Port Embarkation :

Age :

Sibling Spouse :

Parent Child :

Fare Price :
 512.3292

Shiny Input -- sidePanel

A	Action	actionButton (inputId, label, icon, ...)	F	Choose File	fileInput (inputId, label, multiple, accept)
B	Link	actionLink (inputId, label, icon, ...)	G	1	numericInput (inputId, label, value, min, max, step)
C	<input checked="" type="checkbox"/> Check me	checkboxInput (inputId, label, value)	H	passwordInput (inputId, label, value)
D	<input checked="" type="checkbox"/> Choice 1 <input checked="" type="checkbox"/> Choice 2 <input type="checkbox"/> Choice 3	checkboxGroupInput (inputId, label, choices, selected, inline)	I	<input checked="" type="radio"/> Choice A <input type="radio"/> Choice B <input type="radio"/> Choice C	radioButtons (inputId, label, choices, selected, inline)
E		dateInput (inputId, label, value, min, max, format, startview, weekstart, language)	J		dateRangeInput (inputId, label, start, end, min, max, format, startview, weekstart, language, separator)

Shiny Input -- sidePanel

H	<code>submitButton(text, icon)</code> (Prevents reactions across entire app)	J	<code>textInput(inputId, label, value)</code>
I	<code>selectInput(inputId, label, choices, selected, multiple, selectize, width, size) (also selectizeInput())</code>	K	<code>sliderInput(inputId, label, min, max, value, step, round, format, locale, ticks, animate, width, sep, pre, post)</code>

Titanic Survival Prediction

~/Desktop/Shiny/Shiny_Titanic - Shiny

http://127.0.0.1:3857 | Open in Browser | Publish

Titanic Survival Prediction

Passenger Class :

Gender :

Port Embarkation :

Age :
selectInput ↑

Sibling Spouse :

Parent Child :

Fare Price :



The screenshot shows a Shiny web application interface forTitanic Survival Prediction. On the left, there's a sidebar with several input fields: Passenger Class (dropdown with value 1), Gender (dropdown with value female), Port Embarkation (dropdown with value C), Sibling Spouse (dropdown with value 8), Parent Child (dropdown with value 9), and Fare Price (a slider with a range from 0 to 512.3292, currently set at 0). A prominent button labeled "selectInput ↑" is overlaid on the "Age" input field. On the right, there's a large image of a LEGO Titanic sinking, with two minifigures in lifeboats in the foreground. The word "SURVIVED" is written in blue on the side of the sinking ship.

[Code] Titanic Survival Prediction

The screenshot shows the RStudio interface with the ui.R file open. The code defines a Shiny application with a sidebar layout containing three dropdown inputs for Passenger Class, Gender, and Port Embarkation. The 'Port Embarkation' input is highlighted with a pink rounded rectangle.

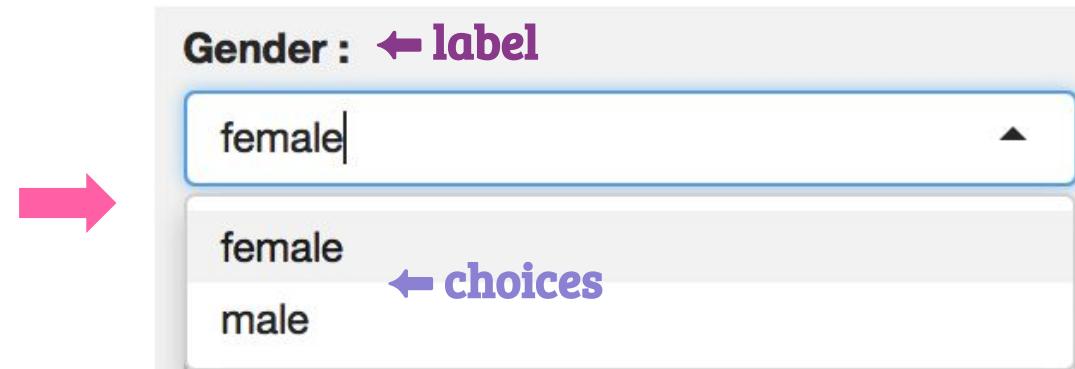
```
1 library(shiny)
2
3 shinyUI(fluidPage(
4   titlePanel("Titanic Survival Prediction"),
5
6   sidebarLayout(
7     sidebarPanel(
8       selectInput("PassengerClass", "Passenger Class : ", choices=pclass),
9       selectInput("Gender", "Gender : ", choices=gender),
10      selectInput("PortEmbarkation", "Port Embarkation : ", choices=embarked)
11    ),
12    mainPanel( imageOutput("result_plot") )
13  )
14 )
15
16 ))
```

[Code] Titanic Survival Prediction-- Example:Gender

```
selectInput("Gender", "Gender : ", choices = gender)
```

inputId label choices

```
gender = c("female", "male")
```



Titanic Survival Prediction

~/Desktop/Shiny/Shiny_Titanic - Shiny

http://127.0.0.1:3857 | Open in Browser | Publish

Titanic Survival Prediction

Passenger Class :

1

Gender :

female

Port Embarkation :

C

Age :

80

Sibling Spouse :

8

Parent Child :

9

Fare Price :

0 51.23 802.466 204.932 307.398 409.864 512.3292

← numericInput

← sliderInput



[Code] Titanic Survival Prediction

```
numericInput( "Age", "Age : ", min = 0, max=100, value=10, step=0.5 )
```

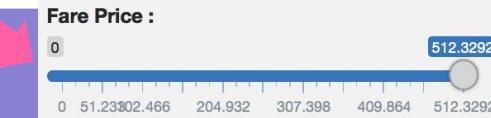
inputId label

min,max: 最大,最小值
value: 第一次啟動時值是多少
step: 間隔多少

```
sliderInput( "FacePrice", "Fare Price : ", min = 0, max=100, value=10, step=0.0001, sep="" )
```

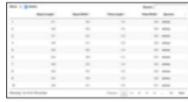
min,max: 最大,最小值
value: 第一次啟動時值是多少
step: 間隔多少
sep: 千位數逗點符號; “” 表示不要有任何符號

Age :

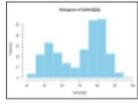


Shiny Output – mainPanel

a

`dataTableOutput(outputId, icon, ...)`

b

`plotOutput(outputId, width, height, click, dblclick, hover, hoverDelay, inline, hoverDelayType, brush, clickId, hoverId)`

c

`uiOutput(outputId, inline, container, ...)`

d

```
'data.Frame': 3 obs. of  2 variables:  
$ Sepal.Length: num  5.1 4.9 4.7  
$ Sepal.Width : num  3.5 3 3.2
```

`verbatimTextOutput(outputId)`

e

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.10	3.50	1.60	0.10	Iris-setosa
2	4.90	3.00	1.40	0.30	Iris-setosa
3	4.70	3.20	1.30	0.20	Iris-setosa
4	4.50	2.30	1.00	0.20	Iris-setosa
5	4.50	2.30	1.40	0.30	Iris-setosa
6	4.50	2.00	1.30	0.00	Iris-setosa
7	4.60	3.10	1.50	0.10	Iris-setosa
8	5.00	3.60	1.40	0.20	Iris-versicolor
9	5.10	3.80	1.30	0.20	Iris-versicolor
10	5.00	3.90	1.50	0.20	Iris-versicolor

`tableOutput(outputId)`

f

`imageOutput(outputId, width, height, click, dblclick, hover, hoverDelay, inline, hoverDelayType, brush, clickId, hoverId)`

g

foo

`textOutput(outputId, container, inline)`

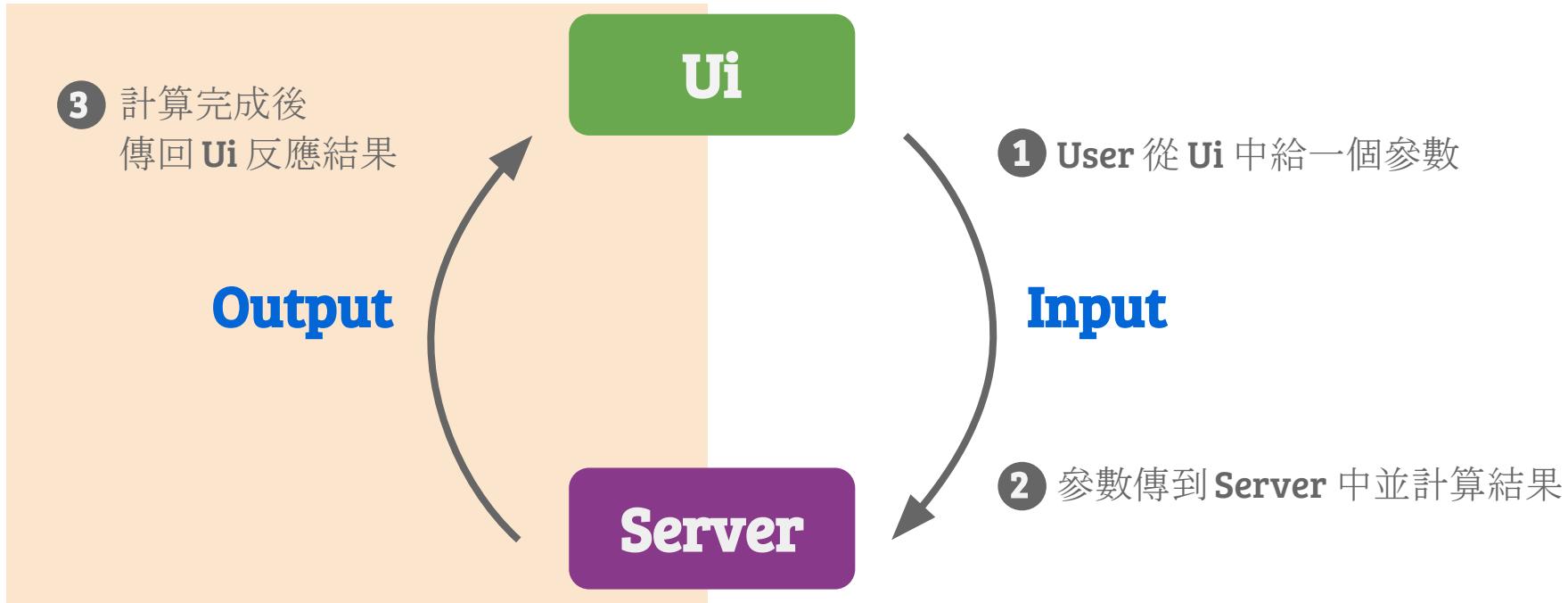
h

`htmlOutput(outputId, inline, container, ...)``plotOutput("hist")`the type of output
to displayname to give to the
output object

[Code] Titanic Survival Prediction -- Original Server.R

```
1 library(shiny)
2
3 server <- function(input, output) {
4
5 }
```

[Review] How does Shiny work



Output Object Name

Ui.R

```
48     numericInput( inputId = "Age",
49                     label = "Age : ",
50                     min = age_min, max =
51
52     numericInput( inputId = "SiblingSpouse",
53                     label = "Sibling Spouse :
54                     min = sibSp_min, max = sib
55
56     numericInput( inputId = "ParentChild",
57                     label = "Parent Child : ",
58                     min = parch_min, max = par
59
60     sliderInput( inputId = "FarePrice",
61                     label = "Fare Price : ",
62                     min = fare_min, max = fare_
63     ),
64     mainPanel( imageOutput("result_plot") )
65   )
66
67 )
```

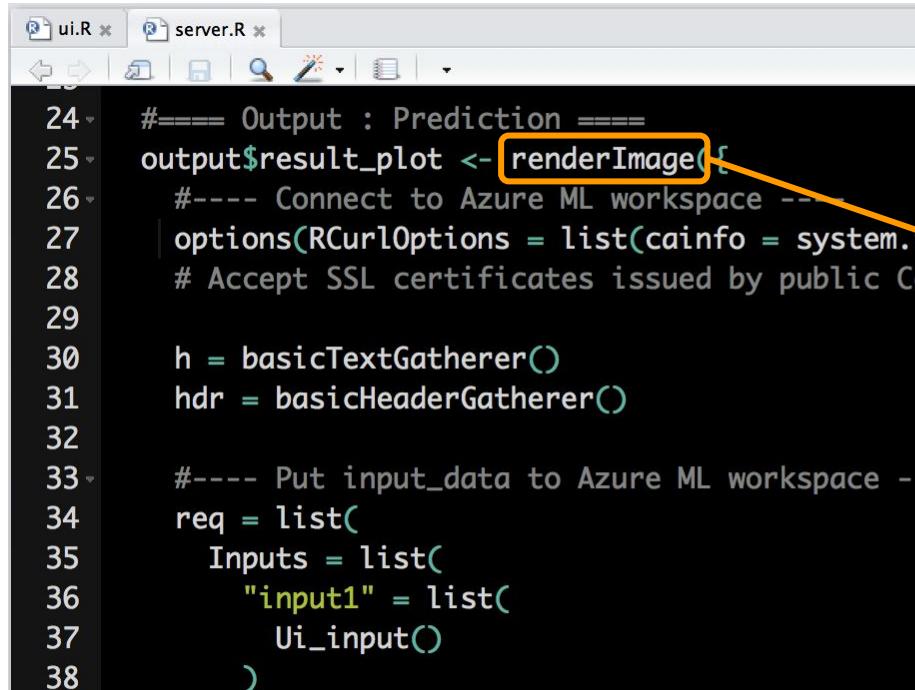
OuputId:
result_plot

Server.R

```
26
27
28
29
30
31
32
33
34
35
36
37
38
#==== Output : Prediction ====
output$result_plot <- renderImage{
  #---- Connect to Azure ML workspace ----
  | options(RCurlOptions = list(cainfo = system.fi
  # Accept SSL certificates issued by public Cer
  |
  h = basicTextGatherer()
  hdr = basicHeaderGatherer()

  #---- Put input_data to Azure ML workspace ---
  req = list(
    Inputs = list(
      "input1" = list(
        Ui_input()
      )
    )
  )
}
```

Build Output Object



The screenshot shows an RStudio interface with two files open: ui.R and server.R. The server.R file contains R code for building a Shiny application. An orange arrow points from the line of code `output$result_plot <- renderImage()` to a set of three orange question marks (???) located to the right of the code editor.

```
24  #==== Output : Prediction ====
25  output$result_plot <- renderImage()
26  #---- Connect to Azure ML workspace ---
27  options(RCurlOptions = list(cainfo = system.file("bin",
28  # Accept SSL certificates issued by public C
29
30  h = basicTextGatherer()
31  hdr = basicHeaderGatherer()
32
33  #---- Put input_data to Azure ML workspace ---
34  req = list(
35    Inputs = list(
36      "input1" = list(
37        Ui_input()
38    )

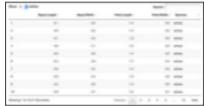
```

???

render*() and Output() functions work together
to add R output to the UI

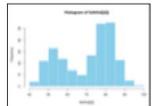
render*() VS. Output()

Server.R



`DT::renderDataTable(expr, options, callback, escape, env, quoted)`

`renderImage(expr, env, quoted, deleteFile)`



`renderPlot(expr, width, height, res, ..., env, quoted, func)`

`data.Frame[1]: 3 obs. of 2 variables:`
`$ Sepal.Length: num 5.1 4.9 4.7`
`$ Sepal.Width : num 3.5 3 3.2`

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	Iris-setosa
2	4.9	3.0	1.4	0.2	Iris-setosa
3	4.7	3.2	1.3	0.2	Iris-setosa
4	4.6	3.1	1.5	0.2	Iris-setosa
5	5.0	3.4	1.5	0.2	Iris-setosa
6	5.4	3.9	1.7	0.3	Iris-versicolor
7	7.0	3.0	1.0	0.2	Iris-versicolor
8	6.4	3.2	1.1	0.3	Iris-versicolor
9	5.7	3.0	1.6	0.2	Iris-versicolor
10	5.1	3.0	1.9	0.3	Iris-versicolor

foo



`renderPrint(expr, env, quoted, func, width)`

`renderTable(expr,..., env, quoted, func)`

`renderText(expr, env, quoted, func)`

`renderUI(expr, env, quoted, func)`

Ui.R

works
with

`dataTableOutput(outputId, icon, ...)`

`imageOutput(outputId, width, height, click, dblclick, hover, hoverDelay, inline, hoverDelayType, brush, clickId, hoverId)`

`plotOutput(outputId, width, height, click, dblclick, hover, hoverDelay, inline, hoverDelayType, brush, clickId, hoverId)`

`verbatimTextOutput(outputId)`

`tableOutput(outputId)`

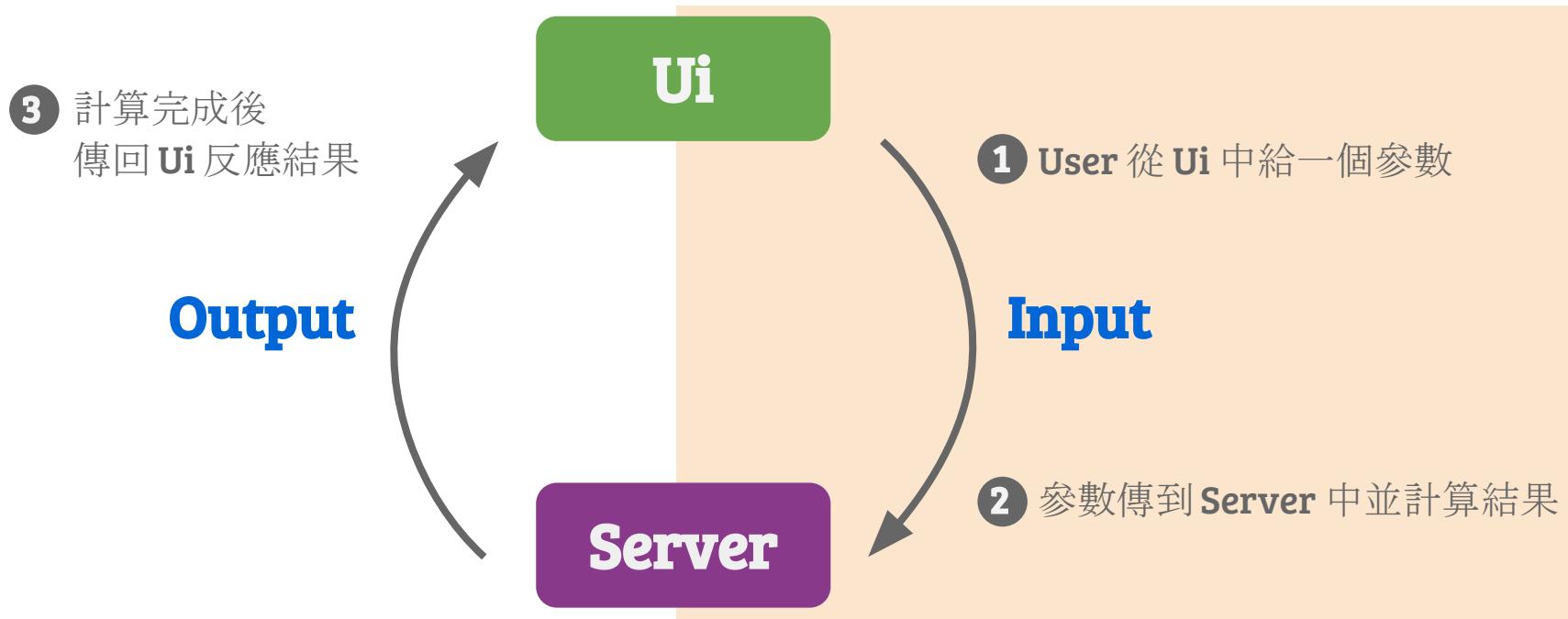
`textOutput(outputId, container, inline)`

`uiOutput(outputId, inline, container, ...)`

`htmlOutput(outputId, inline, container, ...)`

&

[Review] How does Shiny work



Access Input Value

Ui.R

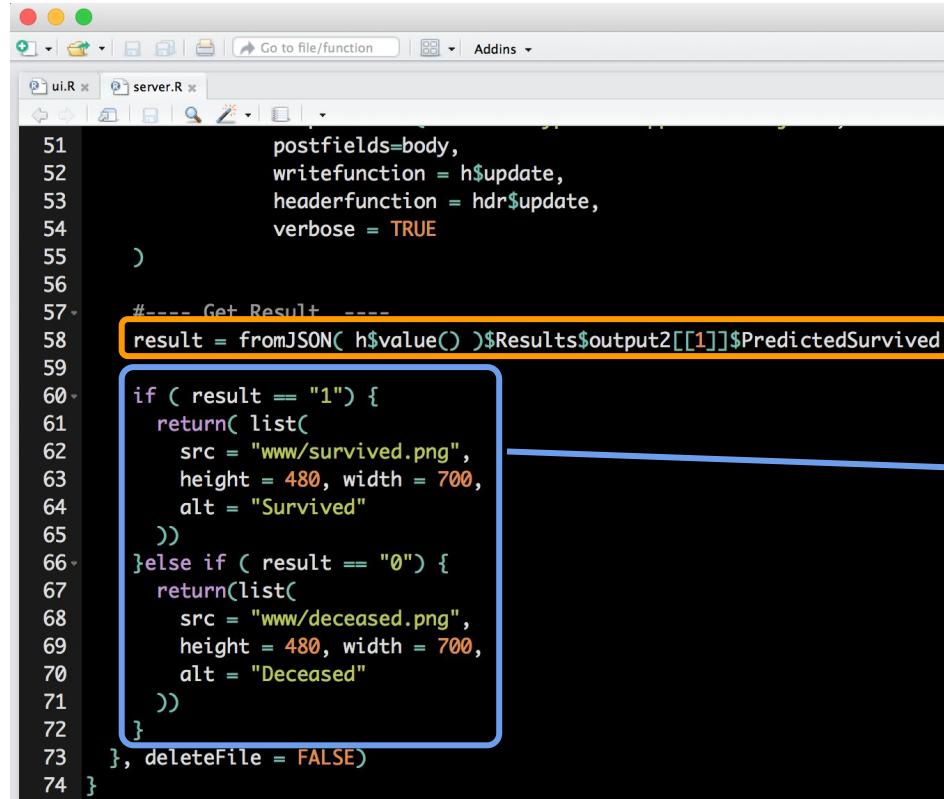


Server.R

```
55
34 sidebarLayout(
35   sidebarPanel(
36     selectInput(inputId = "PassengerClass",
37                 label = "Passenger Class : ",
38                 choices = pclass),
39
40     selectInput(inputId = "Gender",
41                 label = "Gender : ",
42                 choices=gender),
43
44     selectInput(inputId = "PortEmbarkation",
45                 label = "Port Embarkation : ",
46                 choices=embarked),
47
48     numericInput(inputId = "Age",
49                 label = "Age : ",
50                 min = age_min, max = age_max, value = age_mean)
```

```
8
9 #=====
10 # Server
11 #
12 function(input, output) {
13   #---- Get UI.R's input ----
14   Ui_input <- reactive({
15     list(
16       'PassengerClass' = input$PassengerClass,
17       'Gender' = input$Gender,
18       'Age' = as.character(input$Age),
19       'SiblingSpouse' = as.character(input$SibSpouse),
20       'ParentChild' = as.character(input$Parch),
21       'FarePrice' = as.character(input$FarePrice),
22       'PortEmbarkation' = input$PortEmbarkation)}
```

Get AzureML Result & Setting Output Plot



```
51         postfields=body,
52         writefunction = h$update,
53         headerfunction = hdr$update,
54         verbose = TRUE
55     )
56
57     #---- Get Result ----
58     result = fromJSON( h$value() )$Results$output2[[1]]$PredictedSurvived
59
60     if ( result == "1" ) {
61         return( list(
62             src = "www/survived.png",
63             height = 480, width = 700,
64             alt = "Survived"
65         ))
66     } else if ( result == "0" ) {
67         return( list(
68             src = "www/deceased.png",
69             height = 480, width = 700,
70             alt = "Deceased"
71         ))
72     }
73 }, deleteFile = FALSE)
74 }
```

←AzureML Result

```
if ( result == "1" ) {
    Survived Plot
} else if ( result == "0" ) {
    Deceased Plot
}
```

~/Desktop/Shiny/Shiny_Titanic - Shiny

http://127.0.0.1:3857 | [Open in Browser](#)

Titanic Survival Prediction

Passenger Class :

1

Gender :

female

Port Embarkation :

C

Age :

80

Sibling Spouse :

8

Parent Child :

9

Fare Price :

0 512.3292

512.3292

0 51.23802.466 204.932 307.398 409.864 512.3292

KristenChen 94

Get Kaggle Leaderboard with Shiny

Kaggle

Featured Prediction Competition

Home Credit Default Risk

Can you predict how capable each applicant is of repaying a loan?

Home Credit Group · 6,006 teams · 22 days to go (15 days to go until merger deadline)

\$70,000 Prize Money

Join Competition

Overview Data Kernels Discussion Leaderboard Rules Join Competition

Overview

Description Many people struggle to get loans due to insufficient or non-existent credit histories. And, unfortunately, this population is often taken advantage of by untrustworthy lenders.

Evaluation Home Credit strives to broaden financial inclusion for the unbanked population by providing a positive and safe borrowing experience. In order to make sure this underserved population has a positive loan experience, Home Credit makes use of a variety of alternative data--including telco and transactional information--to predict their clients' repayment abilities.

Prizes

Timeline

While Home Credit is currently using various statistical and machine learning methods to make these predictions, they're challenging Kagglers to help them unlock the full potential of their data. Doing so will ensure that clients capable of repayment are not rejected and that loans are given with a principal, maturity, and repayment calendar that will empower their clients to be successful.



Kaggle Leaderboard

Featured Prediction Competition

Home Credit Default Risk

Can you predict how capable each applicant is of repaying a loan?

Home Credit Group · 6,006 teams · 22 days to go (15 days to go until merger deadline)

Overview Data Kernels Discussion **Leaderboard** Rules Team My Submissions Submit Predictions

Public Leaderboard Private Leaderboard

This leaderboard is calculated with approximately 20% of the test data.
The final results will be based on the other 80%, so the final standings may be different.

Raw Data Refresh

#	△1w	Team Name	Kernel	Team Members	Score	Entries	Last
1	▲ 144	SandHill			0.809	193	4h
2	▲ 4	Large hypothesis space			0.809	191	9h
3	▼ 2	vegetable chicken			0.809	307	8h

4



home-credit-default-...oard.csv



Unzip

3



publicleaderboard
data.zip



R-Ladies Kaggle

http://127.0.0.1:5083 | Open in Browser | ~/Desktop/2018_Kaggle_Rank - Shiny

R-Ladies Taipei x Kaggle About Ranking Another No1. Attendees More ▾

`library(dplyr)
r-ladies_taipei %>%
 filter(topic = '2018 R-Ladies Taipei Kaggle')`

R-Ladies Taipei

2018 R- Ladies Taipei Kaggle 大賽

時間: 08/11 11:00 - 08/12 14:30
地點: 台北市中正區市民大道三段2號 11F

R-Ladies Taipei

主辦單位 協辦單位 鑽石級贊助商 黃金級贊助商

#參加kaggle松，保證肉變鬆
#臺灣第一個全女性Kaggle黑客松
R-Ladies Taipei Community

R-Ladies Kaggle

http://127.0.0.1:5083 | Open in Browser | ↗

R-Ladies Taipei x Kaggle About Ranking Another No1. Attendees More ▾

最新戰況

Update Time : 2018-09-02 07:35:51

Show 10 entries

TeamType	TeamName	TeamRole	SubmissionDate	Score
1 Team	Celine	Leader	2018-08-12 04:14:16	0.795
2 Assistant	Pei Lee	Leader	2018-07-07 07:18:53	0.795
3 Team	PepperChang	Leader	2018-08-12 04:56:53	0.789
4 Team	YoungmiHuang	Leader	2018-08-12 04:58:43	0.78
5 Team	hahahahahahah	Leader	2018-08-11 15:41:59	0.778
6 Team	Tricia Tsai	Leader	2018-08-12 04:58:02	0.771
7 Team	Daisy Chou	Leader	2018-08-12 04:33:24	0.766
8 Team	Shirley Chou	Leader	2018-08-12 02:01:38	0.74
9 Team	A.Y. Chang	Leader	2018-08-12 01:53:30	0.723
10 Team	Ching Tseng	Leader	2018-08-12 02:14:07	0.713

Showing 1 to 10 of 16 entries

Previous 1 2 Next

Automatic downloading

Automatic downloading

Note 設定排程 (Azure VM -- Ubuntu)

```
$ vi /etc/crontab
```

```
SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 *      * * *    root    cd / && run-parts --report /etc/cron.hourly
25 6      * * *    root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6      * * 7    root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6      1 * *    root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
00 0-23 * * *    root    bash /srv/shiny-server/2018_Kaggle_Rank/kaggle.sh
#
~
```



Let's Play Shiny

Open Data -- 水質監測資訊

臺北市水質監測資訊

資料集提供機關



臺北自來水事業處

分享至

Google+

Twitter

Facebook

資料集 | 同分類下 | 活動紀錄 | 意見反應 | 友善列印

臺北市水質監測資訊

分類及編號：生活安全及品質

資料集描述：臺北市自來水事業處各水質監測站，所測得水質之濁度、餘氯、酸鹼度等數據15分鐘平均值

主要欄位說明：1.update_date : 日期 2.update_time : 時間 3.qua_id : 監控站ID 4.code_name : 監控站名稱 5.longitude : 經度 6.latitude : 緯度 7.qua_ntu: 濁度(NTU) 8.qua_cl: 餘氯(mg/L) 9.qua_ph: 酸度(pH) 10.偵測值-9係維修中，暫停偵測。

資料集類型：原始資料
最後更新時間：2018-09-03 17:04:02

資料項目

- 水質監測資訊 使用資料
- 水質監測資訊(新版) 使用資料

水質監測

詮釋資料

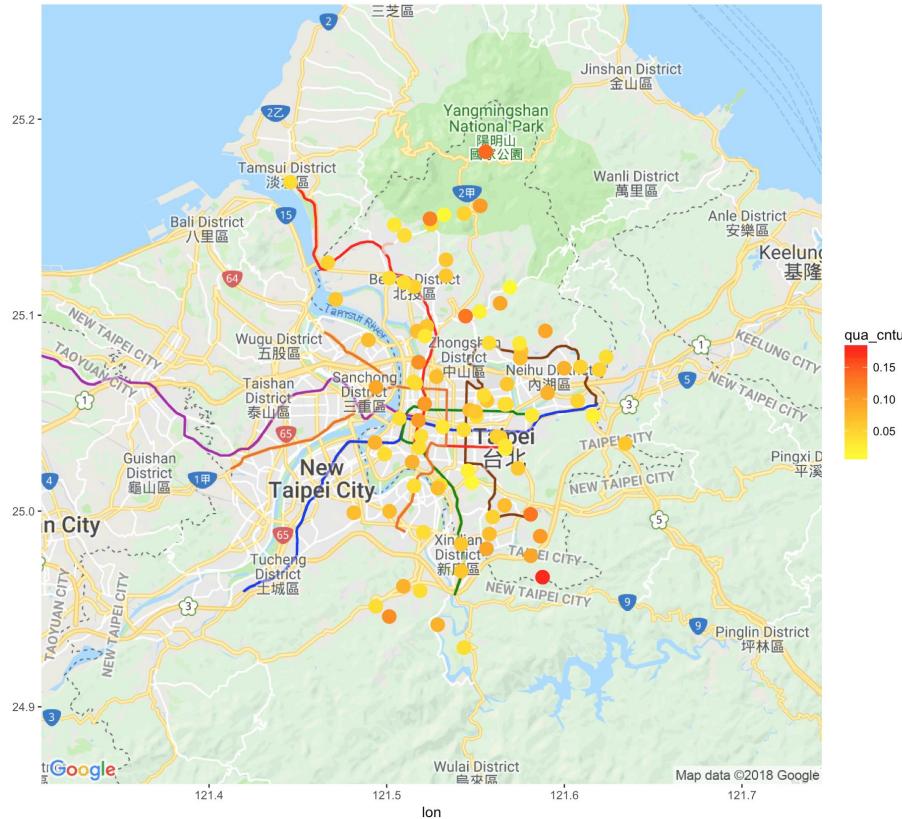
分類及編號	生活安全及品質
資料集描述	臺北市自來水事業處各水質監測站，所測得水質之濁度、餘氯、酸鹼度等數據15分鐘平均值
主要欄位說明	1.update_date : 日期 2.update_time : 時間 3.qua_id : 監控站ID 4.code_name : 監控站名稱 5.longitude : 經度 6.latitude : 緯度 7.qua_ntu: 濁度(NTU) 8.qua_cl: 餘氯(mg/L) 9.qua_ph: 酸度(pH) 10.偵測值-9係維修中，暫停偵測。
資料集類型	原始資料
最後更新時間	2018-09-03 17:04:02
資料量	91
收錄期間(起)	

Open Data -- 水質監測資訊

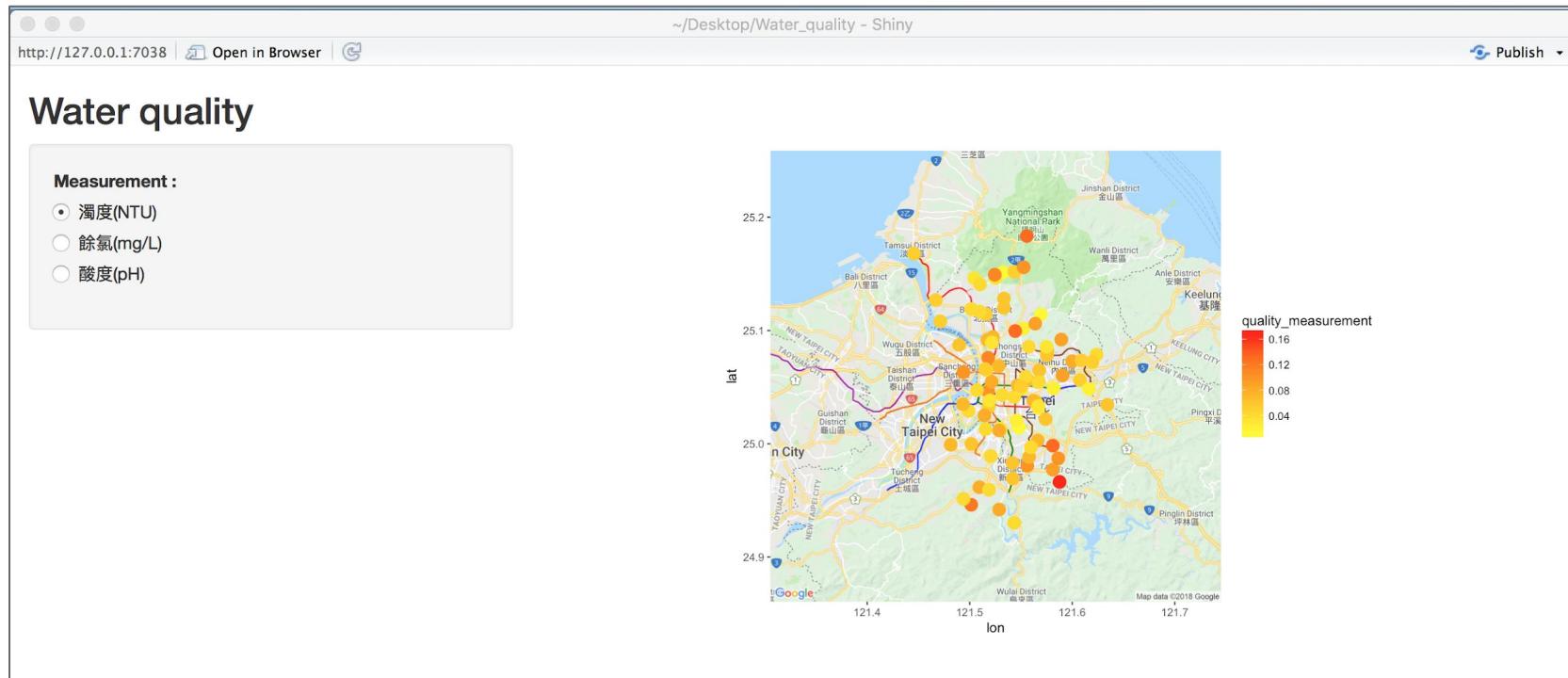
欄位說明

- update_date : 日期
- update_time : 時間
- qua_id : 監控站ID
- code_name : 監控站名稱
- longitude : 經度
- latitude: 緯度
- qua_cntu: 濁度(NTU)
- qua_cl: 餘氯(mg/L)
- qua_ph: 酸度(pH)

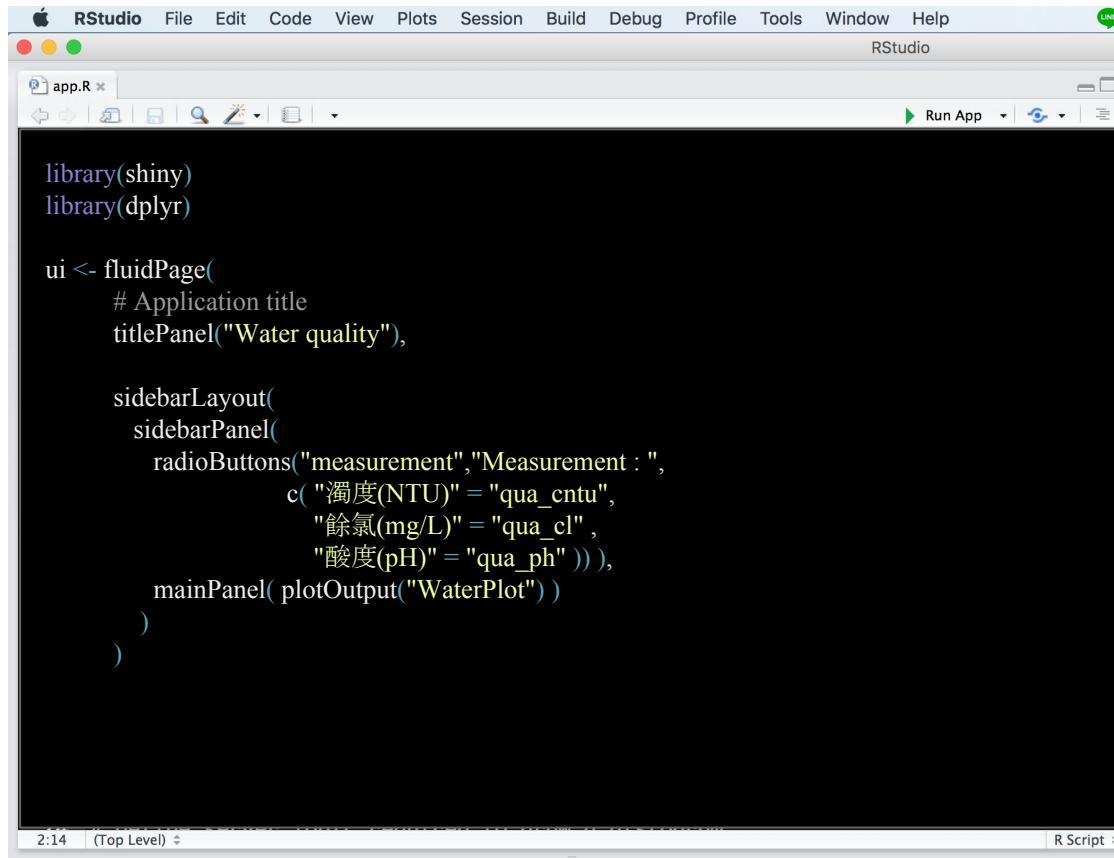
R Plot



Shiny



[Code] Shiny



The screenshot shows the RStudio interface with the following details:

- Toolbar:** RStudio, File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Window, Help.
- Top Bar:** A green "Link" button is visible on the right.
- File Bar:** Shows "app.R" as the active file.
- Toolbars:** Standard RStudio toolbars for file operations (New, Open, Save, Print) and search.
- Code Editor:** The main area contains the following R code:

```
library(shiny)
library(dplyr)

ui <- fluidPage(
  # Application title
  titlePanel("Water quality"),

  sidebarLayout(
    sidebarPanel(
      radioButtons("measurement", "Measurement :",
                  c( "濁度(NTU)" = "qua_ntu",
                     "餘氯(mg/L)" = "qua_cl" ,
                     "酸度(pH)" = "qua_ph" )),

      mainPanel( plotOutput("WaterPlot") )
    )
  )
)
```

The code defines a Shiny application with a title panel and a sidebar layout. The sidebar includes a radio button group for measurement selection (浊度(NTU), 餘氯(mg/L), 酸度(pH)) and a main panel for displaying a plot.

[Code] Shiny

http://data.taipei/opendata/datalist/apiAccess?scope=resourceAquire&rid=190796c8-7c56-42e0-8068-39242b8ec927

```
server <- function(input, output) {  
  WaterData<-fromJSON(getURL("http://data.taipei/opendata/datalist/apiAccess?scope=resourceAquire&rid=190796c8-7c56-42e0-8068-39242b8ec927"))  
  WaterDataFrame<-WaterData$result$results  
  WaterDataFrame$longitude<-as.numeric(WaterDataFrame$longitude)  
  WaterDataFrame$latitude<-as.numeric(WaterDataFrame$latitude)  
  WaterDataFrame$qua_cntu<-as.numeric(WaterDataFrame$qua_cntu)  
  WaterDataFrame$qua_cl<-as.numeric(WaterDataFrame$qua_cl)  
  WaterDataFrame$qua_ph<-as.numeric(WaterDataFrame$qua_ph)  
  
  output$WaterPlot <- renderPlot({  
    WaterData = WaterDataFrame %>%  
      select(longitude,latitude,quality_measurement = input$measurement)  
    ## -- Map ---  
    TaipeiMap = get_map(location = c(121.43,24.93,121.62,25.19), zoom = 11, maptype = 'roadmap')  
    TaipeiMapO = ggmap(TaipeiMap)+  
      geom_point(data=WaterData[WaterData$quality_measurement>=0,],  
                 aes(x=longitude, y=latitude,color=quality_measurement,size=3.5))+  
      scale_color_continuous(low = "yellow",high = "red")+  
      guides(size=FALSE)  
    TaipeiMapO  
  })}  
  #Run the application  
  shinyApp(ui = ui, server = server)
```

Reference

✓ Shiny Cheat Sheet

<https://shiny.rstudio.com/articles/cheatsheet.html>

✓ More Shiny Example

<https://shiny.rstudio.com/gallery/>