AI 實作課程

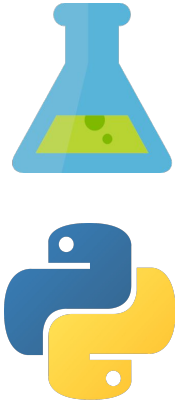
Azure ML X Python

Kristen Chan

2019.05.18

Agenda

- Kaggle
- Microsoft Azure Machine Learning
- Python



Hello !

I am Kristen Chan

Data Scientist
E-Commerce / Telecom

R-Ladies Taipei
Co-Organizer link

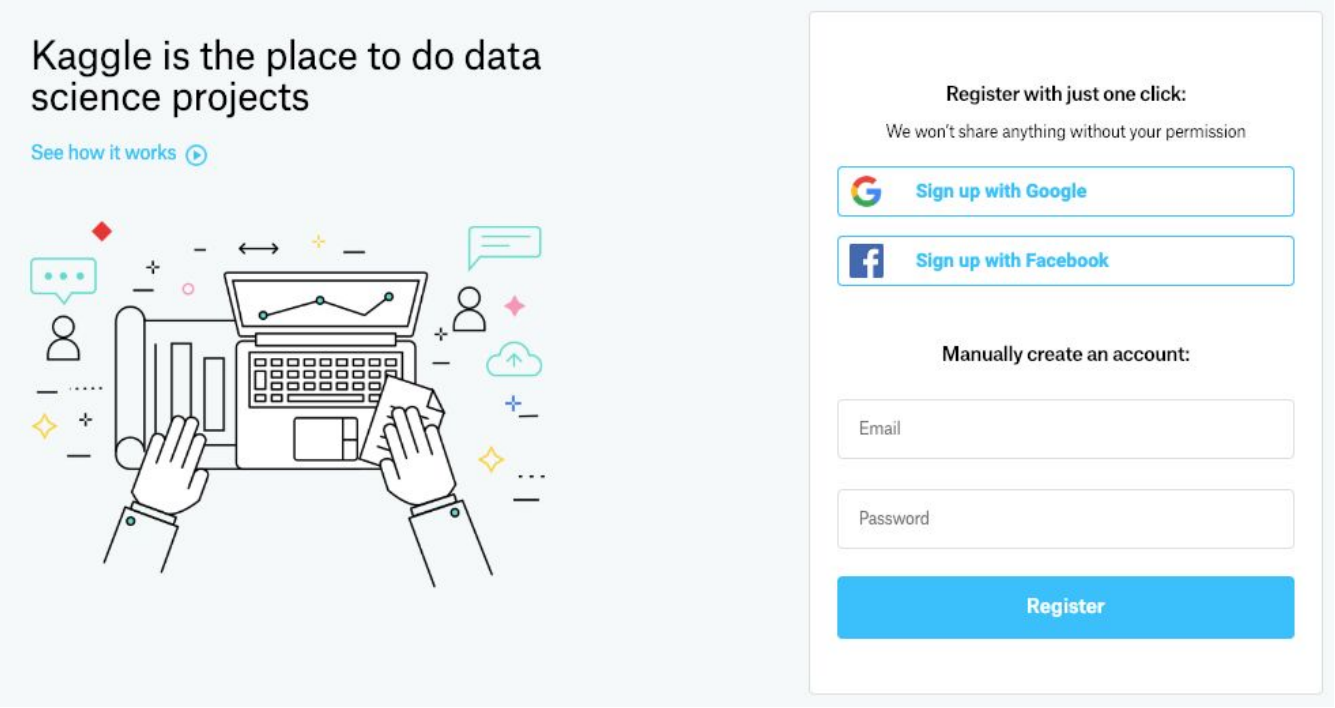
Kaggle Introduction

什麼是 Kaggle？

- Website : https://www.kaggle.com
- 資料科學和機器學習競賽平台
- 目前已累積超過 50萬名、遍布超過194個國家的註冊用戶
- 涵蓋電腦科學、電腦視覺、生物、醫藥
- Kaggle 排行榜更成為業界找尋人才的指標

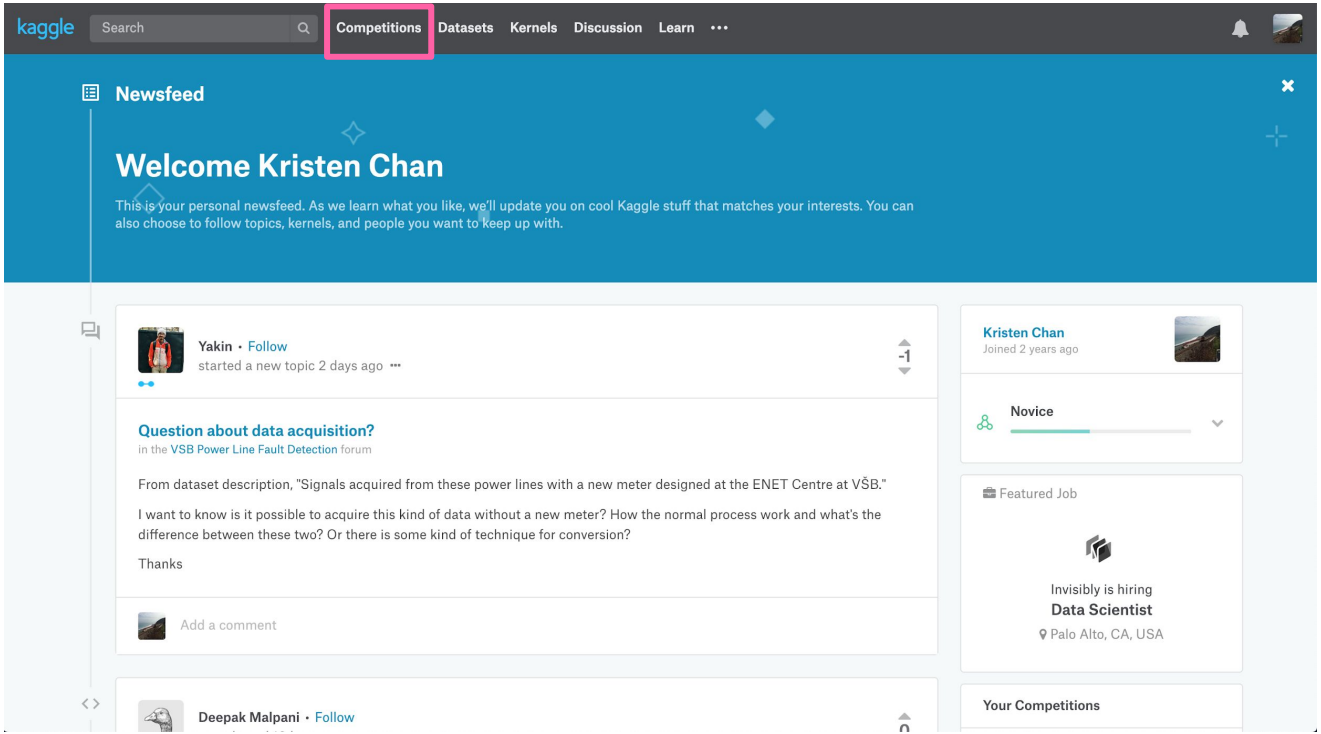
開始前，你需要先...

- Sign in



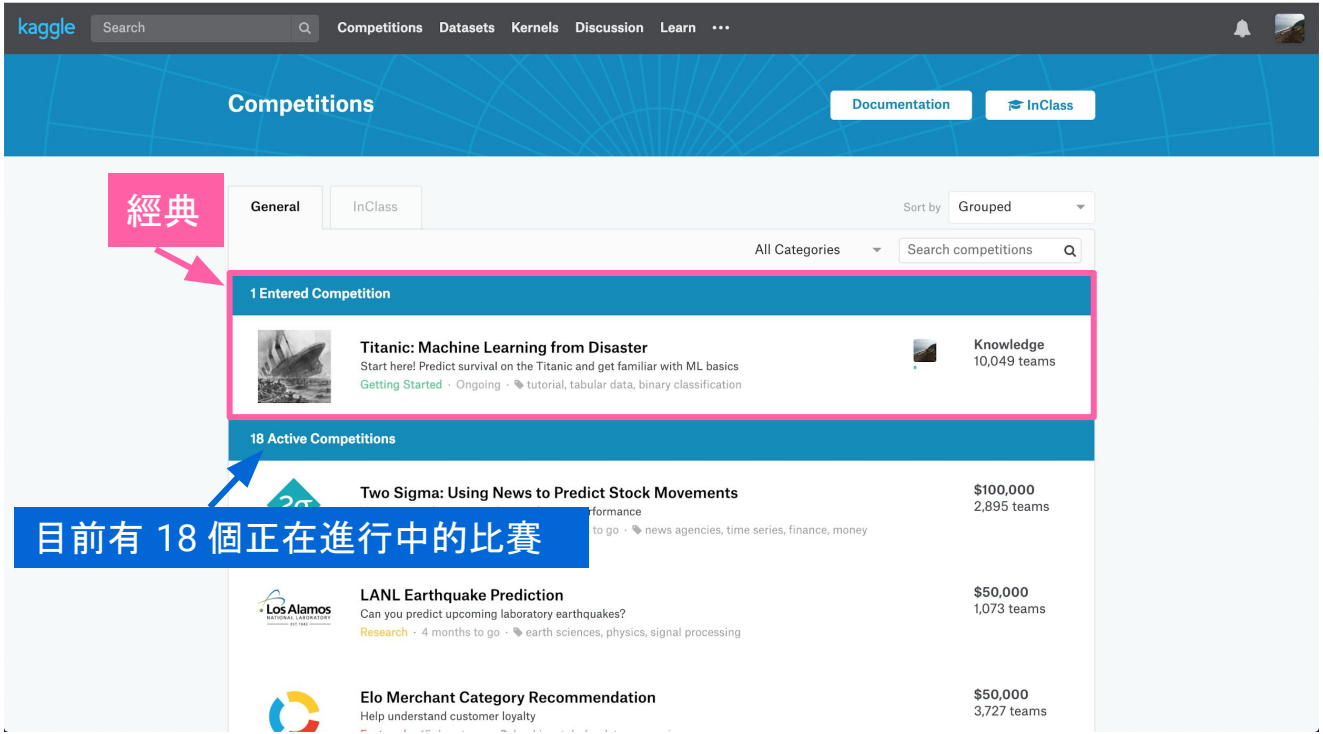
準備開始...

- 找到 Competitions



找一個想要比的競賽

- Titanic: Machine Learning from Disaster



獎牌資格

Competition Medals

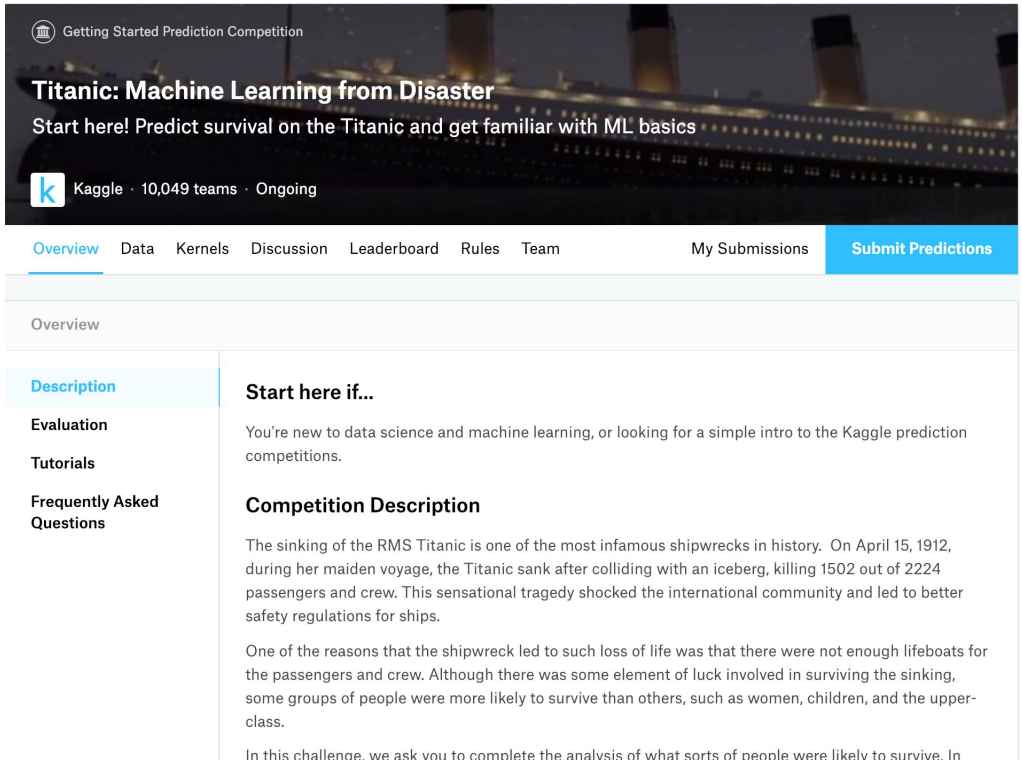
Competition medals are awarded for top competition results. The number of medals awarded per competition varies depending on the size of the competition. Note that InClass, playground, and getting started competitions do not award medals.

	0-99 Teams	100-249 Teams	250-999 Teams	1000+ Teams
Bronze	Top 40%	Top 40%	Top 100	Top 10%
Silver	Top 20%	Top 20%	Top 50	Top 5%
Gold	Top 10%	Top 10	Top 10 + 0.2%*	Top 10 + 0.2%*

* (Top 10 + 0.2%) means that an extra gold medal will be awarded for every 500 additional teams in the competition. For example, a competition with 500 teams will award gold medals to the top 11 teams and a competition with 5000 teams will award gold medals to the top 20 teams.

Titanic Data Sets

Titanic : Machine Learning from Disaster



All Data Sets

資料集	資料維度
A. training.csv	891 X 12
B. testing.csv	418 X 11
C. gender_submission.csv	418 X 2

1. PassengerId	乘客 ID	
2. Survived	是否存活	0 = No, 1 = Yes
3. Pclass	船票的等級	1 = 1st, 2 = 2nd, 3 = 3rd
4. Name	乘客姓名	
5. Sex	性別	
6. Age	年齡	
7. SibSp	兄弟姊妹/配偶數	
8. Parch	父母/小孩數	
9. Ticket	船票編號	
10.Fare	船票價格	
11.Cabin	船艙號碼	
12.Embarked	登岸的港口	C = Cherbourg Q = Queenstown S = Southampton

All Data Sets

資料集	資料維度
A. training.csv	891 X 12
B. testing.csv	418 X 11
C. gender_submission.csv	418 X 2

表示社會經濟地位：1st = Upper,
2nd = Middle,
3rd = Lower

1. PassengerId	乘客 ID	
2. Survived	是否存活	0 = No, 1 = Yes
3. Pclass	船票的等級	1 = 1st, 2 = 2nd, 3 = 3rd
4. Name	乘客姓名	
5. Sex	性別	
6. Age	年齡	
7. SibSp	兄弟姊妹/配偶數	
8. Parch	父母/小孩數	
9. Ticket	船票編號	
10.Fare	船票價格	
11.Cabin	船艙號碼	
12.Embarked	登岸的港口	C = Cherbourg Q = Queenstown S = Southampton

All Data Sets

資料集	資料維度
A. training.csv	891 X 12
B. testing.csv	418 X 11
C. gender_submission.csv	418 X 2

小於 1 歲：會以小數表示
若看到 xx.5 的年齡表示是估計的

1. PassengerId	乘客 ID	
2. Survived	是否存活	0 = No, 1 = Yes
3. Pclass	船票的等級	1 = 1st, 2 = 2nd, 3 = 3rd
4. Name	乘客姓名	
5. Sex	性別	
6. Age	年齡	
7. SibSp	兄弟姊妹/配偶數	
8. Parch	父母/小孩數	
9. Ticket	船票編號	
10.Fare	船票價格	
11.Cabin	船艙號碼	
12.Embarked	登岸的港口	C = Cherbourg Q = Queenstown S = Southampton

All Data Sets

資料集	資料維度
A. training.csv	891 X 12
B. testing.csv	418 X 11
C. gender_submission.csv	418 X 2

Sibling = 兄弟, 姐妹, 繼兄弟, 繼姐妹
Spouse = 丈夫, 妻子（不包括情婦和未婚夫）

1. PassengerId	乘客 ID	
2. Survived	是否存活	0 = No, 1 = Yes
3. Pclass	船票的等級	1 = 1st, 2 = 2nd, 3 = 3rd
4. Name	乘客姓名	
5. Sex	性別	
6. Age	年齡	
7. SibSp	兄弟姊妹/配偶數	
8. Parch	父母/小孩數	
9. Ticket	船票編號	
10.Fare	船票價格	
11.Cabin	船艙號碼	
12.Embarked	登岸的港口	C = Cherbourg Q = Queenstown S = Southampton

Kristen Chiang

All Data Sets

資料集	資料維度
A. training.csv	891 X 12
B. testing.csv	418 X 11
C. gender_submission.csv	418 X 2

Parent = 父親, 母親
Child = 兒子, 女兒, 繼兒子, 繼女兒
[Note] 有些小孩只有保母陪同，所以 Parch =0

1. PassengerId	乘客 ID	
2. Survived	是否存活	0 = No, 1 = Yes
3. Pclass	船票的等級	1 = 1st, 2 = 2nd, 3 = 3rd
4. Name	乘客姓名	
5. Sex	性別	
6. Age	年齡	
7. SibSp	兄弟姊妹/配偶數	
8. Parch	父母/小孩數	
9. Ticket	船票編號	
10.Fare	船票價格	
11.Cabin	船艙號碼	
12.Embarked	登岸的港口	C = Cherbourg Q = Queenstown S = Southampton

All Data Sets

資料集	資料維度
A. training.csv	891 X 12
B. testing.csv	418 X 11
C. gender_submission.csv	418 X 2

Cherbourg : 瑟堡，法國西北屬重要軍港和商
Queenstown : 科芙，愛爾蘭
Southampton : 南安普敦，英國南方 ← Titanic 出航

1. PassengerId	乘客 ID	
2. Survived	是否存活	0 = No, 1 = Yes
3. Pclass	船票的等級	1 = 1st, 2 = 2nd, 3 = 3rd
4. Name	乘客姓名	
5. Sex	性別	
6. Age	年齡	
7. SibSp	兄弟姊妹/配偶數	
8. Parch	父母/小孩數	
9. Ticket	船票編號	
10.Fare	船票價格	
11.Cabin	船艙號碼	
12.Embarked	登岸的港口	C = Cherbourg Q = Queenstown S = Southampton

All Data Sets

資料集	資料維度
A. training.csv	891 X 12
B. testing.csv	418 X 11
C. gender_submission.csv	418 X 2

1. PassengerId	乘客 ID	
2. Pclass	船票的等級	1 = 1st, 2 = 2nd, 3 = 3rd
3. Name	乘客姓名	
4. Sex	性別	
5. Age	年齡	
6. SibSp	兄弟姊妹/配偶數	
7. Parch	父母/小孩數	
8. Ticket	船票編號	
9.Fare	船票價格	
10.Cabin	船艙號碼	
11.Embarked	登岸的港口	C = Cherbourg Q = Queenstown S = Southampton

All Data Sets

資料集	資料維度
A. training.csv	891 X 12
B. testing.csv	418 X 11
C. gender_submission.csv	418 X 2

最後上傳的格式

1. PassengerId 乘客 ID
2. Survived 是否存活 0 = No, 1 = Yes

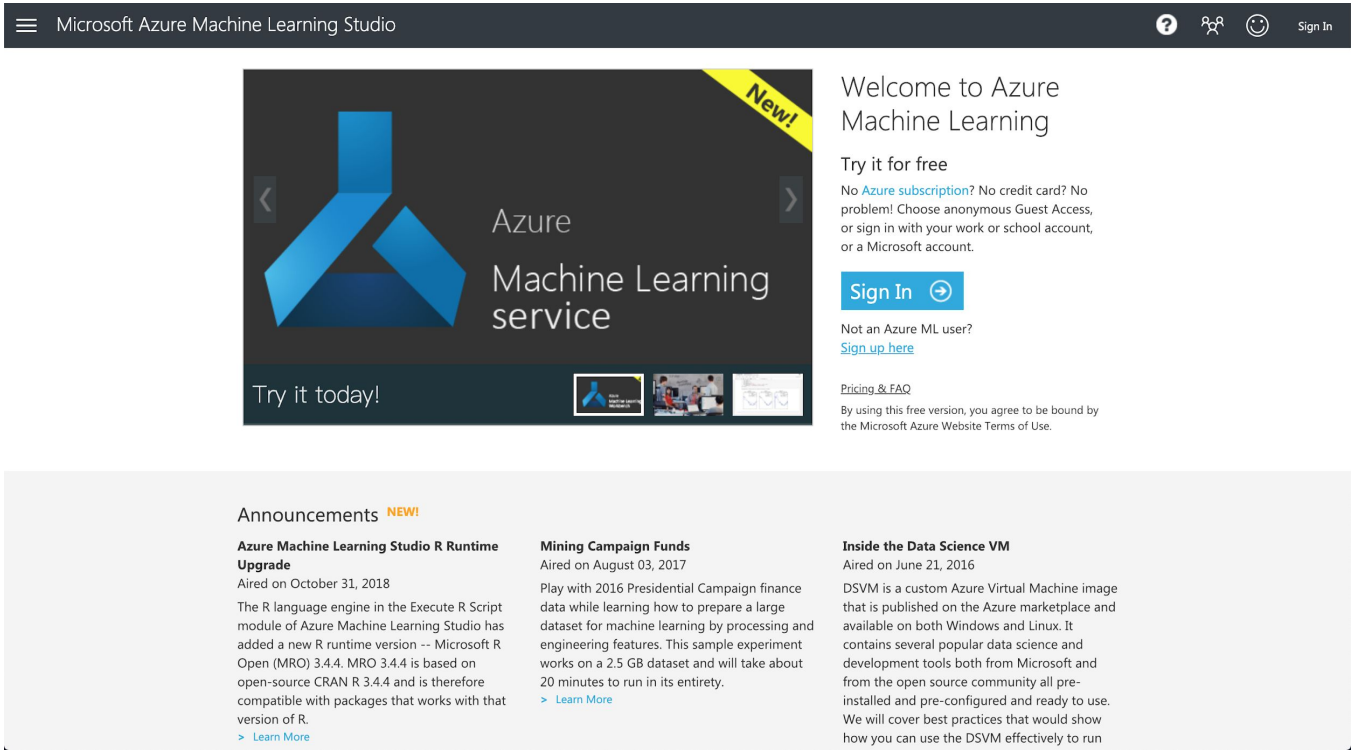
Azure ML Studio

什麼是 Microsoft Azure Machine Learning

- 不用安裝
- 不用寫程式
  - 資料清洗
  - 機器學習演算法
- 也可以支援 Python 或 R 加強運算
- 可佈署成 Web Services，分享給他人使用

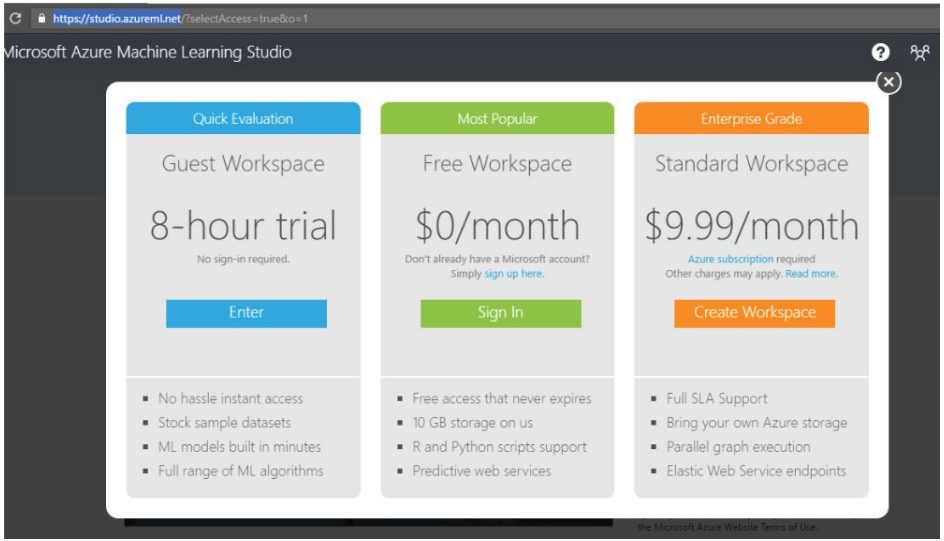
Microsoft Azure Machine Learning Studio

- Website : https://studio.azureml.net/



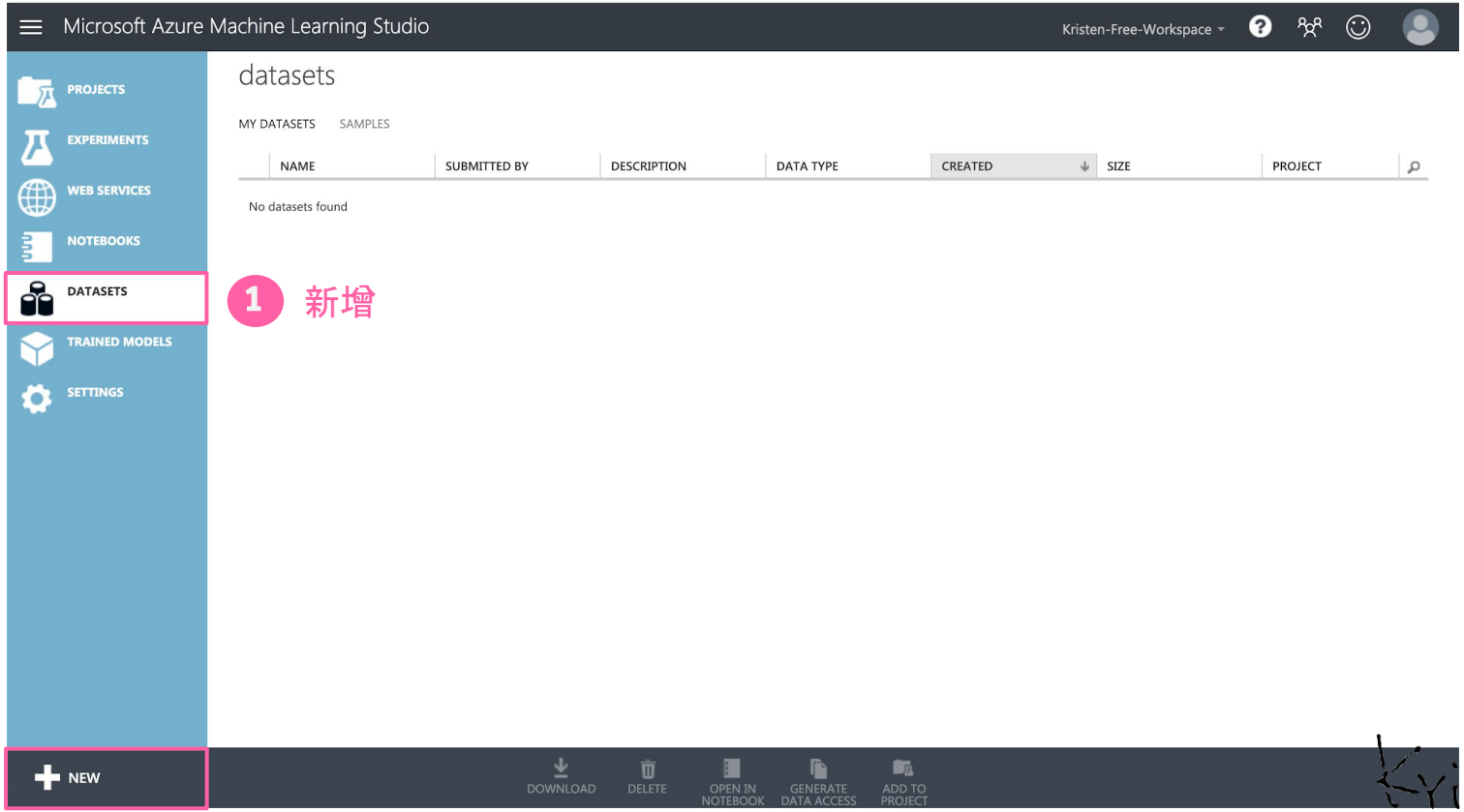
Microsoft Azure Machine Learning Studio

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

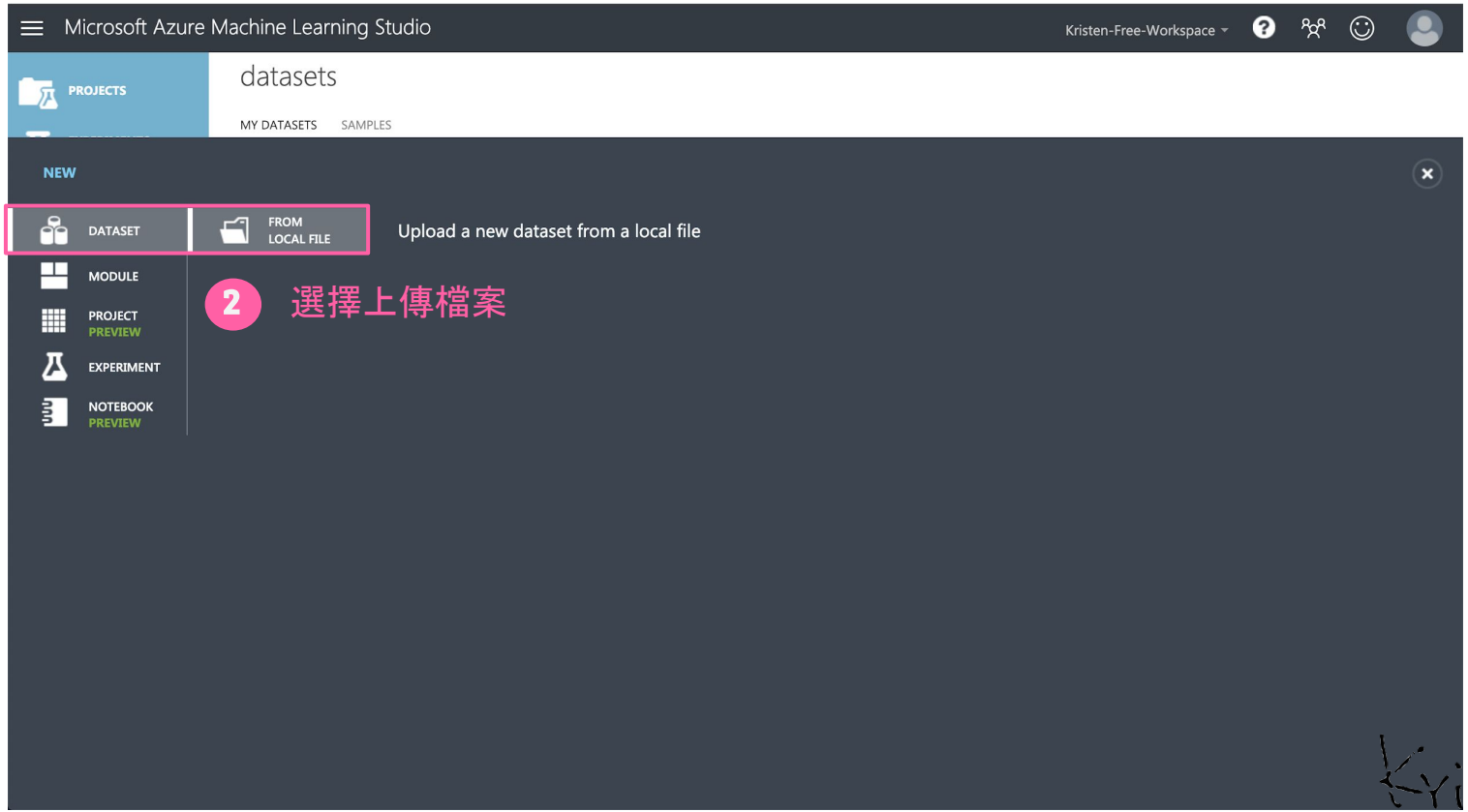


Source : https://docs.microsoft.com/zh-tw/azure/machine-learning/studio/what-is-ml-studio

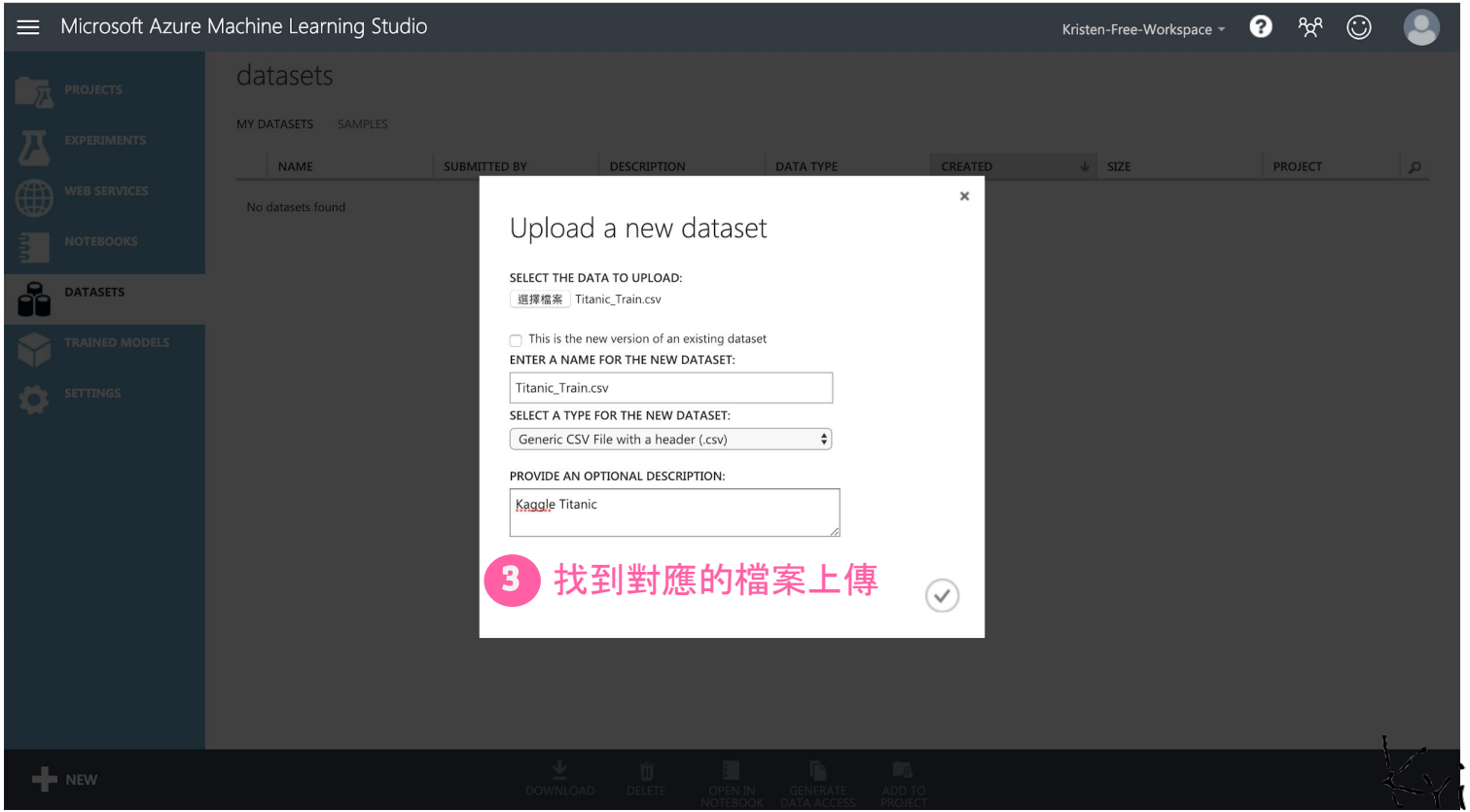
建立 Dataset


Microsoft Azure Machine Learning Studio
Kristen-Free-Workspace
PROJECTS
EXPERIMENTS
WEB SERVICES
NOTEBOOKS
DATASETS
TRAINED MODELS
SETTINGS
datasets
MY DATASETS
SAMPLES
NAME
SUBMITTED BY
DESCRIPTION
DATA TYPE
CREATED
SIZE
PROJECT
No datasets found
1
新增
NEW
DOWNLOAD
DELETE
OPEN IN NOTEBOOK
GENERATE DATA ACCESS
ADD TO PROJECT

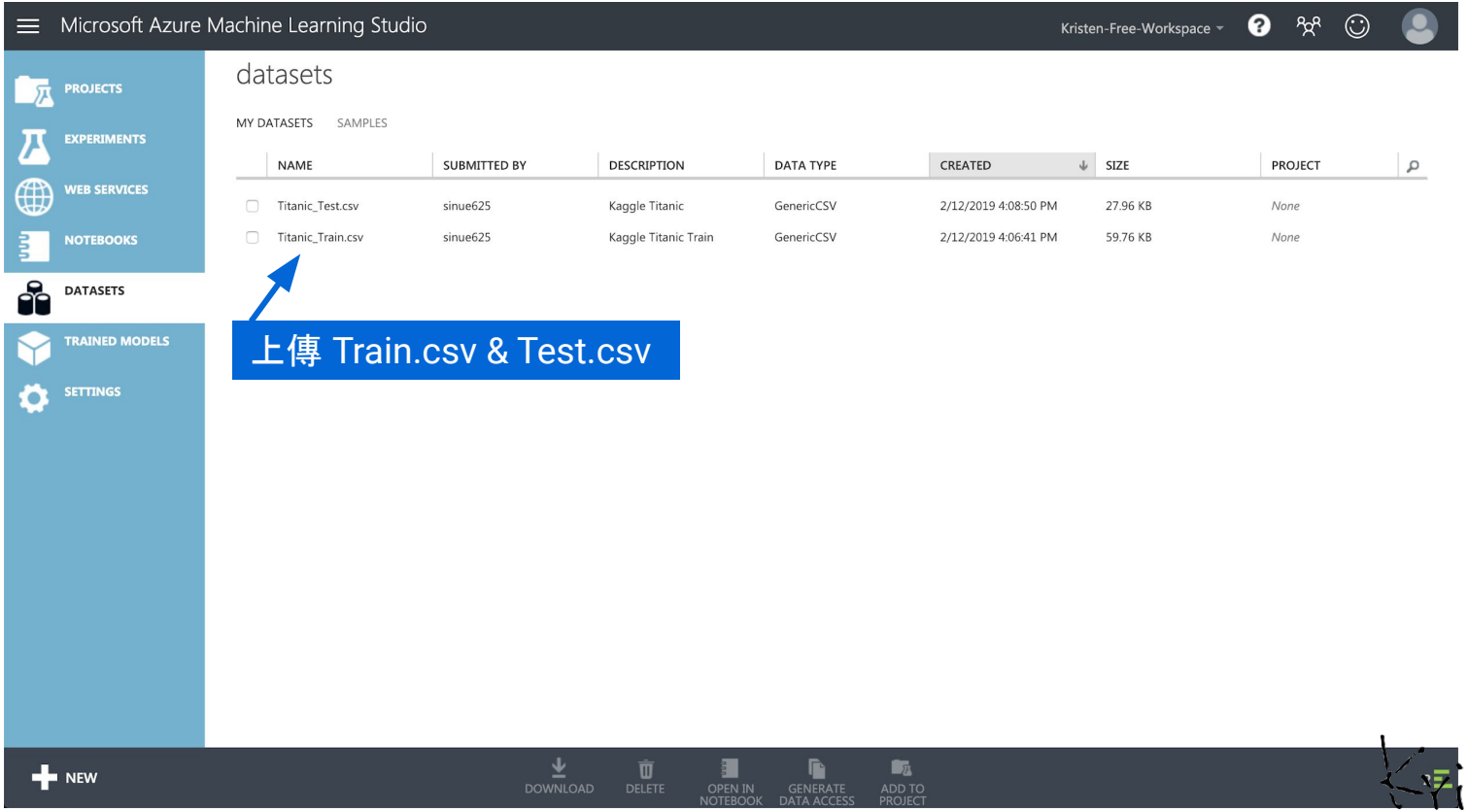
建立 Dataset


Microsoft Azure Machine Learning Studio
Kristen-Free-Workspace
PROJECTS
datasets
MY DATASETS
SAMPLES
NEW
DATASET
FROM LOCAL FILE
Upload a new dataset from a local file
MODULE
PROJECT PREVIEW
EXPERIMENT
NOTEBOOK PREVIEW
2
選擇上傳檔案

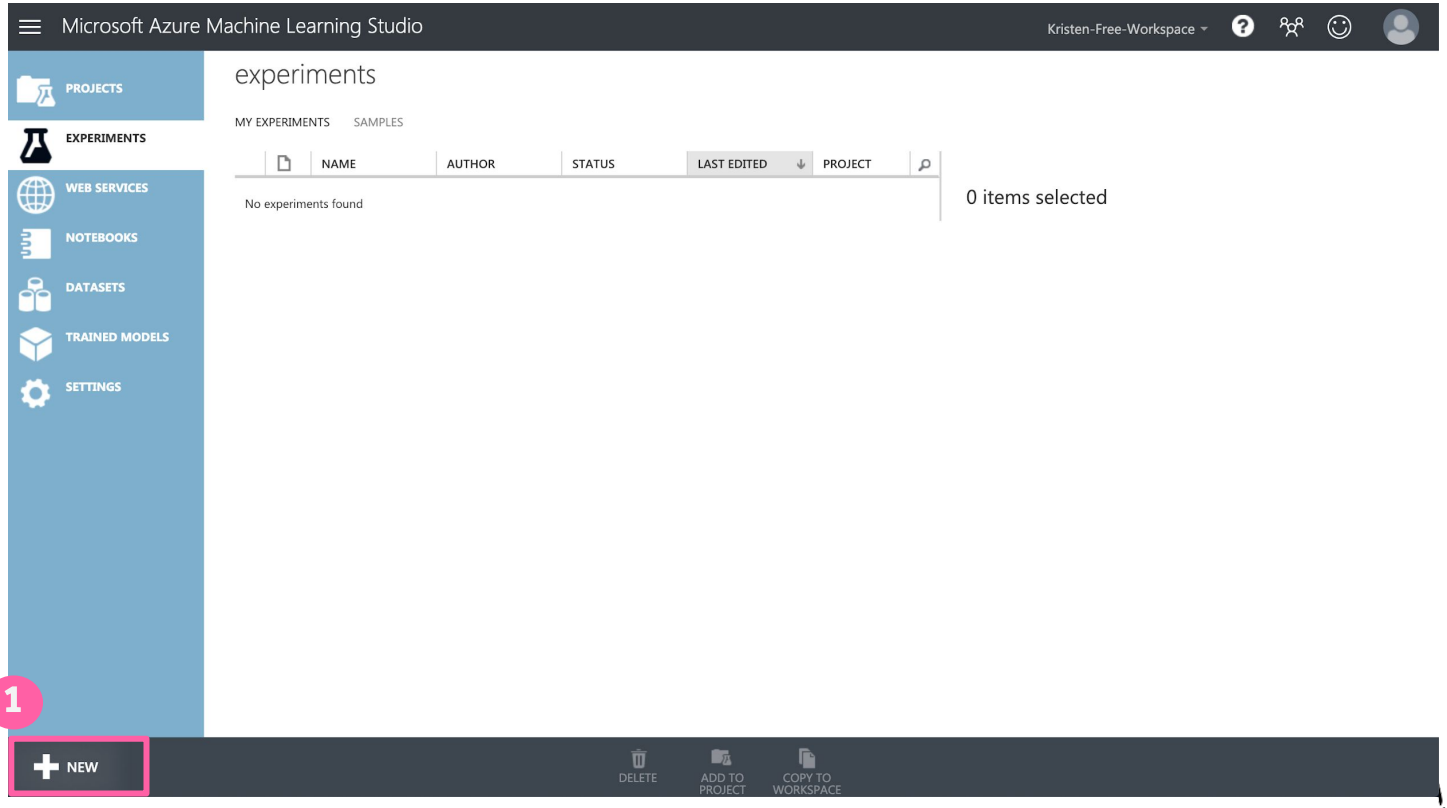
建立 Dataset


Microsoft Azure Machine Learning Studio
Kristen-Free-Workspace
PROJECTS
EXPERIMENTS
WEB SERVICES
NOTEBOOKS
DATASETS
TRAINED MODELS
SETTINGS
datasets
MY DATASETS
SAMPLES
NAME
SUBMITTED BY
DESCRIPTION
DATA TYPE
CREATED
SIZE
PROJECT
No datasets found
Upload a new dataset
SELECT THE DATA TO UPLOAD:
選擇檔案
Titanic_Train.csv
This is the new version of an existing dataset
ENTER A NAME FOR THE NEW DATASET:
Titanic_Train.csv
SELECT A TYPE FOR THE NEW DATASET:
Generic CSV File with a header (.csv)
PROVIDE AN OPTIONAL DESCRIPTION:
Kaggle Titanic
3
找到對應的檔案上傳
NEW
DOWNLOAD
DELETE
OPEN IN NOTEBOOK
GENERATE DATA ACCESS
ADD TO PROJECT

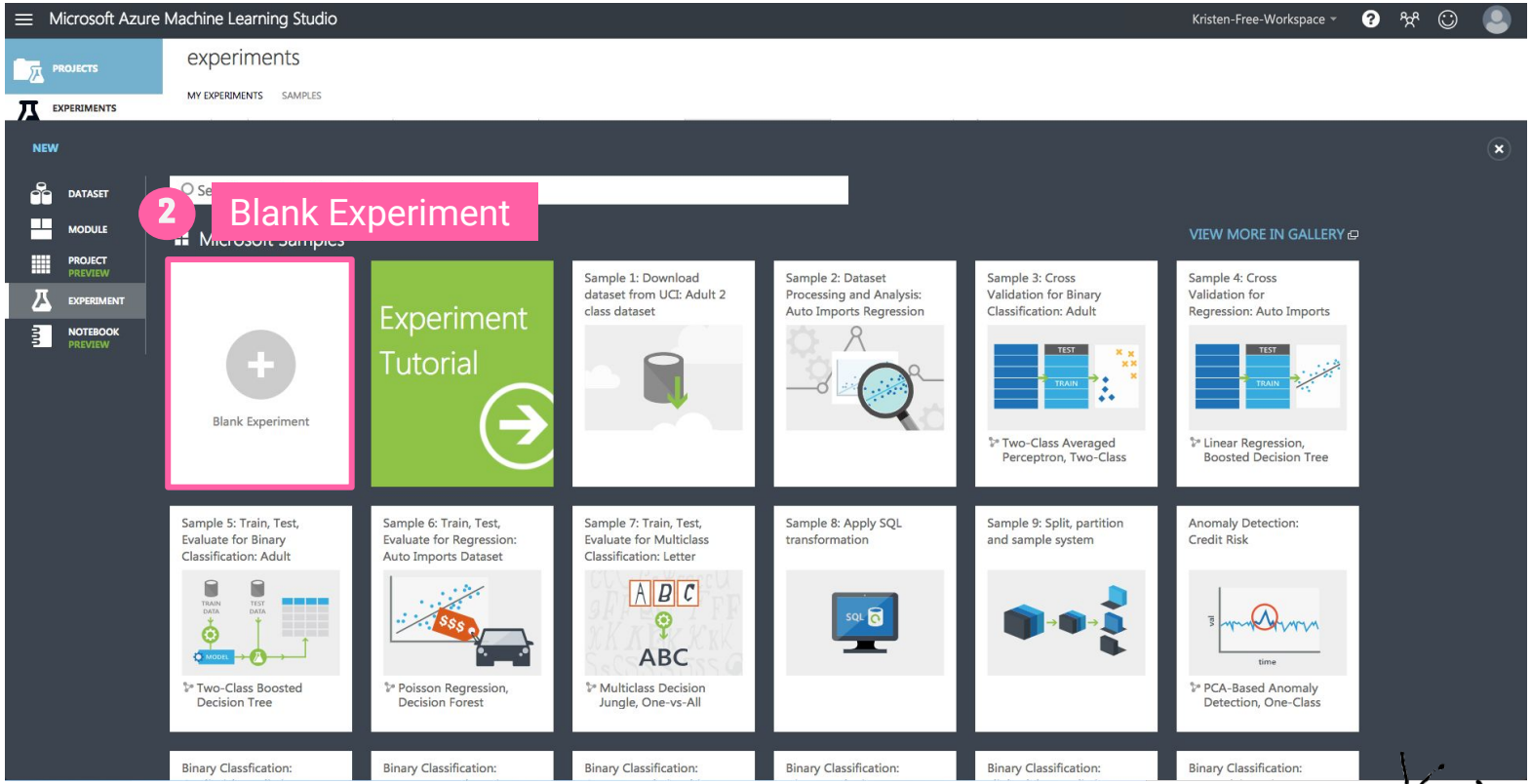
建立 Dataset


Microsoft Azure Machine Learning Studio
Kristen-Free-Workspace
PROJECTS
EXPERIMENTS
WEB SERVICES
NOTEBOOKS
DATASETS
TRAINED MODELS
SETTINGS
datasets
MY DATASETS
SAMPLES
NAME
SUBMITTED BY
DESCRIPTION
DATA TYPE
CREATED
SIZE
PROJECT
Titanic_Test.csv
sinue625
Kaggle Titanic
GenericCSV
2/12/2019 4:08:50 PM
27.96 KB
None
Titanic_Train.csv
sinue625
Kaggle Titanic Train
GenericCSV
2/12/2019 4:06:41 PM
59.76 KB
None
上傳 Train.csv & Test.csv
NEW
DOWNLOAD
DELETE
OPEN IN NOTEBOOK
GENERATE DATA ACCESS
ADD TO PROJECT

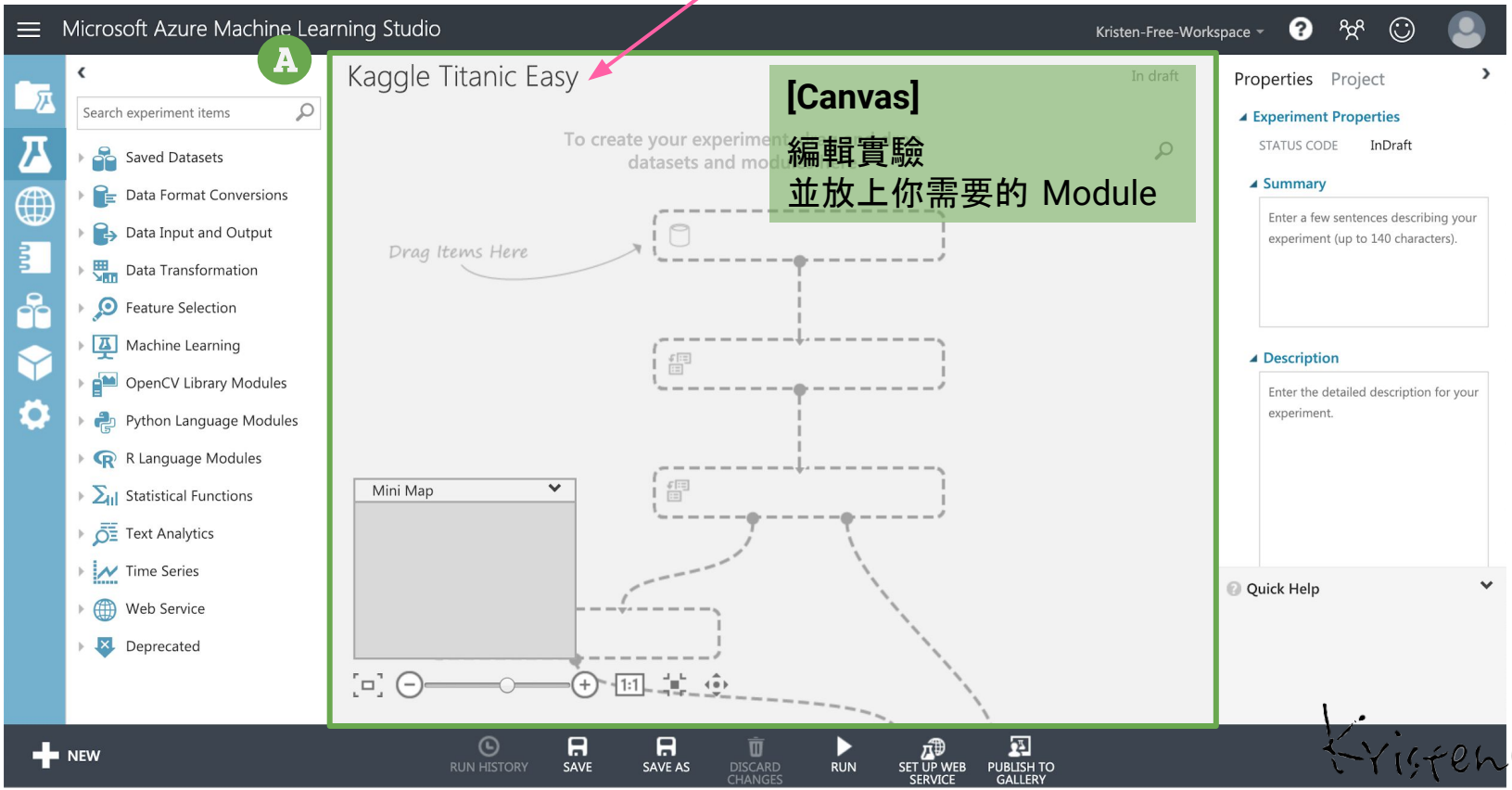
建立 Experiment



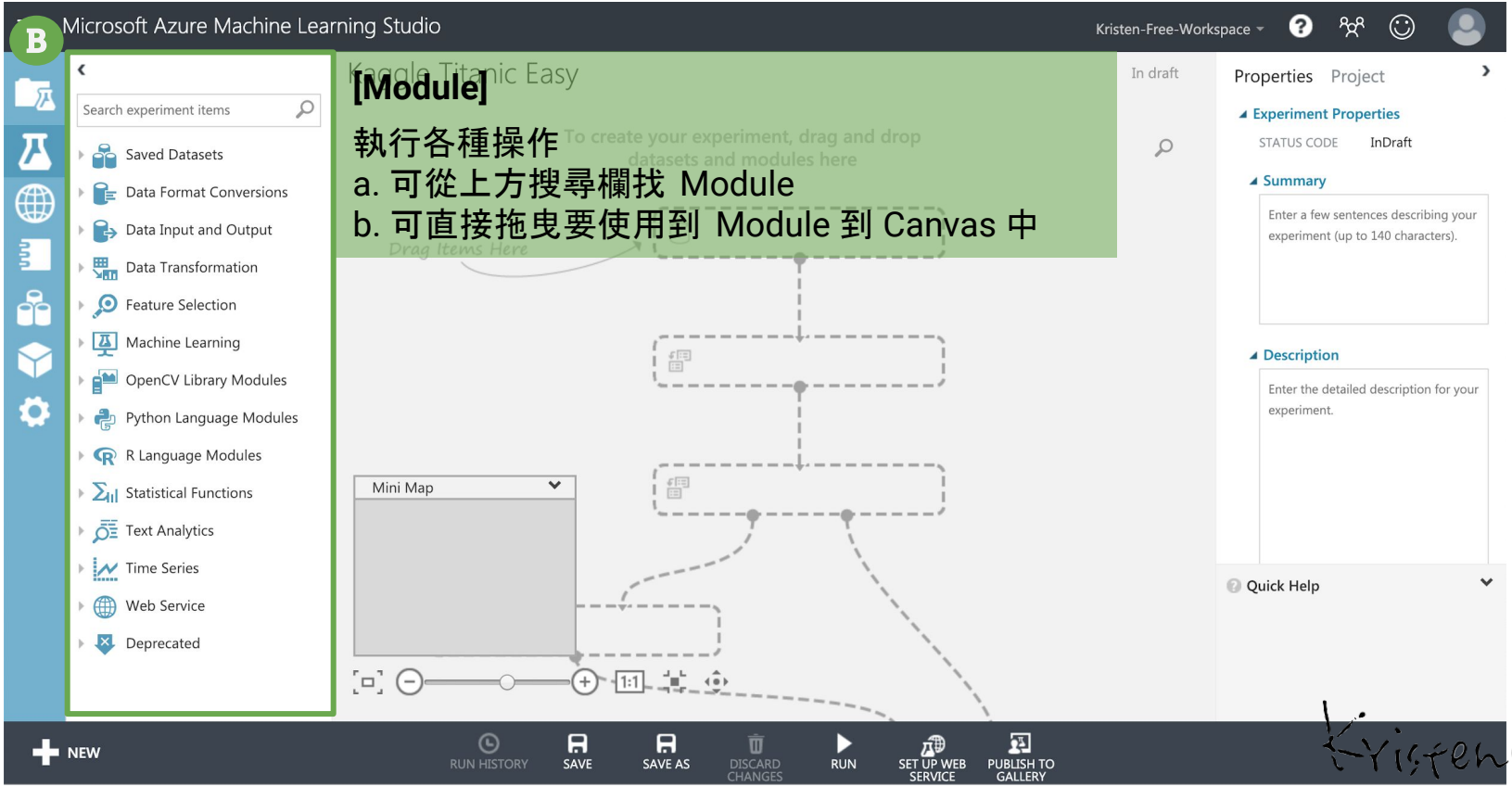
建立 Experiment



介面介紹


可以改標題名稱
Microsoft Azure Machine Learning Studio
Kristen-Free-Workspace
A
Kaggle Titanic Easy
In draft
[Canvas]
編輯實驗
並放上你需要的 Module
Search experiment items
Saved Datasets
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
Drag Items Here
Mini Map
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
NEW
RUN HISTORY
SAVE
SAVE AS
DISCARD CHANGES
RUN
SET UP WEB SERVICE
PUBLISH TO GALLERY

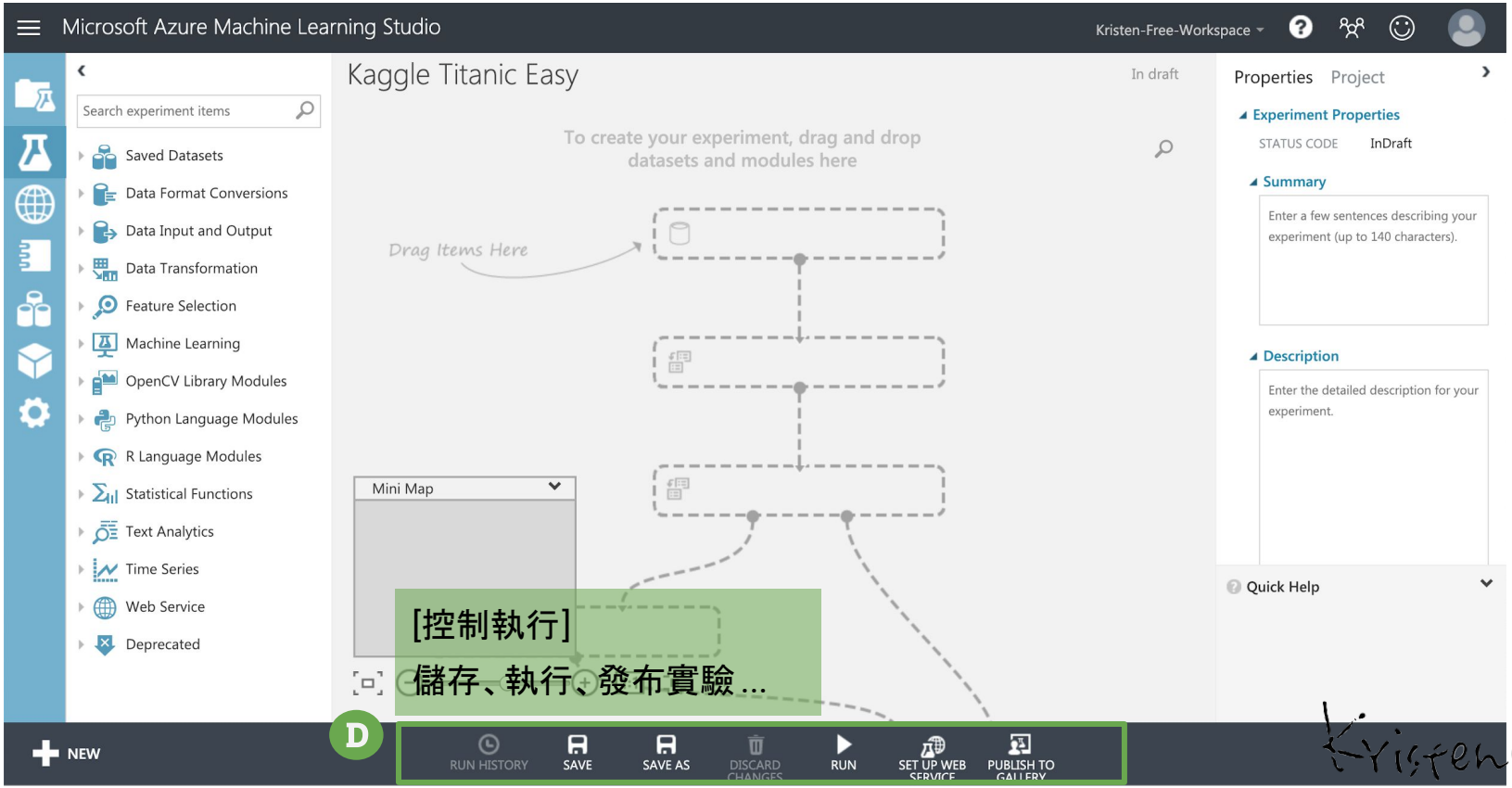
介面介紹


Microsoft Azure Machine Learning Studio
Kristen-Free-Workspace
Search experiment items
Saved Datasets
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
Kaggle Titanic Easy
[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
To create your experiment, drag and drop datasets and modules here
Drag Items Here
Mini Map
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
NEW
RUN HISTORY
SAVE
SAVE AS
DISCARD CHANGES
RUN
SET UP WEB SERVICE
PUBLISH TO GALLERY

介面介紹

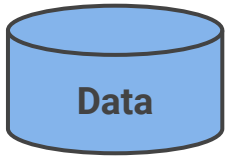

Microsoft Azure Machine Learning Studio
Kristen-Free-Wo
Kaggle Titanic Easy
In draft
[Properties]
當點選 Canvas 上的 Module 時,
Properties會出現對應的參數供調整
[Note] Quick Help
可以幫助你更深入了解個參數意義
Search experiment items
Saved Datasets
Data Format Conversions
Data Input and Output
Data Transformation
Feature Selection
Machine Learning
OpenCV Library Modules
Python Language Modules
R Language Modules
Statistical Functions
Text Analytics
Time Series
Web Service
Deprecated
Drag Items Here
Mini Map
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
NEW
RUN HISTORY
SAVE
SAVE AS
DISCARD CHANGES
RUN
SET UP WEB SERVICE
PUBLISH TO GALLERY

介面介紹

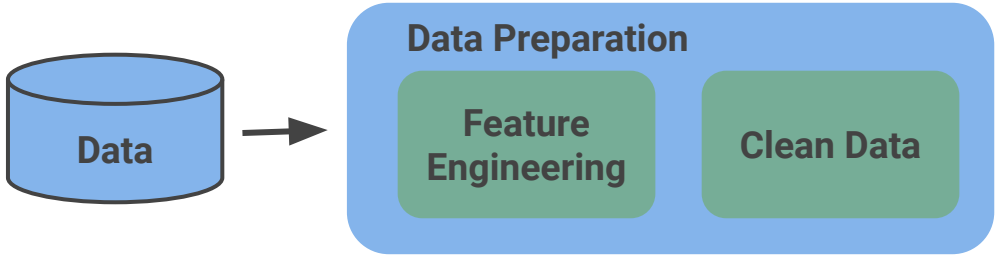

Microsoft Azure Machine Learning Studio
Kristen-Free-Workspace
Kaggle Titanic Easy
In draft
Search experiment items
Saved Datasets
Data Format Conversions
Data Input and Output
Data Transformation
Feature Selection
Machine Learning
OpenCV Library Modules
Python Language Modules
R Language Modules
Statistical Functions
Text Analytics
Time Series
Web Service
Deprecated
To create your experiment, drag and drop datasets and modules here
Drag Items Here
Mini Map
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
[控制執行]
儲存、執行、發布實驗...
D
NEW
RUN HISTORY
SAVE
SAVE AS
DISCARD CHANGES
RUN
SET UP WEB SERVICE
PUBLISH TO GALLERY

Analyze data with Azure ML

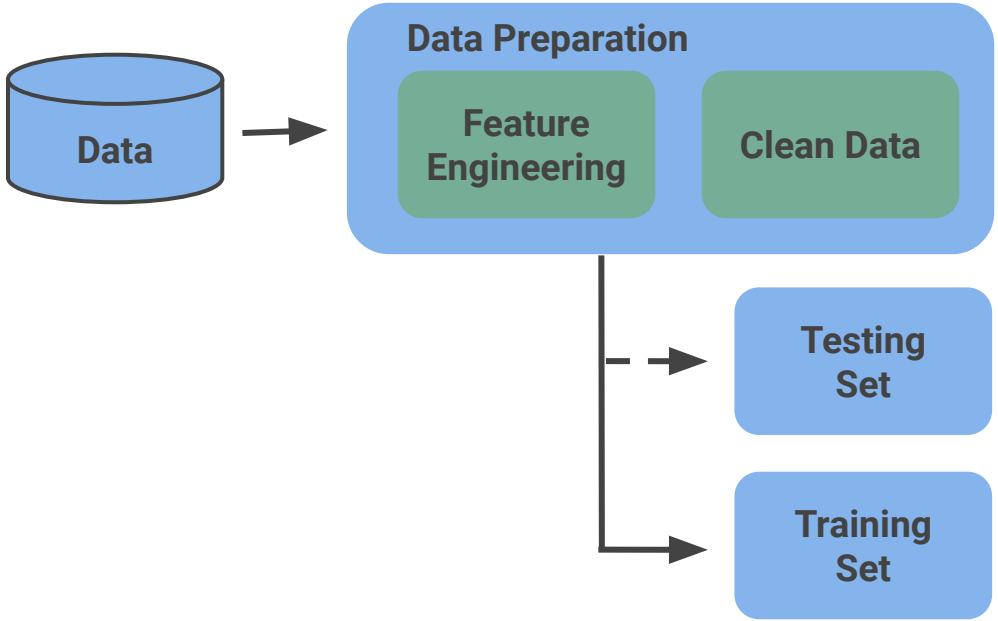
Data processing pipeline



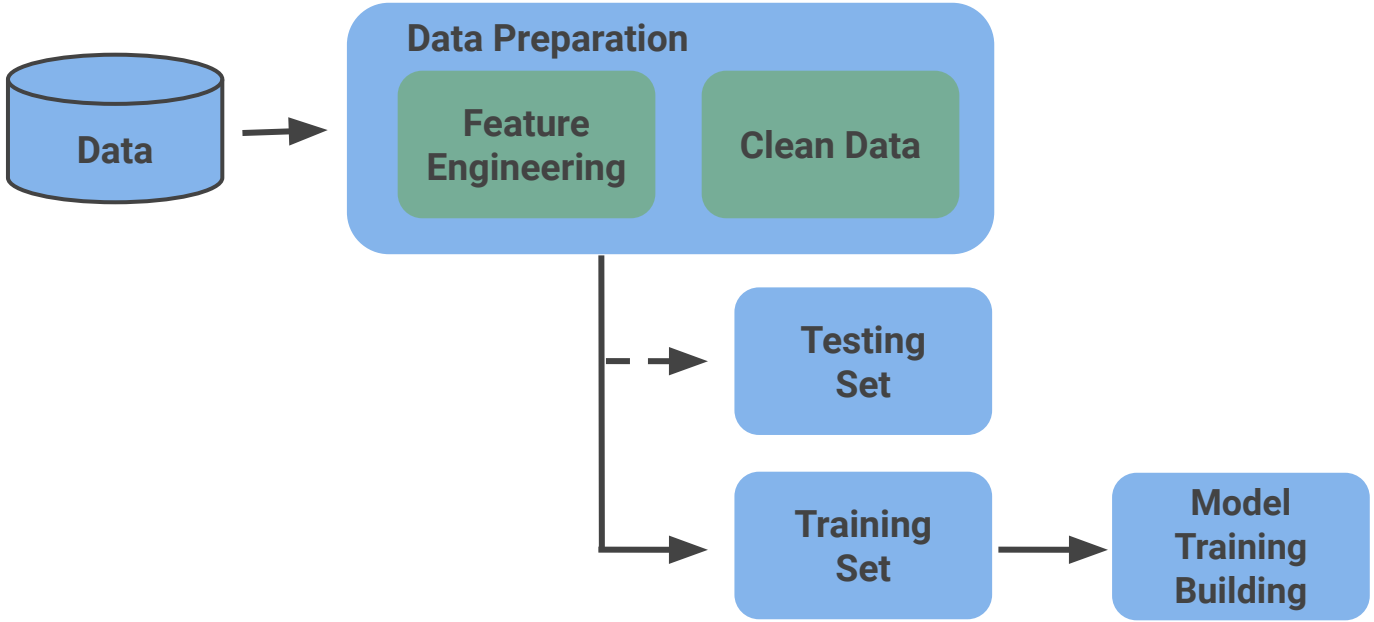
Data processing pipeline



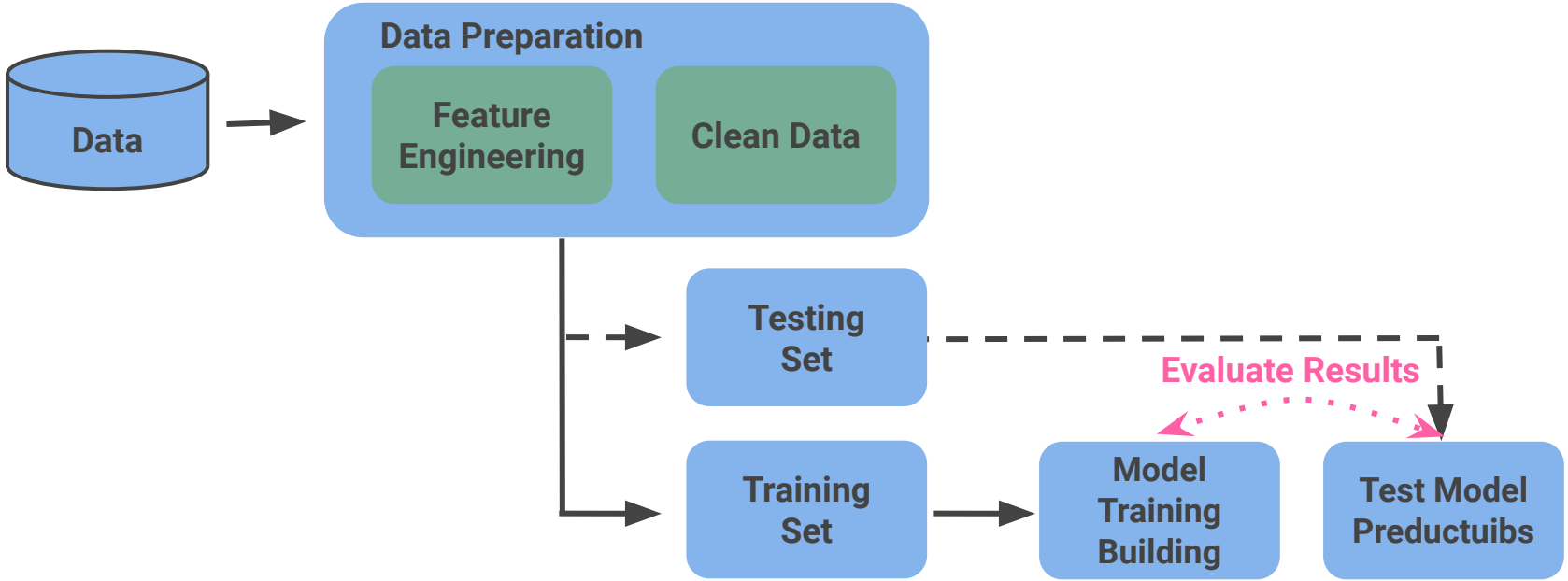
Data processing pipeline



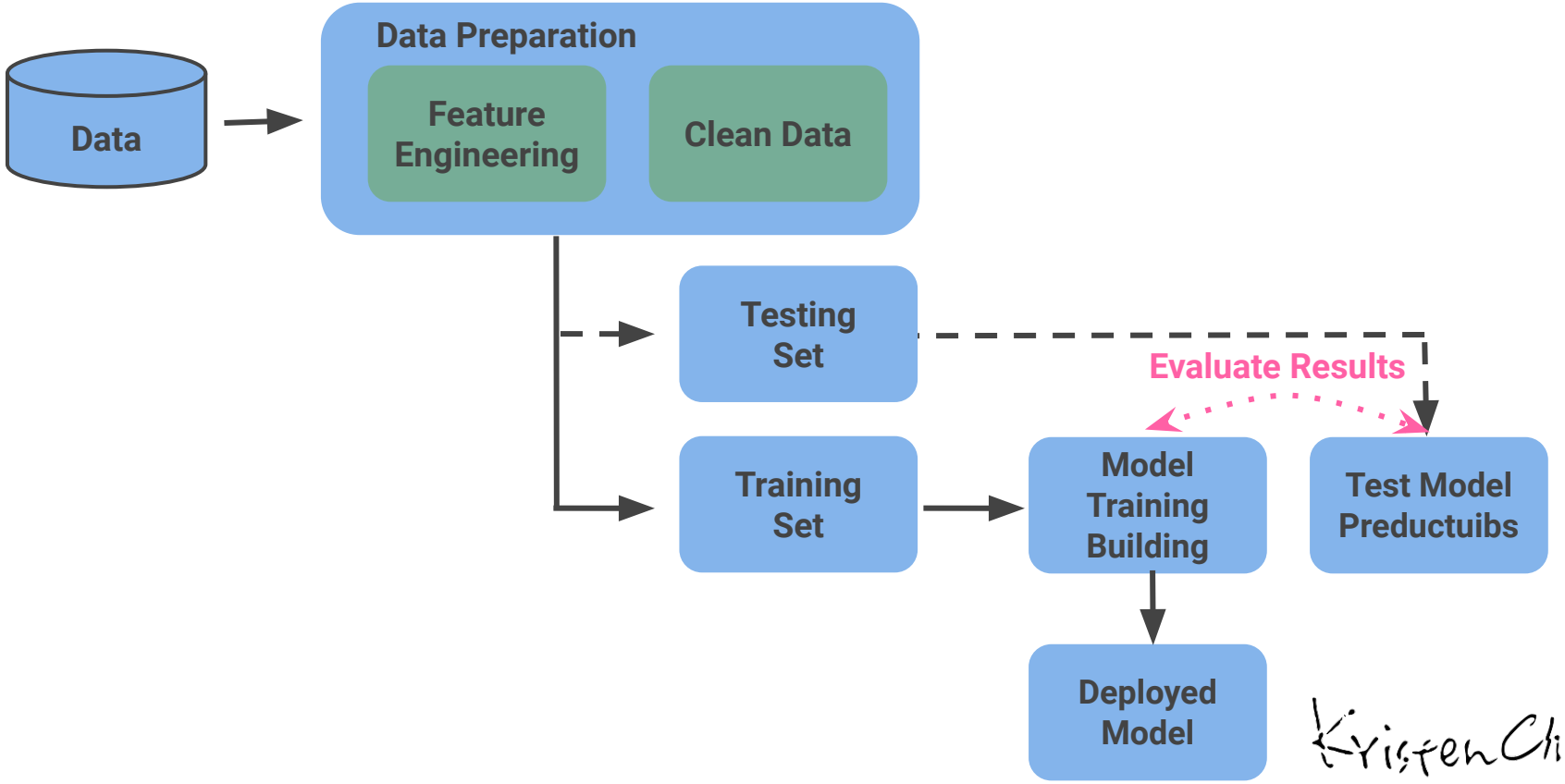
Data processing pipeline



Data processing pipeline



Data processing pipeline



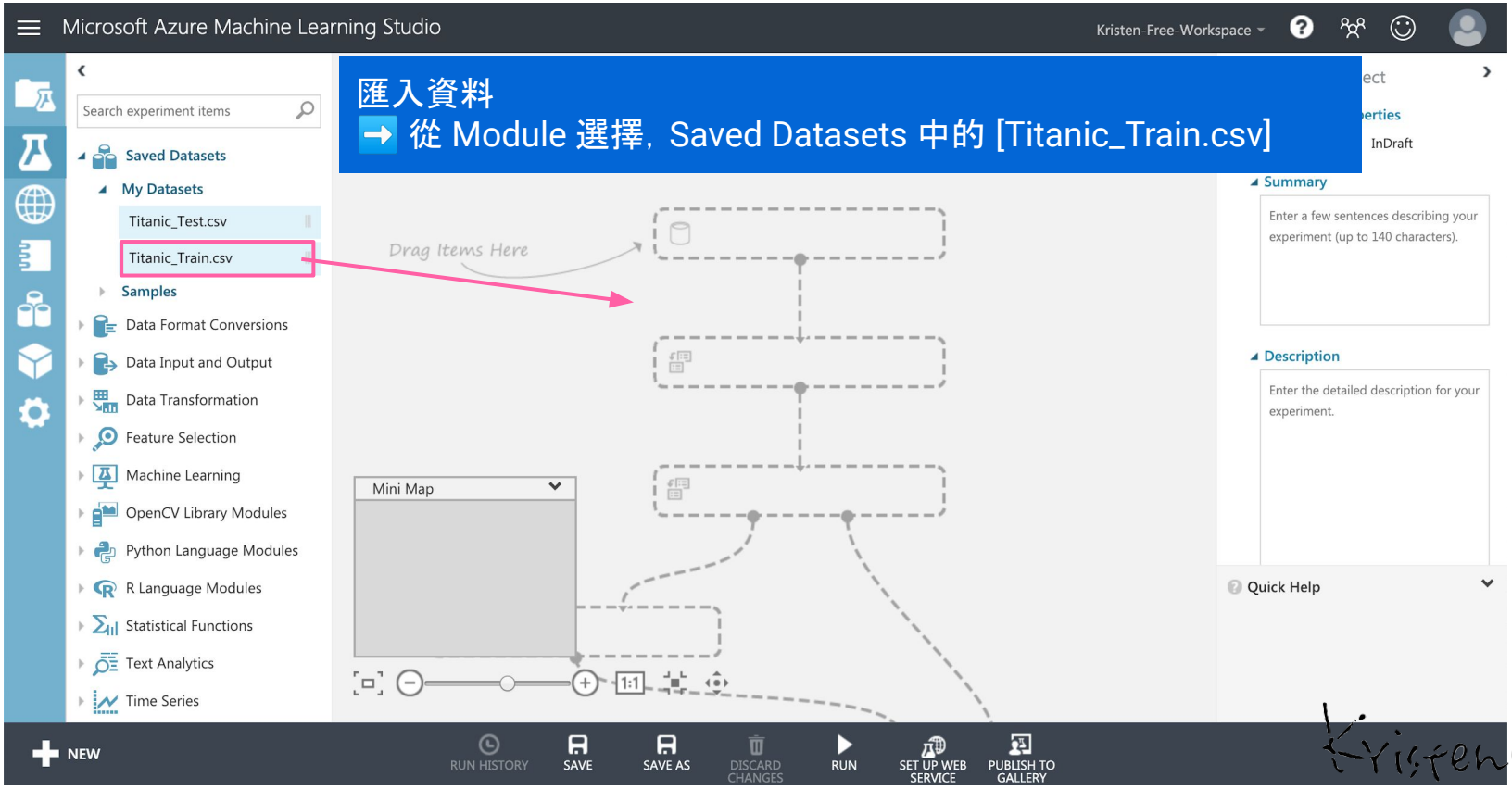
All Data Sets

資料集	資料維度
A. training.csv	891 X 12
B. testing.csv	418 X 11
C. gender_submission.csv	418 X 2

剛剛Kaggle 的資料集

1. PassengerId	乘客 ID	
2. Survived	是否存活	0 = No, 1 = Yes
3. Pclass	船票的等級	1 = 1st, 2 = 2nd, 3 = 3rd
4. Name	乘客姓名	
5. Sex	性別	
6. Age	年齡	
7. SibSp	兄弟姊妹/配偶數	
8. Parch	父母/小孩數	
9. Ticket	船票編號	
10.Fare	船票價格	
11.Cabin	船艙號碼	
12.Embarked	登岸的港口	C = Cherbourg Q = Queenstown S = Southampton

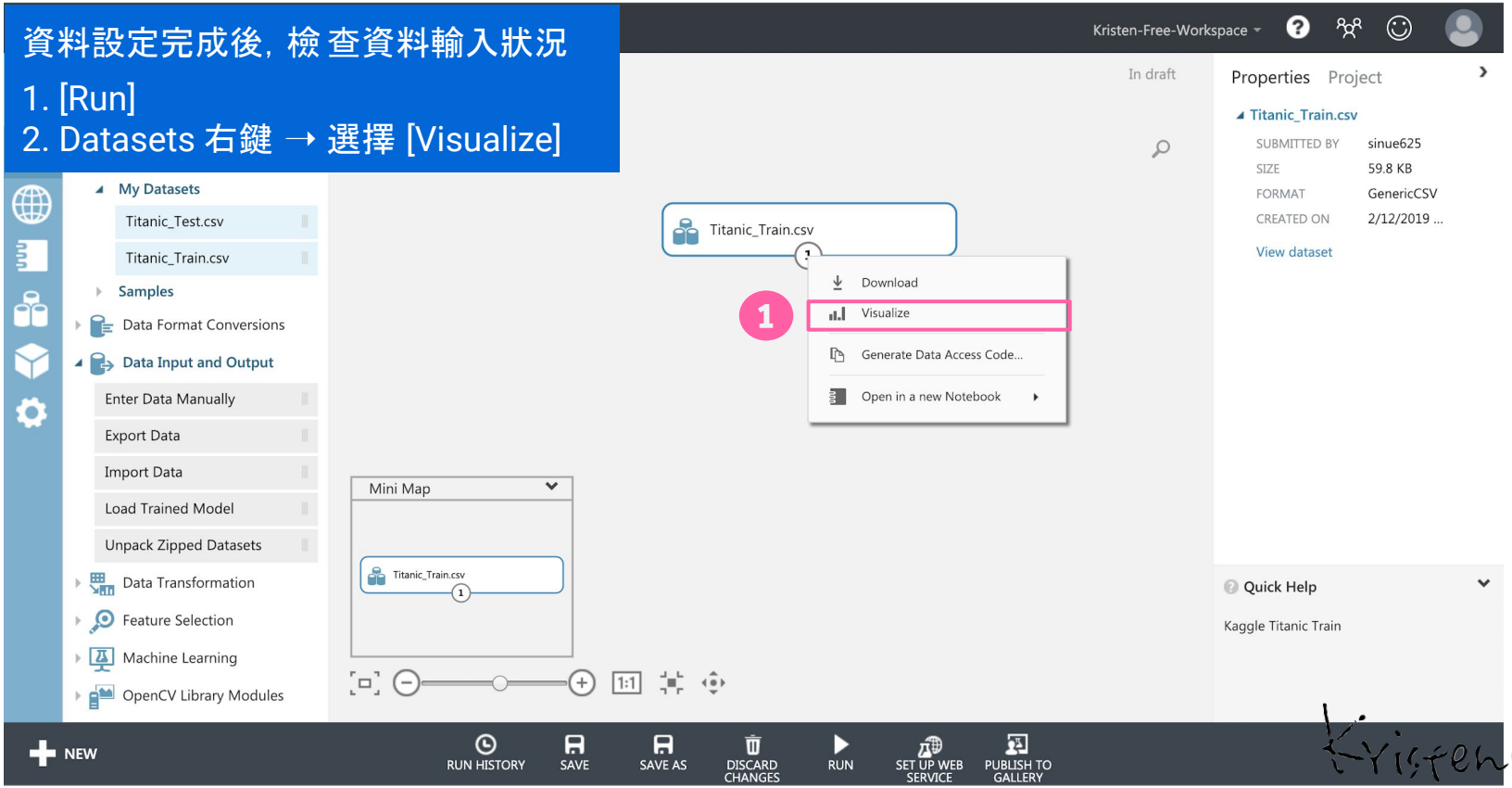
[Method 1] Access Data



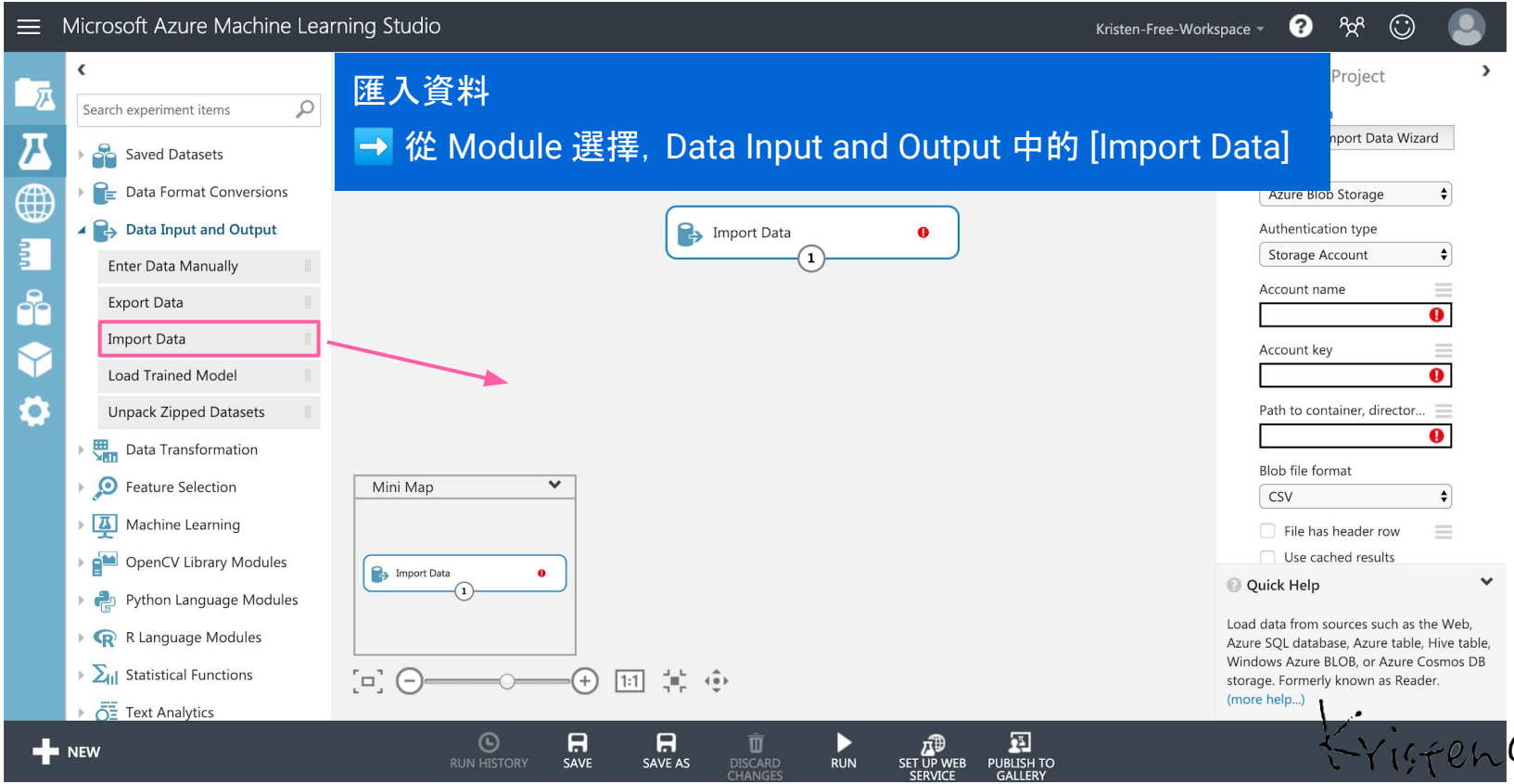
[Method 1] Access Data

資料設定完成後，檢查資料輸入狀況

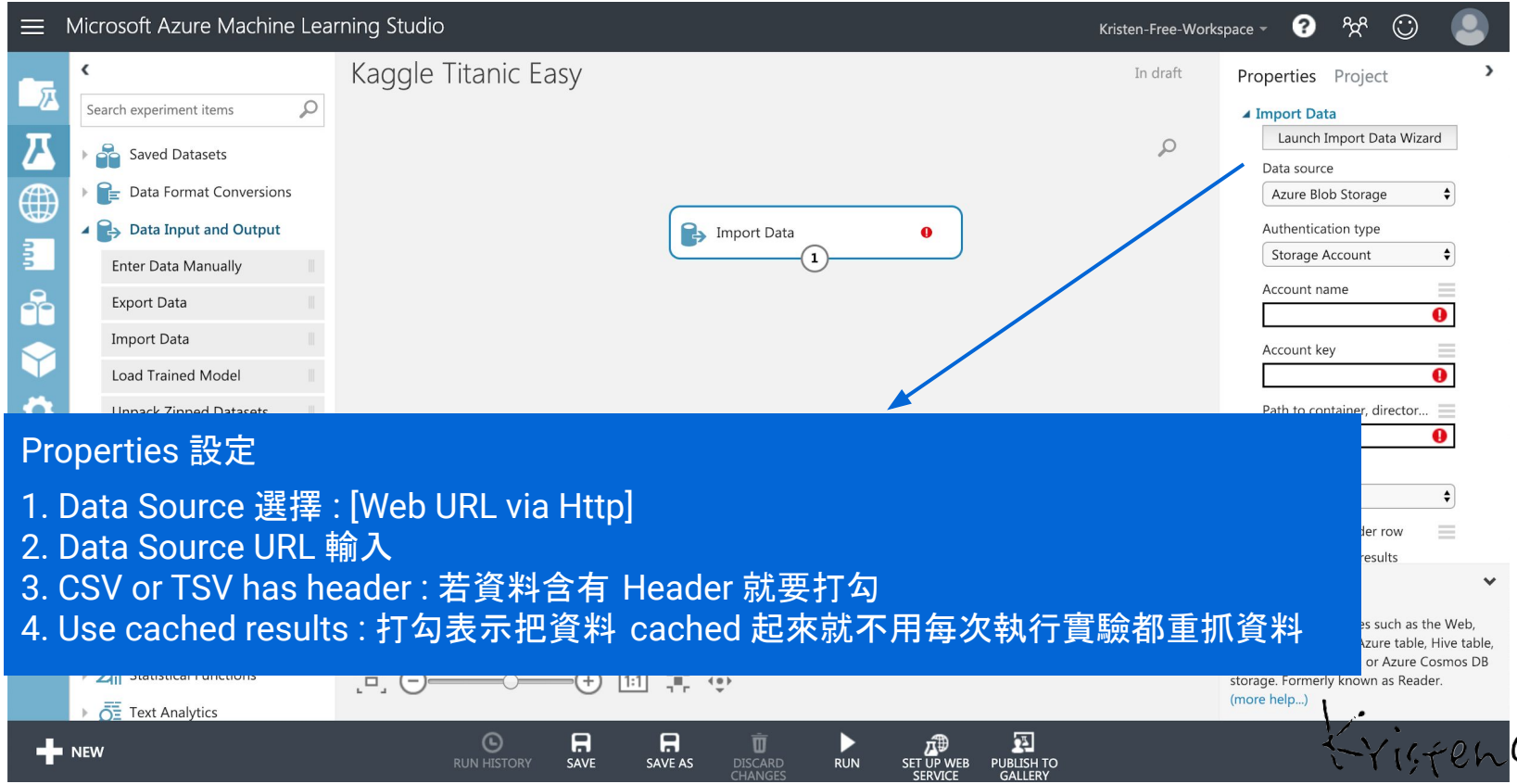
1. [Run]
2. Datasets 右鍵 → 選擇 [Visualize]



[Method 2] Access Data



[Method 2] Access Data

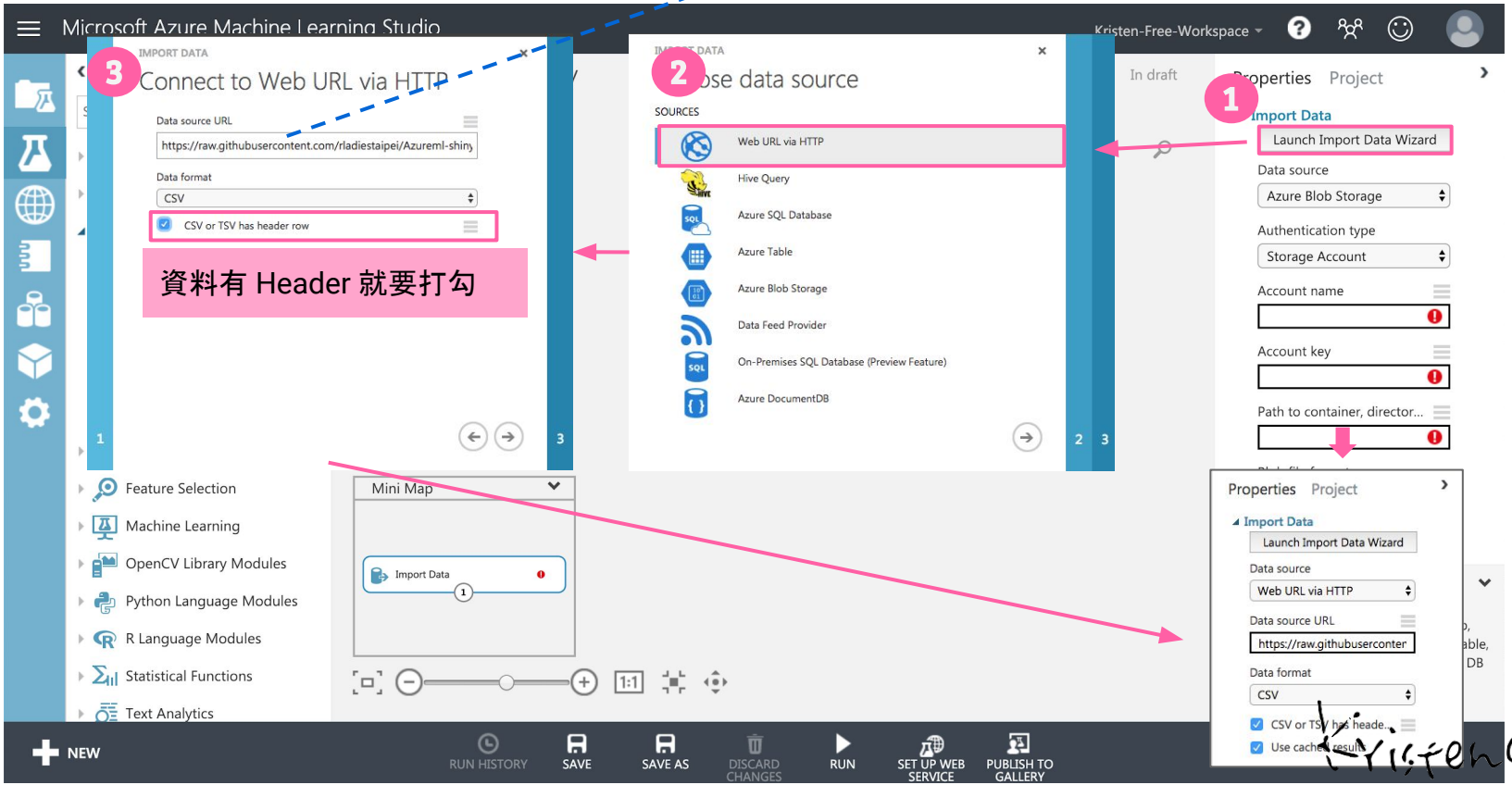


Properties 設定

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

https://raw.githubusercontent.com/kristenchan/Sharing/master/Kaggle_Titanic/Titanic_Train.csv

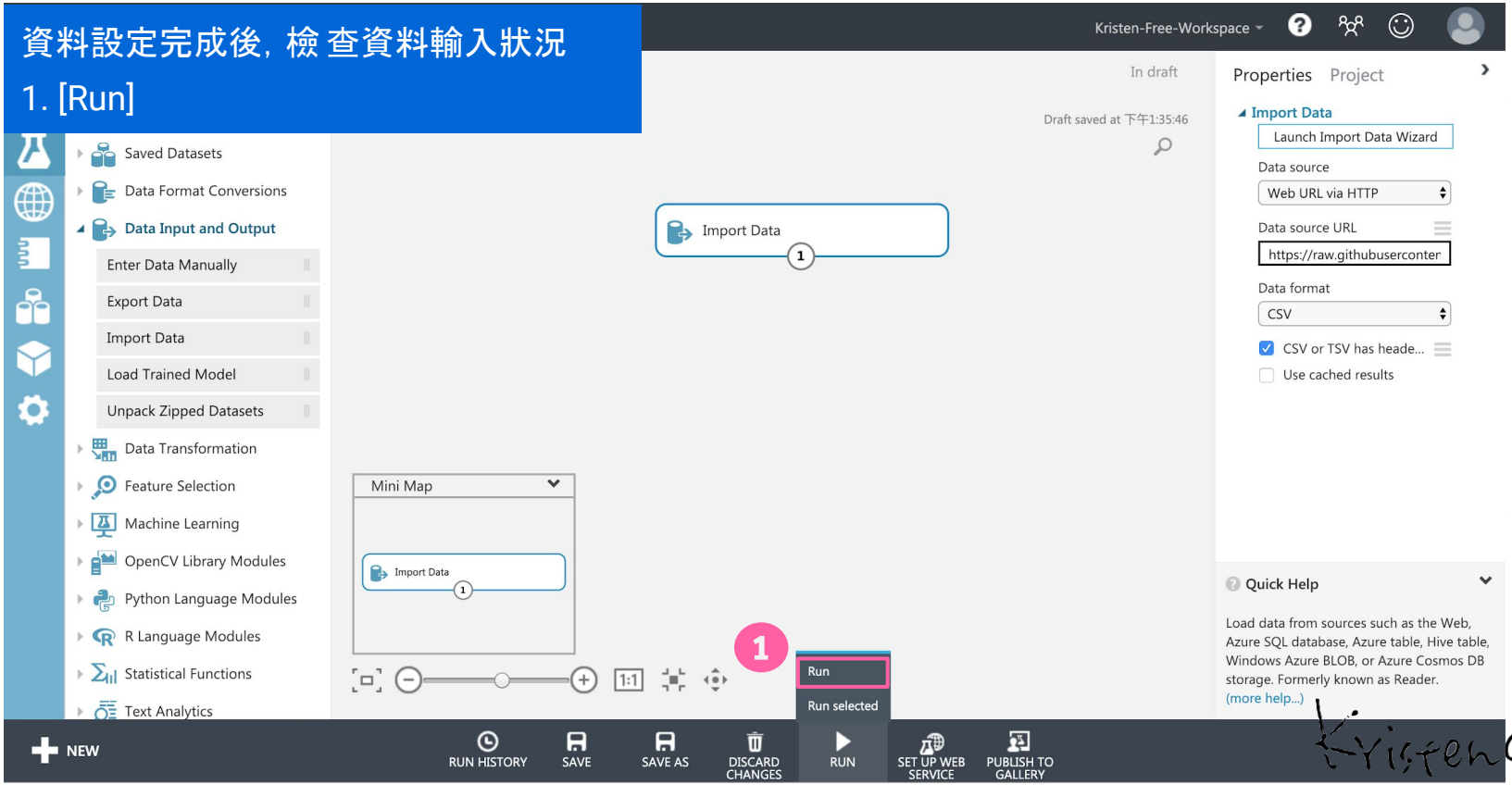
[Method 2] Access Data



[Method 2] Access Data

資料設定完成後，檢查資料輸入狀況

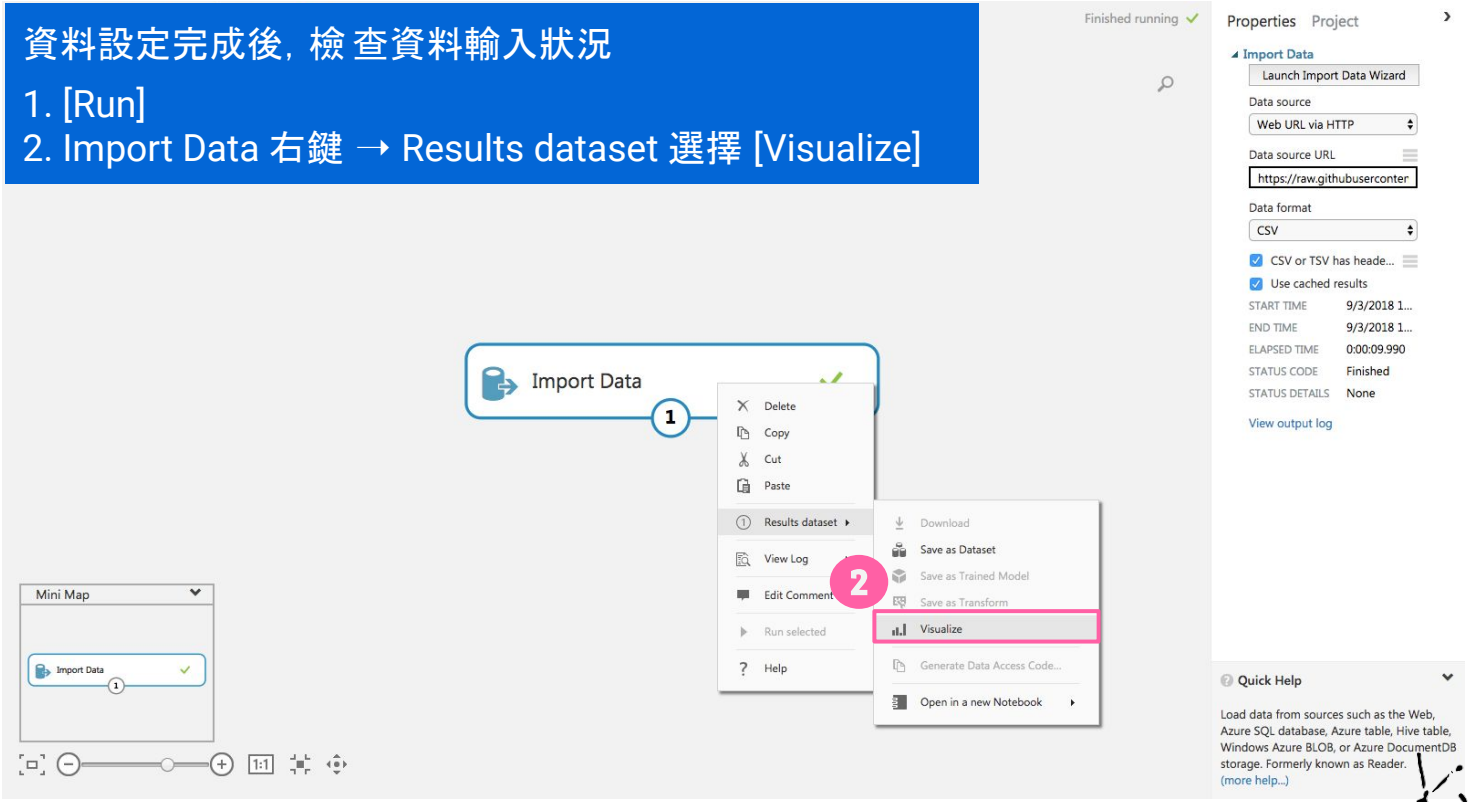
1. [Run]



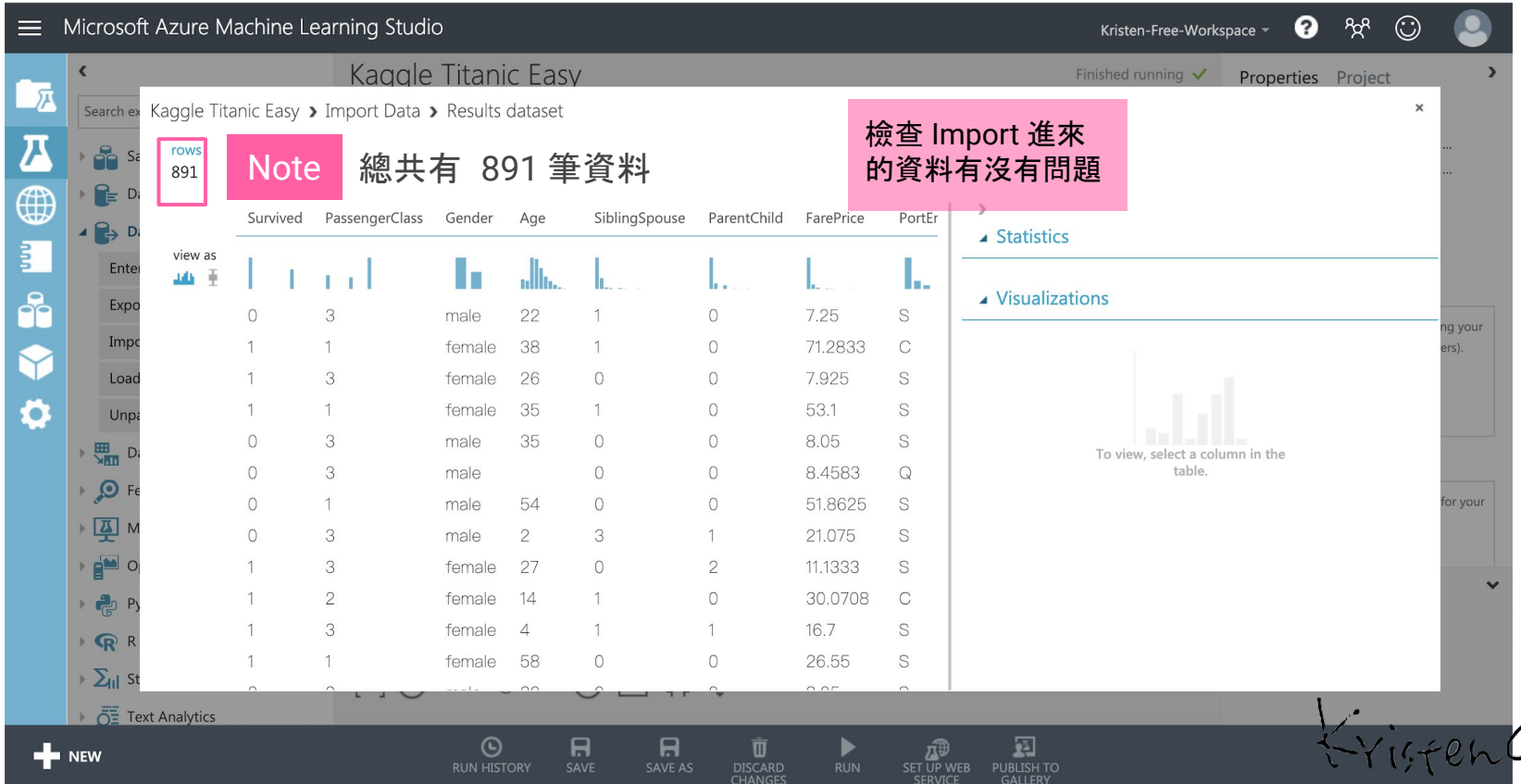
[Method 2] Access Data

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]



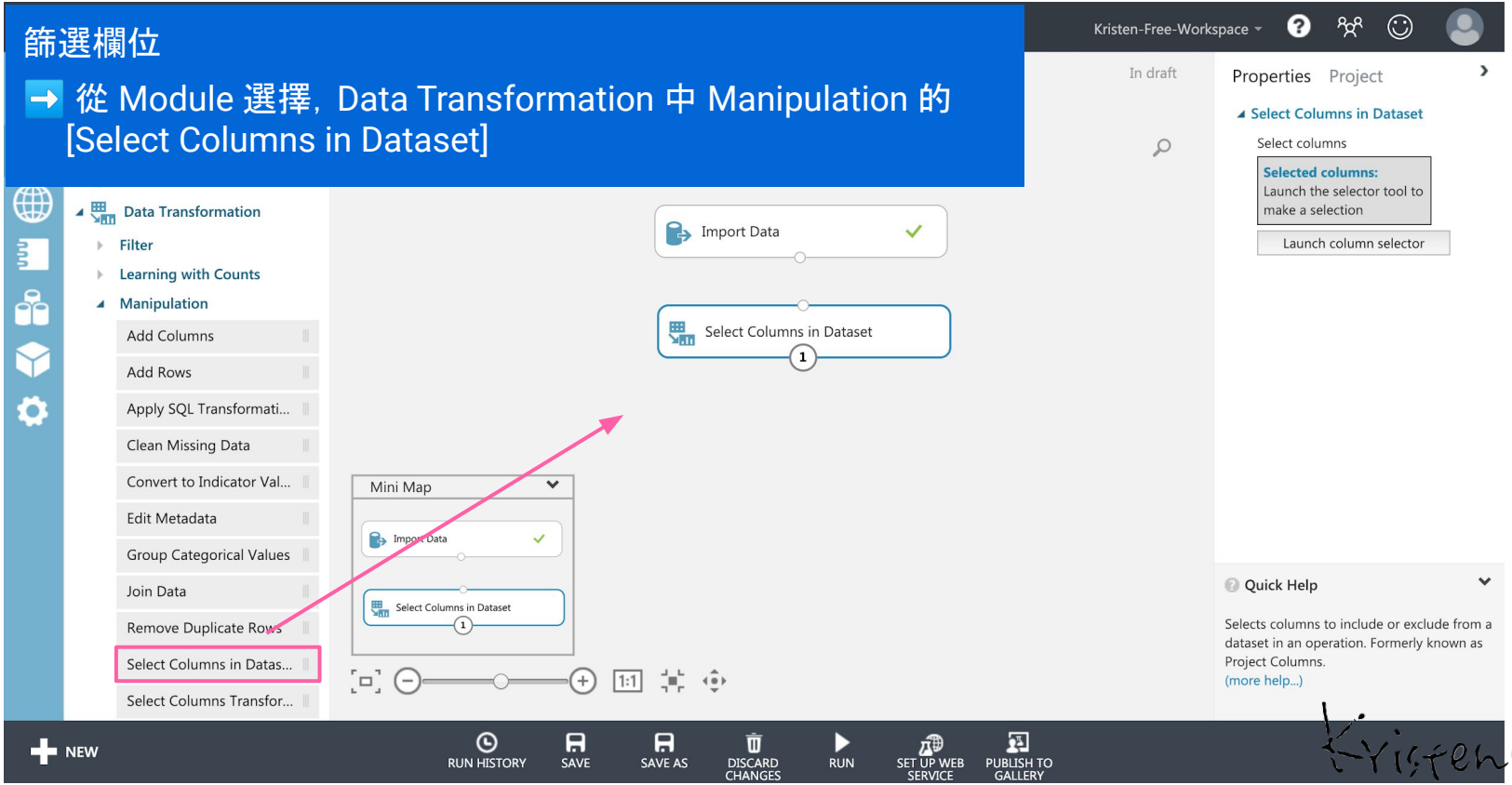
Access Data


Microsoft Azure Machine Learning Studio
Kristen-Free-Workspace
Kaggle Titanic Easy
Finished running
Properties
Project
Kaggle Titanic Easy › Import Data › Results dataset
rows
891
Note
總共有 891 筆資料
檢查 Import 進來的資料有沒有問題
Survived PassengerClass Gender Age SiblingSpouse ParentChild FarePrice PortEr
view as
0 3 male 22 1 0 7.25 S
1 1 female 38 1 0 71.2833 C
1 3 female 26 0 0 7.925 S
1 1 female 35 1 0 53.1 S
0 3 male 35 0 0 8.05 S
0 3 male 0 0 8.4583 Q
0 1 male 54 0 0 51.8625 S
0 3 male 2 3 1 21.075 S
1 3 female 27 0 2 11.1333 S
1 2 female 14 1 0 30.0708 C
1 3 female 4 1 1 16.7 S
1 1 female 58 0 0 26.55 S
Statistics
Visualizations
To view, select a column in the table.
Text Analytics
NEW
RUN HISTORY
SAVE
SAVE AS
DISCARD CHANGES
RUN
SET UP WEB SERVICE
PUBLISH TO GALLERY

篩選資料

篩選欄位

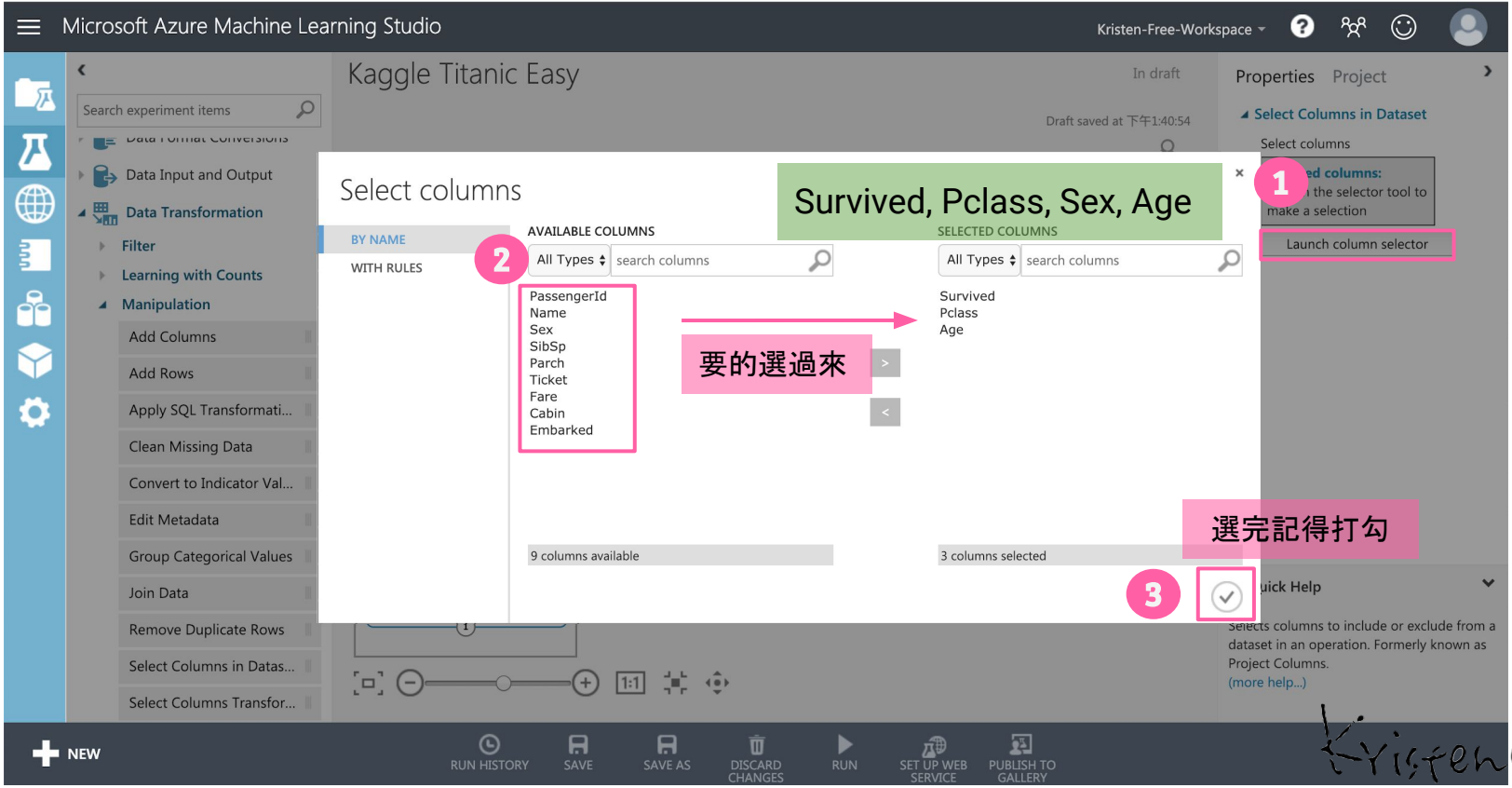
➡ 從 Module 選擇，Data Transformation 中 Manipulation 的 [Select Columns in Dataset]



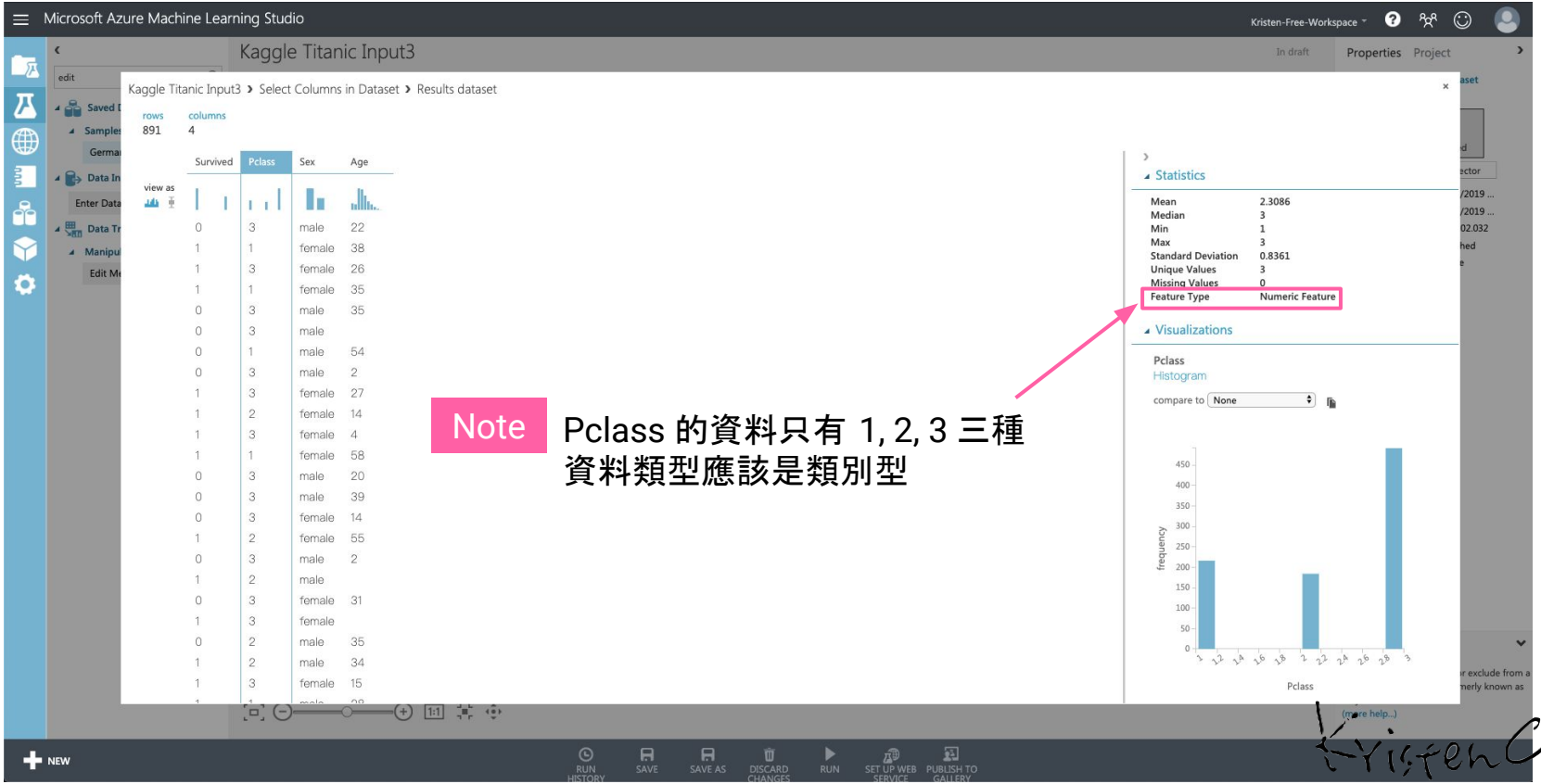
篩選資料


Microsoft Azure Machine Learning Studio
Kristen-Free-Workspace
Kaggle Titanic Easy
In draft
Draft saved at 下午1:39:45
Search experiment items
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...
Import Data
Select Columns in Dataset
1
按著連接點拖到連接處
Note
因為還沒設定 Properties
Mini Map
Properties
Project
Select Columns in Dataset
Select columns
Selected columns:
Launch the selector tool to make a selection
Launch column selector
Quick Help
Selects columns to include or exclude from a dataset in an operation. Formerly known as Project Columns.
(more help...)
NEW
RUN HISTORY
SAVE
SAVE AS
DISCARD CHANGES
RUN
SET UP WEB SERVICE
PUBLISH TO GALLERY

篩選資料

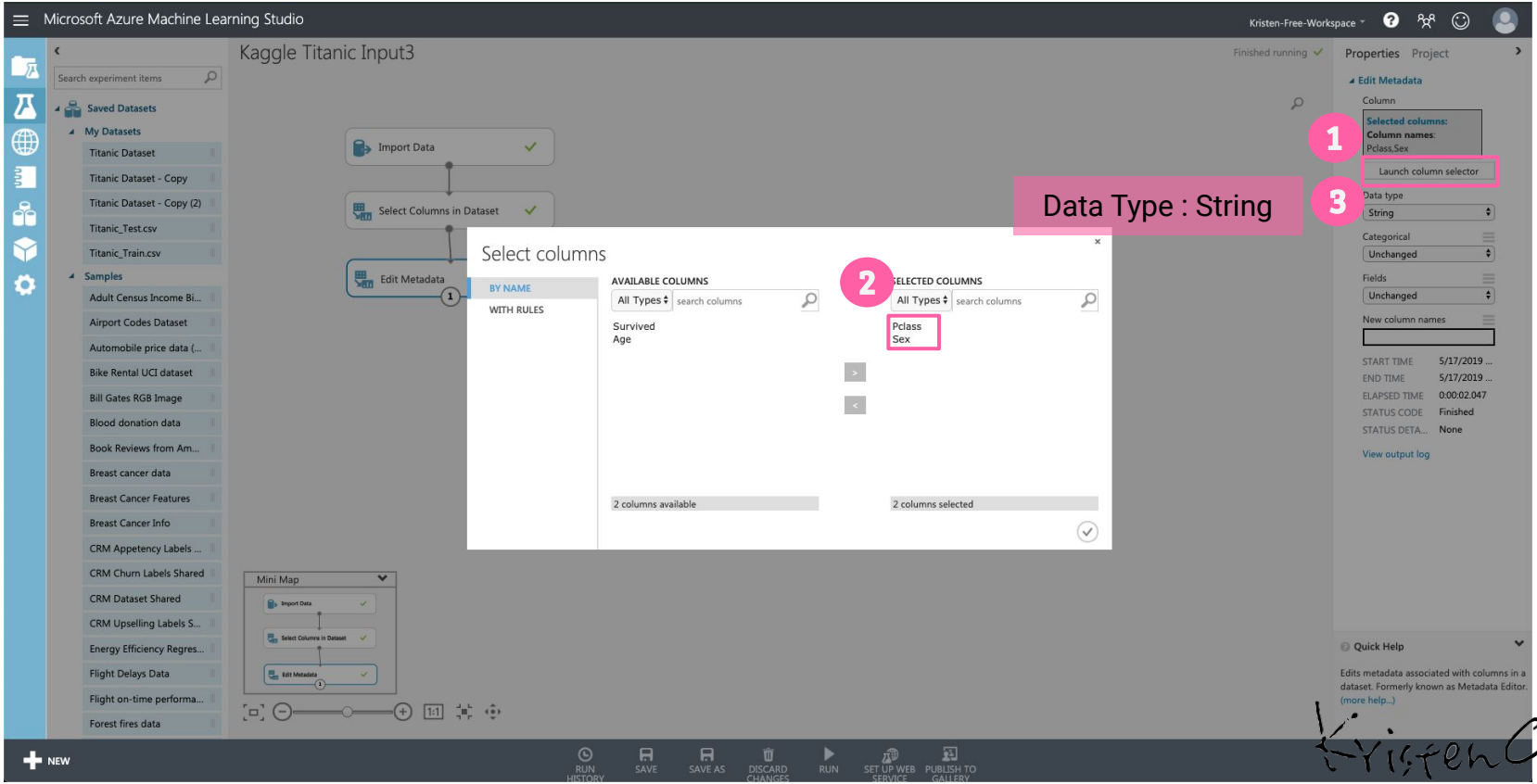

Microsoft Azure Machine Learning Studio
Kristen-Free-Workspace
Kaggle Titanic Easy
In draft
Draft saved at 下午1:40:54
Search experiment items
Data Input and Output
Data Transformation
Filter
Learning with Counts
Manipulation
Add Columns
Add Rows
Apply SQL Transformati...
Clean Missing Data
Convert to Indicator Val...
Edit Metadata
Group Categorical Values
Join Data
Remove Duplicate Rows
Select Columns in Datas...
Select Columns Transfor...
Properties
Project
Select Columns in Dataset
Select columns
Launch column selector
Select columns
Survived, Pclass, Sex, Age
BY NAME
WITH RULES
AVAILABLE COLUMNS
SELECTED COLUMNS
All Types
search columns
PassengerId
Name
Sex
SibSp
Parch
Ticket
Fare
Cabin
Embarked
Survived
Pclass
Age
要的選過來
9 columns available
3 columns selected
選完記得打勾
1
2
3
Quick Help
Selects columns to include or exclude from a dataset in an operation. Formerly known as Project Columns.
(more help...)
NEW
RUN HISTORY
SAVE
SAVE AS
DISCARD CHANGES
RUN
SET UP WEB SERVICE
PUBLISH TO GALLERY

確認資料 -- 確定資料型態

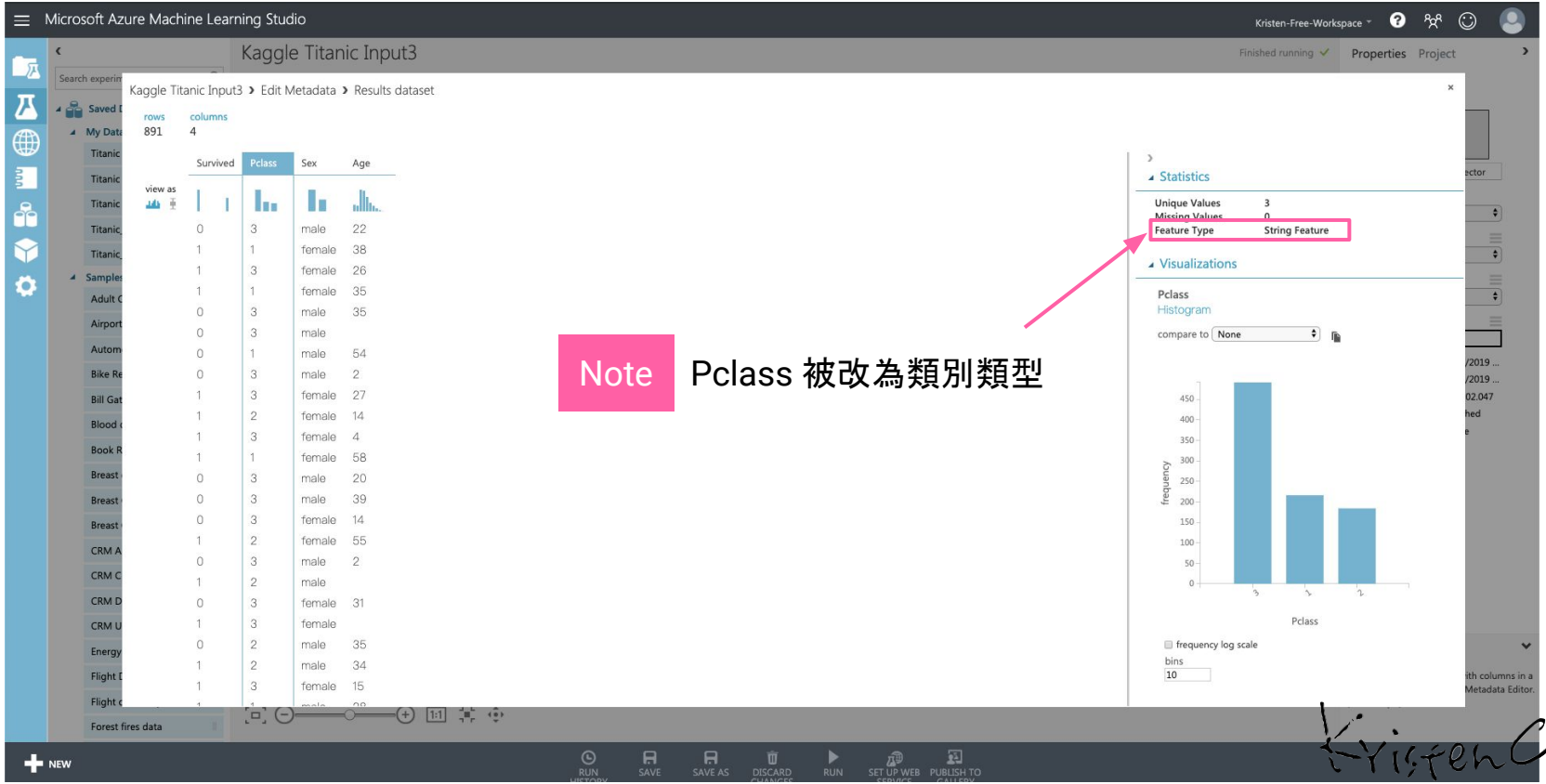


Note Pclass 的資料只有 1, 2, 3 三種
資料類型應該是類別型

確認資料 -- 確定資料型態

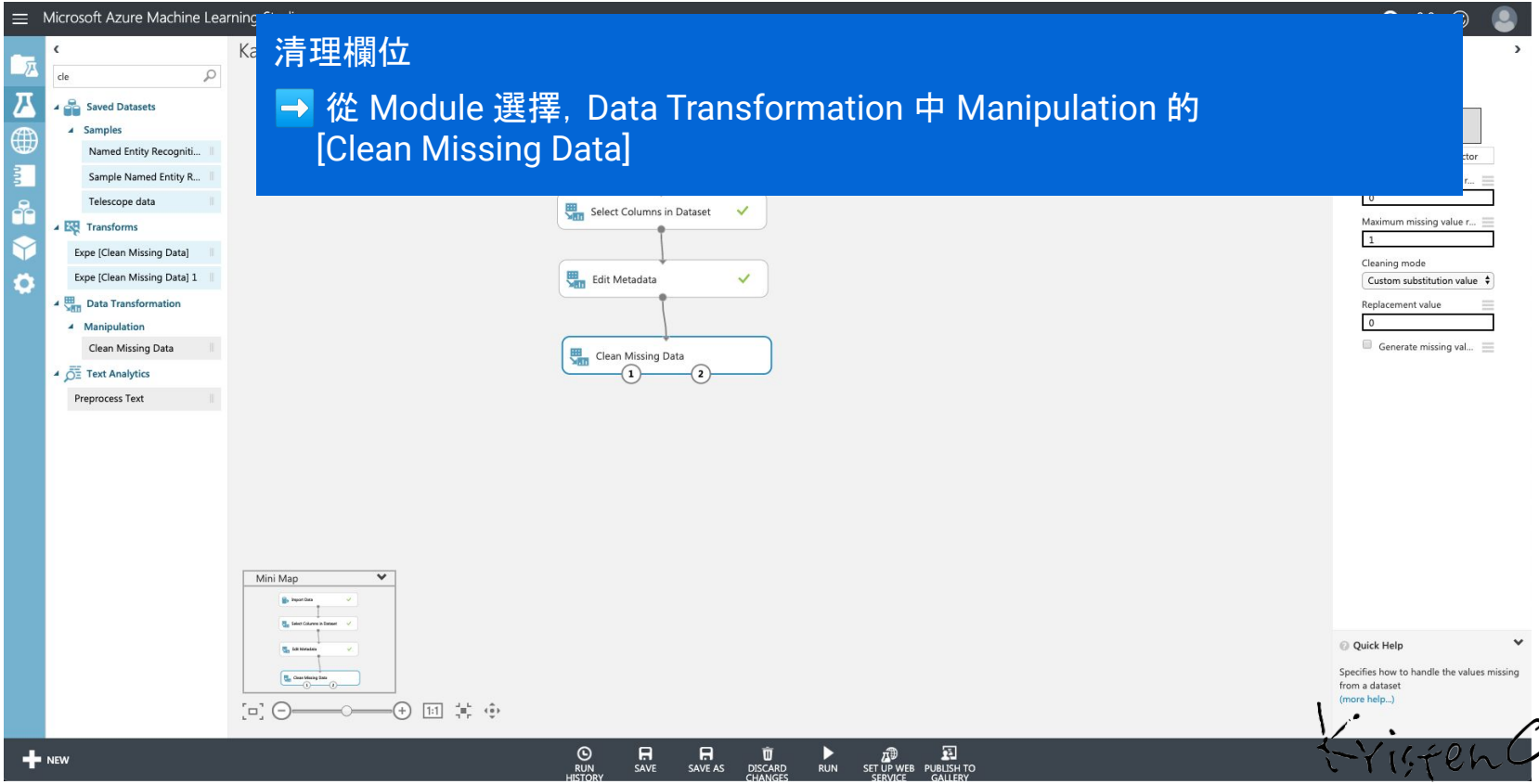

Microsoft Azure Machine Learning Studio
Kristen-Free-Workspace
Kaggle Titanic Input3
Finished running
Properties
Project
Edit Metadata
Column
Selected columns:
Column names:
Pclass,Sex
Launch column selector
Data type
String
Categorical
Unchanged
Fields
Unchanged
New column names
START TIME 5/17/2019 ...
END TIME 5/17/2019 ...
ELAPSED TIME 0:00:02.047
STATUS CODE Finished
STATUS DETA... None
View output log
Quick Help
Edits metadata associated with columns in a dataset. Formerly known as Metadata Editor.
(more help...)
Search experiment items
Saved Datasets
My Datasets
Titanic Dataset
Titanic Dataset - Copy
Titanic Dataset - Copy (2)
Titanic_Test.csv
Titanic_Train.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...
Breast cancer data
Breast Cancer Features
Breast Cancer Info
CRM Appetency Labels ...
CRM Churn Labels Shared
CRM Dataset Shared
CRM Upselling Labels S...
Energy Efficiency Regres...
Flight Delays Data
Flight on-time performa...
Forest fires data
Import Data
Select Columns in Dataset
Edit Metadata
Mini Map
Select columns
BY NAME
WITH RULES
AVAILABLE COLUMNS
All Types
search columns
Survived
Age
2 columns available
SELECTED COLUMNS
Pclass
Sex
2 columns selected
1
2
3
Data Type : String
NEW
RUN HISTORY
SAVE
SAVE AS
DISCARD CHANGES
RUN
SET UP WEB SERVICE
PUBLISH TO GALLERY

確認資料 -- 確定資料型態



Note Pclass 被改為類別類型

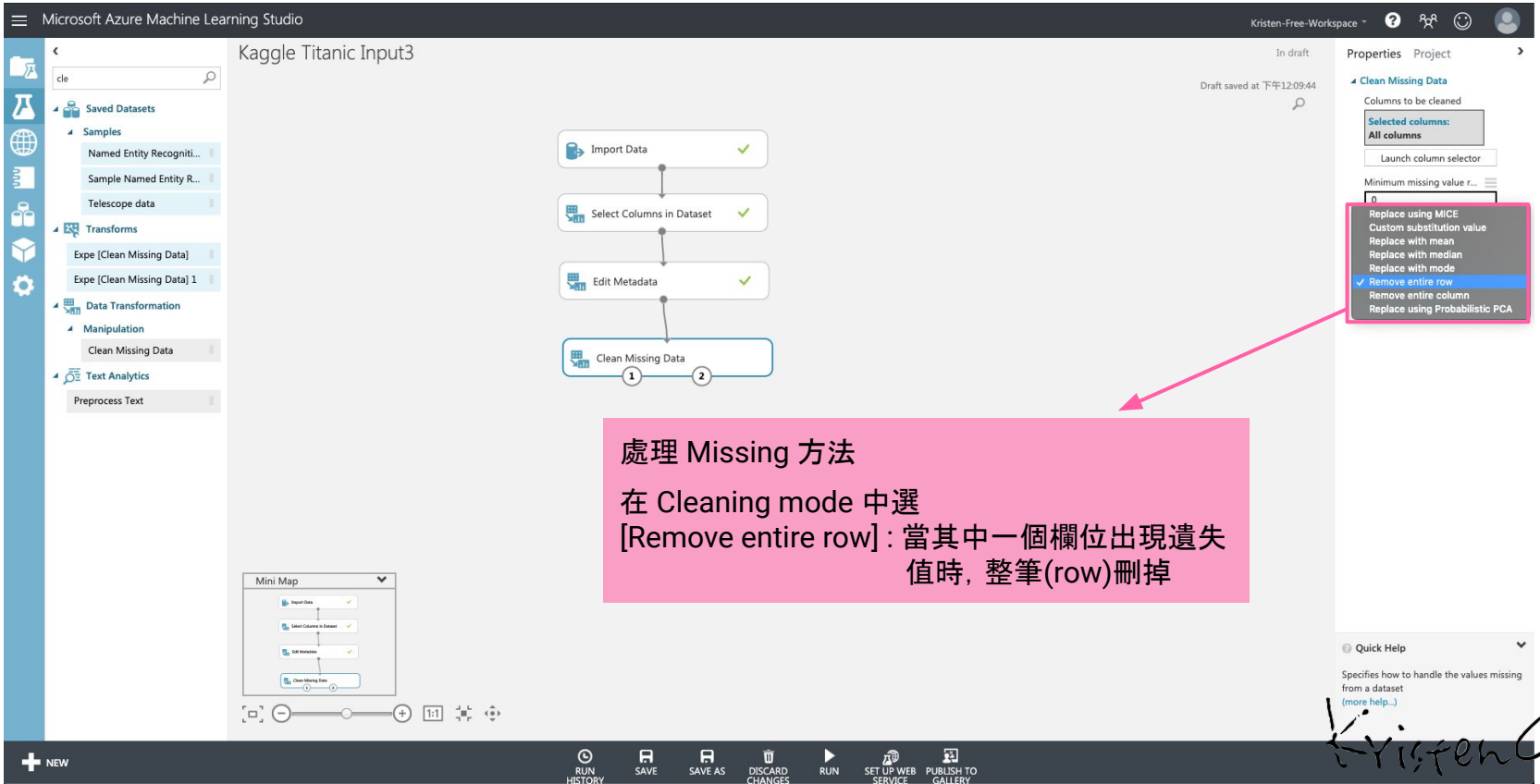
清理資料



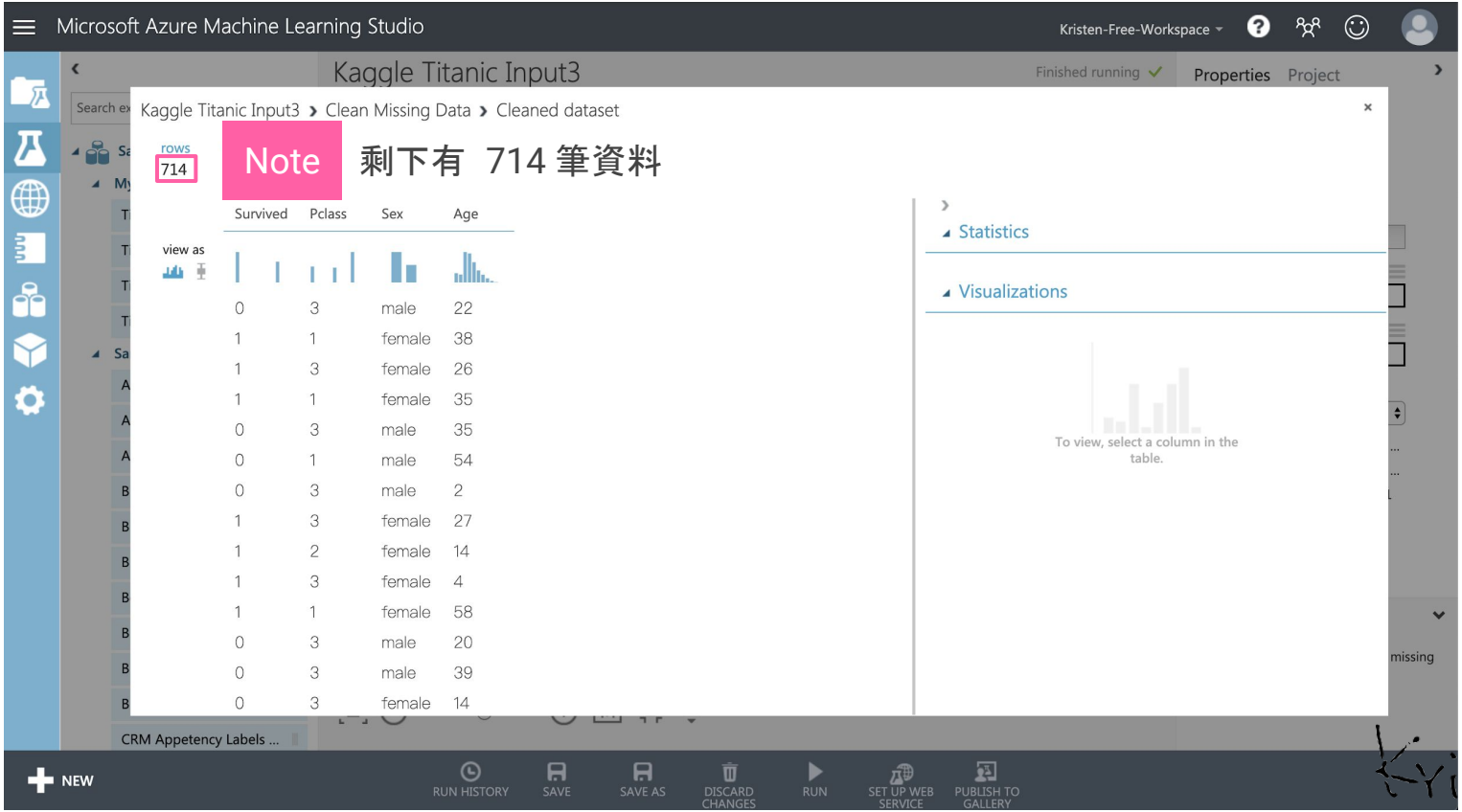
清理欄位

➡ 從 Module 選擇，Data Transformation 中 Manipulation 的 [Clean Missing Data]

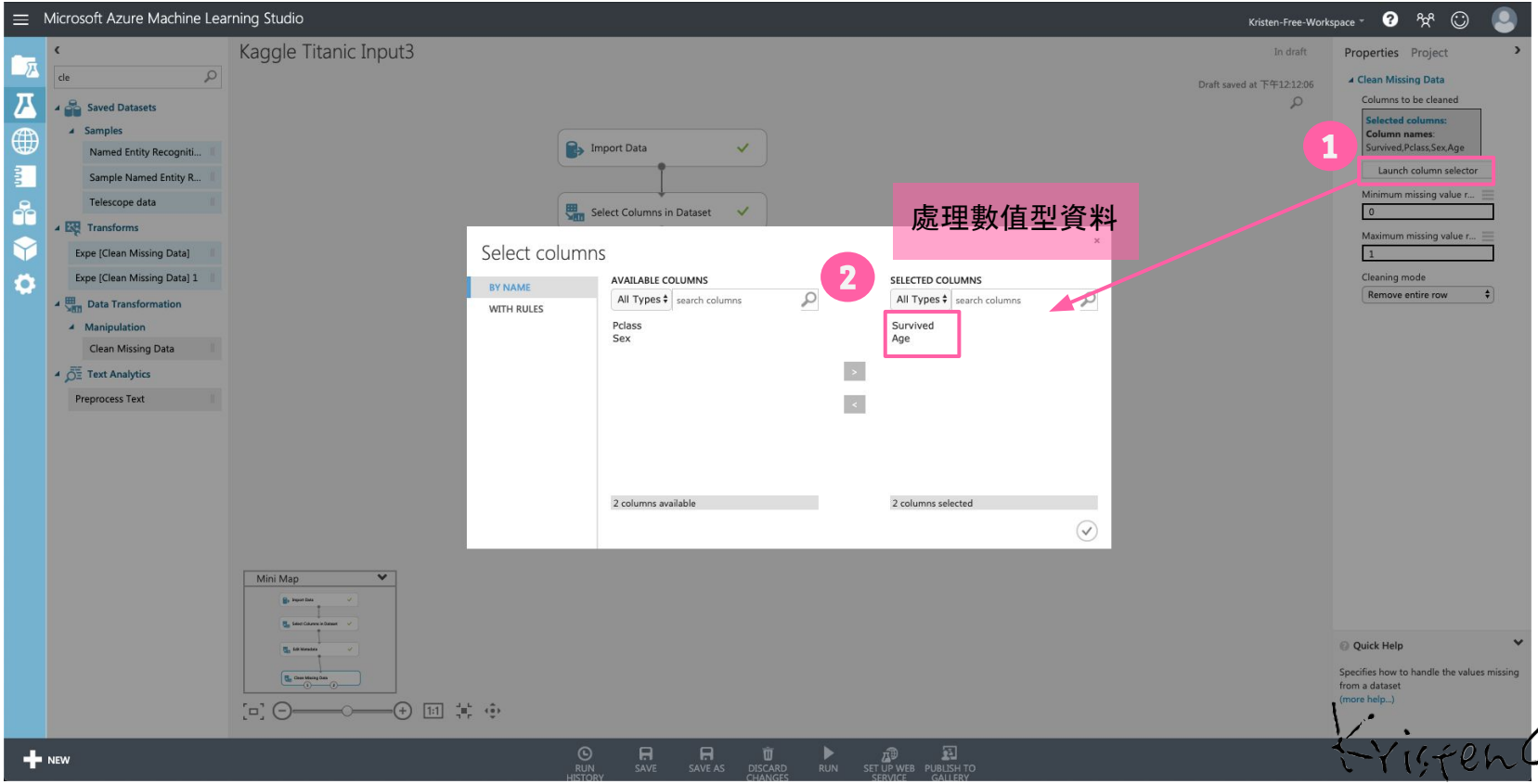
[Method 1] 清理資料 -- 整筆刪掉



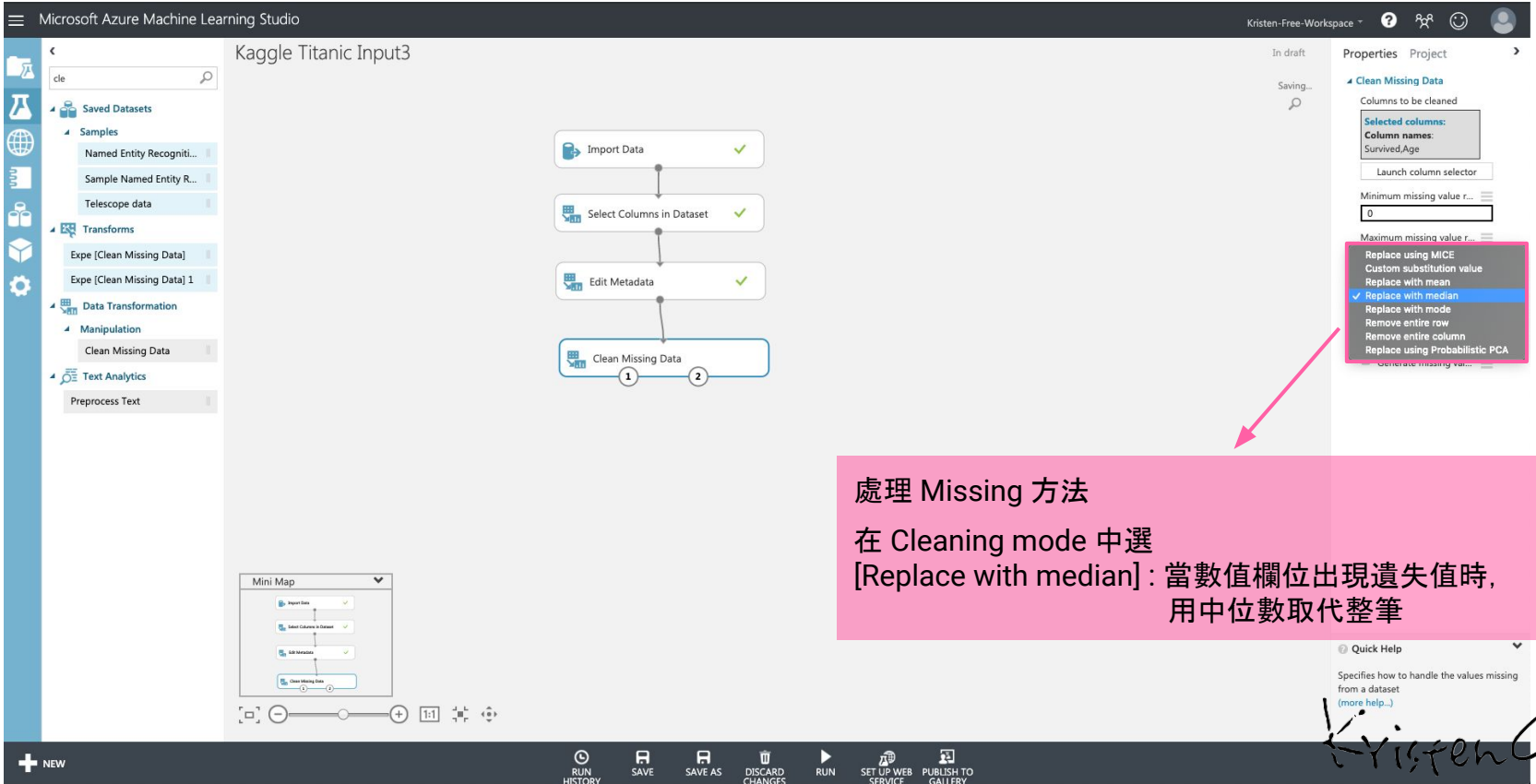
[Method 1] 清理資料 -- 整筆刪掉



[Method 2] 清理資料 -- 數值型資料用中位數取代



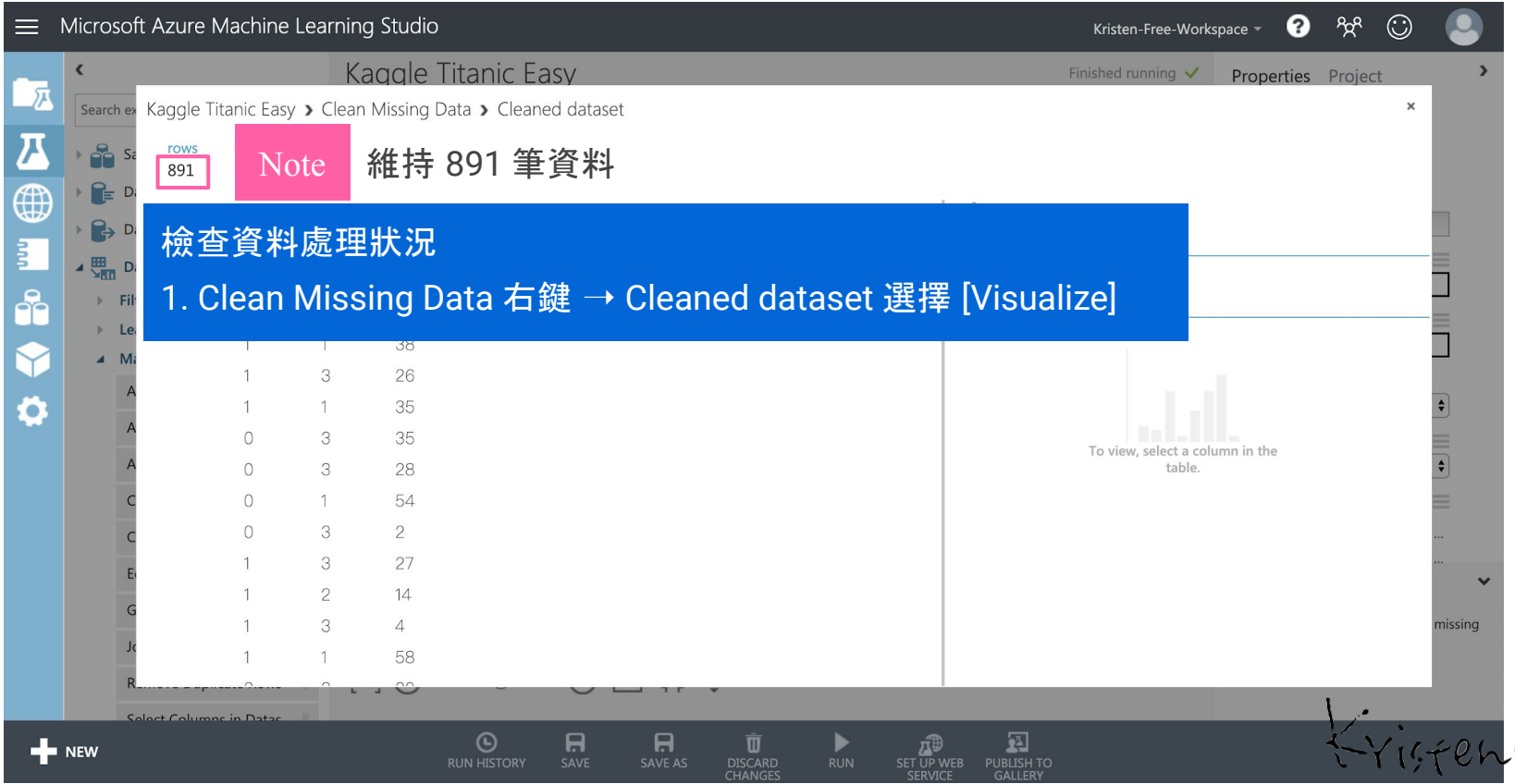
[Method 2] 清理資料 -- 數值型資料用中位數取代



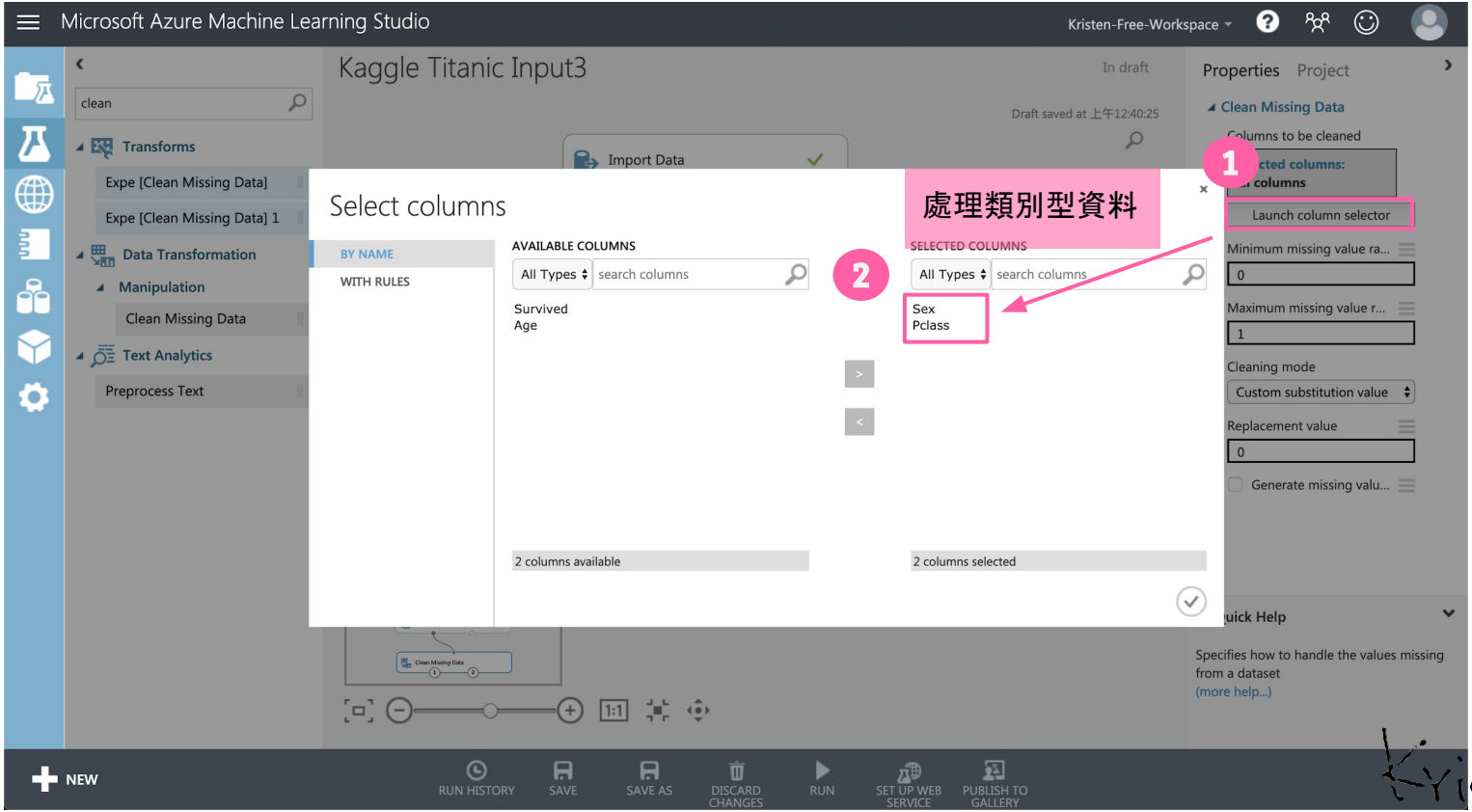
處理 Missing 方法

在 Cleaning mode 中選
[Replace with median] : 當數值欄位出現遺失值時,
用中位數取代整筆

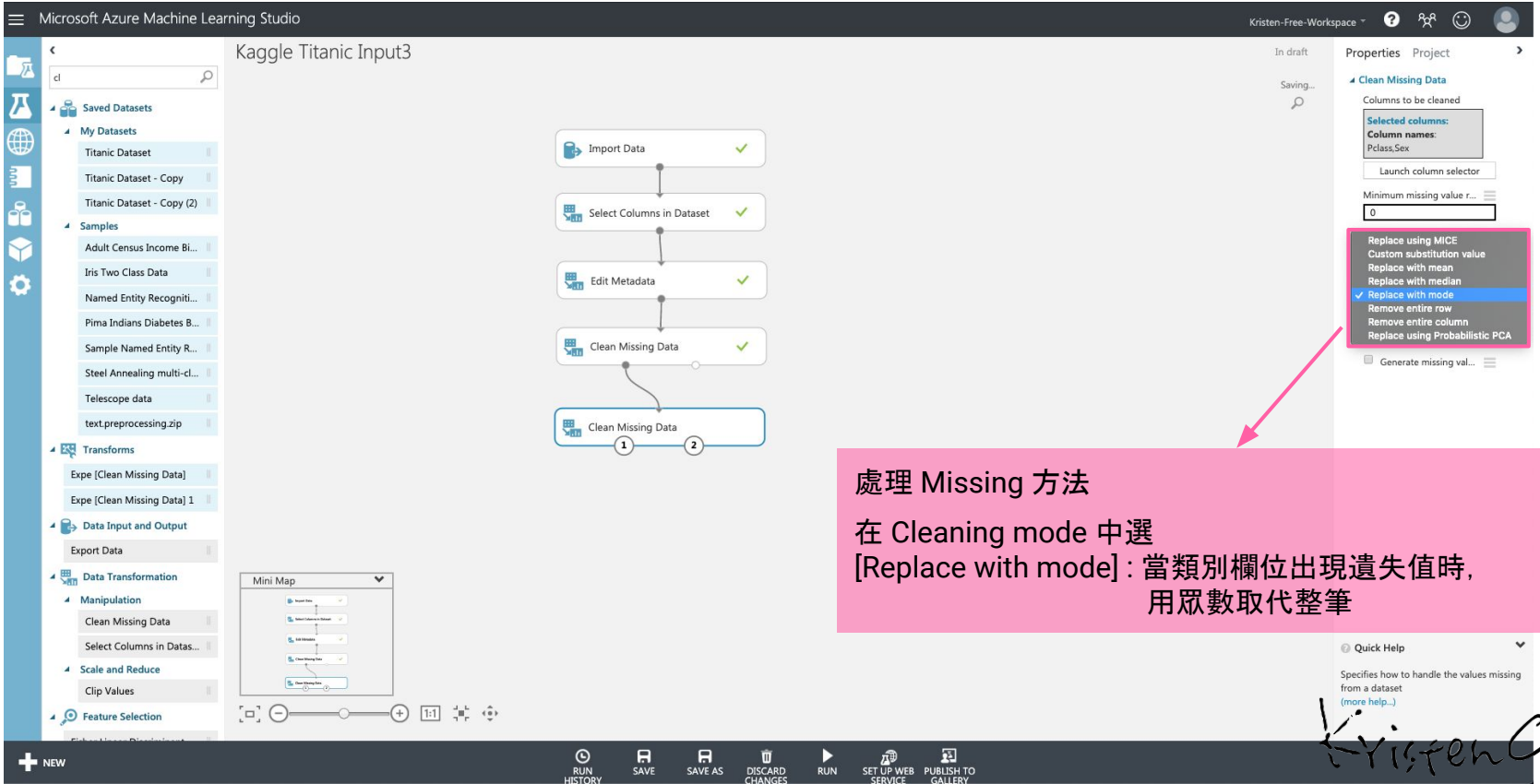
[Method 2] 清理資料 -- 數值型資料用中位數取代



[Method 3] 清理資料 -- 類別型資料用眾數取代



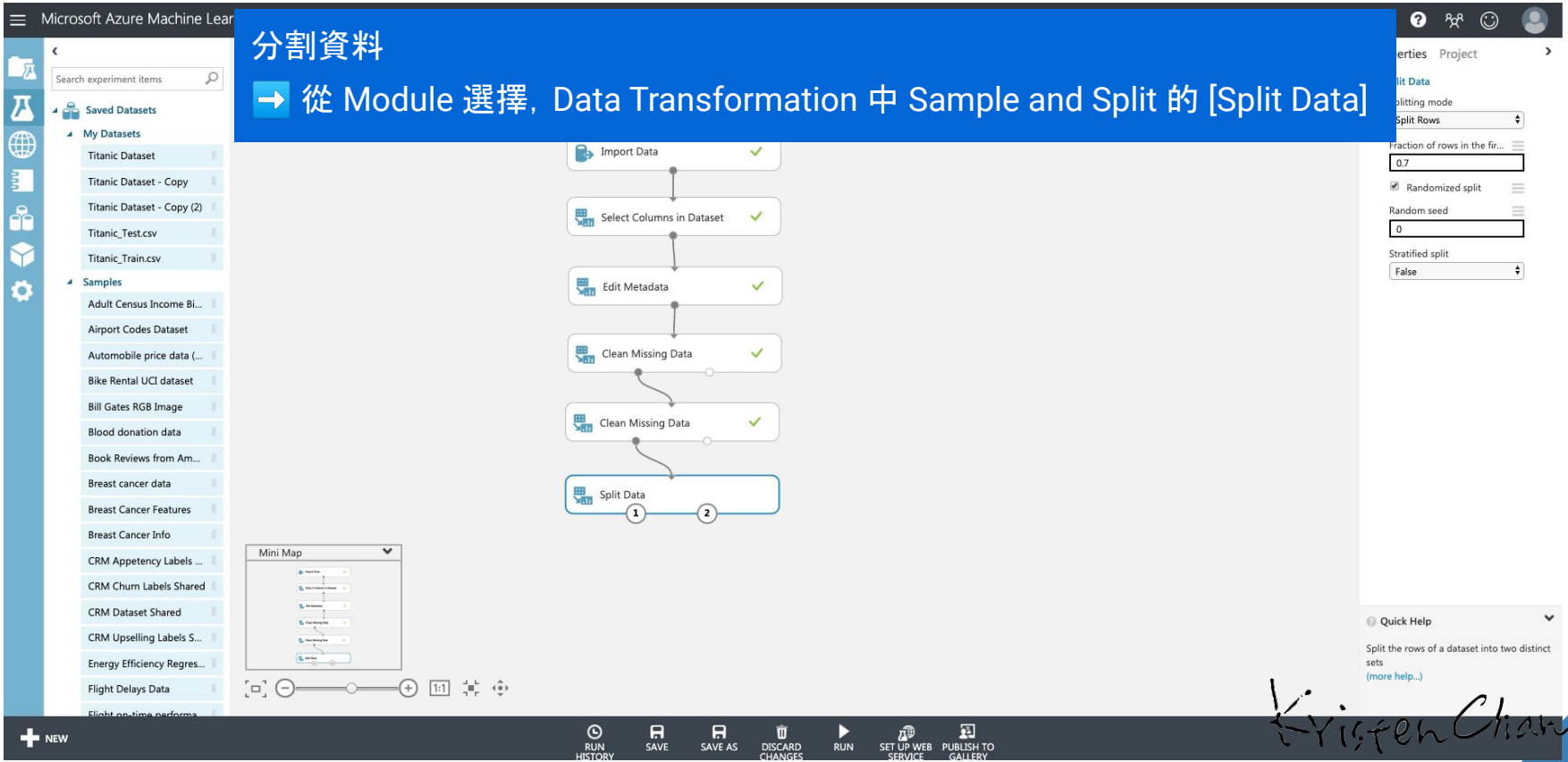
[Method 3] 清理資料 -- 類別型資料用眾數取代



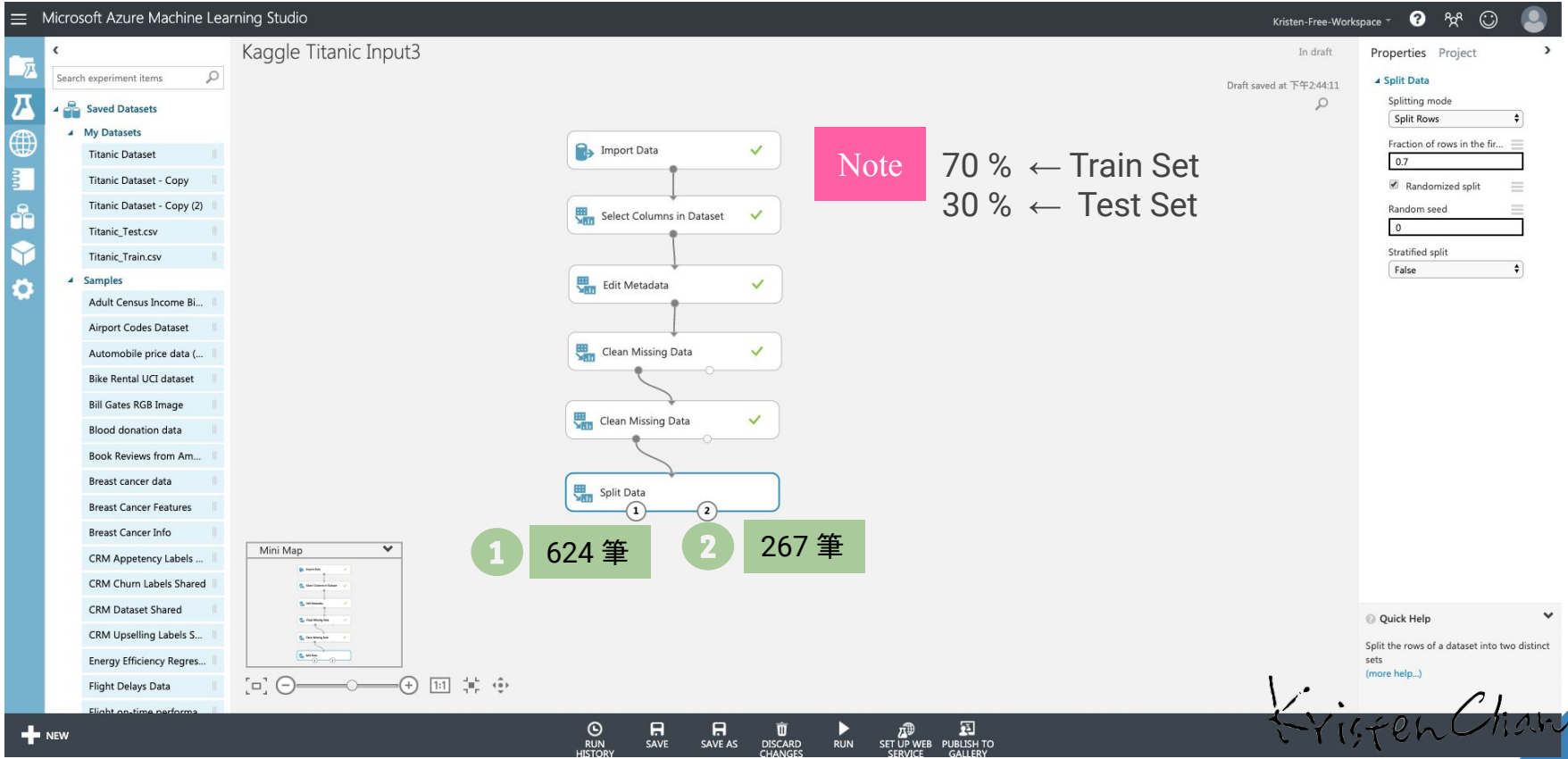
處理 Missing 方法

在 Cleaning mode 中選
[Replace with mode] : 當類別欄位出現遺失值時,
用眾數取代整筆

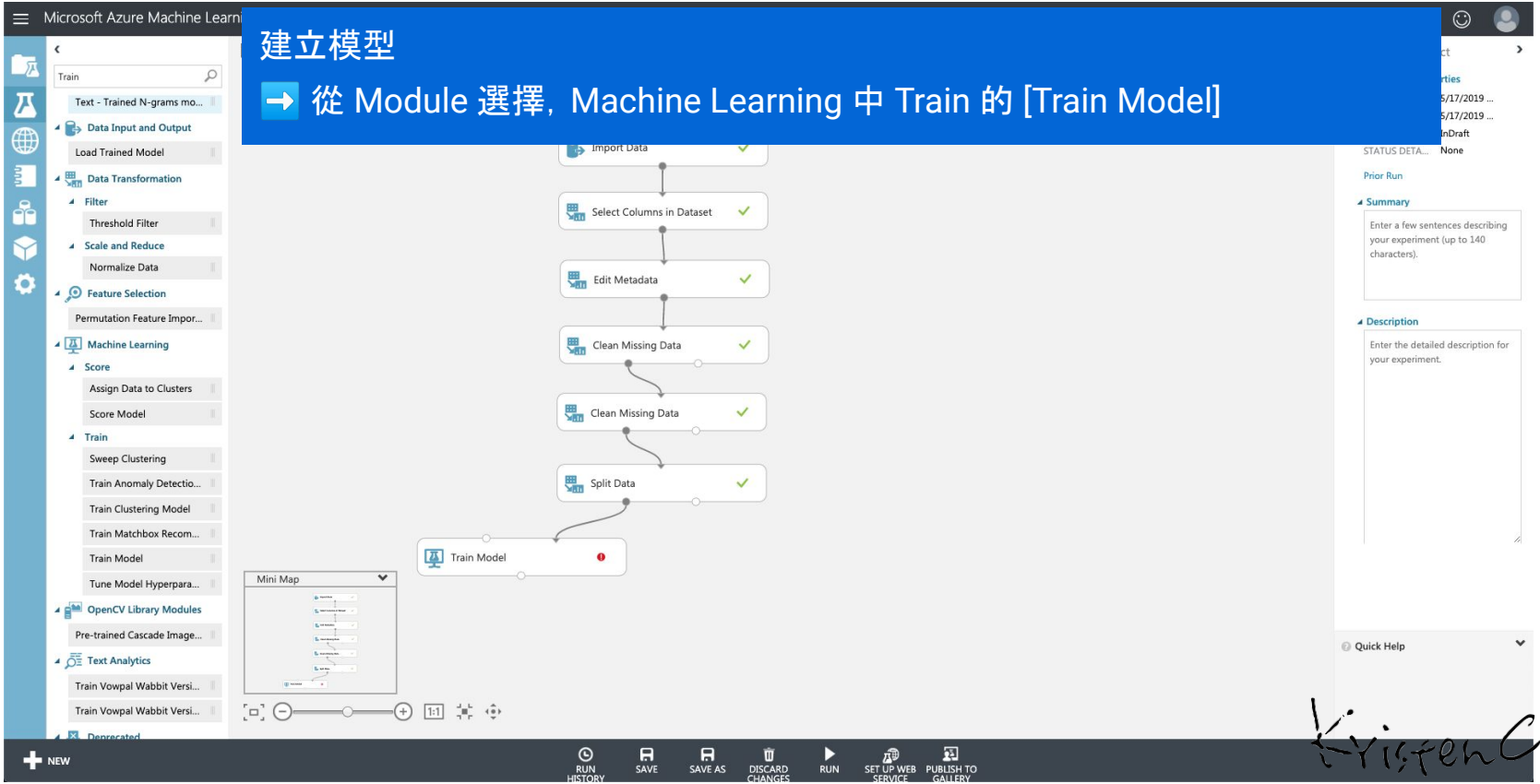
分割資料 -- Train Set / Test Set



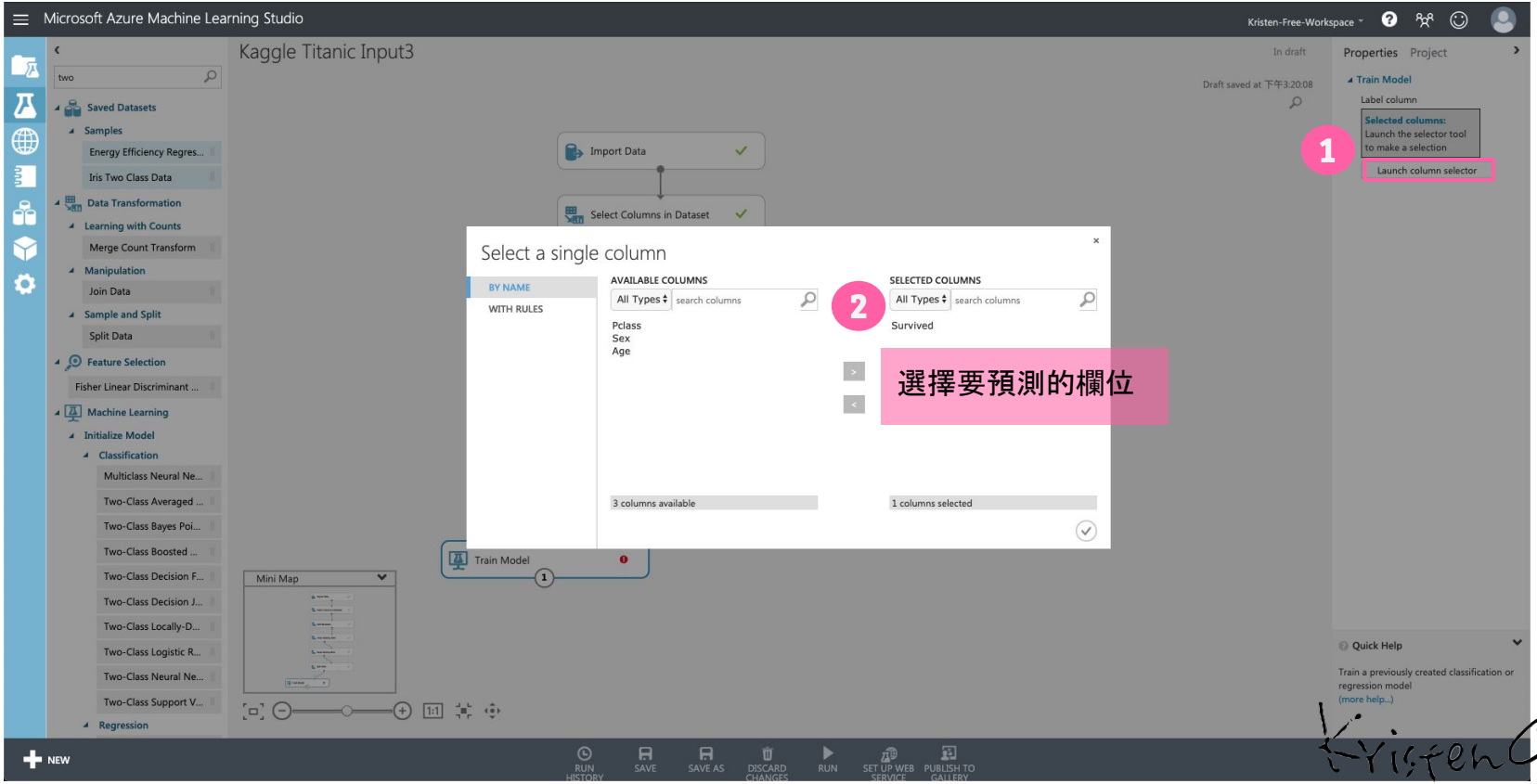
分割資料 -- Train Set / Test Set



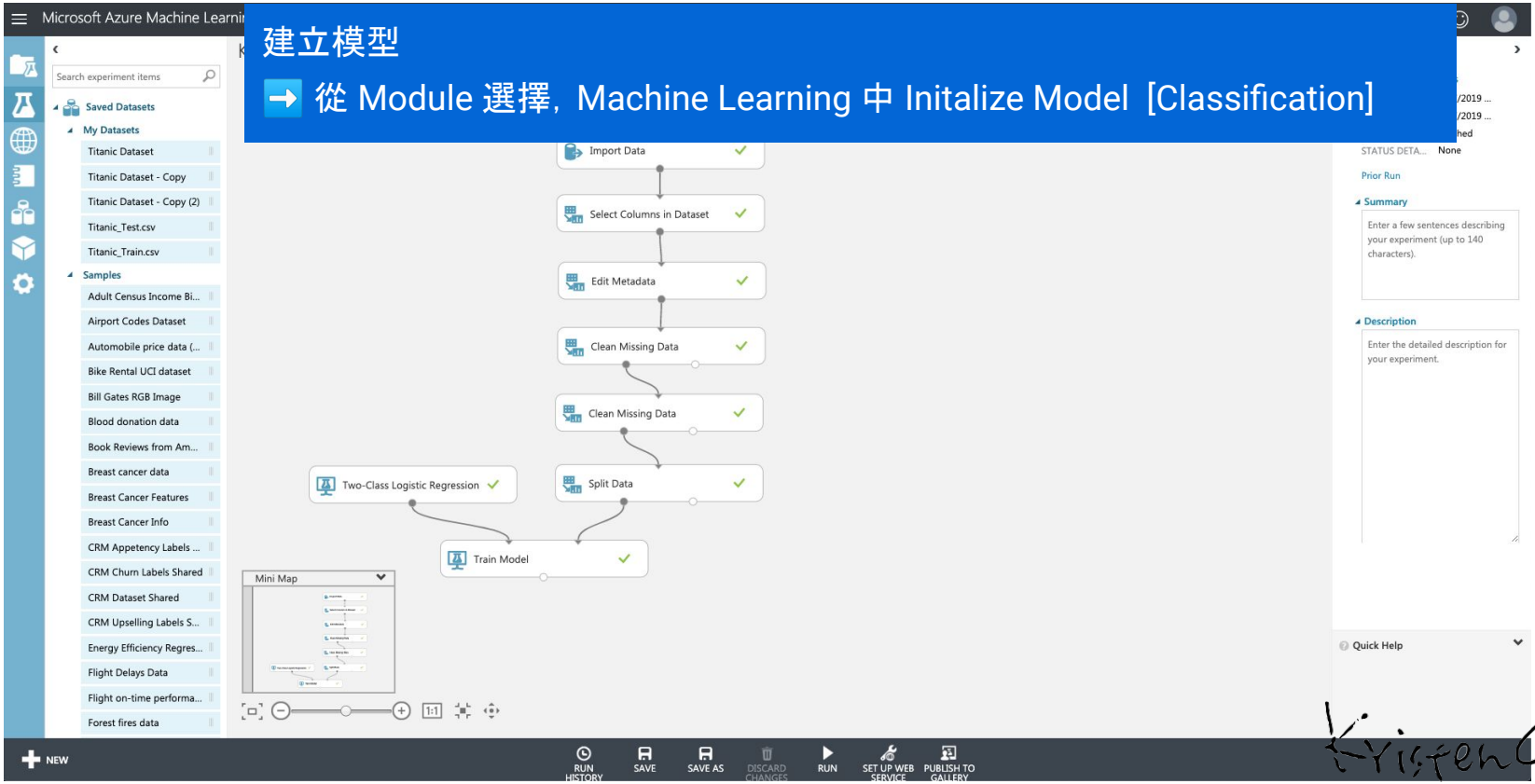
建立 Train Model



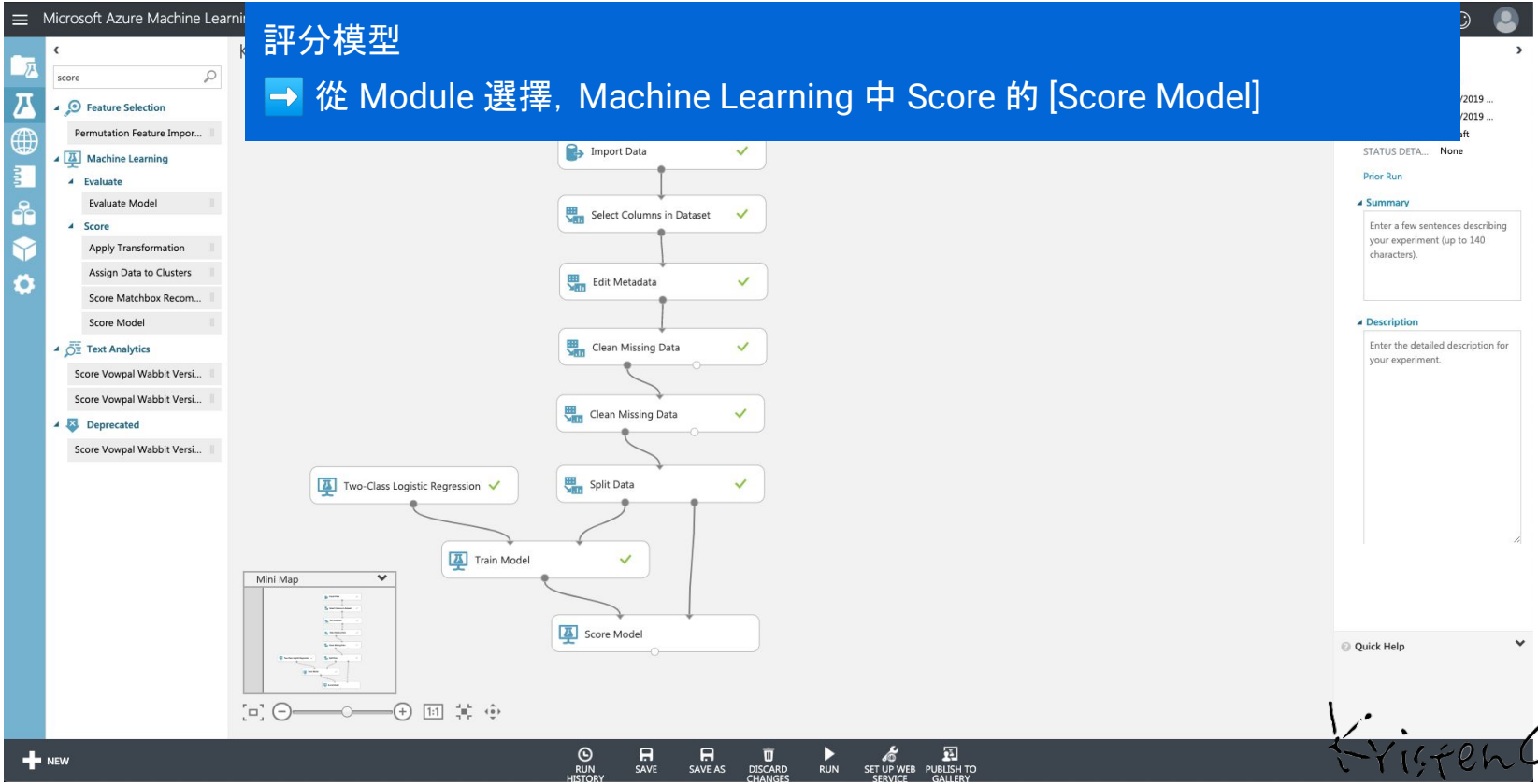
建立 Train Model



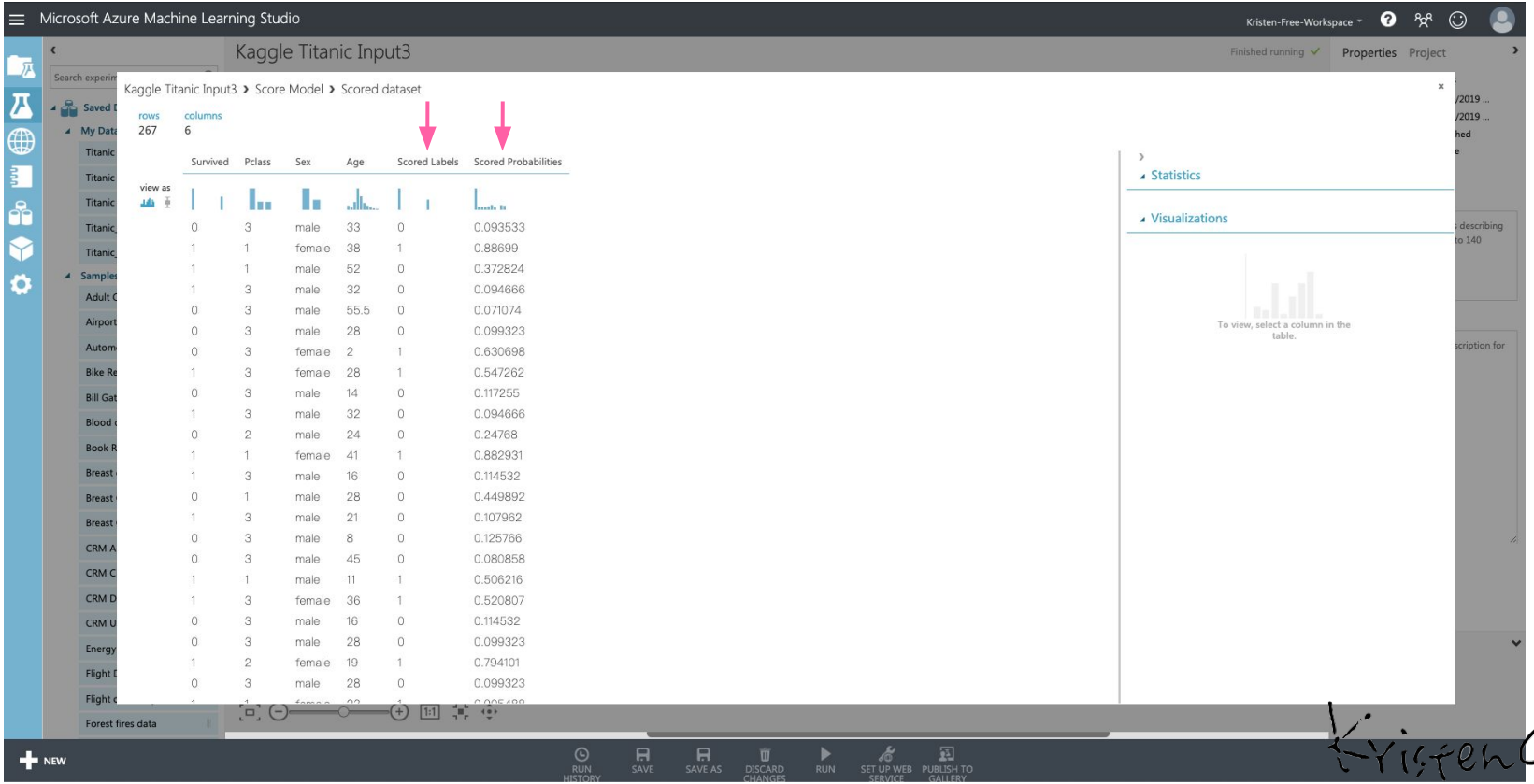
建立 Train Model -- 選擇一個分類方法



將 Model 套用在 Test Set 中



將 Model 套用在 Test Set 中



評估模型好壞

Confuson Matrix	實際 YES	實際 NO
預測 YES	True Positive (TP)	False Positive (FP) Type I Error
預測 NO	False Negative (FN) Type II Error	True Negative (TN)

評估模型好壞

$$\text{Accuracy} = \frac{TP + TN}{\text{Total } N}$$

Confuson Matrix	實際 YES	實際 NO
預測 YES	**True Positive (TP)**	False Positive (FP) Type I Error
預測 NO	False Negative (FN) Type II Error	**True Negative (TN)**

評估模型好壞

Confuson Matrix	實際 YES	實際 NO
預測 YES	True Positive (TP)	False Positive (FP) Type I Error
預測 NO	False Negative (FN) Type II Error	True Negative (TN)

$$\text{Precision} = \frac{TP}{TP + FP}$$

評估模型好壞

Confuson Matrix	實際 YES	實際 NO
預測 YES	True Positive (TP)	False Positive (FP) Type I Error
預測 NO	False Negative (FN) Type II Error	True Negative (TN)

$$\text{Recall} = \frac{TP}{TP + FN}$$

評估模型好壞

Confuson Matrix	實際 YES	實際 NO
預測 YES	True Positive (TP)	False Positive (FP) Type I Error
預測 NO	False Negative (FN) Type II Error	True Negative (TN)

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$\text{F1 Score} = \frac{2}{\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}}$$

評估模型好壞

$$\text{Accuracy} = \frac{TP + TN}{\text{Total } N}$$

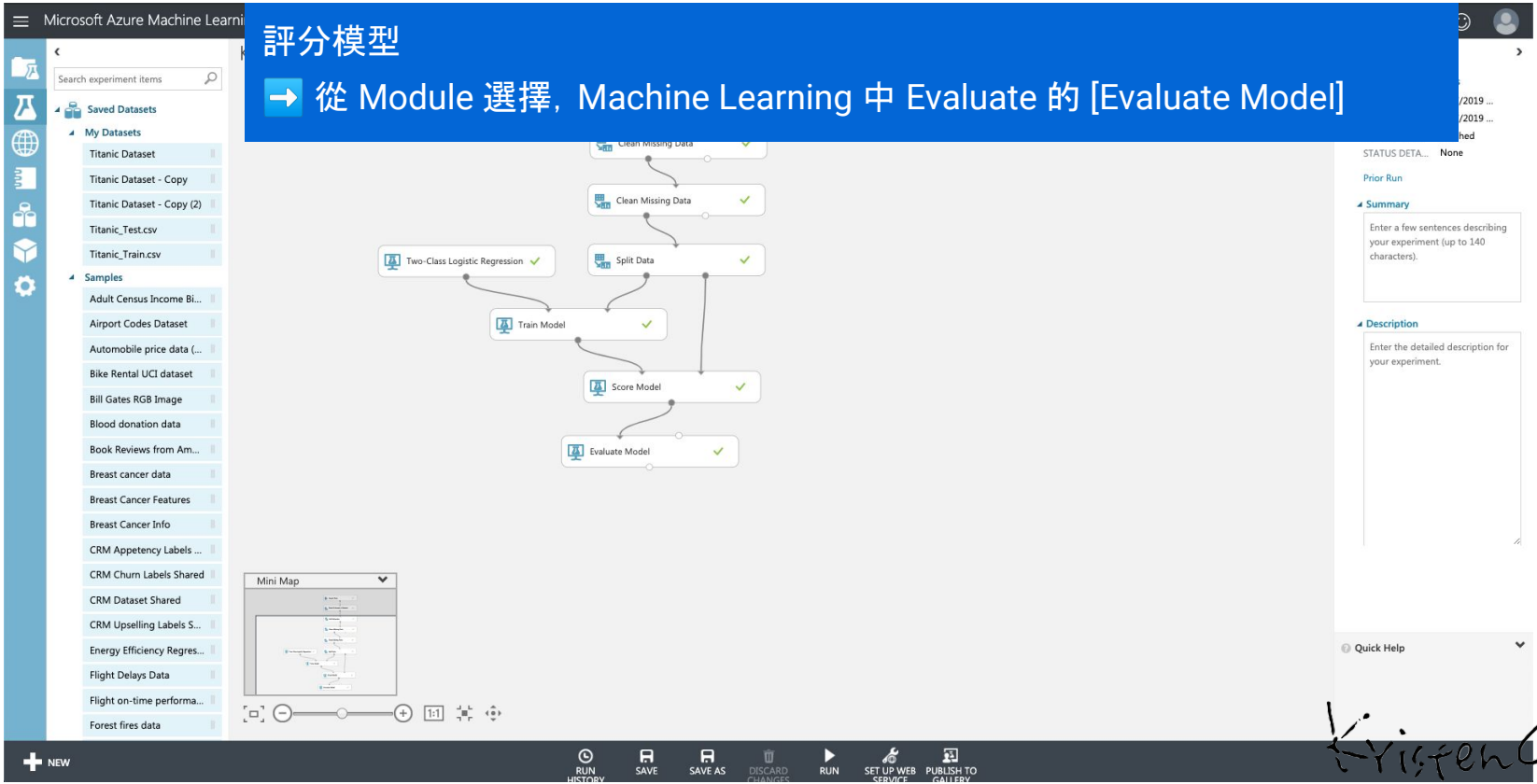
Confuson Matrix	實際 YES	實際 NO
預測 YES	True Positive (TP)	False Positive (FP) Type I Error
預測 NO	False Negative (FN) Type II Error	True Negative (TN)

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$\text{F1 Score} = \frac{2}{\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}}$$

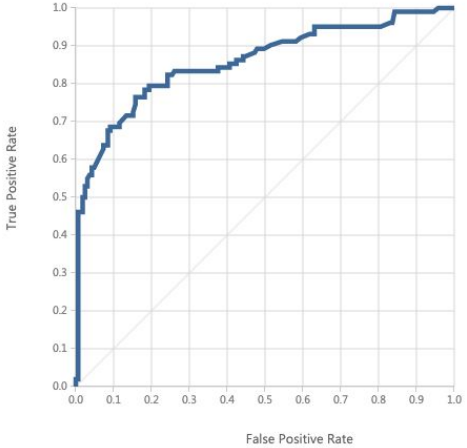
評估模型好壞


評分模型
從 Module 選擇，Machine Learning 中 Evaluate 的 [Evaluate Model]
Microsoft Azure Machine Learni
Search experiment items
Saved Datasets
My Datasets
Titanic Dataset
Titanic Dataset - Copy
Titanic Dataset - Copy (2)
Titanic_Test.csv
Titanic_Train.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...
Breast cancer data
Breast Cancer Features
Breast Cancer Info
CRM Appetency Labels ...
CRM Churn Labels Shared
CRM Dataset Shared
CRM Upselling Labels S...
Energy Efficiency Regres...
Flight Delays Data
Flight on-time performa...
Forest fires data
Clean Missing Data
Clean Missing Data
Two-Class Logistic Regression
Split Data
Train Model
Score Model
Evaluate Model
Mini Map
STATUS DETA...
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
NEW
RUN HISTORY
SAVE
SAVE AS
DISCARD CHANGES
RUN
SET UP WEB SERVICE
PUBLISH TO GALLERY

評估模型好壞

Kaggle Titanic Input3 › Evaluate Model › Evaluation results

ROC PRECISION/RECALL LIFT



Scored dataset

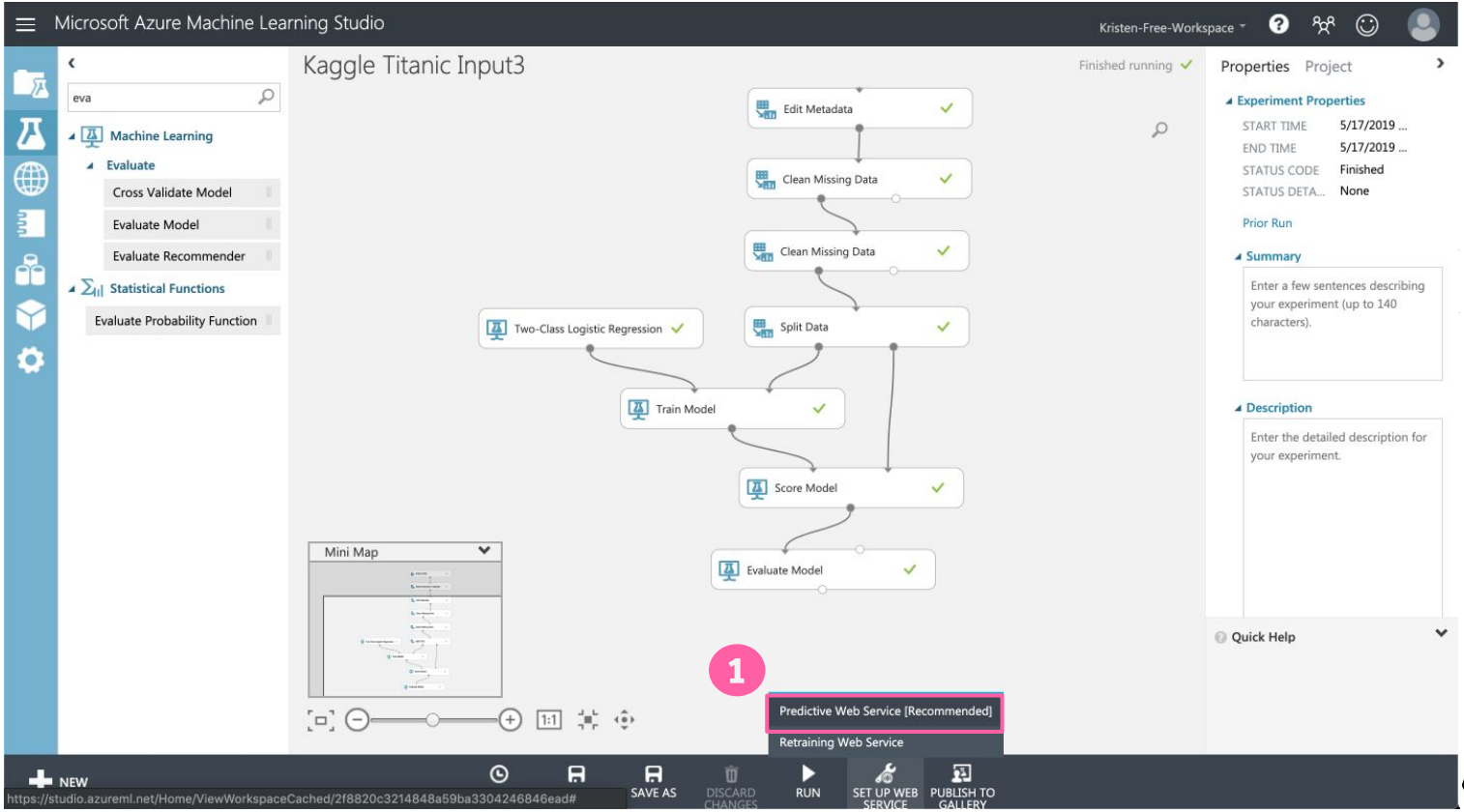
True Positive	False Negative	Accuracy	Precision	Threshold	AUC
70	32	0.820	0.814	0.5	0.858

False Positive	True Negative	Recall	F1 Score
16	149	0.686	0.745

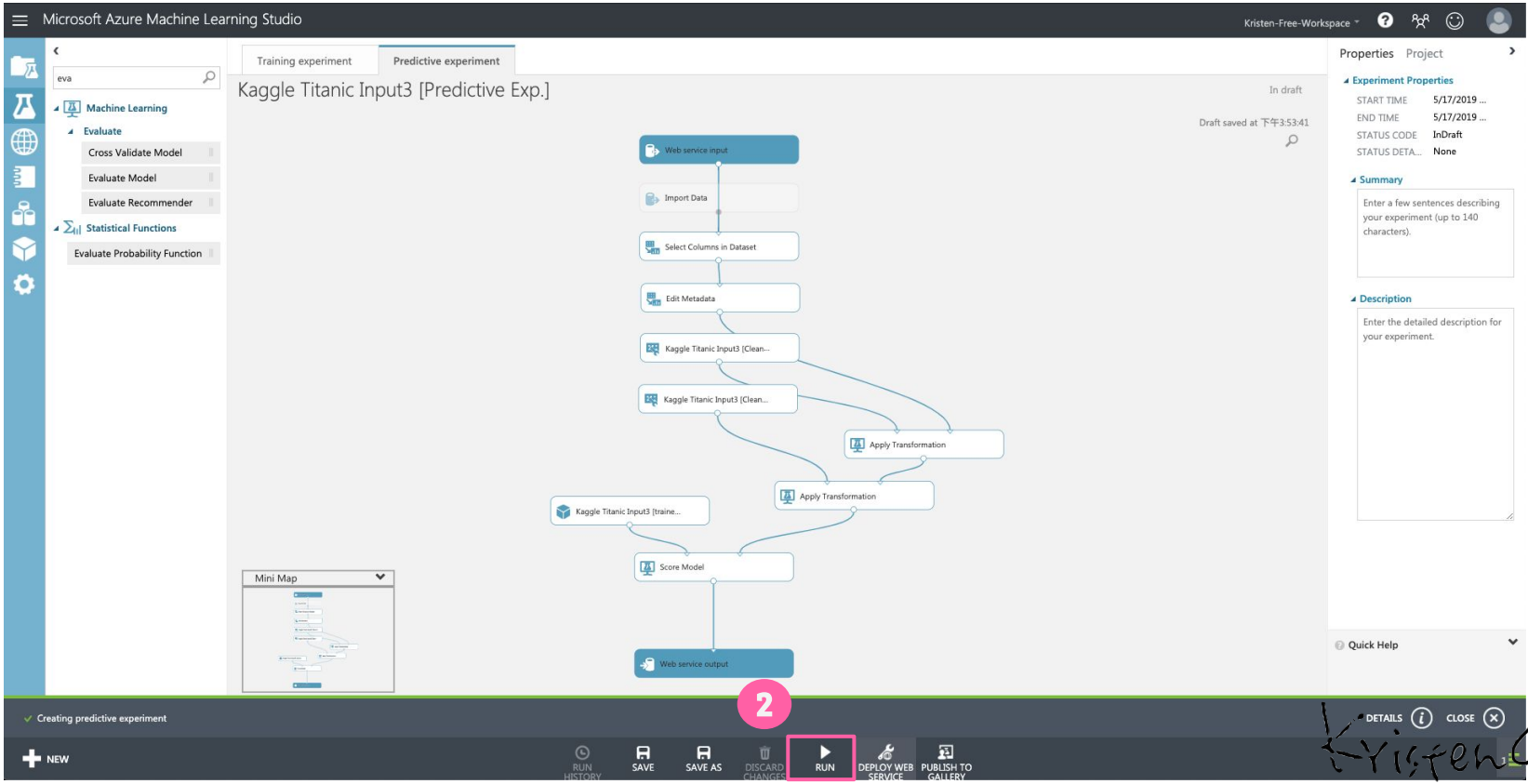
Positive Label	Negative Label
1	0

Score Bin	Positive Examples	Negative Examples	Fraction Above Threshold	Accuracy	F1 Score	Precision	Recall	Negative Precision	Negative Recall	Cumulative AUC
(0.900,1.000]	3	1	0.015	0.625	0.057	0.750	0.029	0.624	0.994	0.000

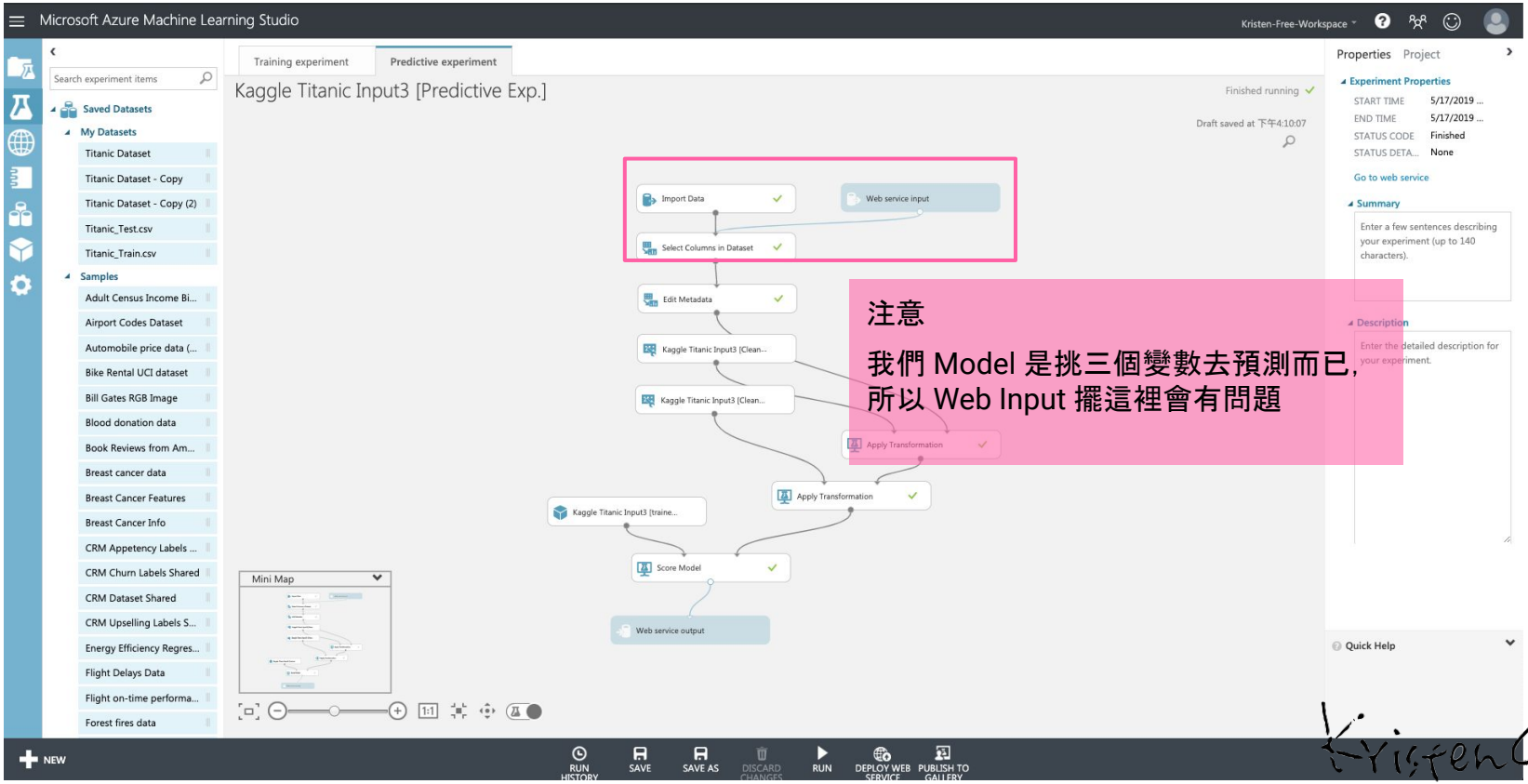
發布 Web Service



發布 Web Service



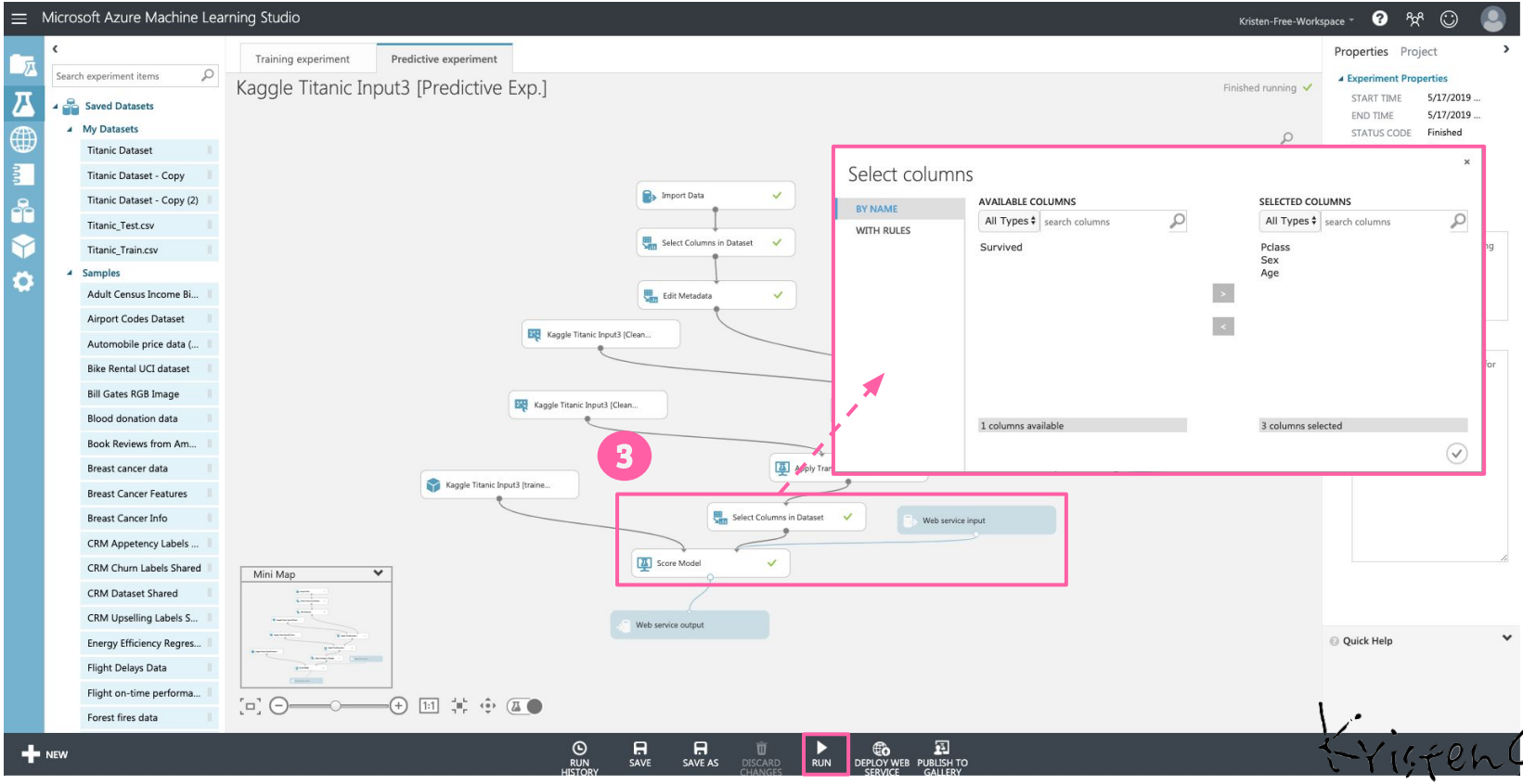
發布 Web Service



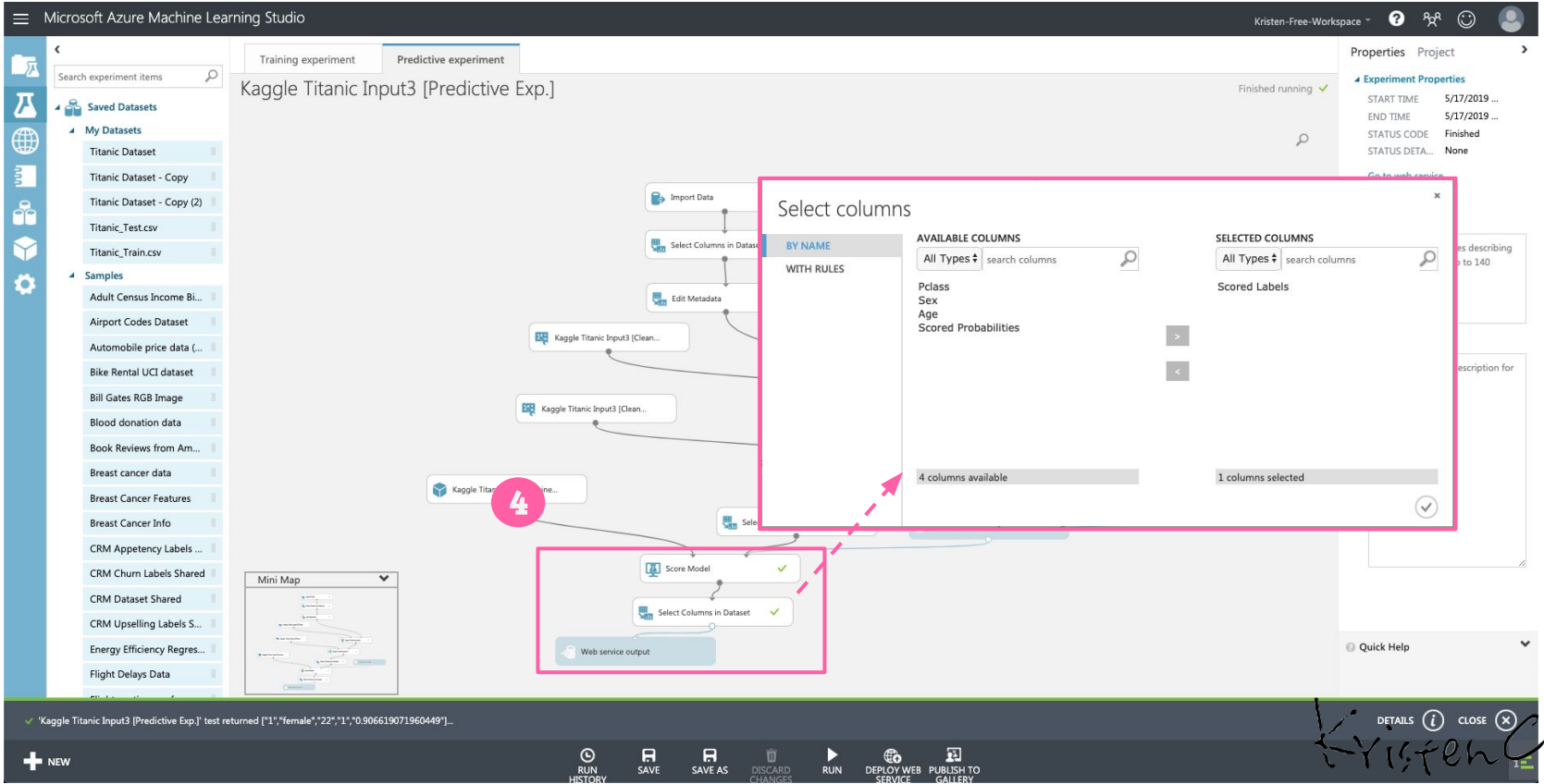
注意

我們 Model 是挑三個變數去預測而已，
所以 Web Input 擺這裡會有問題

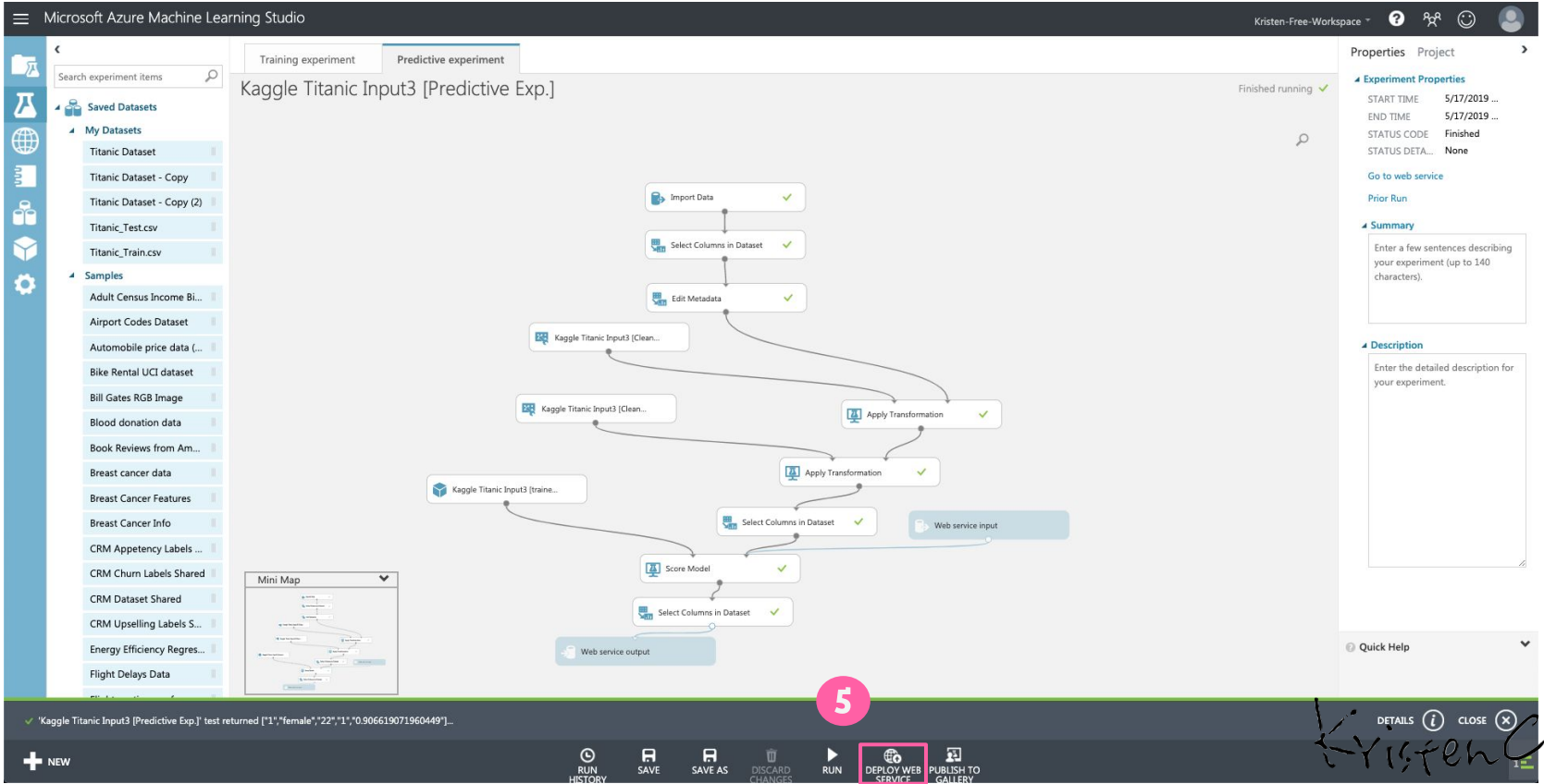
發布 Web Service



發布 Web Service



發布 Web Service



發布 Web Service

Advanced Version -- Python

Enter data to predict 測試

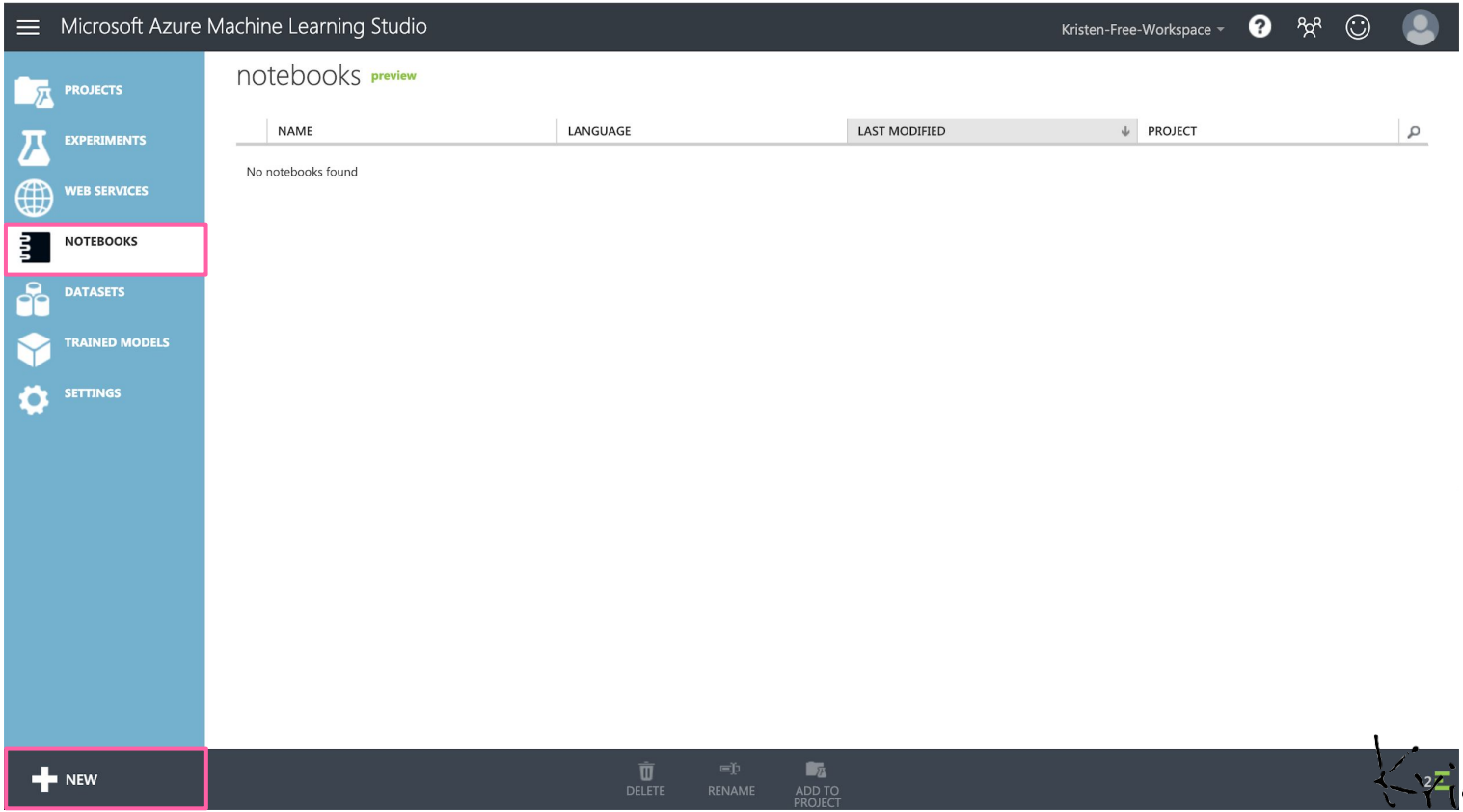
PCLASS
1

SEX
female

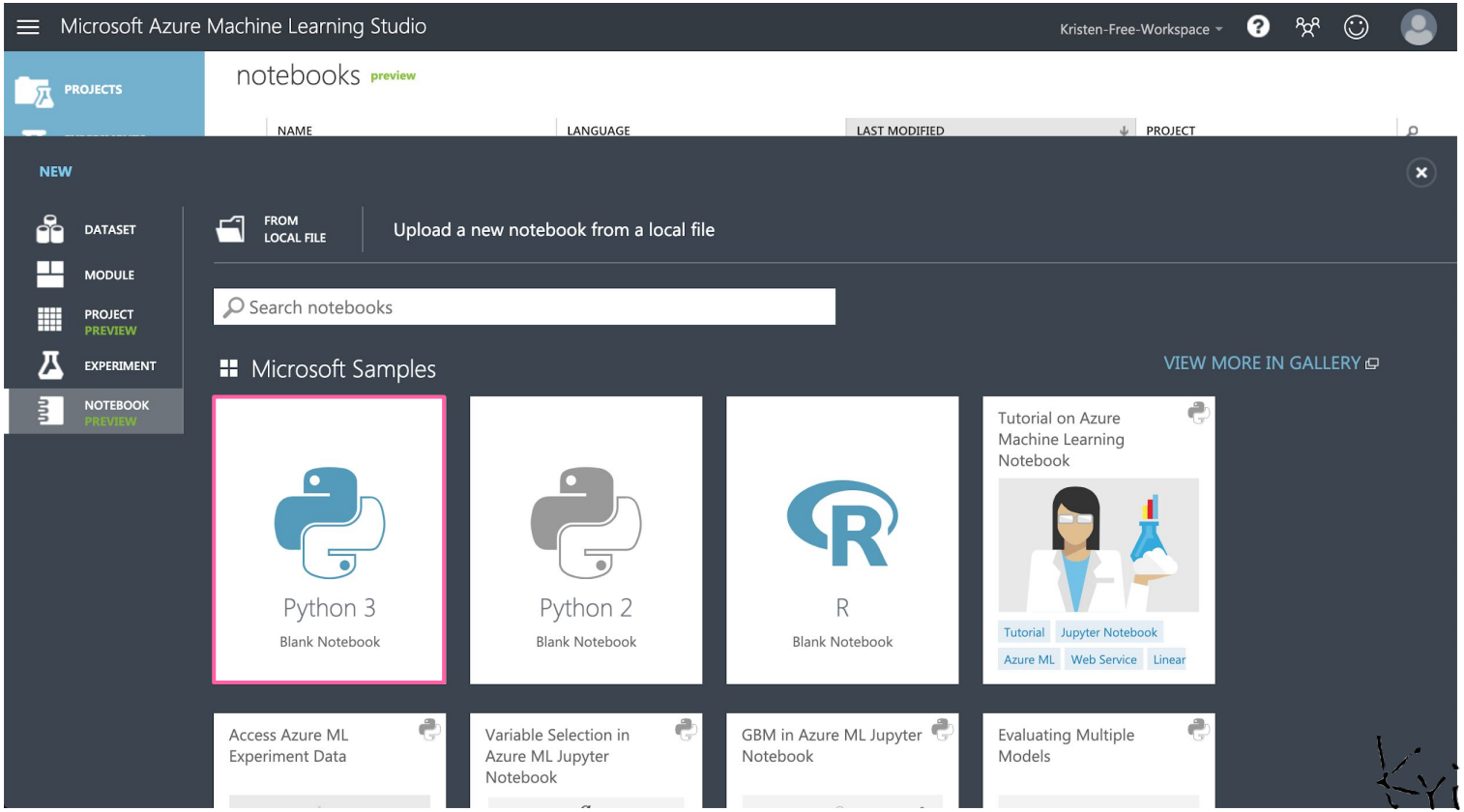
AGE
22

Azure ML Studio with Python Notebook

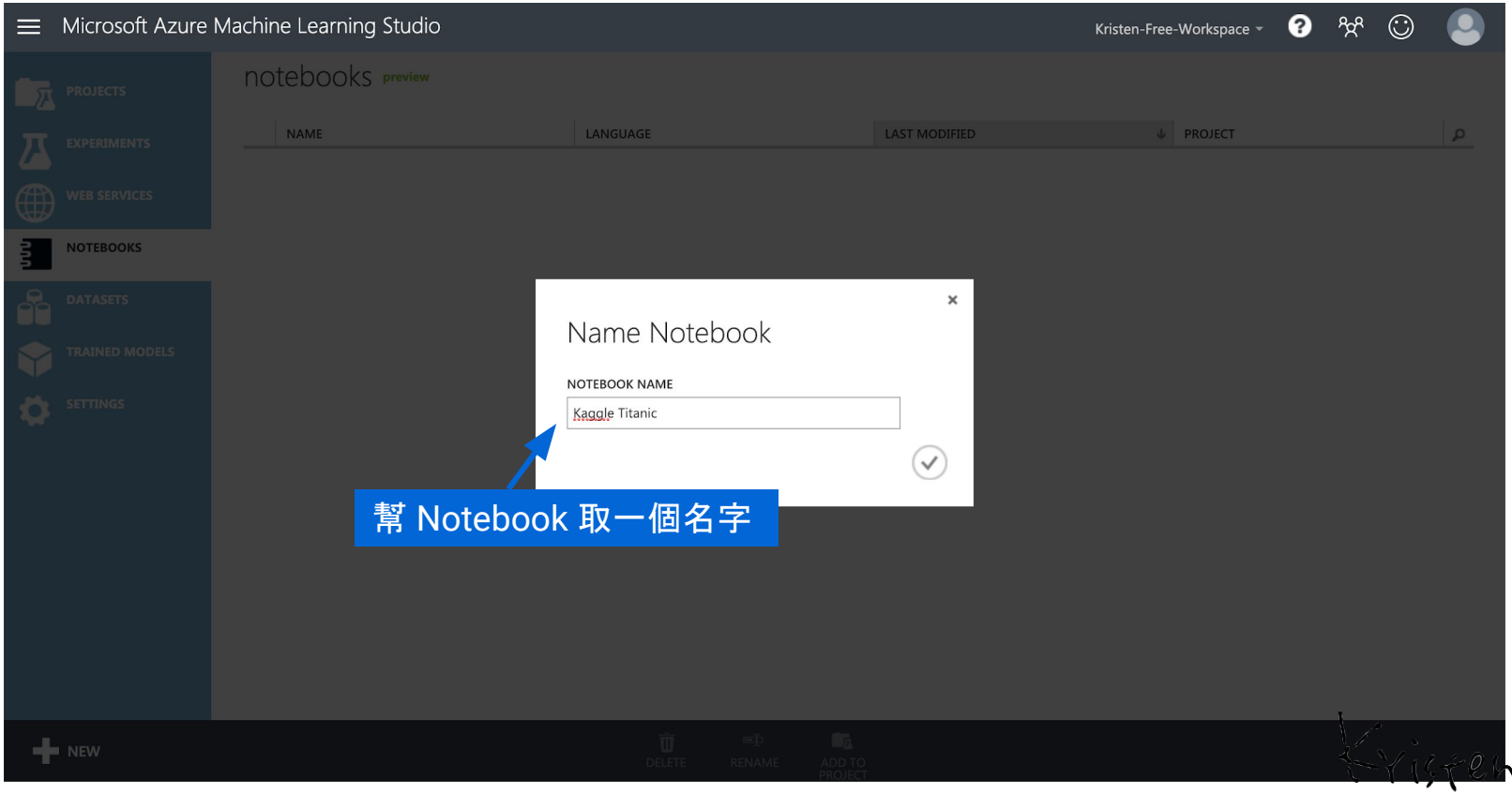
建立一個 Notebook



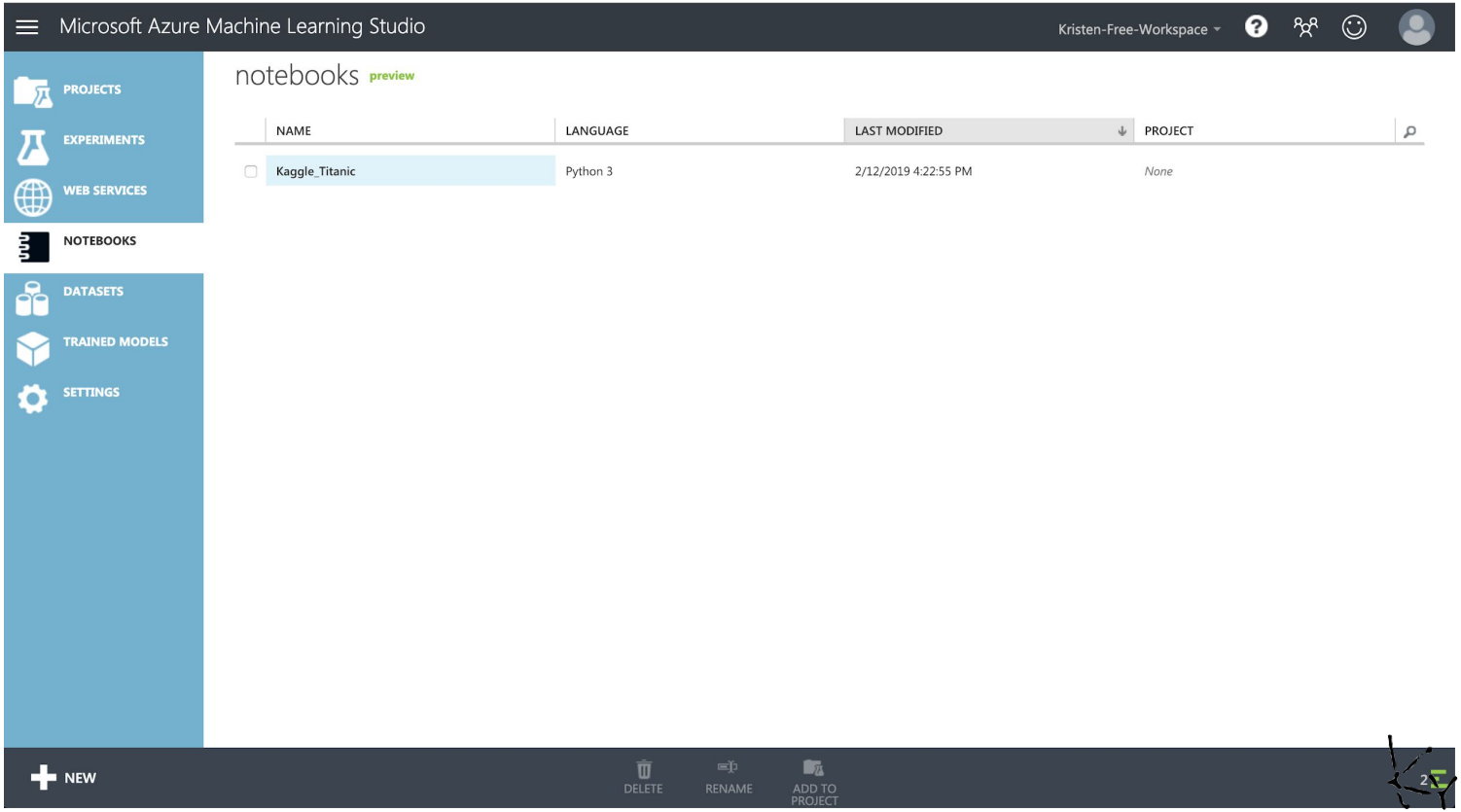
建立一個 Python3 Notebook



建立一個 Python3 Notebook



建立一個 Python3 Notebook



匯入 Package

In [1]:
```
# Warning 不顯示
import warnings
warnings.filterwarnings('ignore')
warnings.filterwarnings('ignore', category=DeprecationWarning)
```

In [2]:
```
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
import seaborn as sns

# 圖在 notebook 中顯示
%matplotlib inline
```

匯入 Data

In [3]:

```
from azureml import Workspace

ws = Workspace()
titanic_train = ws.datasets['Titanic_Train.csv']
titanic_test = ws.datasets['Titanic_Test.csv']

data_train = titanic_train.to_dataframe()
data_test = titanic_test.to_dataframe()
```

從 Workspace 中的資料集

Exploratory Data Analysis

處理遺失值

In [4]:
```
data_train.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
PassengerId    891 non-null int64
Survived       891 non-null int64
Pclass         891 non-null int64
Name           891 non-null object
Sex            891 non-null object
Age            714 non-null float64
SibSp          891 non-null int64
Parch          891 non-null int64
Ticket         891 non-null object
Fare           891 non-null float64
Cabin          204 non-null object
Embarked       889 non-null object
dtypes: float64(2), int64(5), object(5)
memory usage: 83.6+ KB
```

177 個遺失值

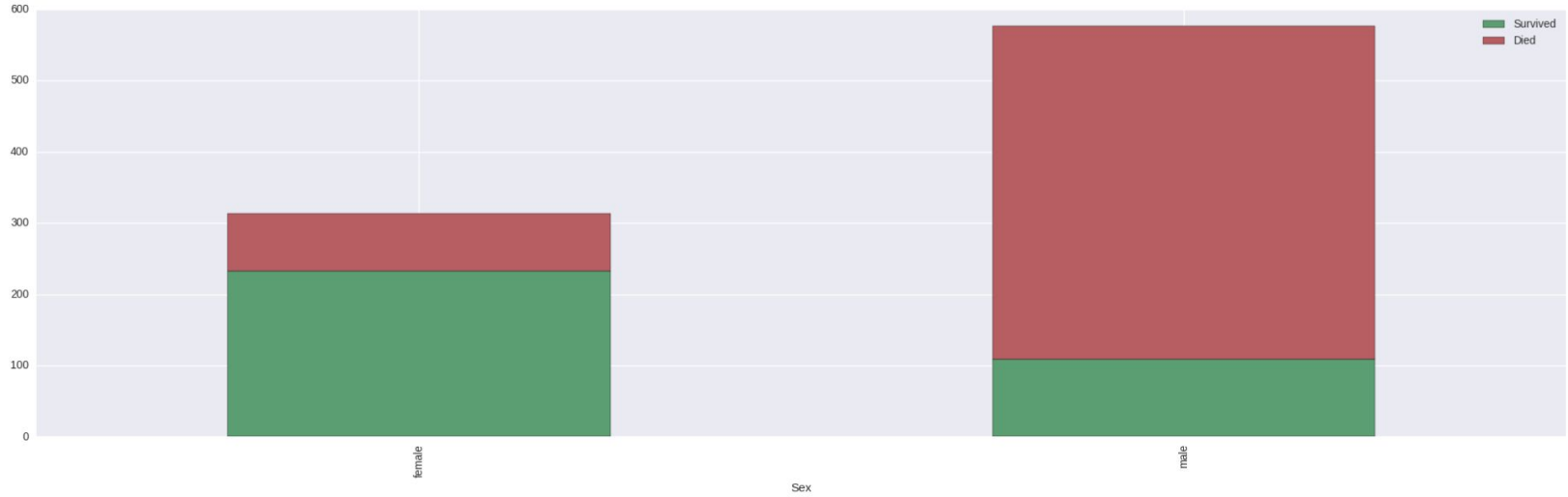
用年齡中位數來做插補

In [5]:
```
# Age : null values with the median age
data_train['Age'] = data_train['Age'].fillna(data_train['Age'].median())
```

Sex VS. Survived (count)

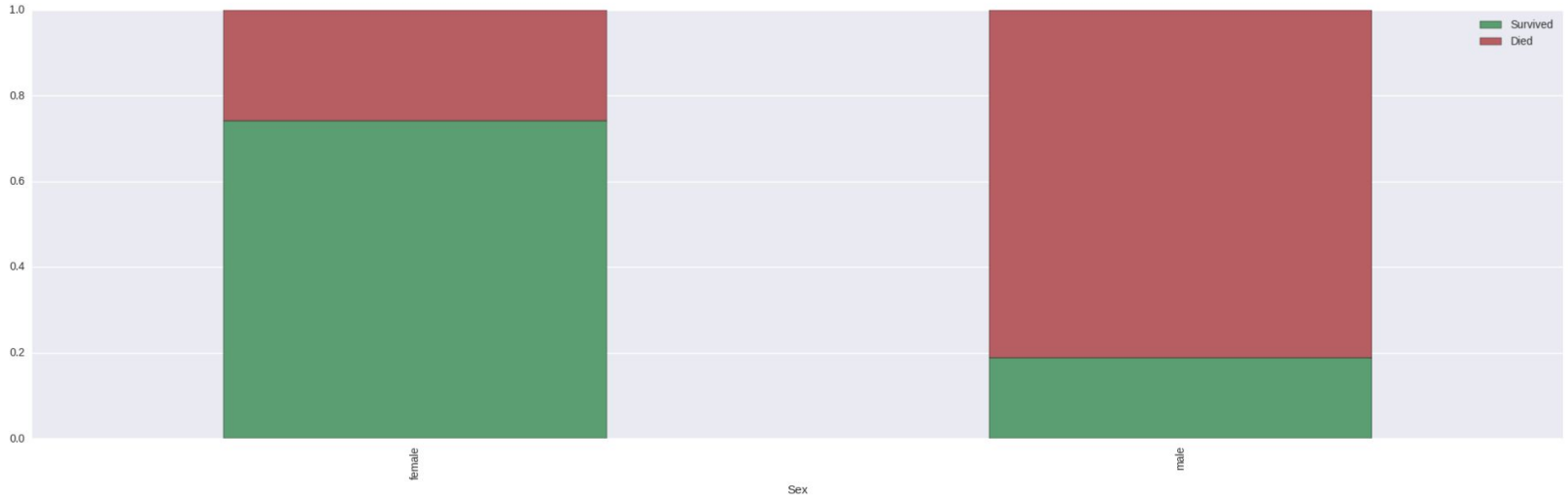
In [6]:
```
# 為了畫圖用建立的變數
data_train['Died'] = 1 - data_train['Survived']
```

In [7]:
```
data_train.groupby('Sex').agg('sum')[['Survived', 'Died']].plot(kind='bar', figsize=(25, 7),
                                                                stacked=True, colors=['g', 'r']);
```



Sex VS. Survived (ratio)

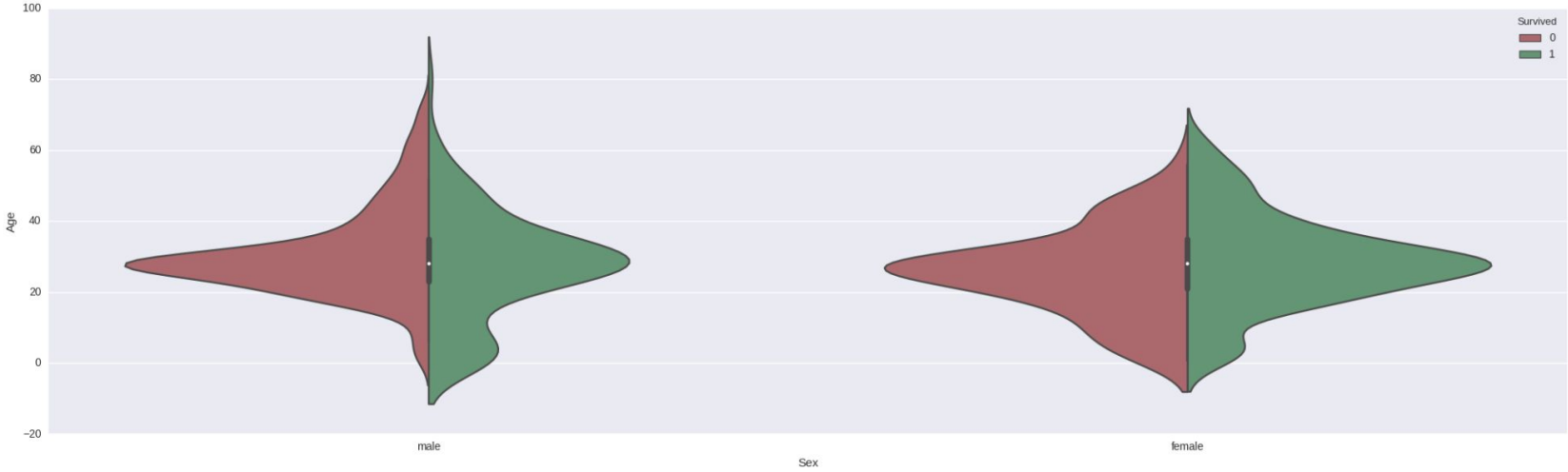
In [8]:
```
data_train.groupby('Sex').agg('mean')[['Survived', 'Died']].plot(kind='bar', figsize=(25, 7),
                                                                 stacked=True, colors=['g', 'r']);
```



Age VS. Sex VS. Survived

In [9]:

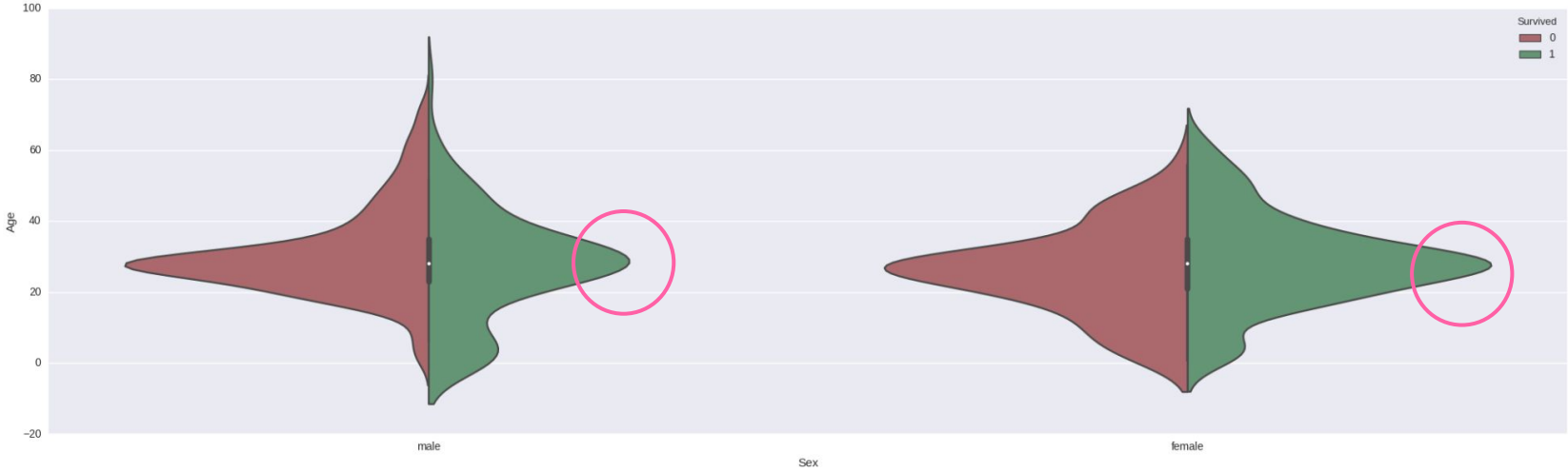
```
fig = plt.figure(figsize = (25, 7))
sns.violinplot(x = 'Sex', y = 'Age',
               hue = 'Survived', data = data_train,
               split = True,
               palette = {0: "r", 1: "g"}
              );
```



Age VS. Sex VS. Survived

In [9]:

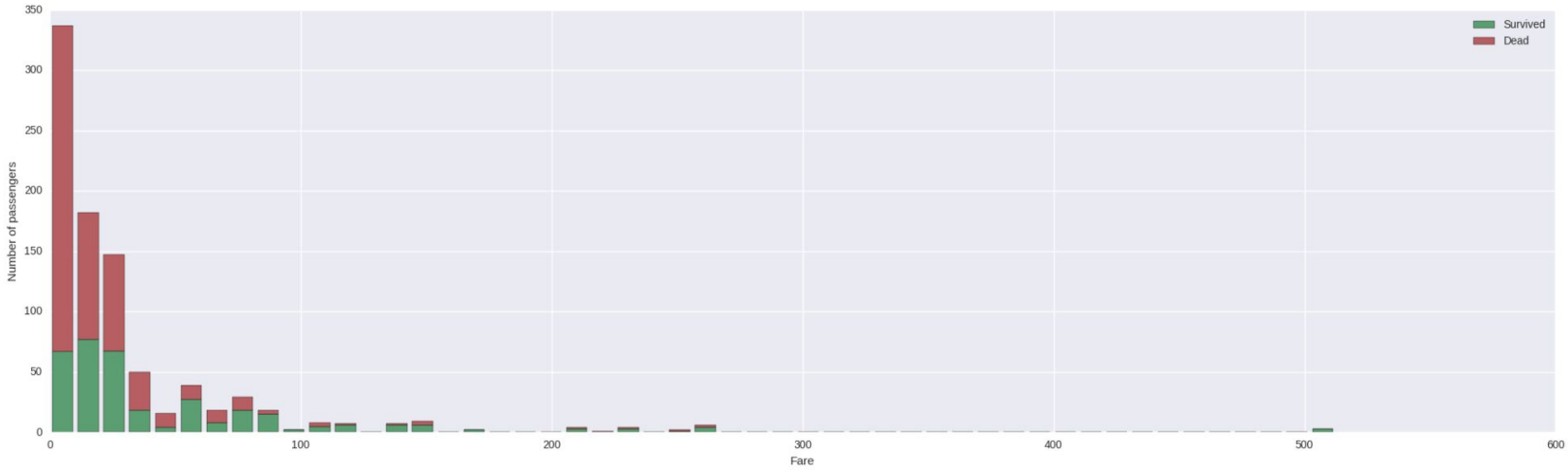
```
fig = plt.figure(figsize = (25, 7))
sns.violinplot(x = 'Sex', y = 'Age',
               hue = 'Survived', data = data_train,
               split = True,
               palette = {0: "r", 1: "g"}
              );
```



Fare VS. Survived

In [10]:

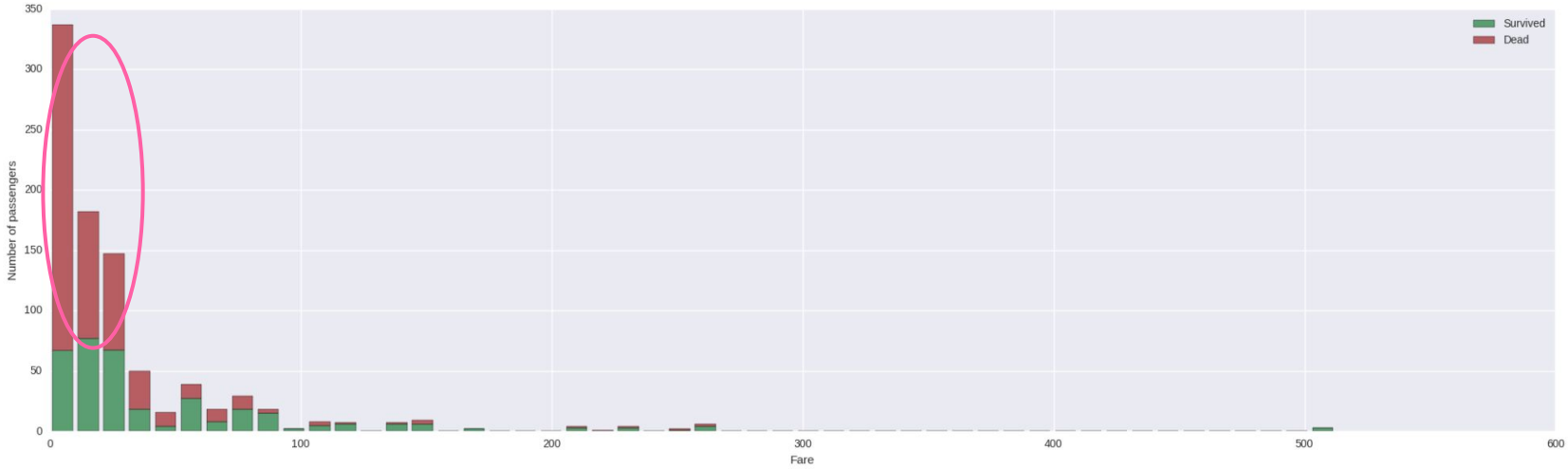
```
figure = plt.figure(figsize = (25, 7))
plt.hist([data_train[data_train['Survived'] == 1]['Fare'], data_train[data_train['Survived'] == 0]['Fare']],
         stacked = True, color = ['g','r'],
         bins = 50, label = ['Survived','Dead'])
plt.xlabel('Fare')
plt.ylabel('Number of passengers')
plt.legend();
```



Fare VS. Survived

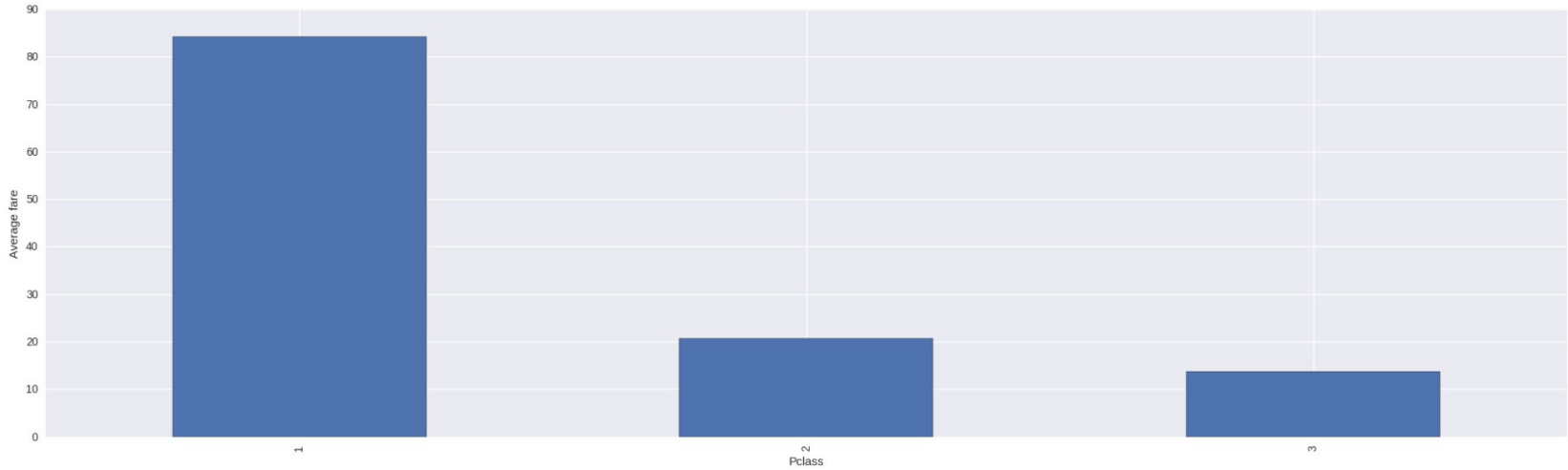
In [10]:

```
figure = plt.figure(figsize = (25, 7))
plt.hist([data_train[data_train['Survived'] == 1]['Fare'], data_train[data_train['Survived'] == 0]['Fare']],
         stacked = True, color = ['g','r'],
         bins = 50, label = ['Survived','Dead'])
plt.xlabel('Fare')
plt.ylabel('Number of passengers')
plt.legend();
```



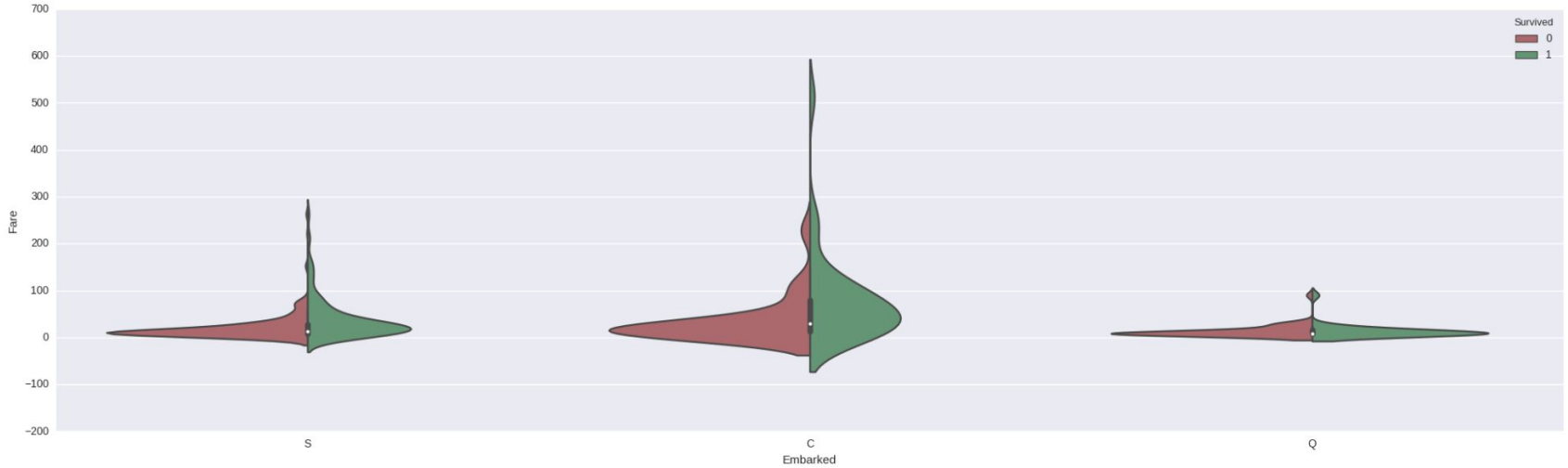
Fare VS. Survived

In [11]:
```
ax = plt.subplot()
ax.set_ylabel('Average fare')
data_train.groupby('Pclass').mean()['Fare'].plot(kind = 'bar', figsize = (25, 7), ax = ax);
```



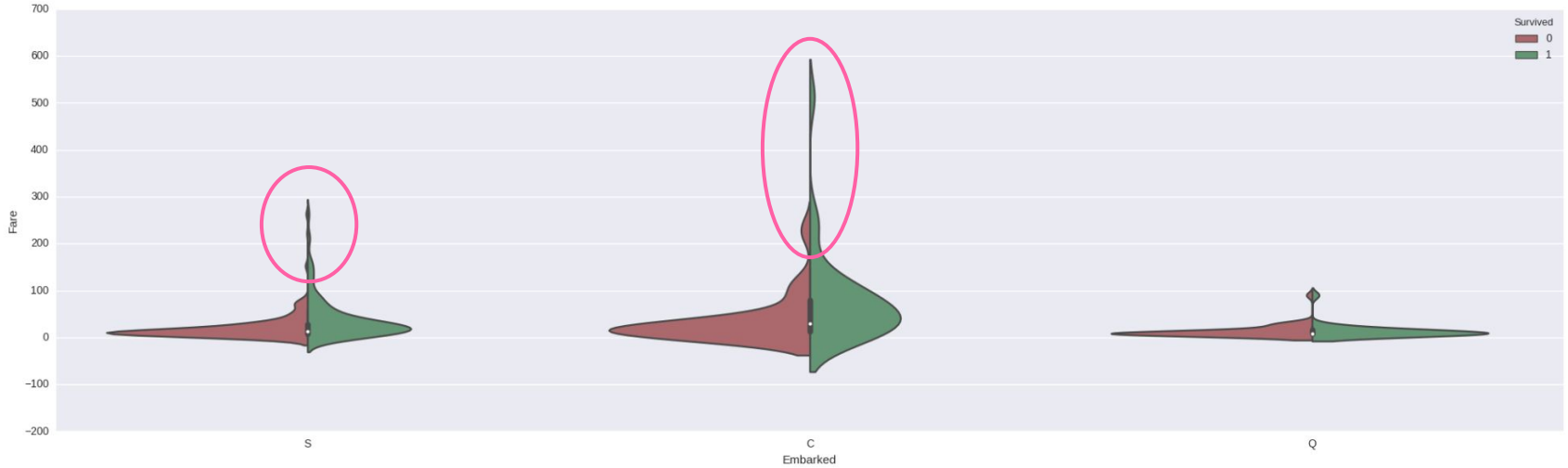
Fare VS. Survived

In [12]:
```
fig = plt.figure(figsize = (25, 7))
sns.violinplot(x = 'Embarked', y = 'Fare', hue = 'Survived', data = data_train, split = True,
               palette = {0: "r", 1: "g"});
```



Embarked VS. Fare VS. Survived

In [12]:
```
fig = plt.figure(figsize = (25, 7))
sns.violinplot(x = 'Embarked', y = 'Fare', hue = 'Survived', data = data_train, split = True,
               palette = {0: "r", 1: "g"});
```



Feature Engineering

合併訓練和測試樣本

In [13]:
```
data_train = titanic_train.to_dataframe()
data_test = titanic_test.to_dataframe()
```

In [14]:
```
targets = data_train.Survived
```

In [15]:
```
data_train.drop(['Survived'], 1, inplace = True)

data_combine = data_train.append(data_test)
data_combine.reset_index(inplace = True)
data_combine.drop(['index', 'PassengerId'], inplace = True, axis = 1)
```

In [16]:
```
data_combine.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1309 entries, 0 to 1308
Data columns (total 11 columns):
Age          1223 non-null float64
Cabin        295 non-null object
Died         891 non-null float64
Embarked     1307 non-null object
Fare         1308 non-null float64
Name         1309 non-null object
Parch        1309 non-null int64
Pclass       1309 non-null int64
Sex          1309 non-null object
SibSp        1309 non-null int64
Ticket       1309 non-null object
dtypes: float64(3), int64(3), object(5)
memory usage: 112.6+ KB
```

891 + 418 = 1309

Passenger Name

Name
Braund, Mr. Owen Harris
Cumings, Mrs. John Bradley (Florence Briggs Th...
Heikkinen, Miss. Laina
Futrelle, Mrs. Jacques Heath (Lily May Peel)
Allen, Mr. William Henry
Moran, Mr. James
McCarthy, Mr. Timothy J
Palsson, Master. Gosta Leonard
Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)
Nasser, Mrs. Nicholas (Adele Achem)

In [17]:

```
titles = set()
for name in data_combine['Name']:
    titles.add(name.split(',')[1].split('.')[0].strip())

print(titles)
```

```
{'Capt',
 'Col',
 'Don',
 'Dona',
 'Dr',
 'Jonkheer',
 'Lady',
 'Major',
 'Master',
 'Miss',
 'Mlle',
 'Mme',
 'Mr',
 'Mrs',
 'Ms',
 'Rev',
 'Sir',
 'the Countess'}
```

Passenger Name

In [18]:
```
Title_Dictionary = {
    "Capt" : "Officer",
    "Col" : "Officer",
    "Major" : "Officer",
    "Jonkheer" : "Royalty",
    "Don" : "Royalty",
    "Dona" : "Royalty",
    "Sir" : "Royalty",
    "Dr" : "Officer",
    "Rev" : "Officer",
    "the Countess" : "Royalty",
    "Mme" : "Mrs",
    "Mlle" : "Miss",
    "Ms" : "Mrs",
    "Mr" : "Mr",
    "Mrs" : "Mrs",
    "Miss" : "Miss",
    "Master" : "Master",
    "Lady" : "Royalty"
}
```

- Officer
- Royalty
- Mr
- Mrs
- Miss
- Master

In [19]:
```
# Split Name
data_combine['Title'] = data_combine['Name'].map(lambda name:name.split(',')[1].split('.')[0].strip())

# Mapping new Title
data_combine['Title'] = data_combine.Title.map(Title_Dictionary)
```

Passenger Name

In [20]:
```
# Encoding in dummy variable
dummy_title = pd.get_dummies(data_combine['Title'], prefix = 'Title')
data_combine = pd.concat([data_combine, dummy_title], axis = 1 )
```

In [21]:
```
data_combine.head()
```

Dummy Variable

Out[21]:

	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked	Title	Title_Master	Title_Miss	Title_Mr	Title_Mrs	Title_Officer	Title_Royalty
0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S	Mr	0	0	1	0	0	0
1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C	Mrs	0	0	0	1	0	0
2	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S	Miss	0	1	0	0	0	0
3	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S	Mrs	0	0	0	1	0	0
4	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S	Mr	0	0	1	0	0	0

Passenger Age

In [22]:
```
print(data_combine['Age'].isnull().sum())
```

```
263
```

train + test 的 missing 共有 263

In [23]:
```
grouped_train = data_combine.iloc[:891].groupby(['Sex','Pclass','Title'])
grouped_train_median = grouped_train.median()
grouped_train_median = grouped_train_median.reset_index()[['Sex', 'Pclass', 'Title', 'Age']]
print(grouped_train_median)
```

	Sex	Pclass	Title	Age
0	female	1	Miss	30.0
1	female	1	Mrs	40.0
2	female	1	Officer	49.0
3	female	1	Royalty	40.5
4	female	2	Miss	24.0
5	female	2	Mrs	31.5
6	female	3	Miss	18.0
7	female	3	Mrs	31.0
8	male	1	Master	4.0
9	male	1	Mr	40.0
10	male	1	Officer	51.0
11	male	1	Royalty	40.0
12	male	2	Master	1.0
13	male	2	Mr	31.0
14	male	2	Officer	46.5
15	male	3	Master	4.0
16	male	3	Mr	26.0

依照 Sex, Pclass, Title 分組計算年齡的中位數

Passenger Age

In [22]:
```
print(data_combine['Age'].isnull().sum())
```

```
263
```

In [23]:
```
grouped_train = data_combine.iloc[:891].groupby(['Sex','Pclass','Title'])
grouped_train_median = grouped_train.median()
grouped_train_median = grouped_train_median.reset_index()[['Sex', 'Pclass', 'Title', 'Age']]
print(grouped_train_median)
```

使用 Training Data

	Sex	Pclass	Title	Age
0	female	1	Miss	30.0
1	female	1	Mrs	40.0
2	female	1	Officer	49.0
3	female	1	Royalty	40.5
4	female	2	Miss	24.0
5	female	2	Mrs	31.5
6	female	3	Miss	18.0
7	female	3	Mrs	31.0
8	male	1	Master	4.0
9	male	1	Mr	40.0
10	male	1	Officer	51.0
11	male	1	Royalty	40.0
12	male	2	Master	1.0
13	male	2	Mr	31.0
14	male	2	Officer	46.5
15	male	3	Master	4.0
16	male	3	Mr	26.0

依照 Sex, Pclass, Title 分組計算年齡的中位數

Passenger Age

In [24]:
```
def fill_age(data):
    fill_id = ( (grouped_train_median['Sex'] == data['Sex']) &
                (grouped_train_median['Title'] == data['Title']) &
                (grouped_train_median['Pclass'] == data['Pclass']) )
    return grouped_train_median[fill_id]['Age'].values[0]
```

In [25]:
```
data_combine['Age'] = data_combine.apply(lambda data: fill_age(data)
                                                      if np.isnan(data['Age']) else data['Age'], axis=1)
```

遇到遺失值，去找對應的組別補上值

Fare

In [26]:
```
print(data_combine['Fare'].isnull().sum())
```

```
1
```

In [27]:
```
data_combine.Fare.fillna(combined.iloc[:891].Fare.mean(), inplace = True)
```

補平均票價

Embarked

In [28]:
```
print(data_combine['Embarked'].isnull().sum())
```

```
2
```

In [29]:
```
data_combine.loc[:891,'Embarked'].value_counts()
```

Out[29]:
```
S    644
C    168
Q     78
Name: Embarked, dtype: int64
```

In [30]:
```
data_combine.Embarked.fillna('S', inplace = True)
```

In [31]:
```
# Encoding in dummy variable
dummy_embarked = pd.get_dummies(data_combine['Embarked'], prefix = 'Embarked')
data_combine = pd.concat([data_combine, dummy_embarked], axis = 1 )
```

- 補出現最多的港口
- Add Dummy Variable

Cabin

In [32]:
```
print(data_combine['Cabin'].isnull().sum())
```

```
1014
```

In [33]:
```
cabins = set()
for c in data_combine.iloc[:891]['Cabin']:
    cabins.add(c)

print(cabins)
```

```
{nan, 'A32', 'E36', 'A16', 'D47', 'B49', 'F2', 'B4', 'D56', 'E50', 'A14', 'D45', 'D9', 'F G63', 'C110', 'C99', 'B96 B
98', 'C62 C64', 'D26', 'C86', 'B41', 'C54', 'C22 C26', 'C124', 'C128', 'B28', 'C82', 'B82 B84', 'F33', 'C106', 'B39',
'B5', 'D15', 'E17', 'B86', 'B37', 'D11', 'D28', 'A23', 'G6', 'E34', 'C47', 'B42', 'B71', 'D6', 'B77', 'C103', 'B35',
'D33', 'C101', 'B30', 'D7', 'A19', 'C45', 'B3', 'C49', 'F4', 'B18', 'B73', 'D19', 'C50', 'C65', 'B58 B60', 'C93', 'F3
8', 'D37', 'E67', 'C148', 'E58', 'B80', 'B102', 'A7', 'E25', 'C85', 'C87', 'C123', 'E40', 'D30', 'A26', 'C52', 'D36',
'D10 D12', 'C90', 'E68', 'B22', 'B94', 'E63', 'E38', 'A36', 'B57 B59 B63 B66', 'E44', 'A20', 'A34', 'E12', 'C2', 'D4
8', 'E33', 'C30', 'B51 B53 B55', 'C118', 'D49', 'C78', 'D', 'C111', 'C83', 'E46', 'C95', 'F G73', 'A5', 'E8', 'C70',
'C23 C25 C27', 'C125', 'E24', 'B20', 'B69', 'A24', 'E77', 'A6', 'C126', 'B38', 'C68', 'E31', 'C91', 'A10', 'E49', 'D2
1', 'E10', 'D46', 'C32', 'D50', 'C7', 'E121', 'B78', 'C92', 'E101', 'A31', 'B101', 'B79', 'B50', 'F E69', 'D20', 'T',
'C104', 'D17', 'B19', 'D35', 'C46'}
```

所有艙號

Cabin

In [34]:
```
# Missing : M
data_combine.Cabin.fillna('M', inplace = True)

# Get Each Cabin First letter
data_combine['Cabin'] = data_combine['Cabin'].map(lambda l: l[0])
```

In [35]:
```
# Encoding in dummy variable
dummy_cabin = pd.get_dummies(data_combine['Cabin'], prefix = 'Cabin')
data_combine = pd.concat([data_combine, dummy_cabin], axis = 1 )
```

- Missing 的用 M 來代表 Missing
- 取 Cabin 的第一個英文字
- Add Dummy Variable

Passenger Sex

In [36]:
```
data_combine['Sex'] = data_combine['Sex'].map({'male':1, 'female':0})
```

Female : 0
Male : 1

Pclass

In [37]:
```
# Encoding in dummy variable
dummy_pclass = pd.get_dummies(data_combine['Pclass'], prefix = 'Pclass')
data_combine = pd.concat([data_combine, dummy_pclass], axis = 1 )
```

Add Dummy Variable

Ticket

	Ticket
0	A/5 21171
1	PC 17599
2	STON/O2. 3101282
3	113803
4	373450
5	330877
6	17463
7	349909
8	347742
9	237736

In [38]:

```
def cleanTicket(ticket):
    ticket = ticket.replace('.', '')
    ticket = ticket.replace('/', '')
    ticket = ticket.split()
    ticket = map(lambda t : t.strip(), ticket)
    ticket = list(filter(lambda t : not t.isdigit(), ticket))
    if len(ticket) > 0:
        return ticket[0]
    else:
        return 'Null'
```

In [39]:

```
tickets = set()
for t in data_combine['Ticket']:
    tickets.add(cleanTicket(t))

print(tickets)
```

```
{'Fa', 'SCA3', 'WEP', 'SCParis', 'AS', 'CASOTON', 'SCOW', 'SOTONOQ', 'SOC', 'STONO', 'Null', 'CA', 'C', 'SCPARIS', 'SP', 'STONO2', 'PPP', 'STONOQ', 'LINE', 'FCC', 'SWPP', 'FC', 'SOTONO2', 'LP', 'PC', 'A', 'AQ3', 'PP', 'SCA4', 'AQ4', 'SCAH', 'SOPP', 'A5', 'A4', 'SOP', 'WC', 'SC'}
```

取出票號前的英文，若沒有則顯示 Null

Ticket

In [40]:
```
data_combine['Ticket'] = data_combine['Ticket'].map(cleanTicket)
```

In [41]:
```
# Encoding in dummy variable
dummy_tickets = pd.get_dummies(data_combine['Ticket'], prefix = 'Ticket')
data_combine = pd.concat([data_combine, dummy_tickets], axis = 1 )
```

Add Dummy Variable

Family

In [42]:

```
data_combine['Family_size'] = data_combine['Parch'] + data_combine['SibSp'] + 1

# Introducing other features based on the family size
data_combine['Single_family'] = data_combine['Family_size'].map(lambda s: 1 if s == 1 else 0)
data_combine['Small_family'] = data_combine['Family_size'].map(lambda s: 1 if 2 <= s <= 4 else 0)
data_combine['Big_family'] = data_combine['Family_size'].map(lambda s: 1 if 5 <= s else 0)
```

新增變數

- Family Size : 兄弟姊妹/配偶 + 父母/小孩
- Single Family : Family Size = 1 ← 獨自一人
- Small Family : Family Size = 2~4 ← 小家庭
- Big Family : Family Size >= 5 ← 大家庭

Final Data

In [43]:
```
data_combine.drop(['Name','Title','Embarked','Cabin','Pclass','Ticket'], axis = 1, inplace = True)
```

In [44]:
```
data_combine.shape
```

Out[44]:
```
(1309, 67)
```

In [45]:
```
data_combine.head()
```

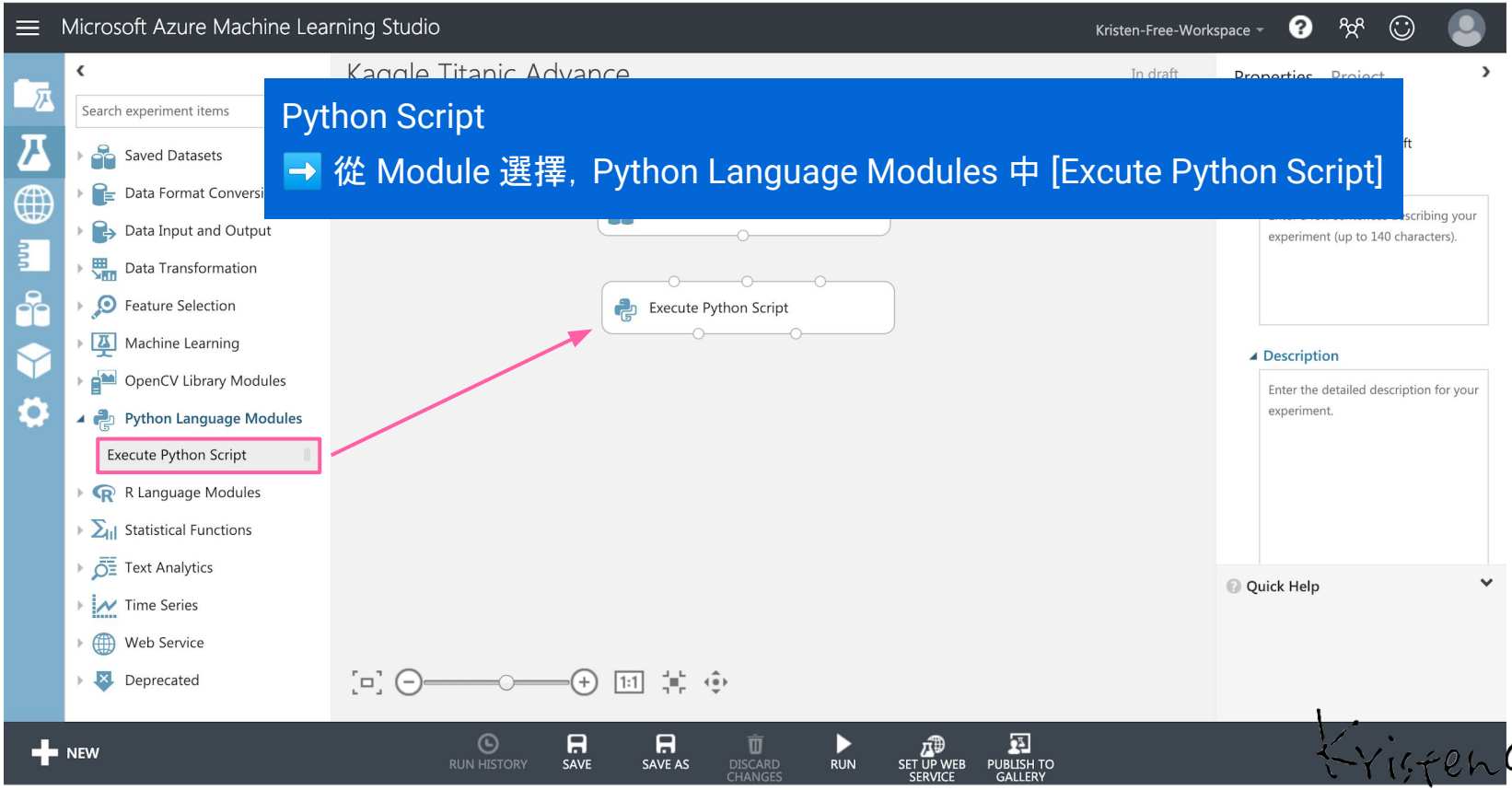
Out[45]:

	Sex	Age	SibSp	Parch	Fare	Title_Master	Title_Miss	Title_Mr	Title_Mrs	Title_Officer	...	Ticket_STONO	Ticket_STONO2	Ticket_STONOQ	Ticket_SW
0	1	22.0	1	0	7.2500	0	0	1	0	0	...	0	0	0	
1	0	38.0	1	0	71.2833	0	0	0	1	0	...	0	0	0	
2	0	26.0	0	0	7.9250	0	1	0	0	0	...	0	1	0	
3	0	35.0	1	0	53.1000	0	0	0	1	0	...	0	0	0	
4	1	35.0	0	0	8.0500	0	0	1	0	0	...	0	0	0	

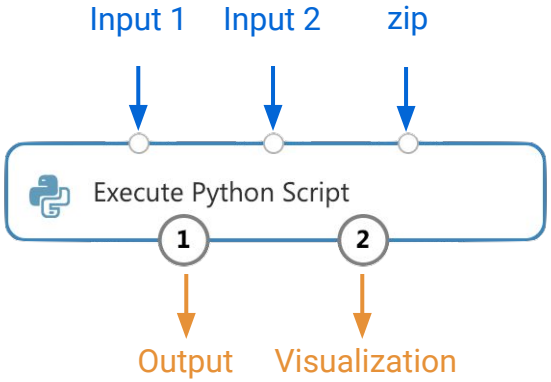
5 rows × 67 columns

Azure ML Studio
Execute Python Script

介紹 Excute Python Script



介紹 Excute Python Script

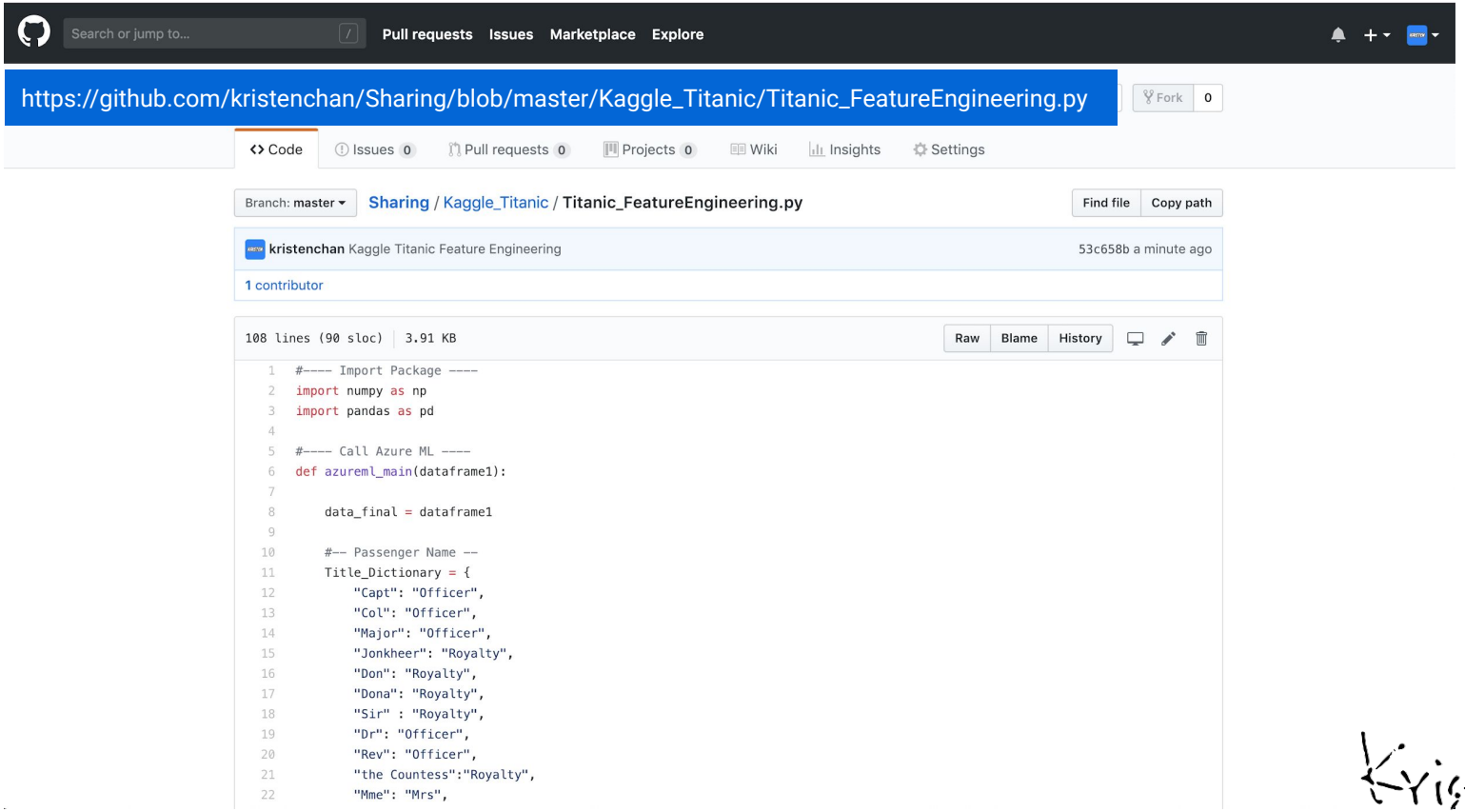


Input 1 Input 2

```
1  import pandas as pd
2
3  def azureml_main(dataframe1,dataframe2):
4
5      result = pd.DataFrame(...)
6
7      return result
8
```

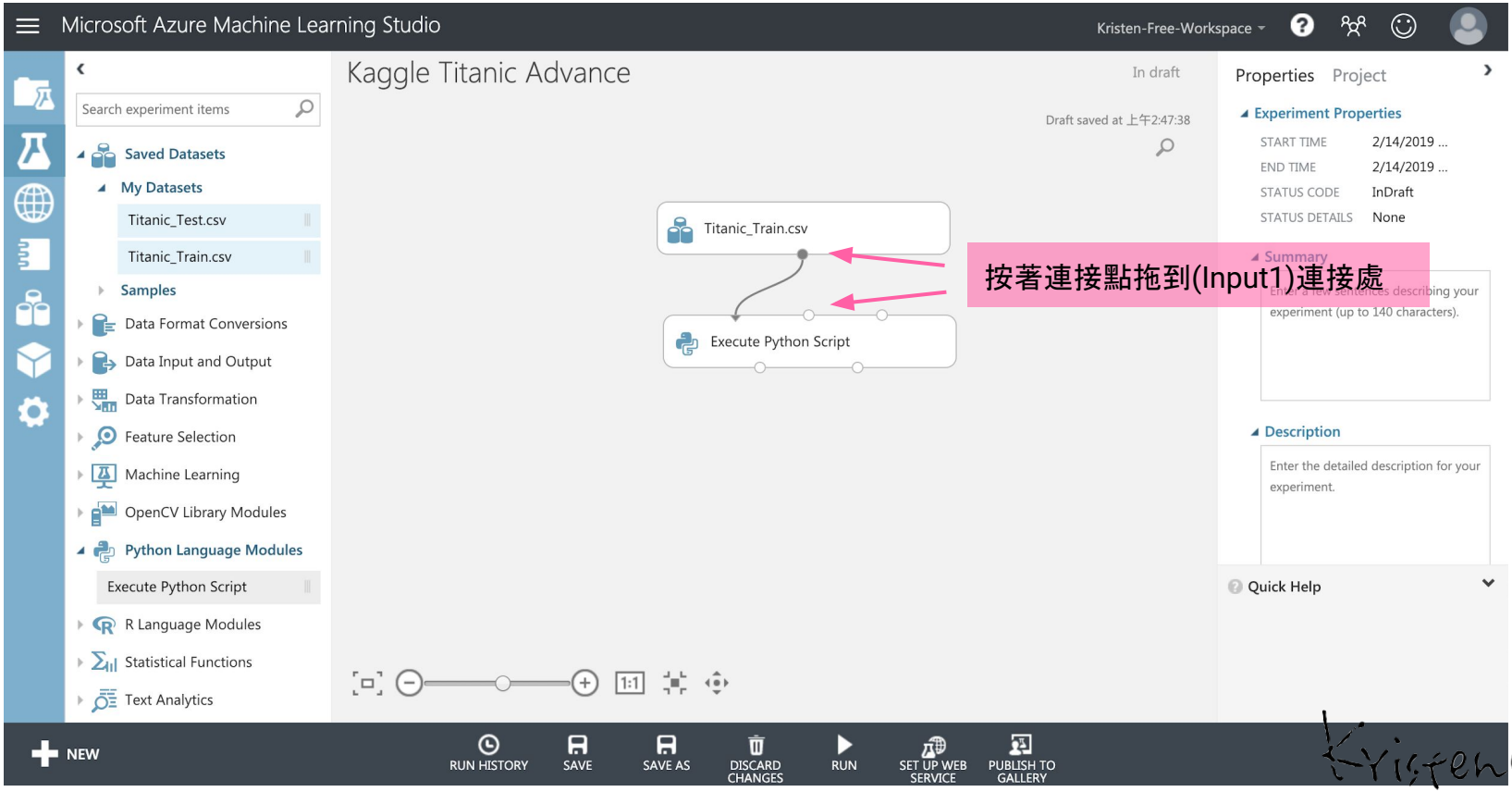
Output

Python Script -- Titanic Feature Engineering

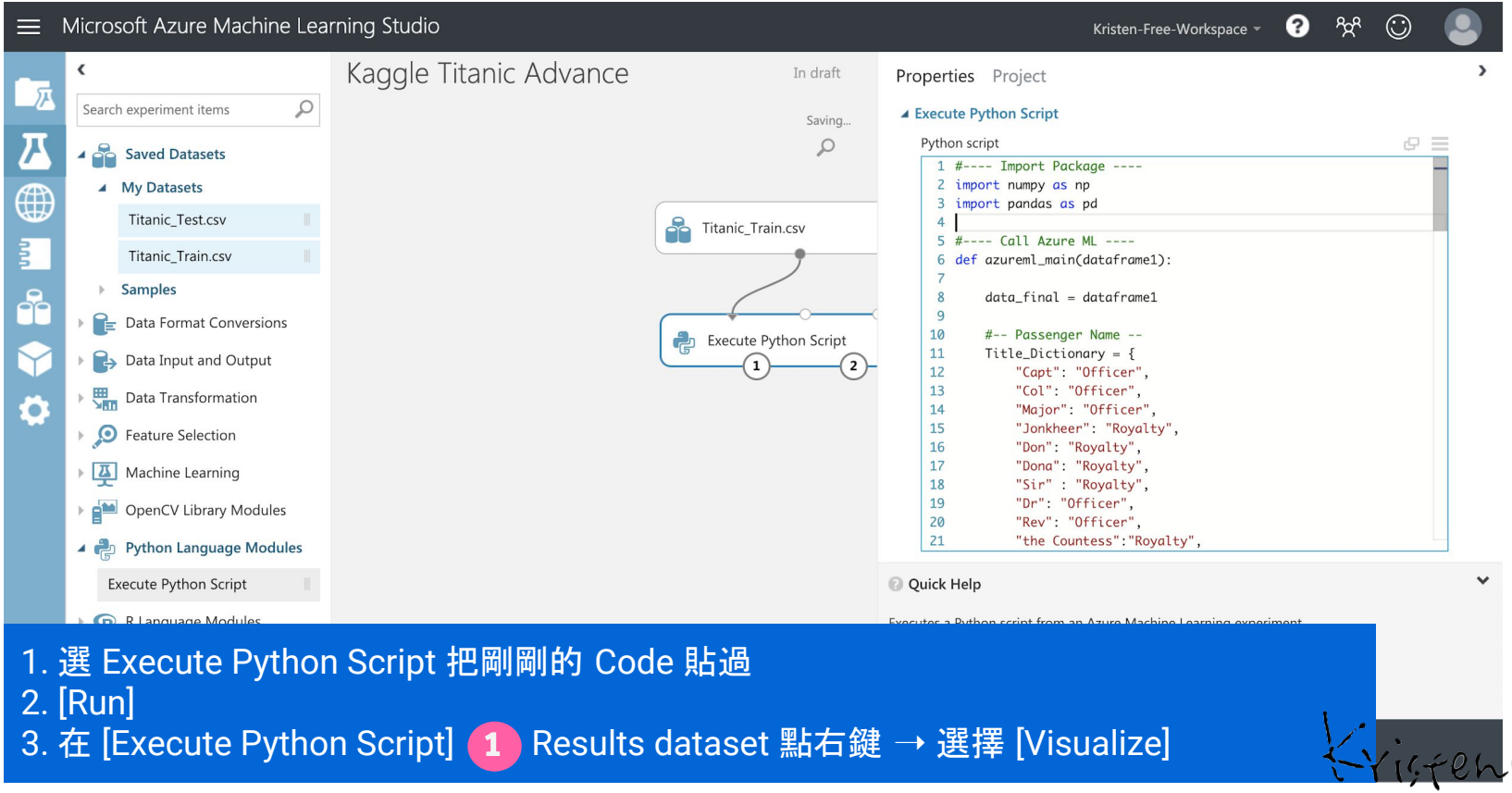



```python
#---- Import Package ----
import numpy as np
import pandas as pd

#---- Call Azure ML ----
def azureml_main(dataframe1):

    data_final = dataframe1

    #-- Passenger Name --
    Title_Dictionary = {
        "Capt": "Officer",
        "Col": "Officer",
        "Major": "Officer",
        "Jonkheer": "Royalty",
        "Don": "Royalty",
        "Dona": "Royalty",
        "Sir" : "Royalty",
        "Dr": "Officer",
        "Rev": "Officer",
        "the Countess":"Royalty",
        "Mme": "Mrs",
```

Excute Python Script

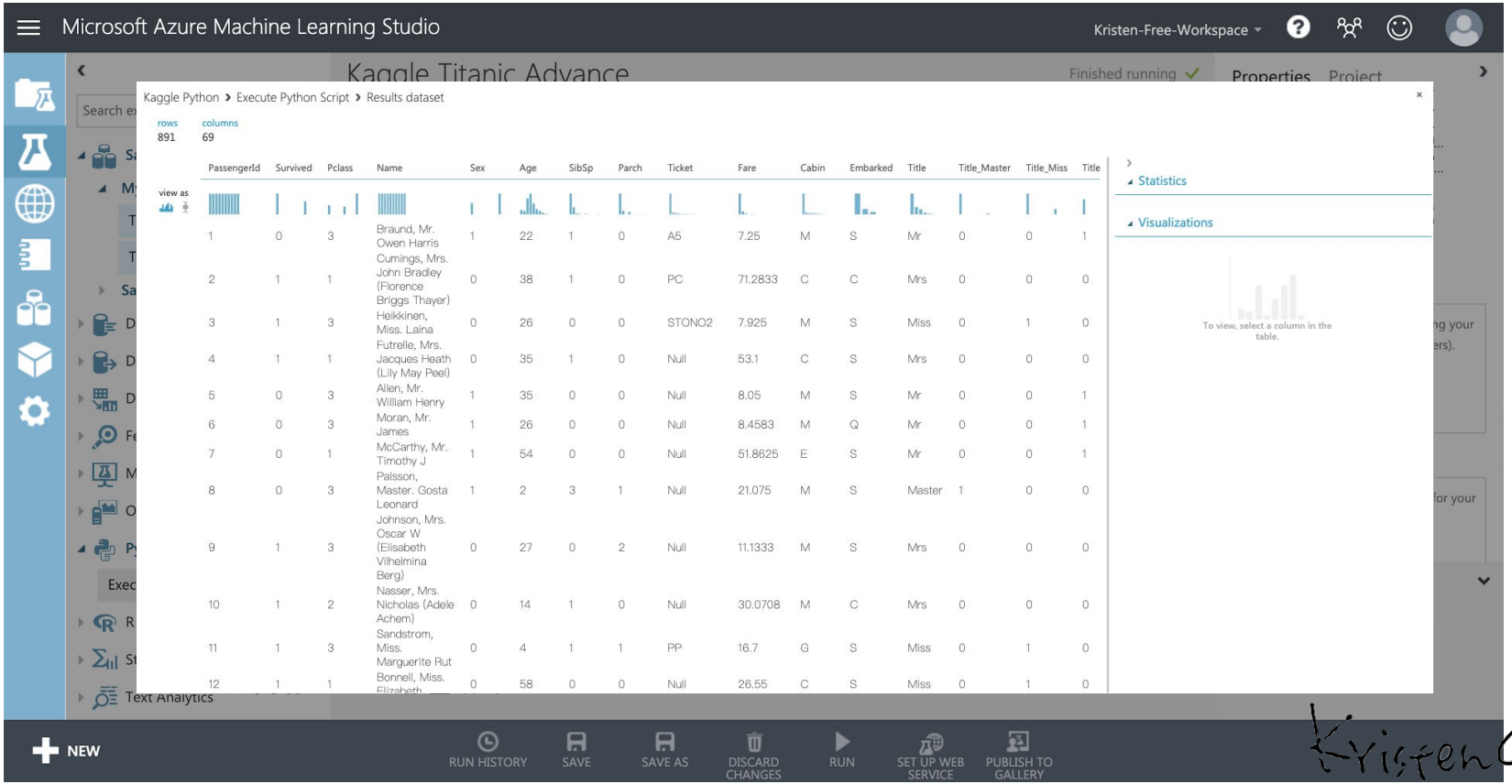


Excute Python Script

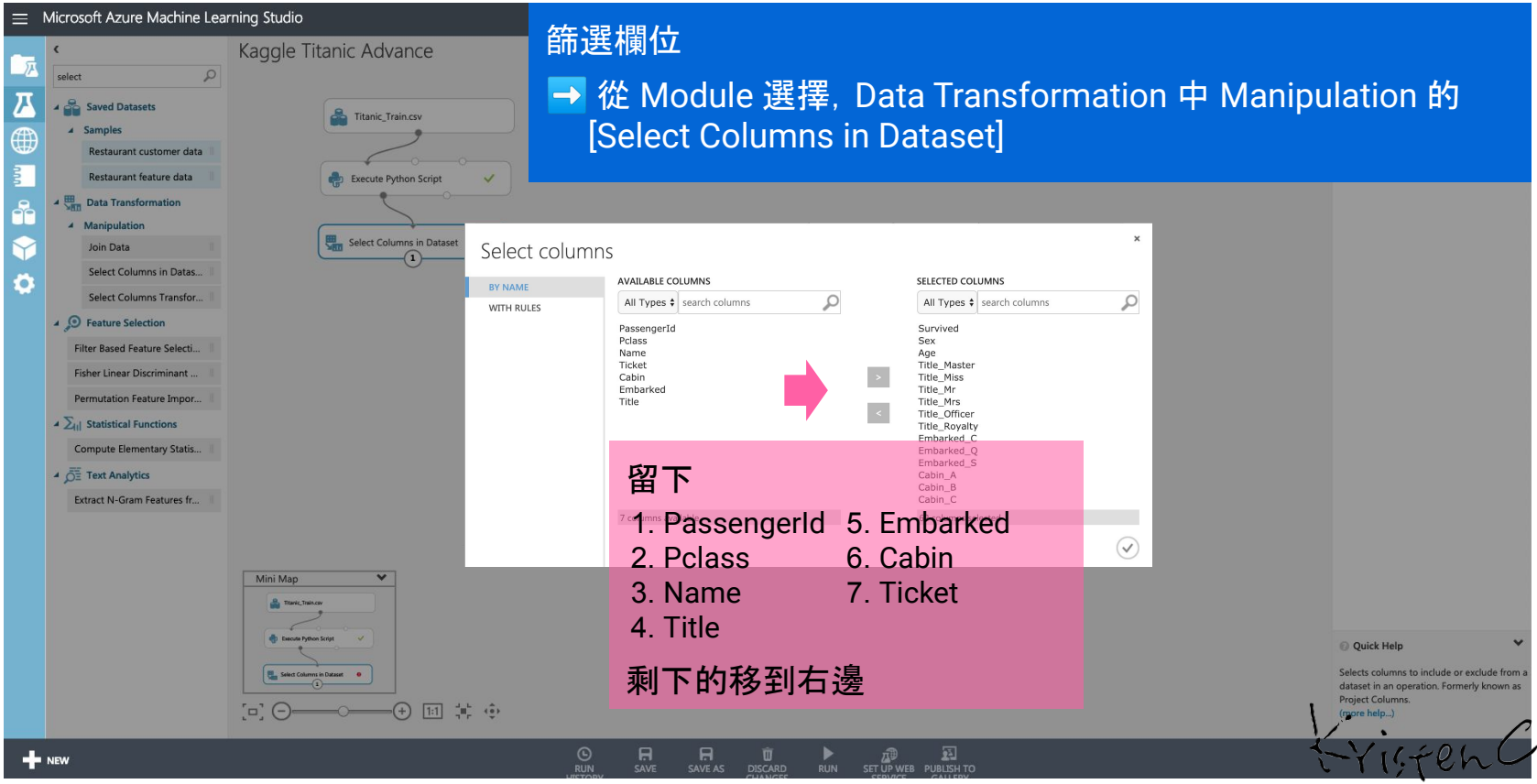


1. 選 Execute Python Script 把剛剛的 Code 貼過
2. [Run]
3. 在 [Execute Python Script] (1) Results dataset 點右鍵 → 選擇 [Visualize]

Excute Python Script



Advanced Version -- Python



篩選欄位

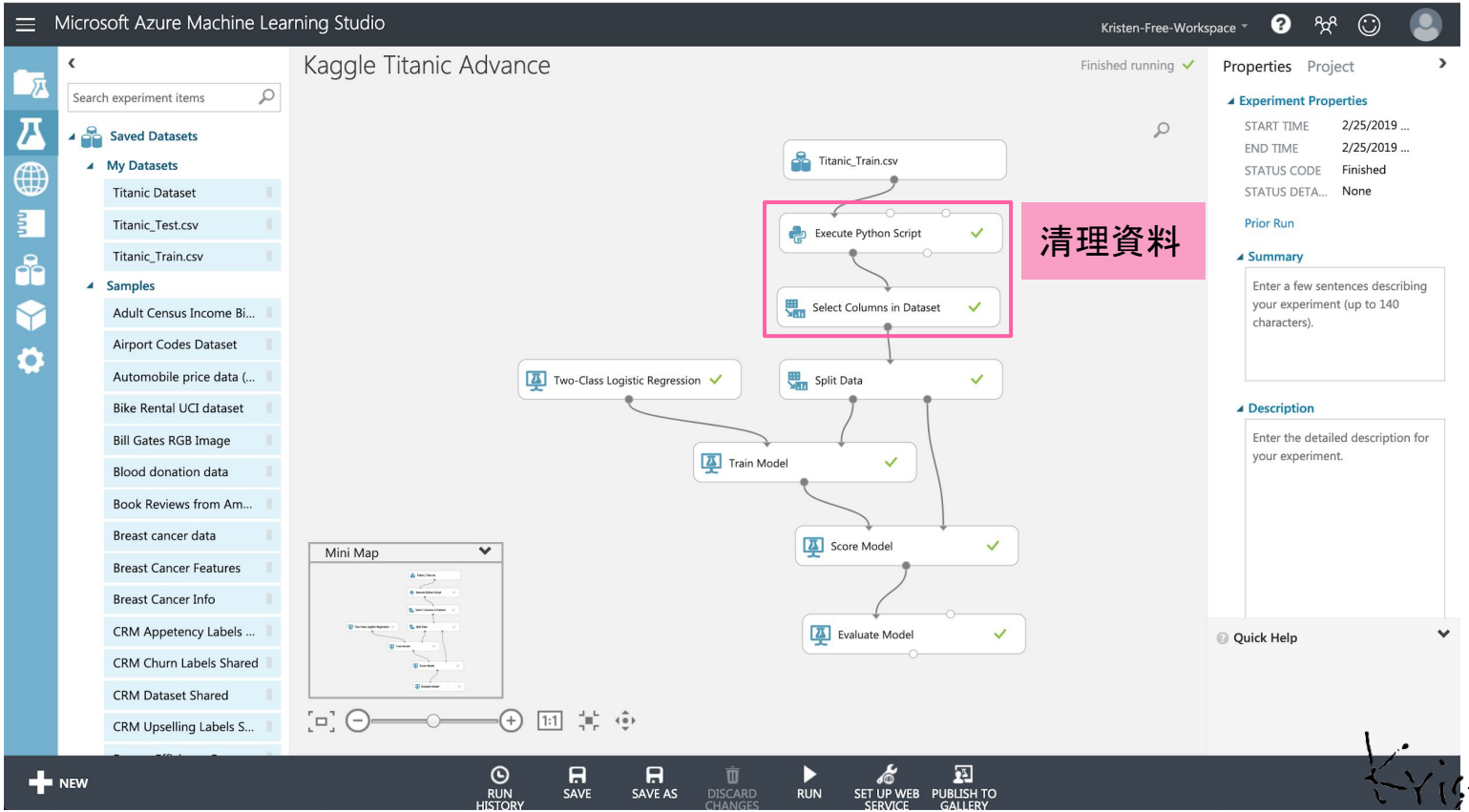
➡️ 從 Module 選擇，Data Transformation 中 Manipulation 的 [Select Columns in Dataset]

留下

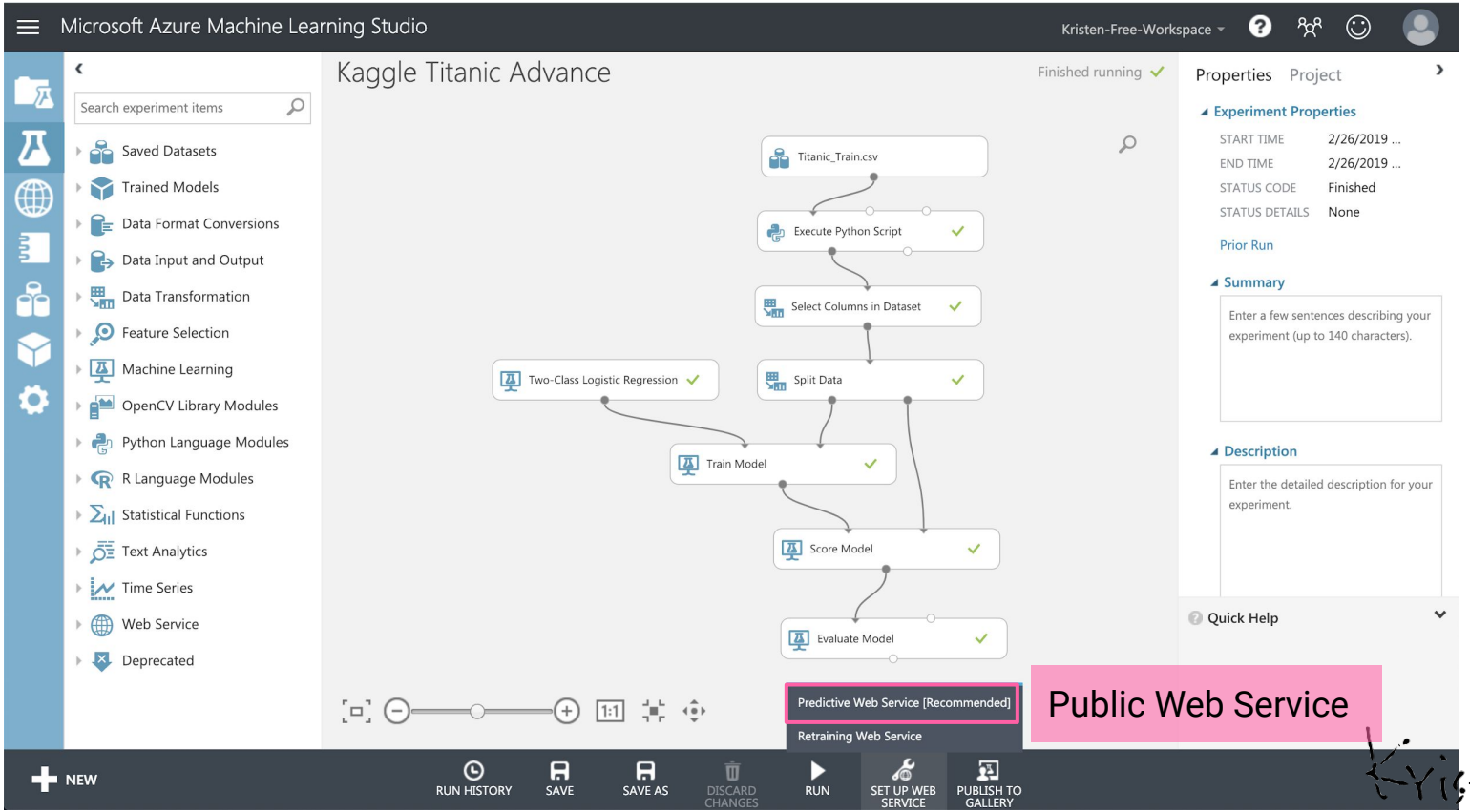
1. PassengerId
2. Pclass
3. Name
4. Title
5. Embarked
6. Cabin
7. Ticket

剩下的移到右邊

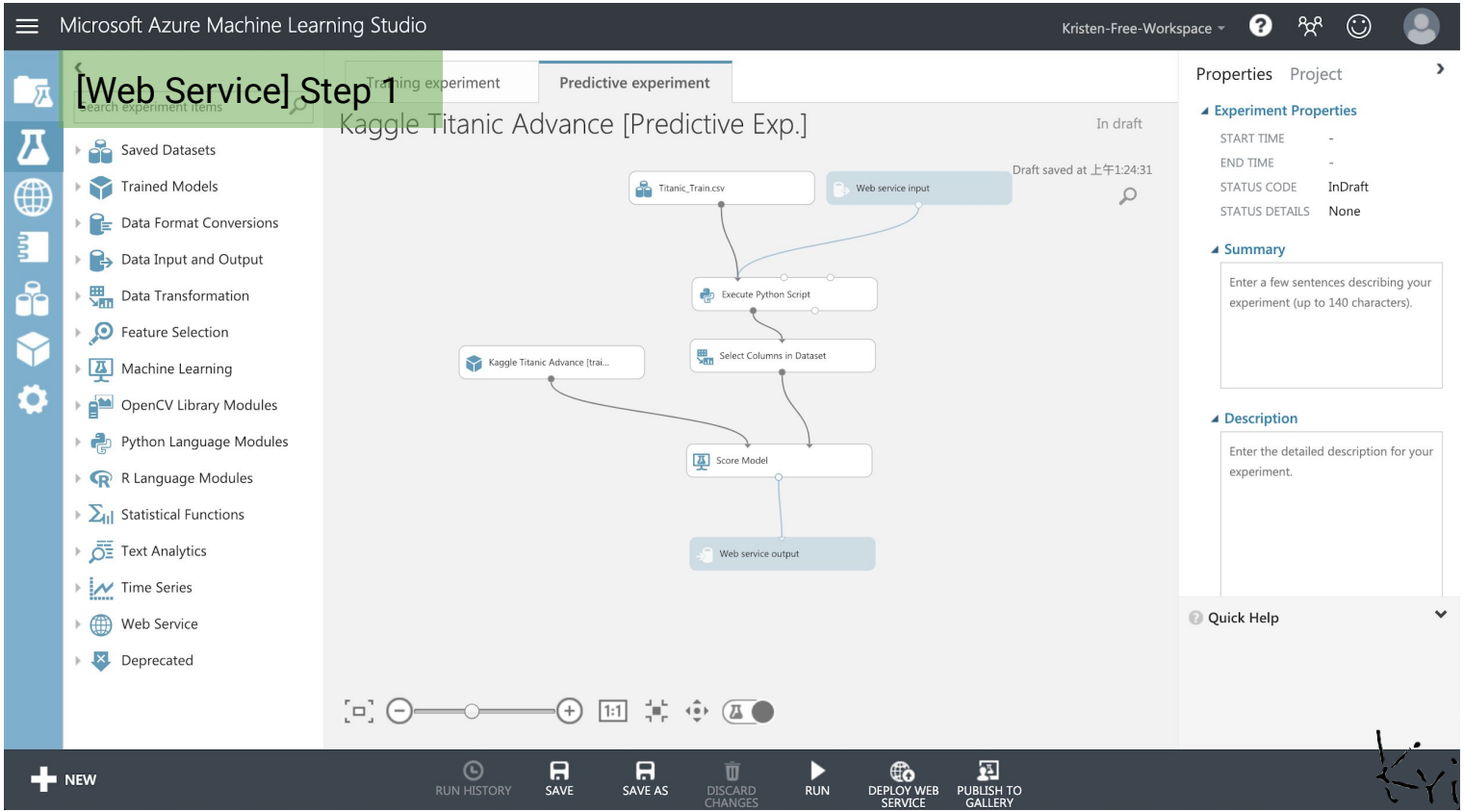
Advanced Version -- Python



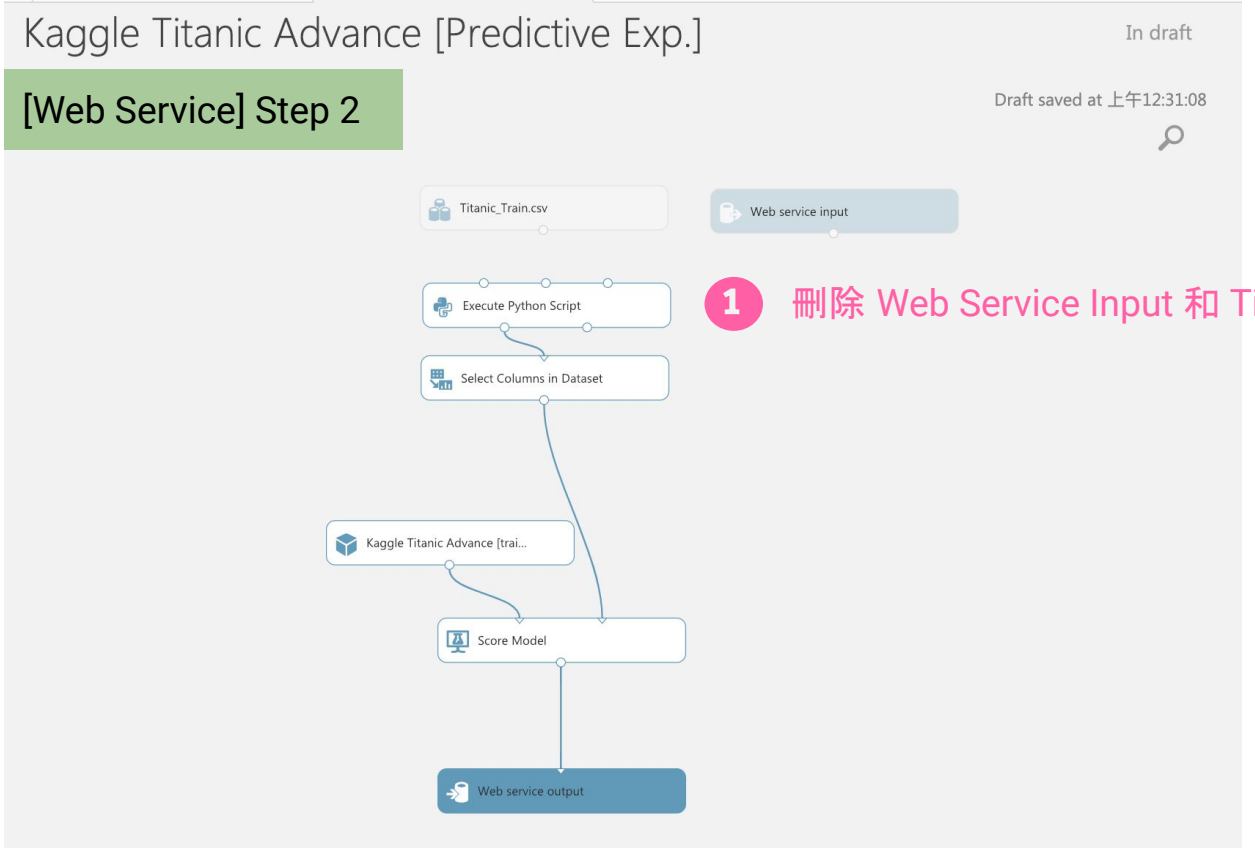
Advanced Version -- Python



Advanced Version -- Python

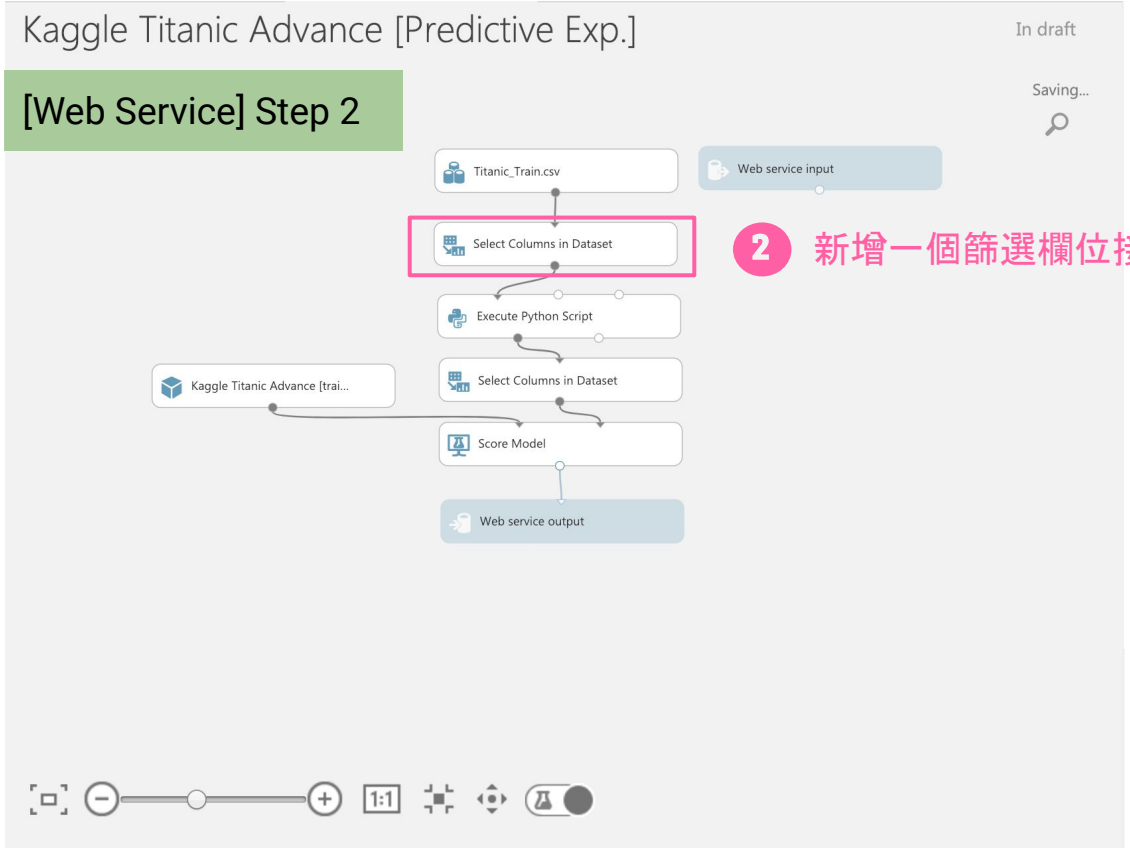


Advanced Version -- Python



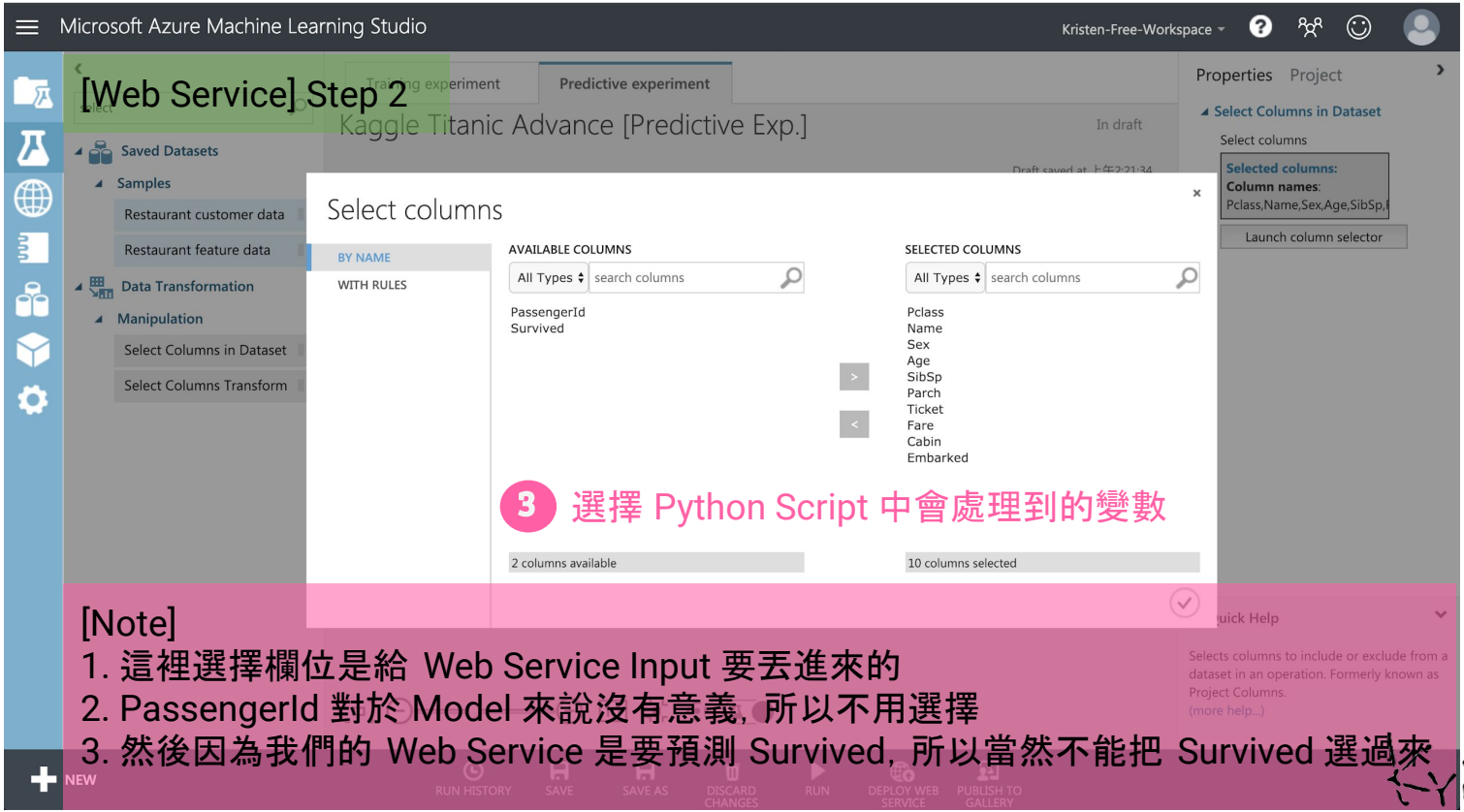
1 刪除 Web Service Input 和 Titanic_Train 的連結

Advanced Version -- Python

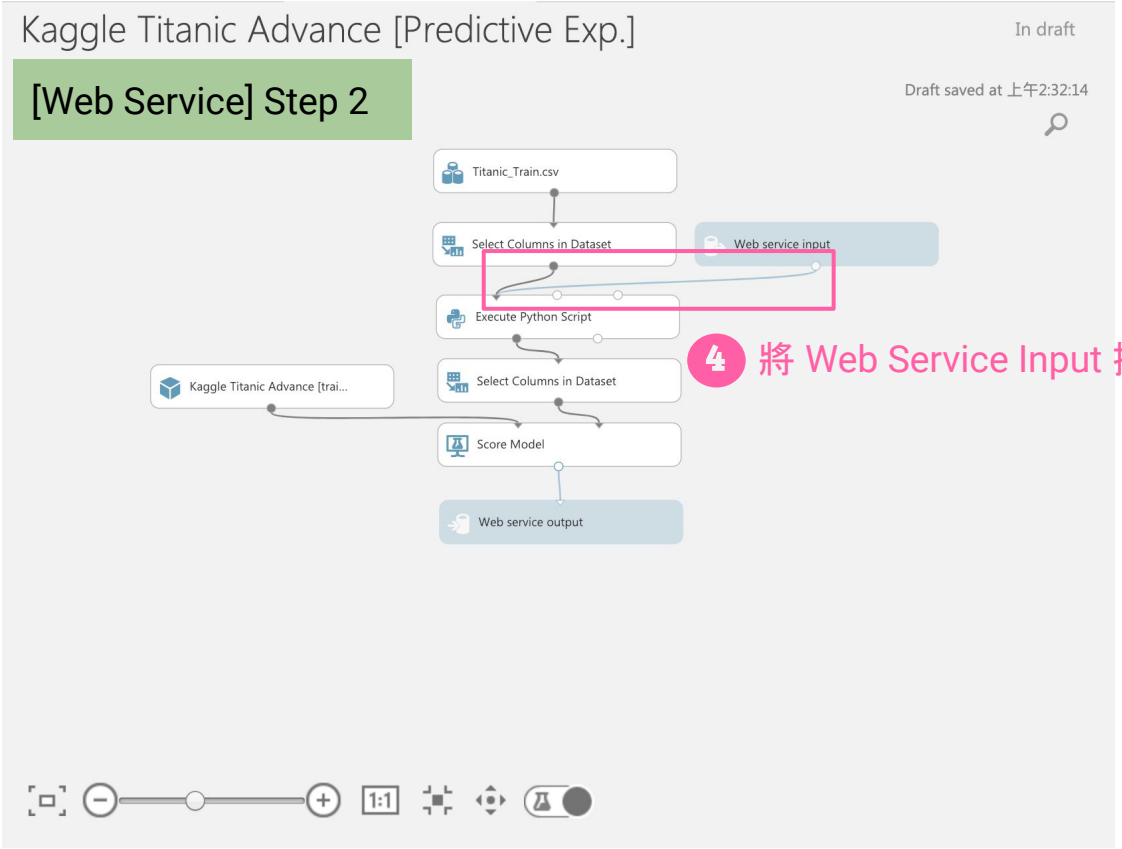


2 新增一個篩選欄位接上 Titanic_Train

Advanced Version -- Python

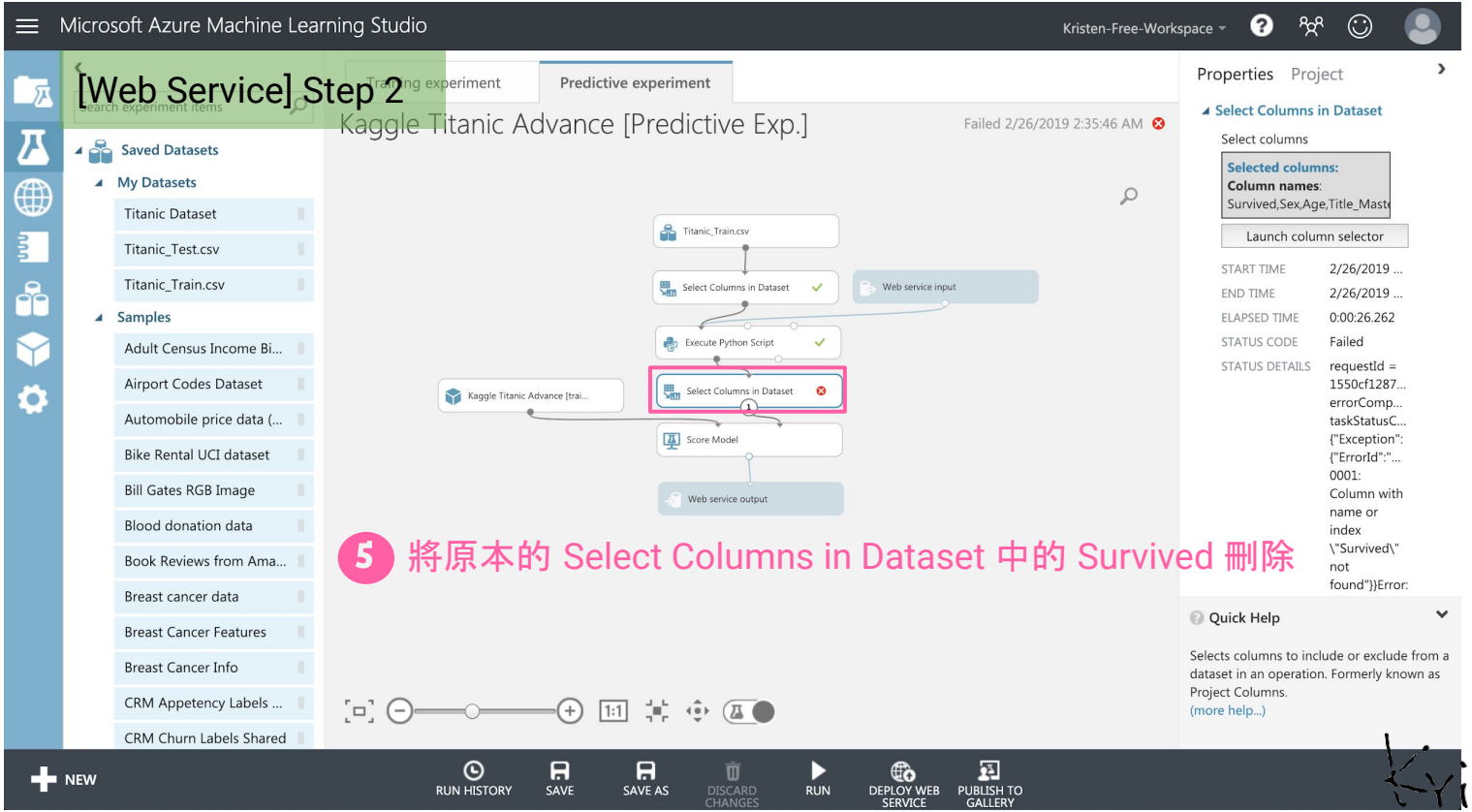


Advanced Version -- Python

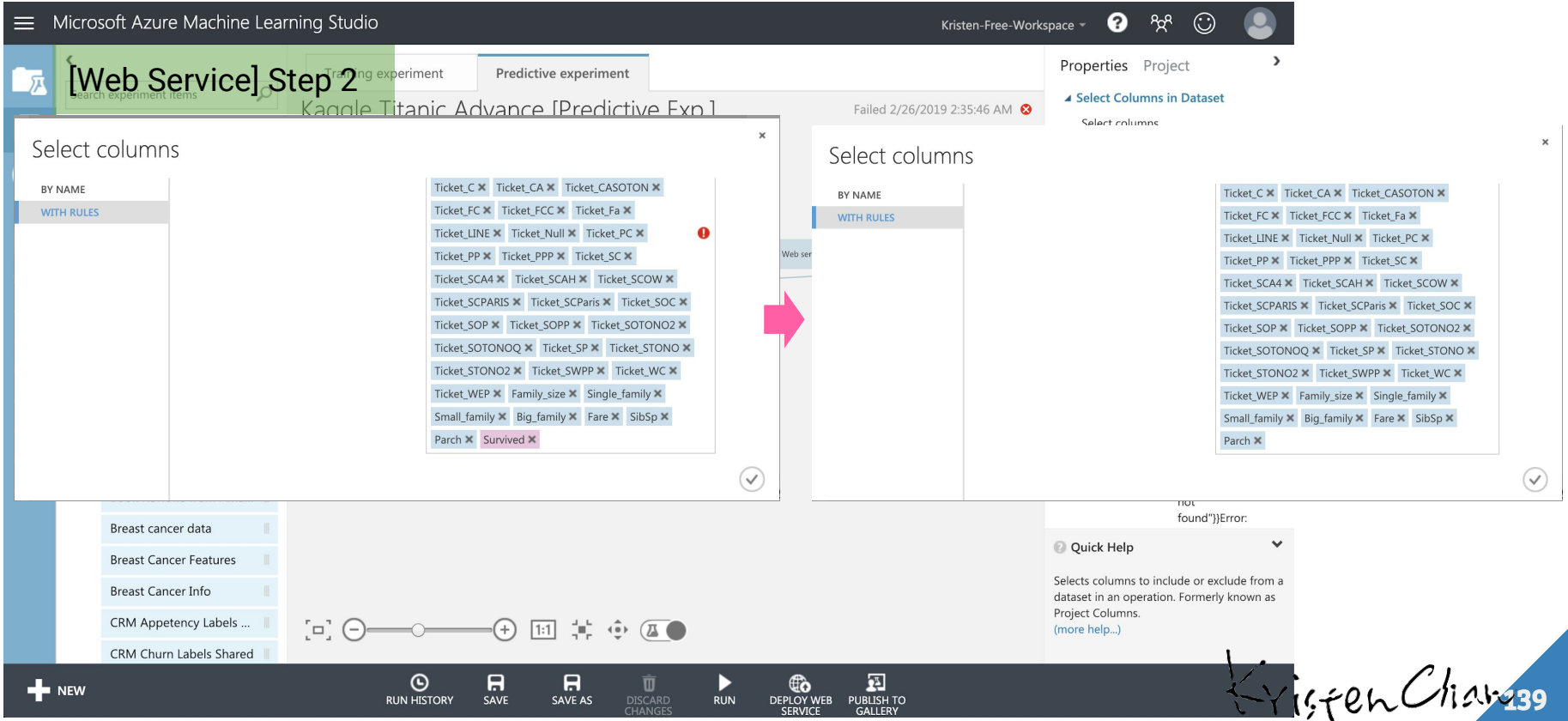


4 將 Web Service Input 接上 Excute Python Script

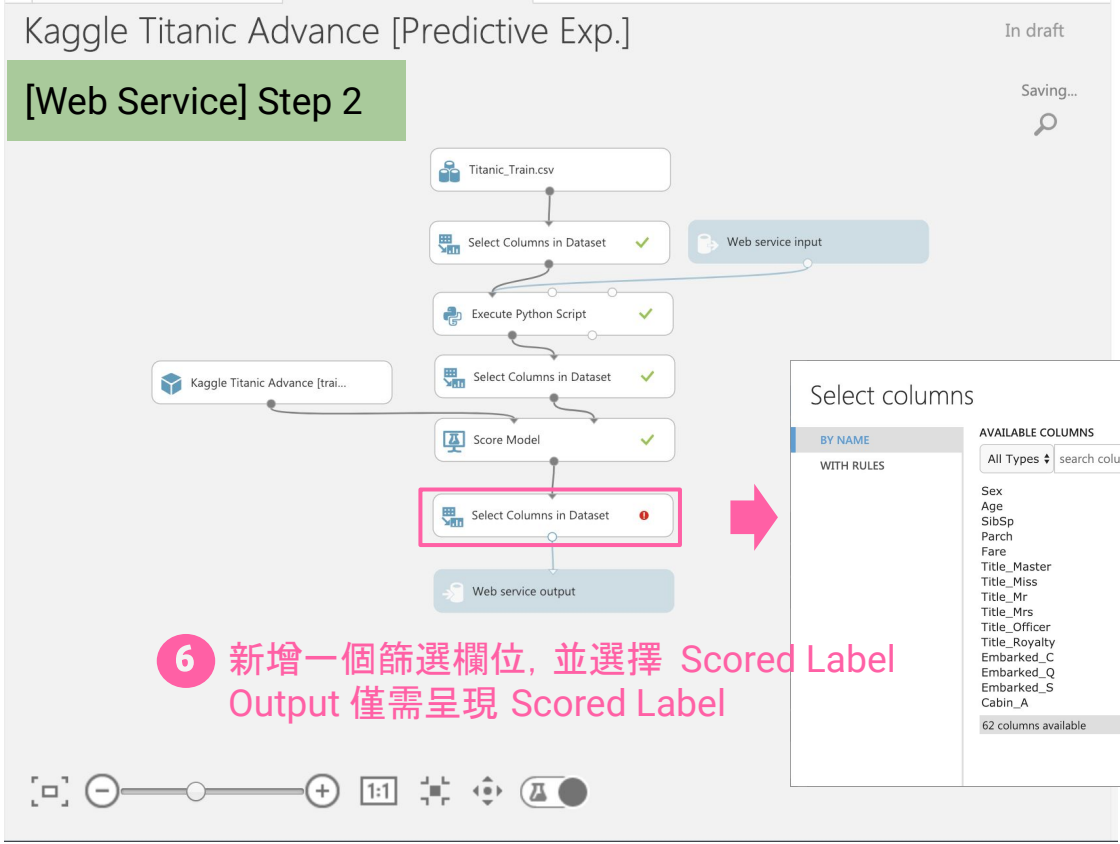
Advanced Version -- Python

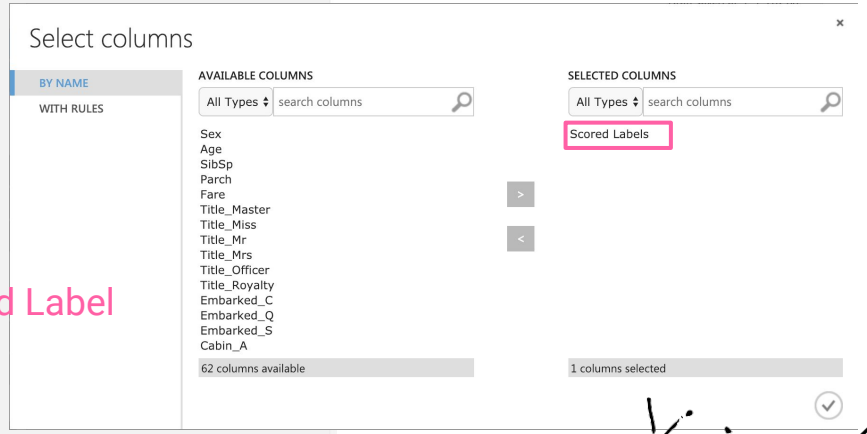


Advanced Version -- Python

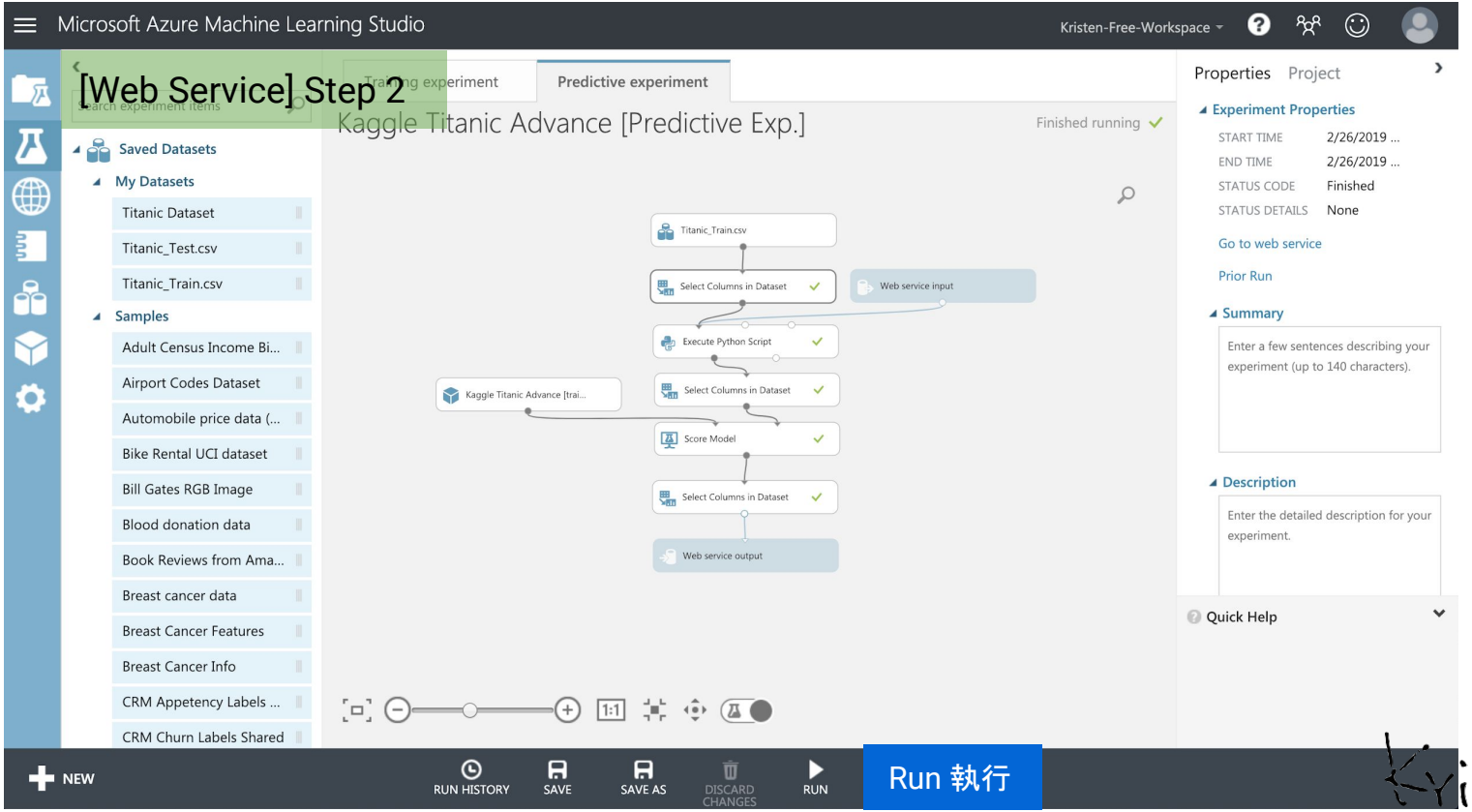


Advanced Version -- Python

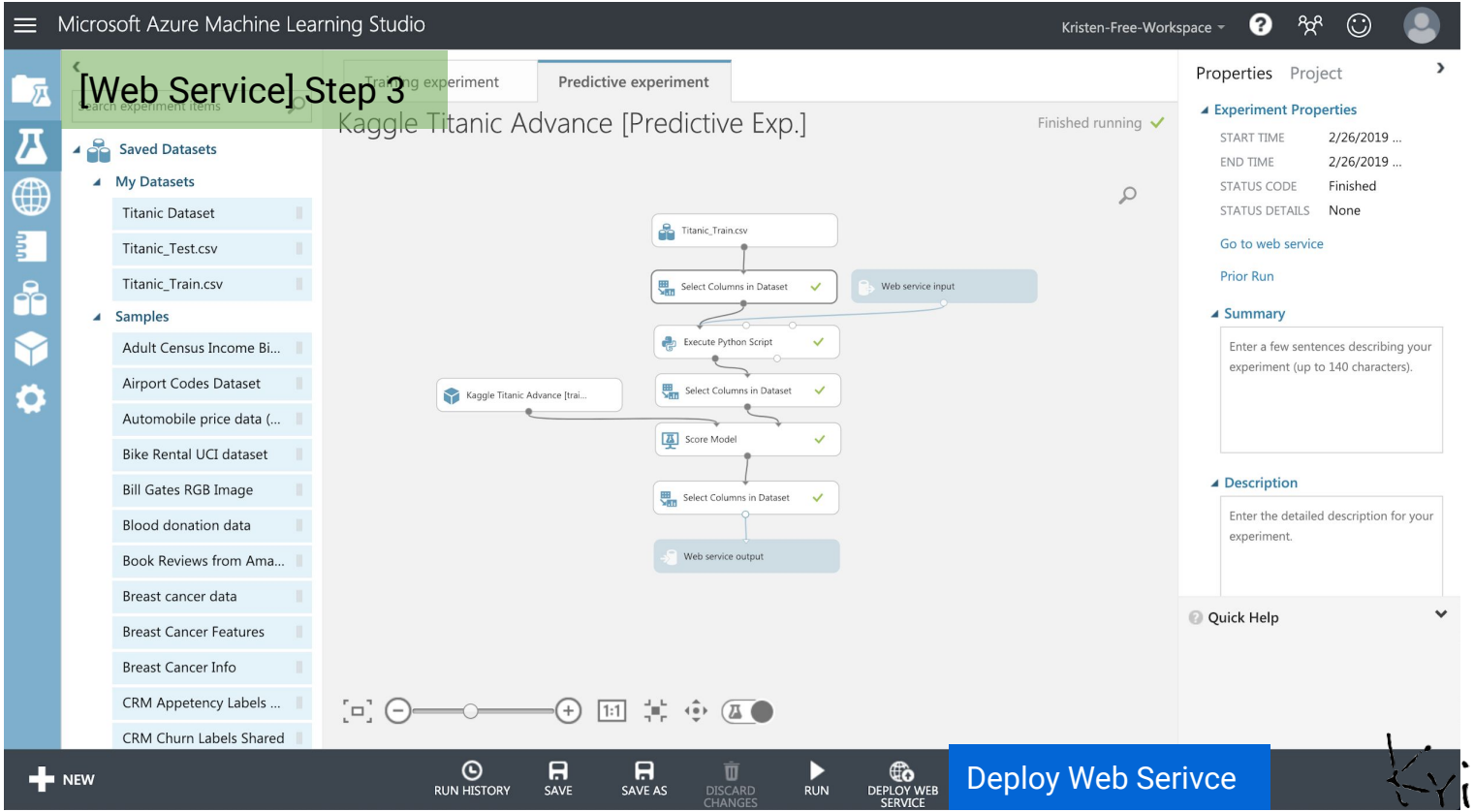




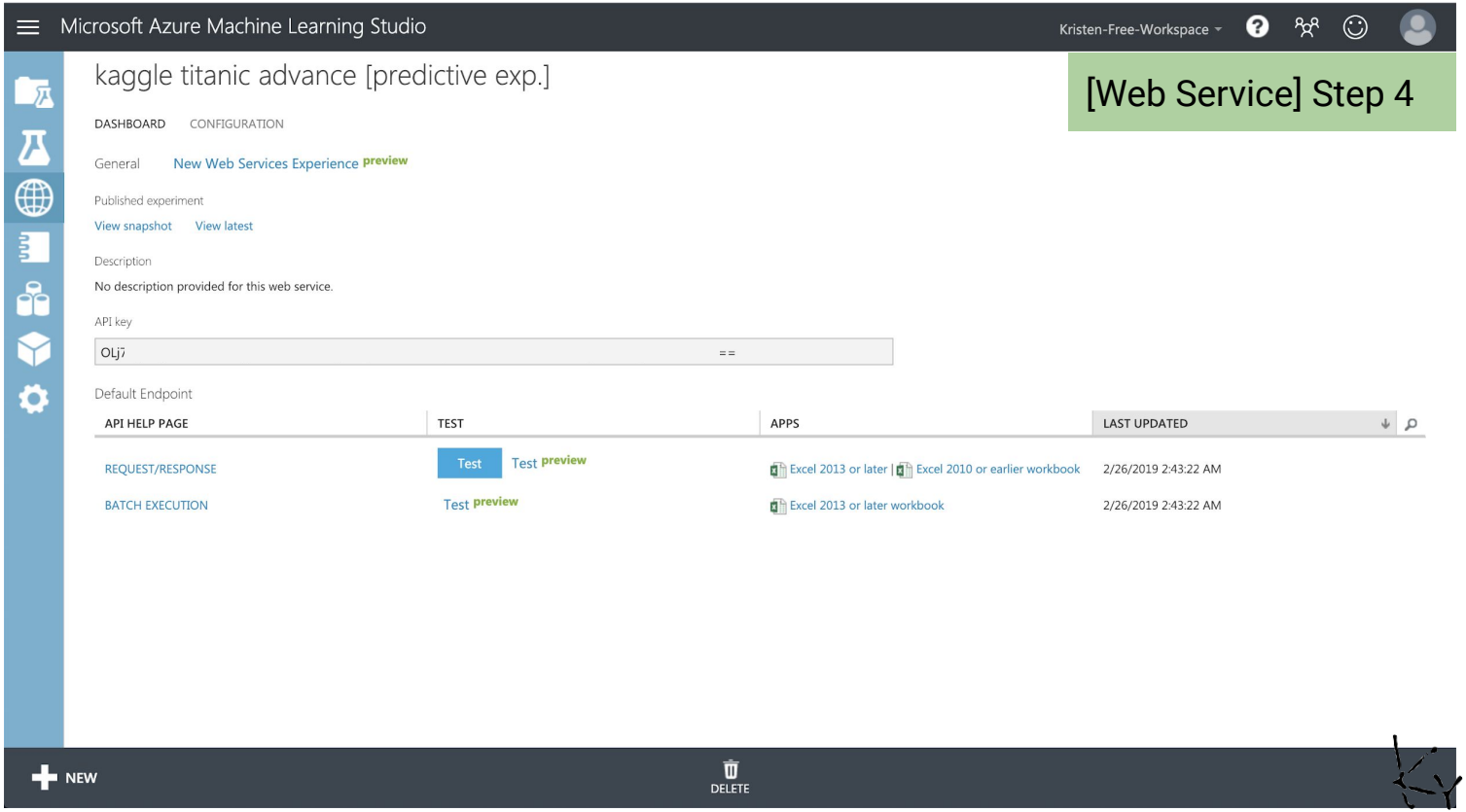
Advanced Version -- Python



Advanced Version -- Python



Advanced Version -- Python



Advanced Version -- Python

Advanced Version -- Python

Enter data to predict 測試

PCLASS
1

NAME
Ostby, Miss. Helene Ragnhild

SEX
female

AGE
22

SIBSP
0

PARCH
1

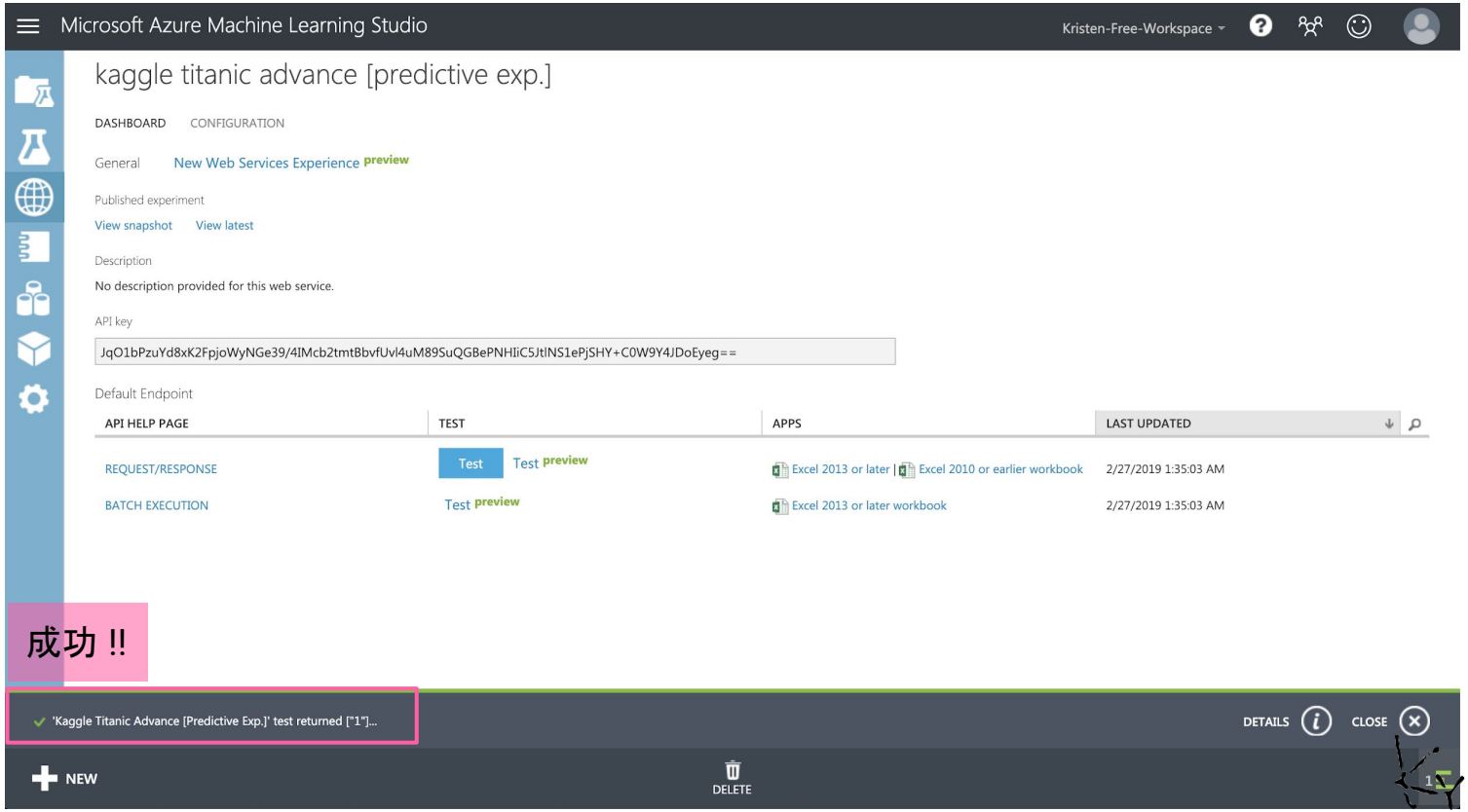
TICKET
113509

FARE
61.9792

CABIN
B36

EMBARKED
C

Advanced Version -- Python



Note

Reference

- Kaggle Titanic
  - https://github.com/ahmedbesbes/How-to-score-0.8134-in-Titanic-Kaggle-Challenge/blob/master/article_1.ipynb
- Cross Validation
  - https://blog.contactsunny.com/data-science/different-types-of-validations-in-machine-learning-cross-validation
- Azure ML Excute Python Script
  - https://docs.microsoft.com/zh-tw/azure/machine-learning/studio/execute-python-scripts