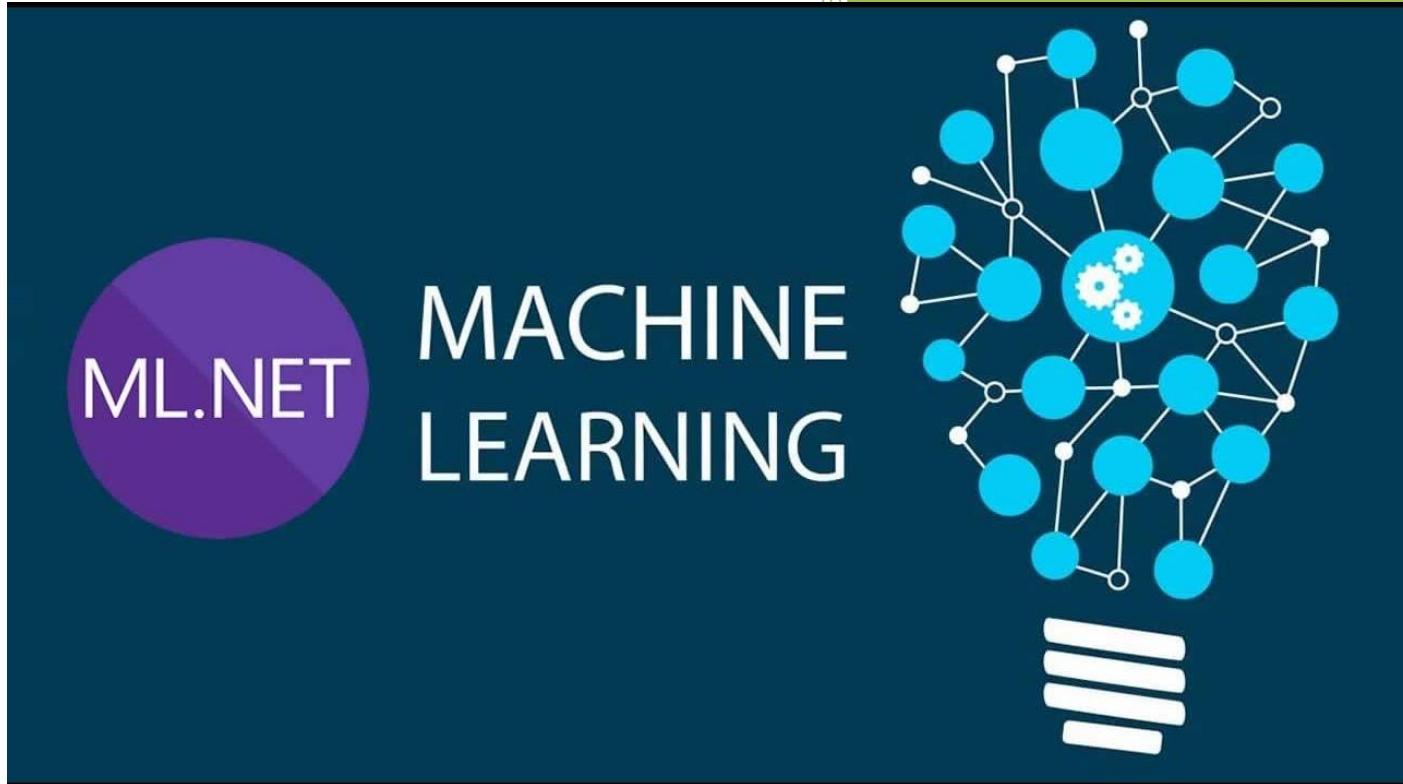


# LonGreat ML.NET

教学范例

Community V1.0.0



# 目 录

## 内容

分析情感(二元分类).....	4
建立项目.....	4
准备资料.....	4
加载数据.....	5
数据转换.....	8
新增数据转换.....	8
结果检视.....	8
特征与卷标.....	9
训练模型.....	9
训练器.....	9
执行.....	10
评估.....	11
预测.....	12
纽约市的出租车费用预测价格(回归模型).....	13
建立项目.....	13
准备并了解资料.....	13
加载数据.....	14
数据转换.....	17
新增数据转换.....	17
结果检视.....	19
特征与卷标.....	20
训练模型.....	21
训练器.....	21
执行.....	21
评估.....	22
鸢尾花花卉分类.....	24
了解问题.....	24
选取适当的机器学习工作.....	24
准备资料.....	24
鸢尾花(多元分类).....	24
建立项目.....	24
加载数据.....	25

数据转换.....	28
特征与卷标.....	29
训练模型.....	29
评估.....	30
鸢尾花(群集).....	32
建立项目.....	32
加载数据.....	32
特征与卷标.....	35
训练模型.....	35
评估.....	36
推荐的电影(矩阵分解模型).....	38
选取适当的机器学习工作.....	38
建立项目.....	38
加载您的数据.....	39
数据转换.....	43
新增数据转换.....	43
结果检视.....	43
特征与卷标.....	44
训练模型.....	45
训练器.....	45
执行.....	45
其他建议算法.....	47
评估.....	48
花辨识(影像分类).....	49
影像分类传输学习范例概观.....	49
了解问题.....	49
ML.NET 影像分类 API.....	49
什么是传输学习？.....	49
训练程序.....	50
瓶颈阶段.....	50
训练阶段.....	50
了解预先定型的模型.....	51
准备并了解资料.....	51
加载数据.....	53
数据转换.....	56
随机资料列.....	56
结果检视.....	57
特征与卷标.....	58
训练模型.....	58

# LonGreat ML.NET Community

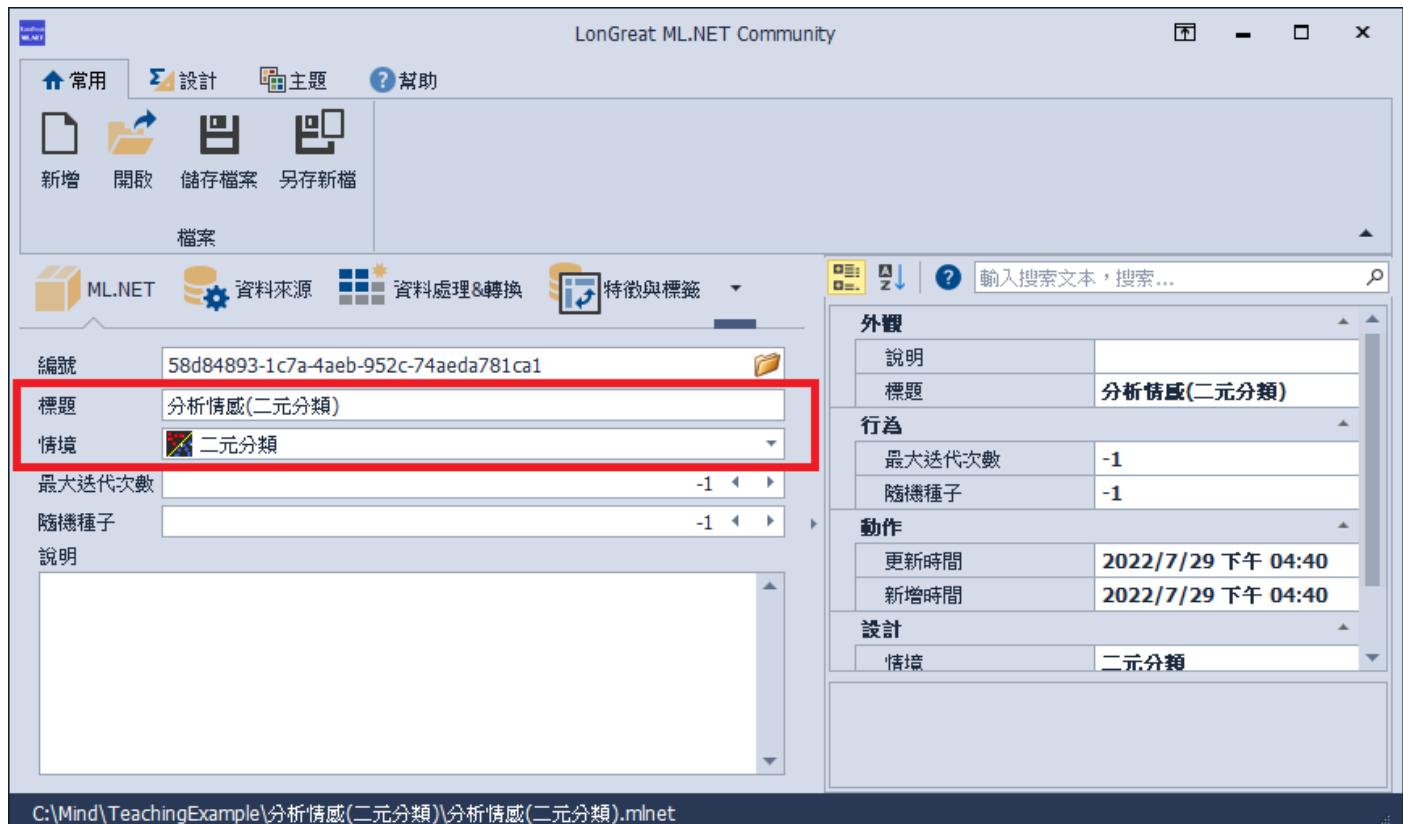
---

训练器.....	58
执行.....	59
评估.....	60
预测.....	61
相关连结.....	61

## 分析情感(二元分类)

### 建立项目

1. 输入标题 => 分析情感(二元分类)
2. 选择情境 => 二元分类



### 准备资料

输入数据集类别 SentimentData 具有使用者评论 (SentimentText) 的 string，以及代表情感的 bool (Sentiment) 值 1 (正面) 或 0 (负面)。其描述每个字段的数据文件顺序。此外，Sentiment 属性具有 [ColumnName](#) 属性来将它指定为 Label 字段。下列范例档案没有标头数据列，且看起来像这样：

SentimentText	Label
女服务生的服务速度有点慢。	0
不够酥脆。	0

哇。。。喜欢这个位置。	1
服务很迅速。	1

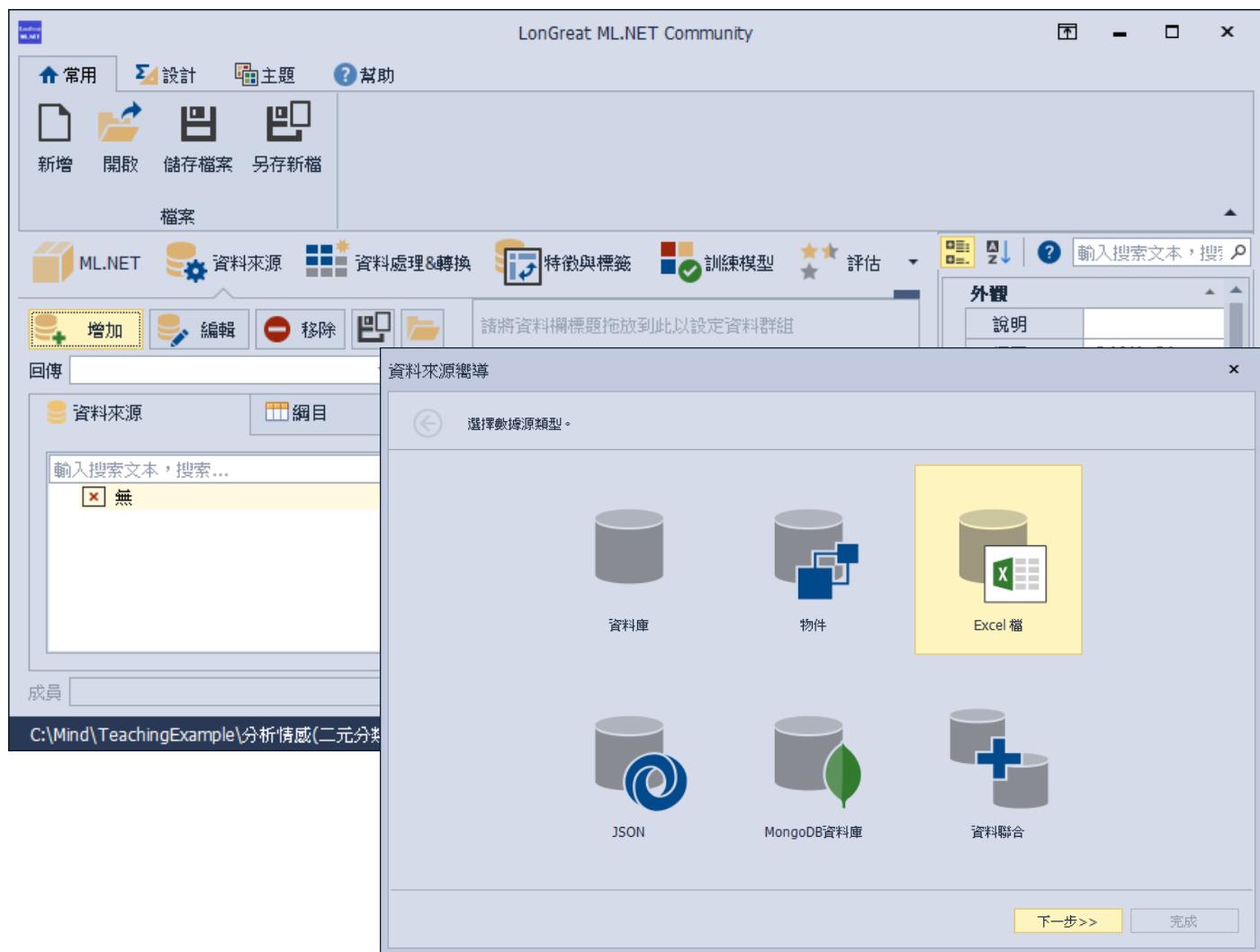
SentimentPrediction 是在模型定型后所使用的预测类别。 它继承自 SentimentData，以便输入 SentimentText 可以和输出预测一起显示。 Prediction 布尔值是在提供新输入 SentimentText 时，模型预测的值。

输出类别 SentimentPrediction 包含模型计算的两个其他属性：Score – 模型计算的原始分数，和 Probability – 针对文字具有正面情感之可能性所校正的分数。

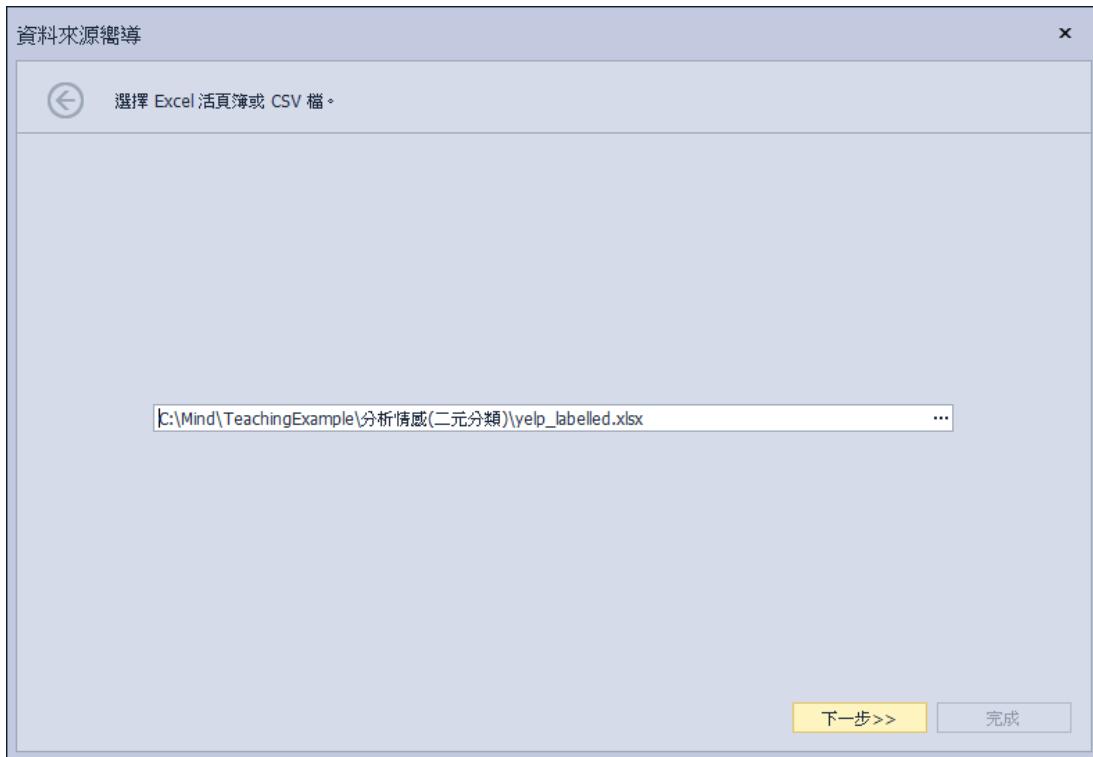
本教学课程中最重要的属性是 Prediction。

## 加载数据

数据源->增加->Excel 檔



选取 yelp\_labelled.xlsx



This screenshot displays two overlapping windows of the 'Data Source Wizard'.

The top window, titled '資料來源嚮導' (Data Source Wizard), is on 'Step 2: 指定導入設置' (Specify import settings). It contains several checked options under 'Import Settings': '使用 first 行的值為欄位名稱' (Use the value of the first row as column names), 'Skip empty rows', 'Skip hidden rows', and 'Skip hidden columns'. At the bottom right are 'Next >' and 'Finish' buttons.

The bottom window, also titled '資料來源嚮導' (Data Source Wizard), is on 'Step 3: 選擇所需的範圍' (Select the required range). It shows a preview of the data with a single row selected, labeled 'Sheet'. At the bottom right are 'Next >' and 'Finish' buttons.

## 按完成



## 检视资料

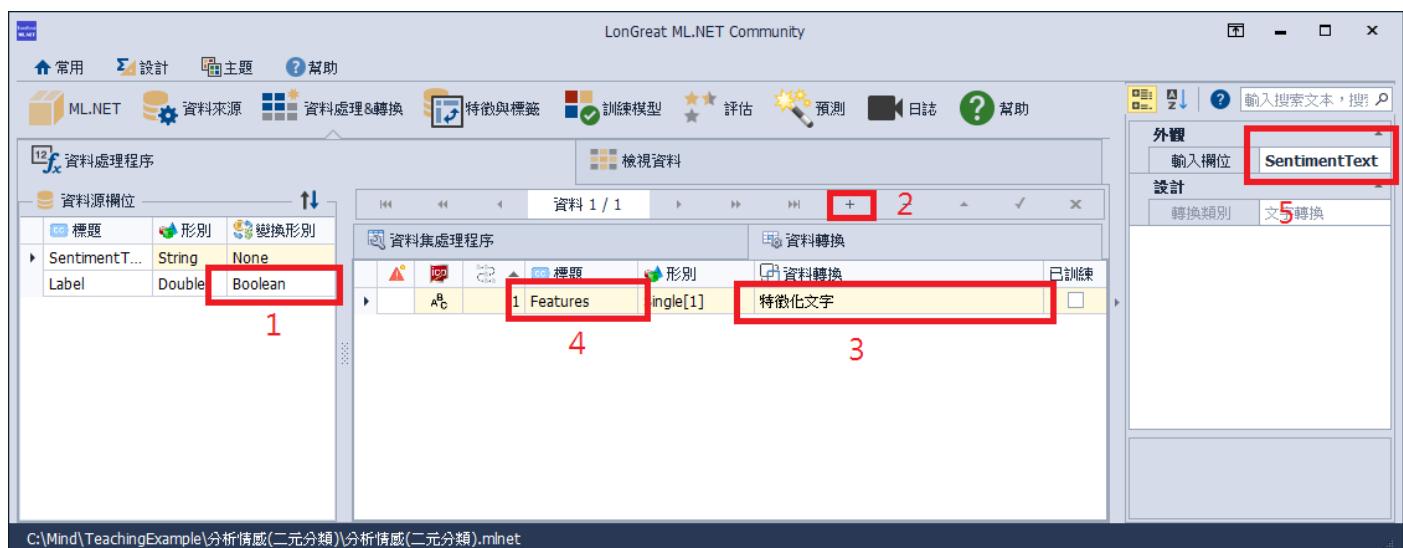
Sentiment Text	Label
Wow... Loved this place.	1
Crust is not good.	0
Not tasty and the texture was just nasty.	0
Stopped by during the late May bank holiday off Rick Steve recommendation and loved it.	1
The selection on the menu was great and so were the prices.	1
Now I am getting angry and I want my damn pho.	0
Honesty it didn't taste THAT fresh.)	0
The potatoes were like rubber and you could tell they had been made up ahead of time being kept und...	0
The fries were great too.	1
A great touch.	1
Service was very prompt.	1
Would not go back.	0
The cashier had no care what so ever on what I had to say it still ended up being wayyy overpriced.	0
I tried the Cape Cod ravioli, chicken,with cranberry...mmmm!	1
I was disgusted because I was pretty sure that was human hair.	0
I was shocked because no signs indicate cash only.	0
Highly recommended.	1
Waitress was a little slow in service.	0
This place is not worth your time, let alone Vegas.	0
did not like at all.	0
The Burritos Blah!	0

## 数据转换

### 新增数据转换

将字段 SentimentText 特征化，使用特征化文字(FeaturizeText)函式

1. Label 变换形别=>Boolean
2. +新增数据转换
3. 选择数据转换函式=>特征化文字(FeaturizeText)
4. 定义新字段标题名称=>Featurizes
5. 选择输入字段=>SentimentText



### 结果检视

1. 至检视资料页签
2. 选择显示笔数
3. 按充满资料

SentimentText	Label	Features
女服务生的服务速度有点慢。	0	[0.76, 0.65, 0.44, ...]
不够酥脆。	0	[0.98, 0.43, 0.54, ...]
哇。。。喜欢这个位置。	1	[0.35, 0.73, 0.46, ...]
服务很迅速。	1	[0.39, 0, 0.75, ...]

# LonGreat ML.NET Community

The screenshot shows the ML.NET Studio interface with the 'Data View' tab selected. A red box highlights the '檢視資料' (View Data) button at the top center. Another red box highlights the '10000' dropdown next to it. A third red box highlights the '充滿資料' (Fill Data) button on the right side.

## 特征与卷标

拖拉 Features=>特征

拖拉 Label => 标签

The screenshot shows the '特征与標籤' (Features & Labels) tab in the ML.NET Studio interface. It displays four columns:

- 忽略 (無) (Ignore (None))
- 特徵 (一或多) \$ (Features (Single[1]))
- 標籤 (只有一) \$ (Label (Boolean))
- 權重 (無或一) \$ (Score (Probability))

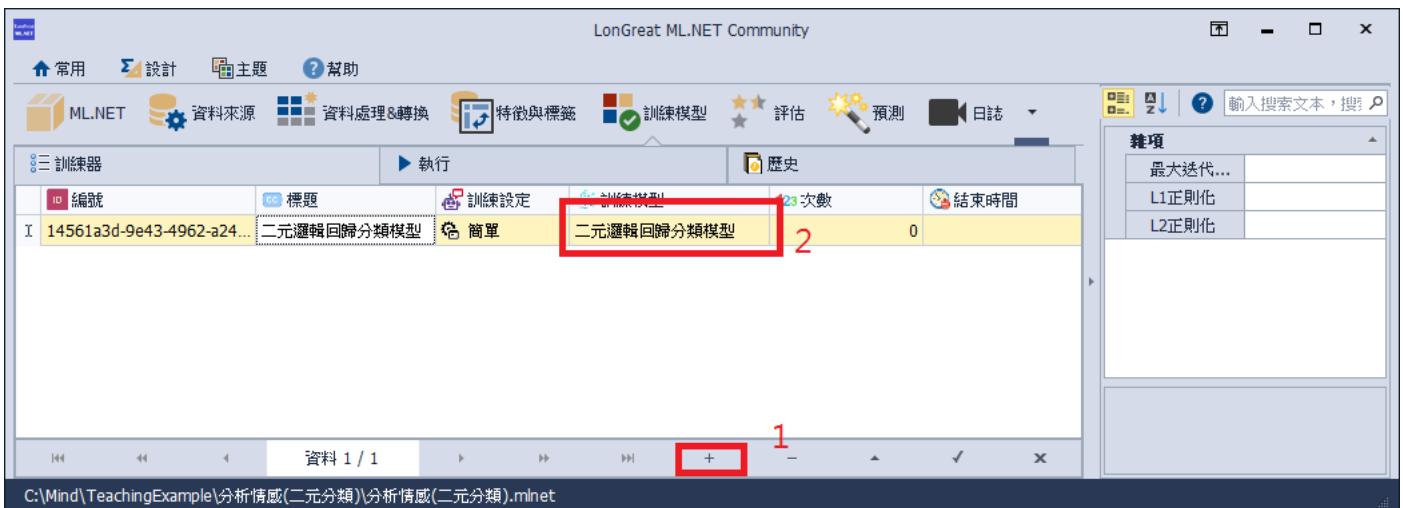
## 训练模型

### 训练器

按+新增

选择二元逻辑回归分类模型(SdcaLogisticRegressionBinary)

# LonGreat ML.NET Community

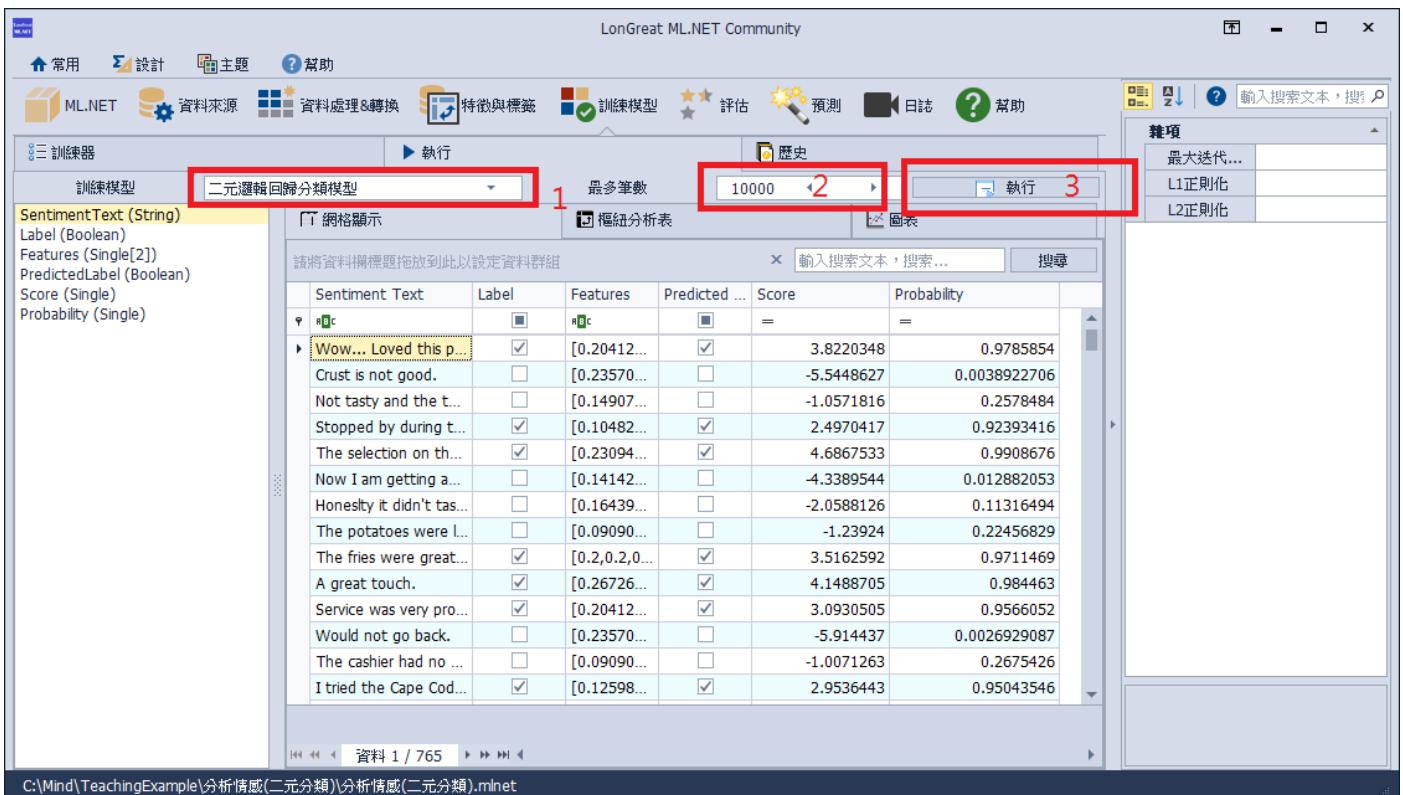


## 执行

1. 选择训练模型
2. 要显示最多笔数
3. 执行

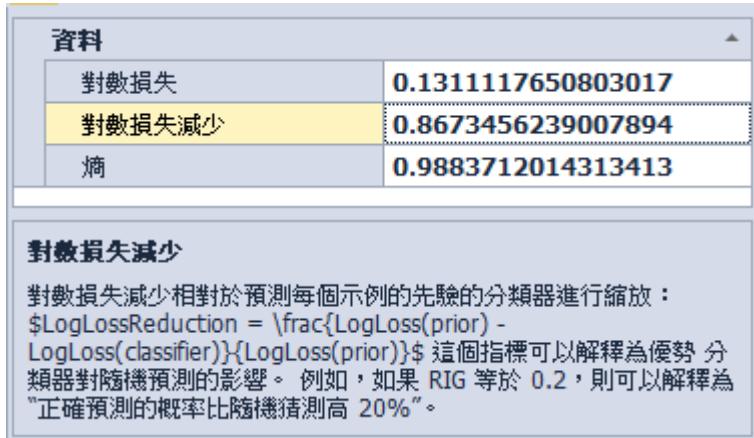
结果多三个字段

- 预测标签 PredictedLabel
- 或然率 Probability
- 分数 Score



## 评估

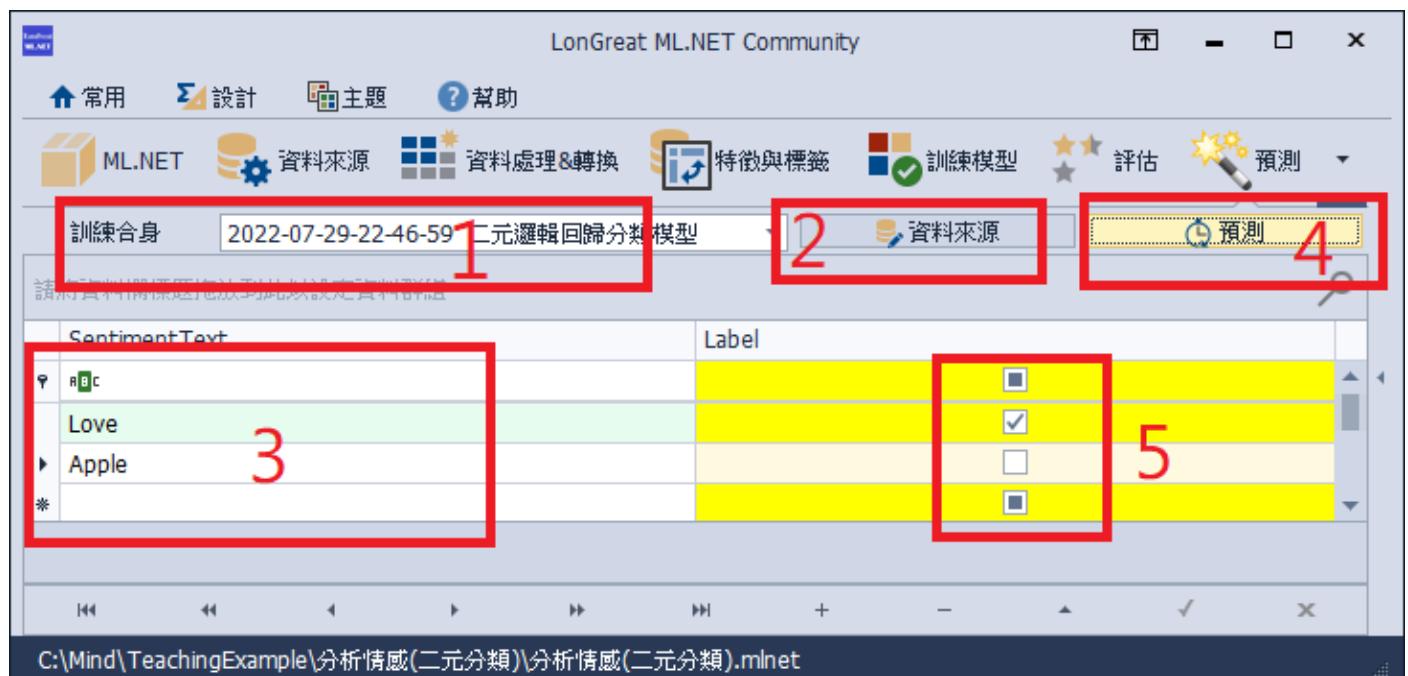
1. 选取以训练完合身的模型
2. 选取设定数据须与原来相同 SentimentText 与 Label
3. 点击预测按钮=>产生出批次预测结果
4. 点击评估按钮=>评估结果



	對數損失	0.131111765...
對數損失減少	0.867345623...	
熵	0.988371201...	

## 预测

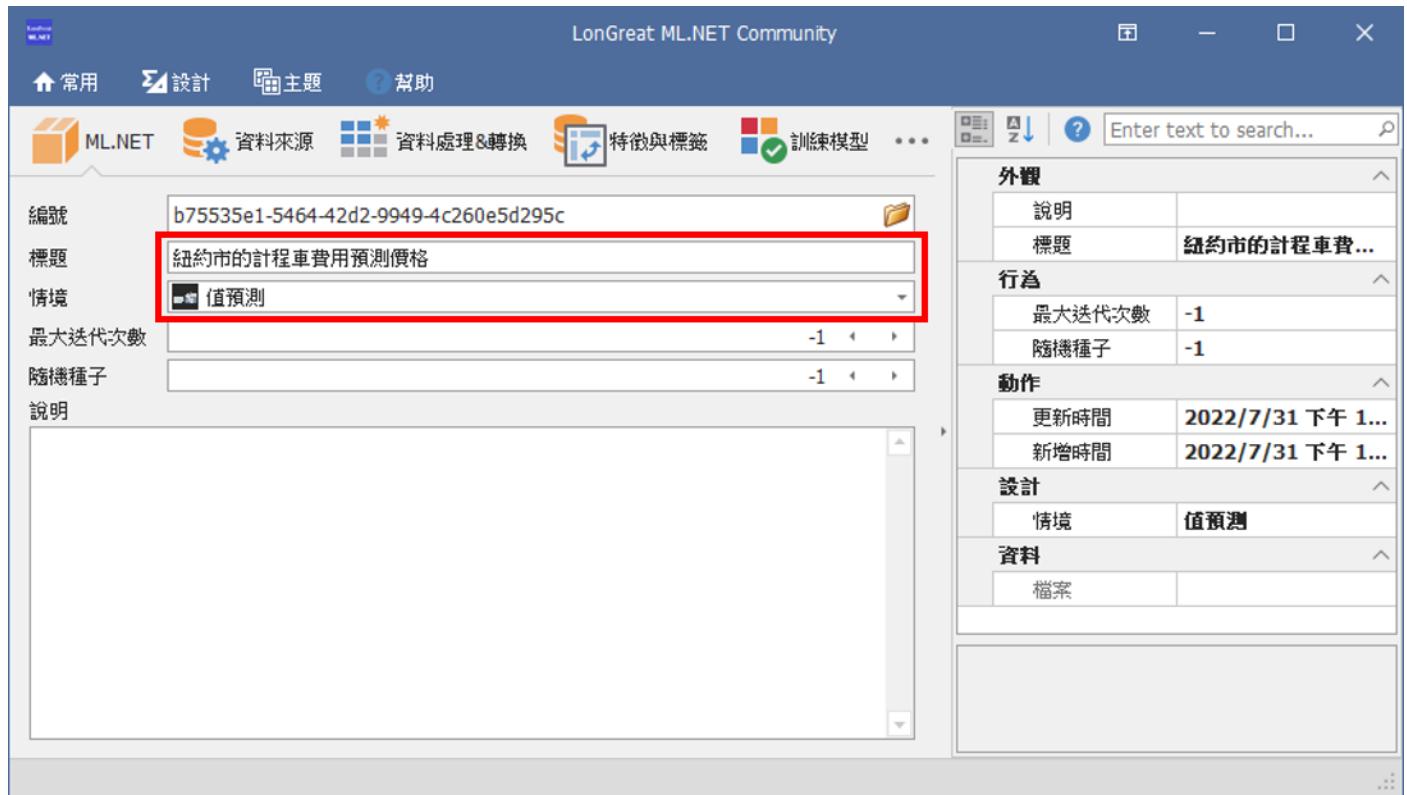
1. 选取以训练完合身的模型
2. 点击数据源按钮=>产生数据表
3. 输入数据表 SentimentText 字段
4. 点击预测按钮
5. 产生出 Label 预测结果



## 纽约市的出租车费用预测价格(回归模型)

### 建立项目

1. 输入标题 => 纽约市的出租车费用预测价格
2. 选择情境 => 值预测



### 准备并了解资料

1. 下载 [taxi-fare-train.csv](#) 和 [taxi-fare-test.csv](#) 数据集，并将它们储存至您在上一个步骤所建立的 *Data* 文件夹。我们可以使用这些数据集将机器学习模型定型，然后评估模型的准确程度。这些数据集原先来自 [NYC TLC Taxi Trip 数据集](#)。
2. 开启 [taxi-fare-train.csv](#) 数据集，然后查看第一个数据列中的数据行标头。请查看每个资料行。了解数据，并决定哪些数据行是 **features**，以及哪一个数据行是 **label**。

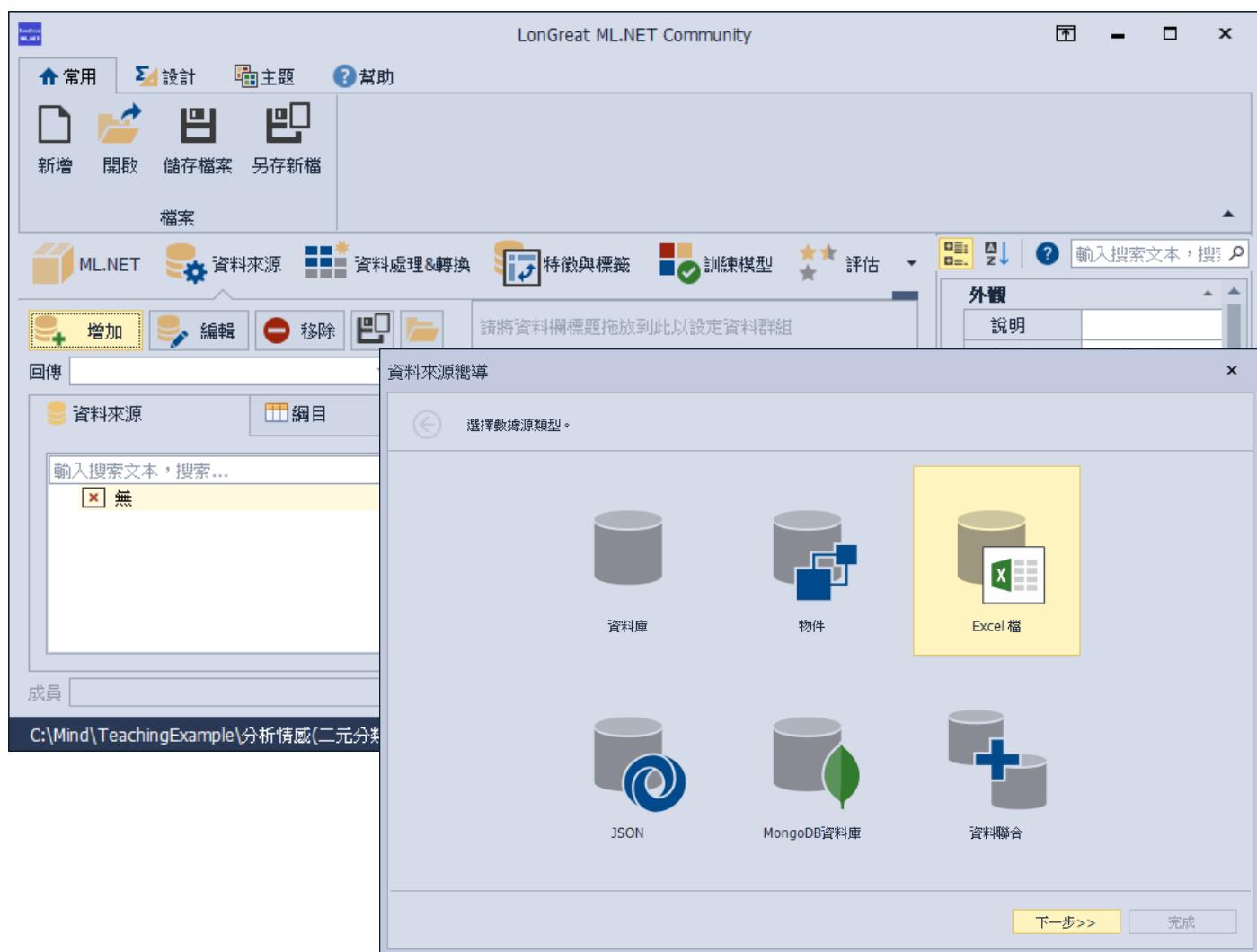
**label** 是您希望进行预测的数据行。识别的 **Features** 是您提供模型来预测 **Label** 的输入。

提供的数据集包含下列数据行：

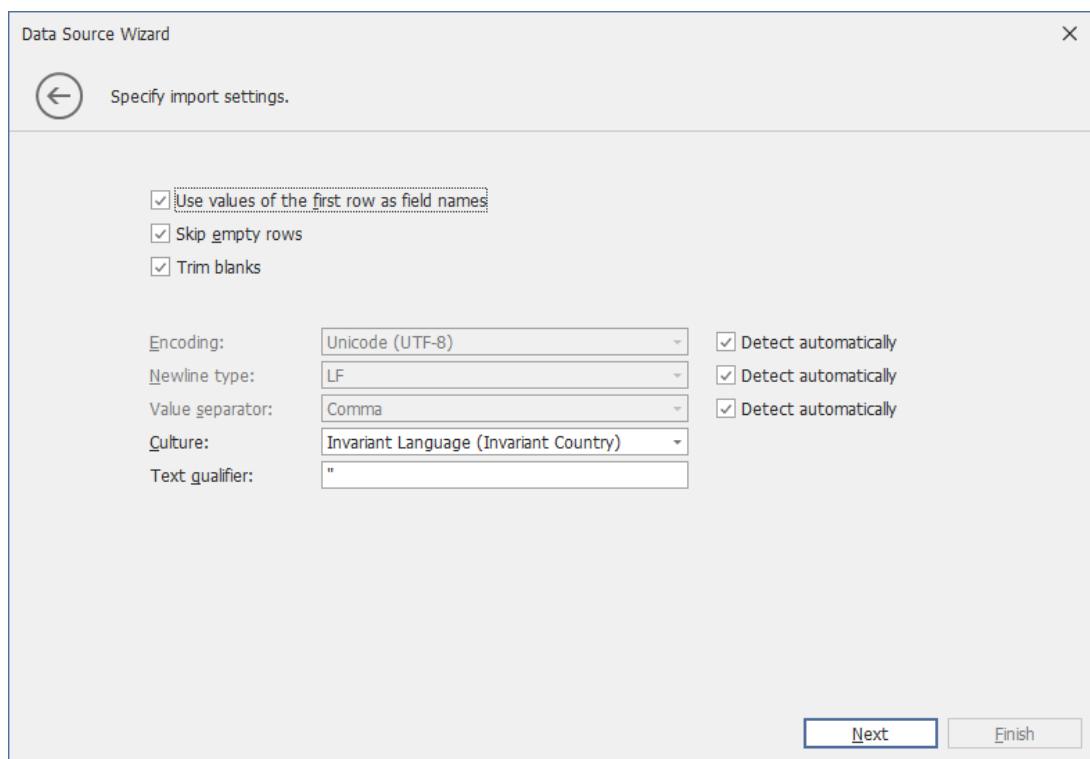
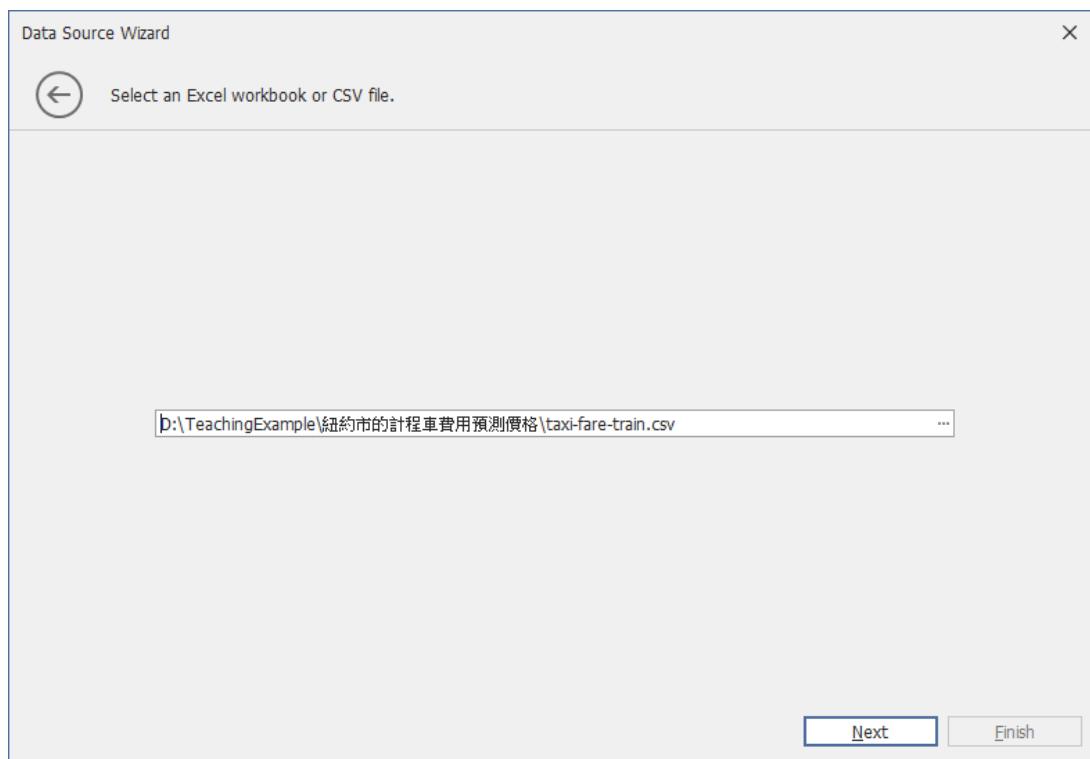
- **vendor\_id:** 出租车厂商的标识符是一项特征。
- **rate\_code:** 出租车行程的费率类型是一项特征。
- **passenger\_count:** 行程的乘客数目是一项特征。
- **trip\_time\_in\_secs:** 行程所花费的时间长度。您想要在行程结束之前预测行程的车资。届时，您不知道行程会花多少时间。因此，行程时间不是一项特征，您将从模型中排除这个数据行。
- **trip\_distance:** 行程的距离是一项特征。
- **payment\_type:** 付款方式（现金或信用卡）是一项特征。
- **fare\_amount:** 出租车车资总计是标签。

## 加载数据

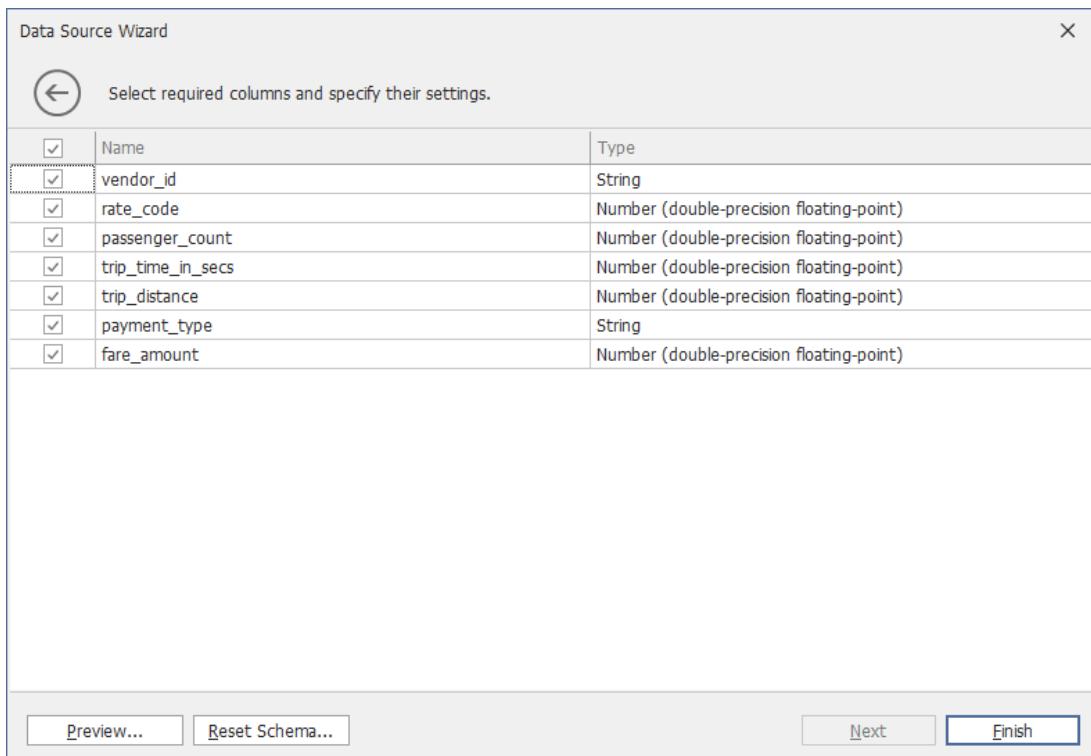
数据源->增加->Excel 檔



选取 taxi-fare-train.csv



## 按完成



## 检视资料

The screenshot shows the ML.NET Studio interface with the following components:

- Toolbar:** Includes '常用' (Common), '設計' (Design), '主題' (Theme), '幫助' (Help), 'ML.NET' icon, '資料來源' (DataSource), '資料處理&轉換' (Data Processing & Transformation), '特徵與標籤' (Features & Labels), '訓練模型' (Train Model), '評估' (Evaluate), '預測' (Predict), '日誌' (Log), and a question mark icon.
- DataSource1 Properties:** A properties panel on the right side with sections for '外觀' (Appearance), '行為' (Behavior), '動作' (Actions), '設計' (Design), and '資料' (Data). It shows details like '說明' (Description: 紐約市的計程...), '標題' (Title: -1), '最大值...' (Max Value: -1), '新增時間' (Created At: 2022/7/31 ...), and '新增時間' (Created At: 2022/7/31 ...).
- Data Grid:** A large table displaying data from 'excelDataSource1'. The columns are: vendor\_id, rate\_code, passenger\_c..., trip\_time\_in..., trip\_distance, payment\_type, fare\_amount. The data consists of approximately 100,000 rows, with the last row showing values: 2, 1, 1, 3, 0, NOC, 52.
- DataSource1 Properties:** A properties panel on the left showing the connection details for 'excelDataSource1'.

## 数据转换

### 新增数据转换

1. 最常编码向量 => 将 vendor\_id 转换为编码向量

The screenshot shows the ML.NET Data Processing Studio interface. On the left, the 'Source Fields' pane lists fields: vendor\_id, rate\_code, passenger\_c..., trip\_time\_in..., trip\_distance, payment\_type, fare\_amount. The 'Transformations' pane shows a list of steps:

步驟	標題	形別	資料轉換	已訓練
1	VendorIdEncoded	Single[1]	最常編碼向量	<input type="checkbox"/>
2	RateCodeEncoded	Single[1]	最常編碼向量	<input type="checkbox"/>
3	PaymentTypeEncoded	Single[1]	最常編碼向量	<input type="checkbox"/>
4	PassengerCountMV	PassengerCountMV	均值方差標準化	<input type="checkbox"/>
5	TripTimeMV	TripTimeMV	均值方差標準化	<input type="checkbox"/>
6	TripDistanceMV	TripDistanceMV	均值方差標準化	<input type="checkbox"/>
7	Label	Label	複製資料欄	<input type="checkbox"/>

The 'Output Column' search bar at the top right contains 'vendor\_id'. The 'Behavior' section on the right shows: Maximum Keys = 1000000, Output Type = Indicator, Key Sequence = ByOccurrence.

2. 最常编码向量 => 将 rate\_code 转换为编码向量

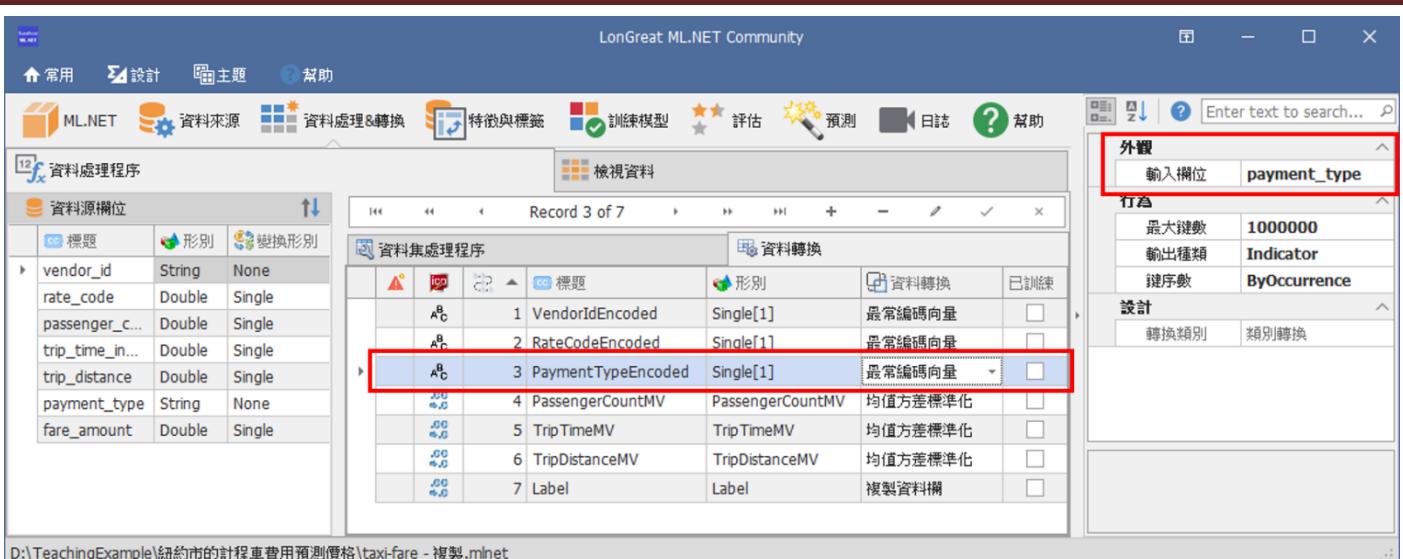
The screenshot shows the ML.NET Data Processing Studio interface. The 'Source Fields' pane is identical to the previous screenshot. The 'Transformations' pane shows a list of steps:

步驟	標題	形別	資料轉換	已訓練
1	VendorIdEncoded	Single[1]	最常編碼向量	<input type="checkbox"/>
2	RateCodeEncoded	Single[1]	最常編碼向量	<input type="checkbox"/>
3	PaymentTypeEncoded	Single[1]	最常編碼向量	<input type="checkbox"/>
4	PassengerCountMV	PassengerCountMV	均值方差標準化	<input type="checkbox"/>
5	TripTimeMV	TripTimeMV	均值方差標準化	<input type="checkbox"/>
6	TripDistanceMV	TripDistanceMV	均值方差標準化	<input type="checkbox"/>
7	Label	Label	複製資料欄	<input type="checkbox"/>

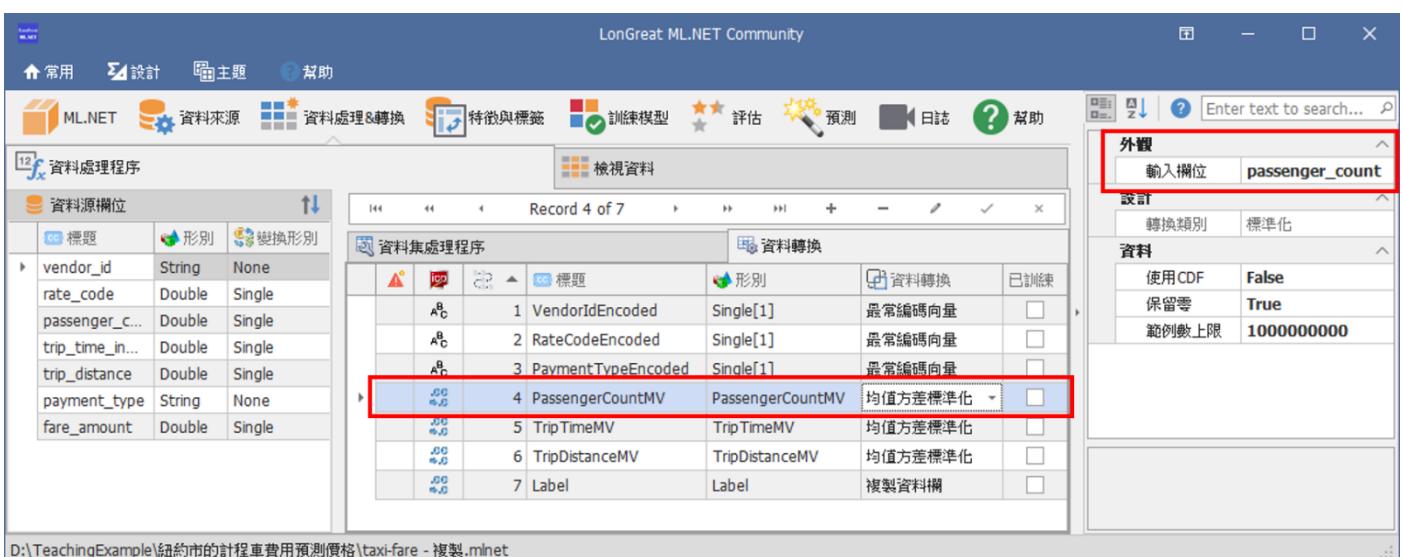
The 'Output Column' search bar at the top right contains 'rate\_code'. The 'Behavior' section on the right shows: Maximum Keys = 1000000, Output Type = Indicator, Key Sequence = ByOccurrence.

3. 最常编码向量 => 将 payment\_type 转换为编码向量

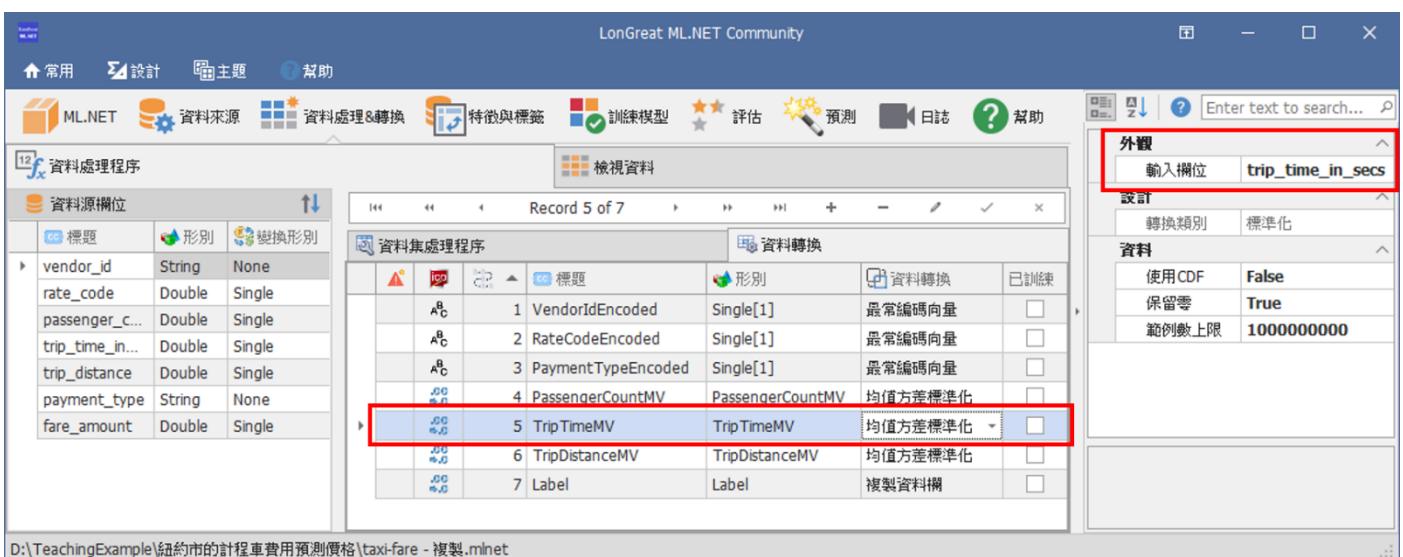
# LonGreat ML.NET Community



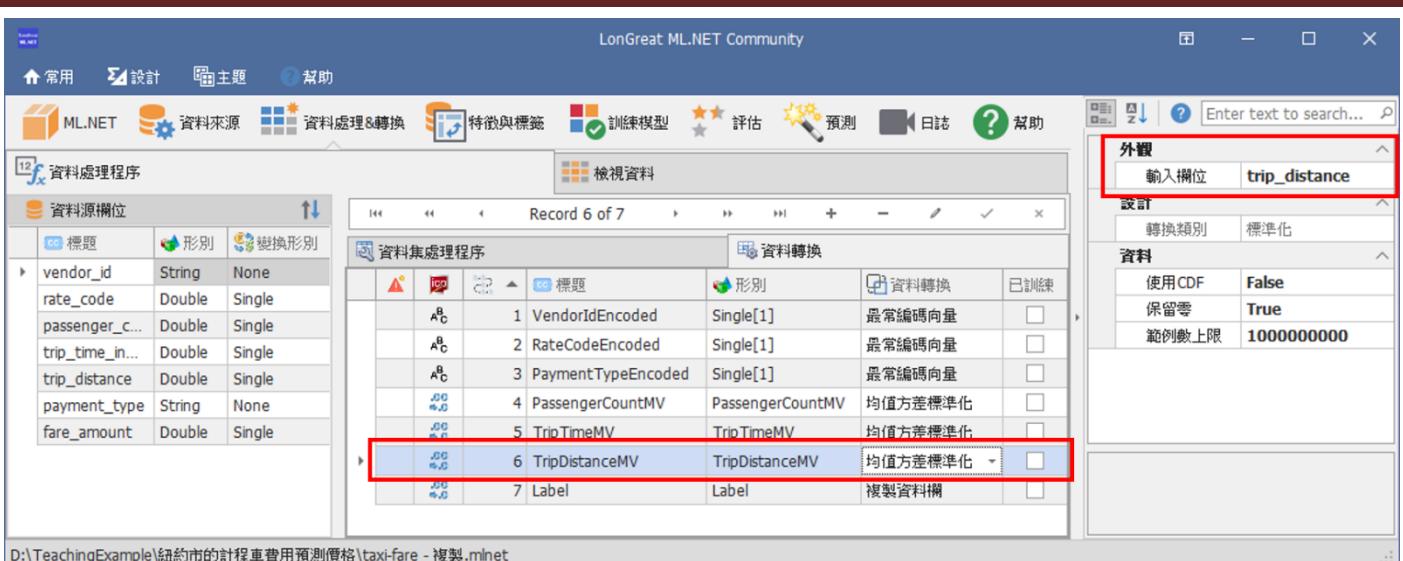
4. 均值方差标准化 => 将 passenger\_count 转换为均值方差标准化



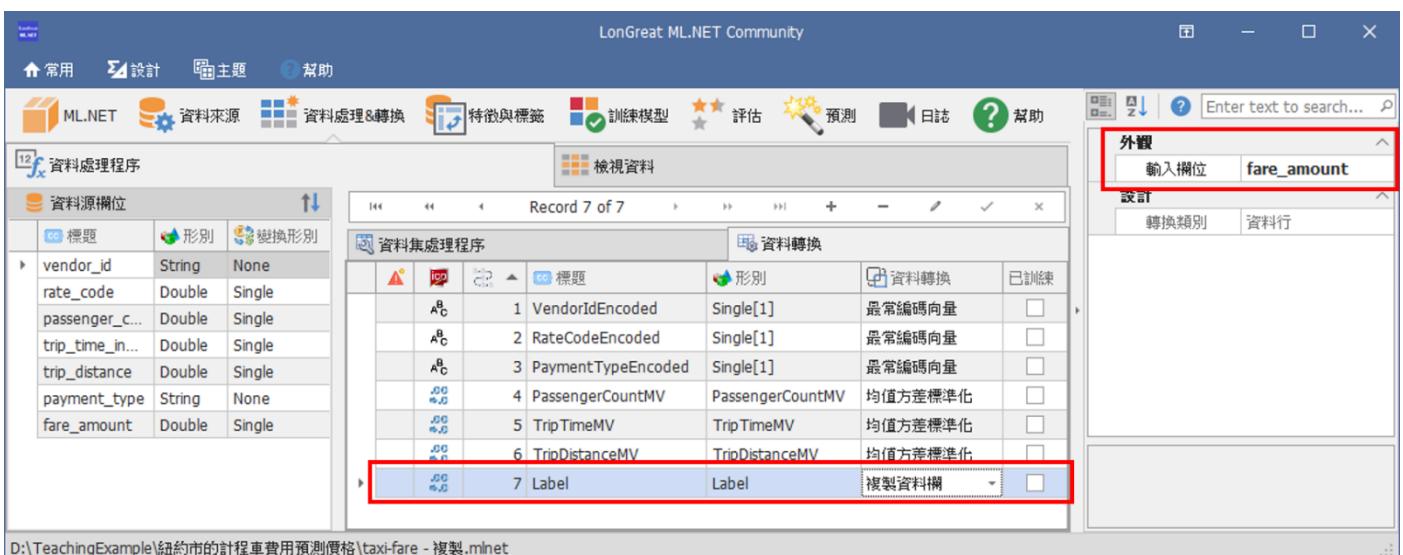
5. 均值方差标准化 => 将 trip\_time\_in\_secs 转换为均值方差标准化



6. 均值方差标准化 => 将 trip\_distance 转换为均值方差标准化



## 7. 复制数据域 => 将 fare\_amount 复制



## 结果检视

- 至检视资料页签
- 选择显示笔数
- 按充满资料

# LonGreat ML.NET Community

The screenshot shows the ML.NET Studio environment. At the top, there's a navigation bar with tabs like '常用' (Common), '設計' (Design), '題主題' (Topic), '幫助' (Help), 'ML.NET', '資料來源' (Data Source), '資料處理&轉換' (Data Processing & Transformation), '特徵與標籤' (Features & Labels), '訓練模型' (Train Model), '評估' (Evaluate), '預測' (Predict), '日誌' (Log), and '幫助' (Help). Below the navigation bar is a toolbar with icons for '資料處理程序' (Data Processor), '檢視資料' (View Data), and '充滿資料' (Fill Data). The main area contains a data grid titled '10000' which displays 10,000 rows of taxi fare data. The columns include vendor\_id, rate\_code, passenger\_count, trip\_time\_in\_secs, trip\_distance, payment\_type, fare\_amount, VendorIdEncoded, RateCodeEncoded, PaymentTypeEncoded, PassengerCountMV, TripTimeMV, TripDistanceMV, Label, and several numerical and categorical columns. The bottom left of the grid shows 'Record 1 of 10000'. The bottom status bar indicates the file path: 'D:\TeachingExample\紐約市的計程車費用預測價格\taxi-fare.mlnet'.

## 特征与卷标

拖拉 VendorIdEncoded=>特征

拖拉 RateCodeEncoded=>特征

拖拉 PaymentTypeEncoded=>特征

拖拉 PassengerCountMV=>特征

拖拉 TripTimeMV=>特征

拖拉 TripDistanceMV=>特征

拖拉 Label =>标签

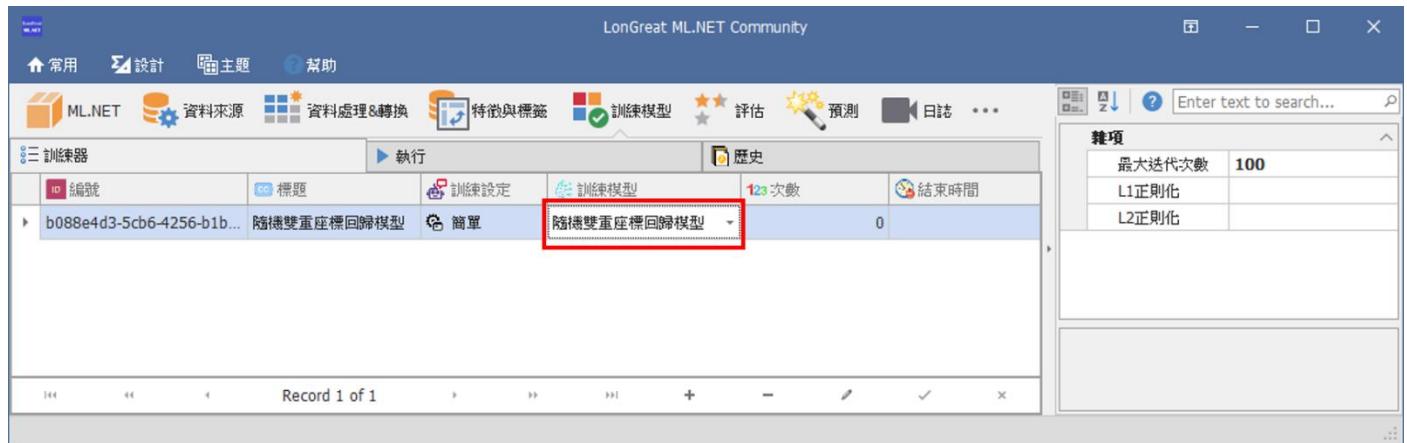
The screenshot shows the 'Features & Labels' section of the ML.NET Studio. It includes four main categories: '忽略 (無)' (Ignore (None)), '特徵 (一或多)' (Feature (One or More)), '標籤 (只有一)' (Label (One)), and '權重 (無或一)' (Weight (None or One)). The '特徵 (一或多)' category lists features such as vendor\_id, rate\_code, passenger\_count, trip\_time\_in\_secs, trip\_distance, payment\_type, fare\_amount, VendorIdEncoded, RateCodeEncoded, PaymentTypeEncoded, PassengerCountMV, TripTimeMV, and TripDistanceMV. The '標籤 (只有一)' category lists the 'Label(Label)' column. The '權重 (無或一)' category is empty. To the right, there's a search bar and a sidebar with various settings and options for training and evaluation.

## 训练模型

### 训练器

按+新增

选择随机双重坐标回归模型(SdcaRegression)

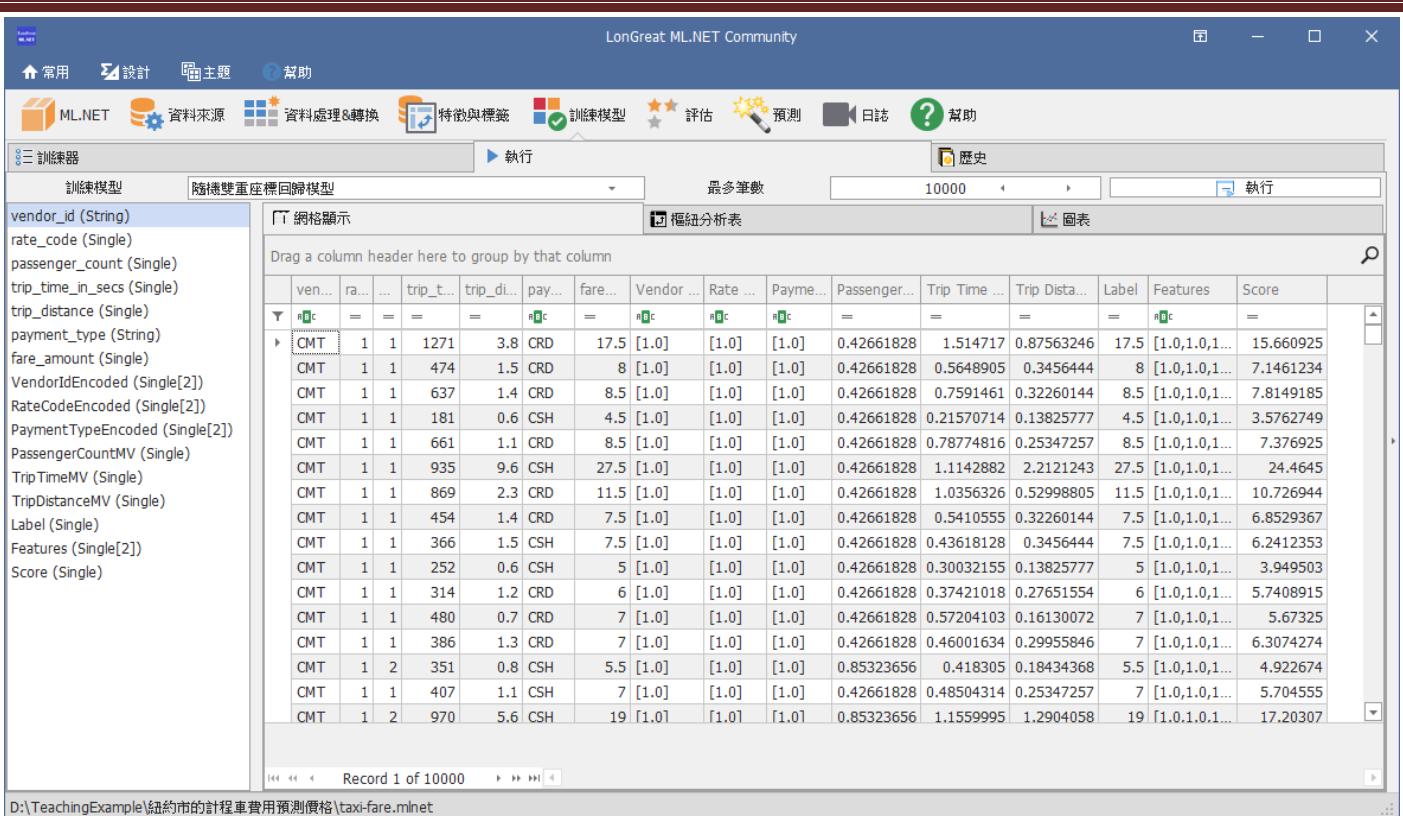


### 执行

1. 选择训练模型
2. 要显示最多笔数
3. 执行

结果多二个字段

- 特征 PredictedLabel
- 分数 Score



## 评估

- A. 选取以训练完合身的模型
- B. 选取设定数据须与原来相同
- C. 点击预测按钮=>产生出批次预测结果
- D. 点击评估按钮=>评估结果

資料	
平均絕對誤差(MAE)	<b>0.956015948805809</b>
判定係數	<b>0.8939507105933373</b>
均方誤差(MSE)	<b>9.539699943138512</b>
根均方誤差(RMSE)	<b>3.0886404684162434</b>
損失函數	<b>9.539699909326917</b>

**平均絕對誤差(MAE)**

絕對損失定義為  $L_1 = \frac{1}{m} \sum_{i=1}^m |y_i - \hat{y}_i|$ , 其中  $m$  是測試集中的實例數， $\hat{y}_i$  是每個實例的預測標籤，和  $y_i$  是每個實例的正確標籤。 $L_1$  損失是一個非負的遞減度量。較小的值表明該指標的模型更好。

# LonGreat ML.NET Community

The screenshot shows the ML.NET Studio interface with the following details:

- Top Bar:** Includes "常用" (Common), "設計" (Design), "暗黑主題" (Dark Theme), and "幫助" (Help).
- Toolbar:** Features icons for "ML.NET", "資料來源" (Data Sources), "資料處理&轉換" (Data Processing & Transformation), "特徵與標籤" (Features & Labels), "訓練模型" (Train Model), "評估" (Evaluate), "預測" (Predict), "日誌" (Log), and "幫助" (Help).
- Left Sidebar:** Shows the "評估" (Evaluate) section with a table of feature names and their types.
- Central Grid View:** Displays a grid of 100,000 records from a taxi fare prediction model. The columns include vendor\_id, rate\_code, passenger\_count, trip\_time\_in\_secs, trip\_distance, payment\_type, fare\_amount, VendorIdEncoded, RateCodeEncoded, PaymentTypeEncoded, PassengerCountMV, TripTimeMV, TripDistanceMV, Label, Features, and Score.
- Right Sidebar:**
  - 資料 (Data):** Shows summary statistics: 平均... 0.956015..., 累定... 0.893950..., 均方... 9.539699..., 根均... 3.088640..., 損失... 9.539699... .
  - 平均絕對誤差 (MAE):** Formula: \$L1 = \frac{1}{m} \sum\_{i=1}^m |y\_i - \hat{y}\_i|\$, where \$y\_i\$ is the test example's label, \$\hat{y}\_i\$ is the predicted value, and \$m\$ is the number of examples.
- Bottom Status Bar:** Shows "Record 1 of 100000" and the file path "D:\TeachingExample\紐約市的計程車費用預測價格\taxi-fare.mlnet".

## 鸢尾花花卉分类

### 了解问题

这个问题是关于将一组鸢尾花按照花卉特征分成不同的群组。这些特征是萼片的长度和宽度以及花瓣的长度和宽度。本教学课程中假设不知道每个花卉的类型。您想要从特征了解数据集的结构，还要预测数据实例如何符合此结构。

### 选取适当的机器学习工作

因为您不知道每个花卉属于哪个群组，所以您选择[非监督式机器学习](#)工作。若要按照类似元素归于同一群组的方式来分割群组中的数据集，请使用[群集](#)机器学习工作。

### 准备资料

1. *Iris.xlsx* 数据集，如需有关鸢尾花数据集的详细信息，请参阅[鸢尾花数据集](#)维基百科页面和[鸢尾花数据集](#)页面（也就是数据集的来源）。

*Iris.xlsx* 档案包含五个资料行，分别表示：

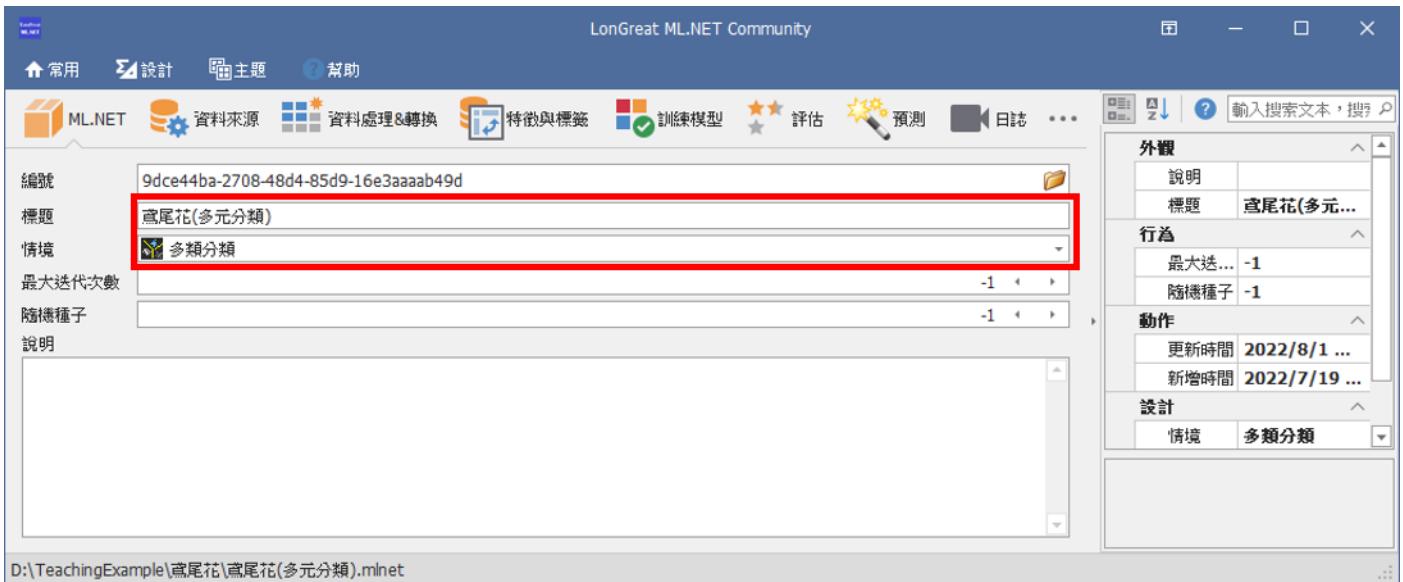
- 萼片长度 SepalLength (以公分为单位)
- 萼片宽度 SepalWidth (以公分为单位)
- 花瓣长度 PetalLength (以公分为单位)
- 花瓣宽度 PetalWidth (以公分为单位)
- 鸢尾花的类型 Class

### 鸢尾花(多元分类)

#### 建立项目

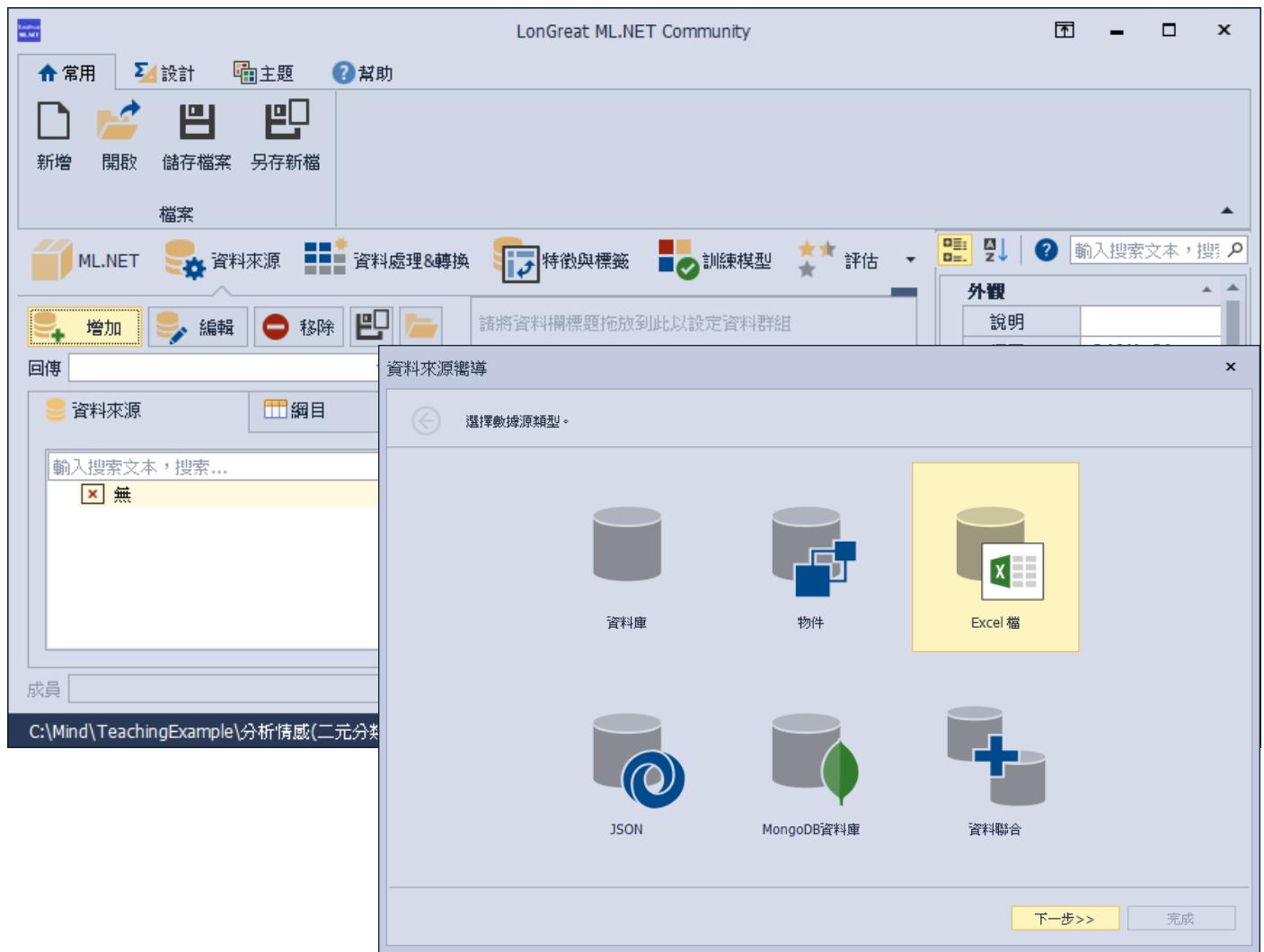
1. 输入标题 => 鸢尾花(多元分类)
2. 选择情境 => 多类分类

# LonGreat ML.NET Community

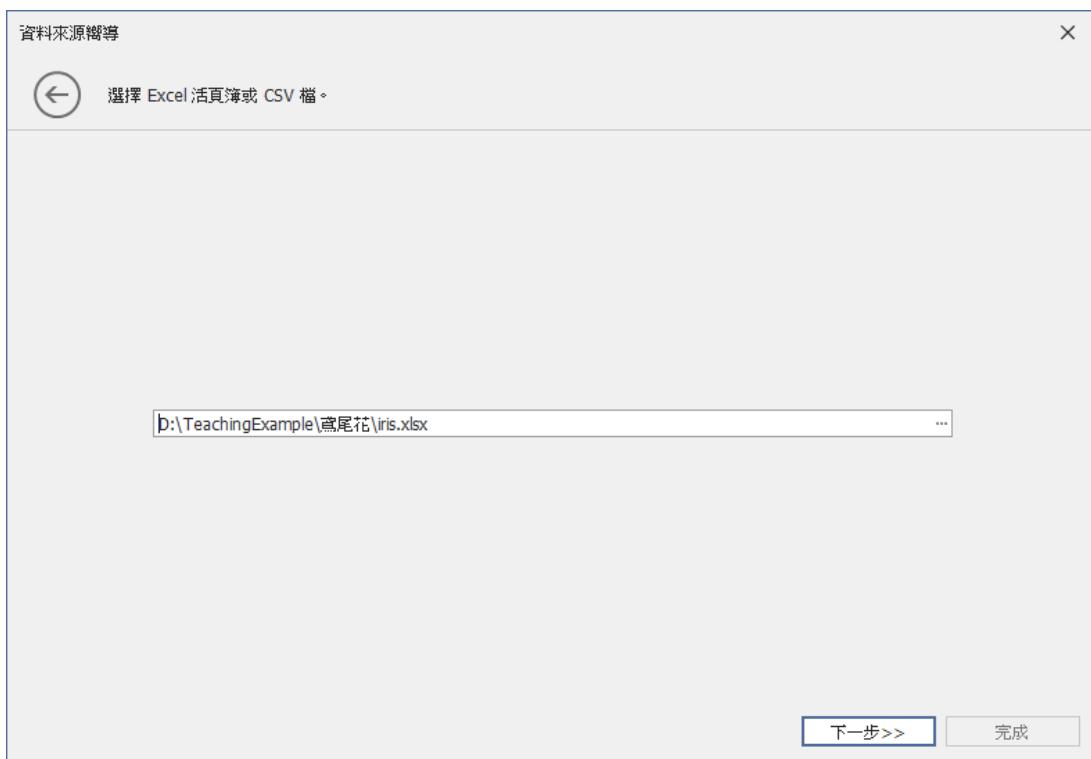


## 加载数据

数据源->增加->Excel 檔



选取 iris.xlsx

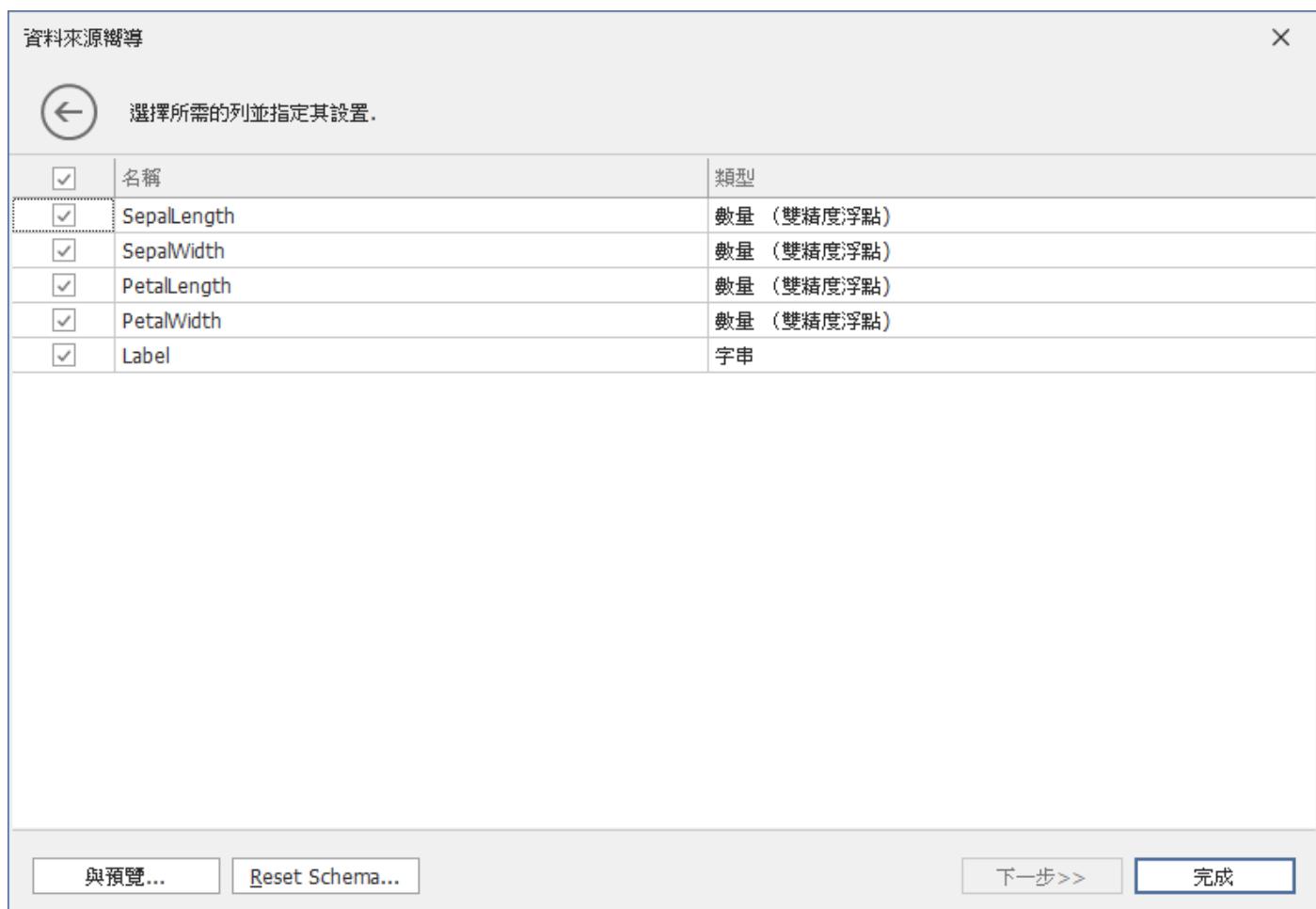


This screenshot shows the second and third steps of the 'Data Source Wizard'.  
Step 2: '指定導入設置。' (Specify import settings). It includes several checked options:

- 使用 first 行的值為欄位名稱 (Use the value of the first row as column names)
- Skip empty rows
- Skip hidden rows
- Skip hidden columns

  
Step 3: '資料來源嚮導' (Data Source Wizard). It displays a list of available sheets in the current Excel file, with 'iris' selected. Below the list are two buttons: '下一步>>' (Next >) and '完成' (Finish).

## 按完成



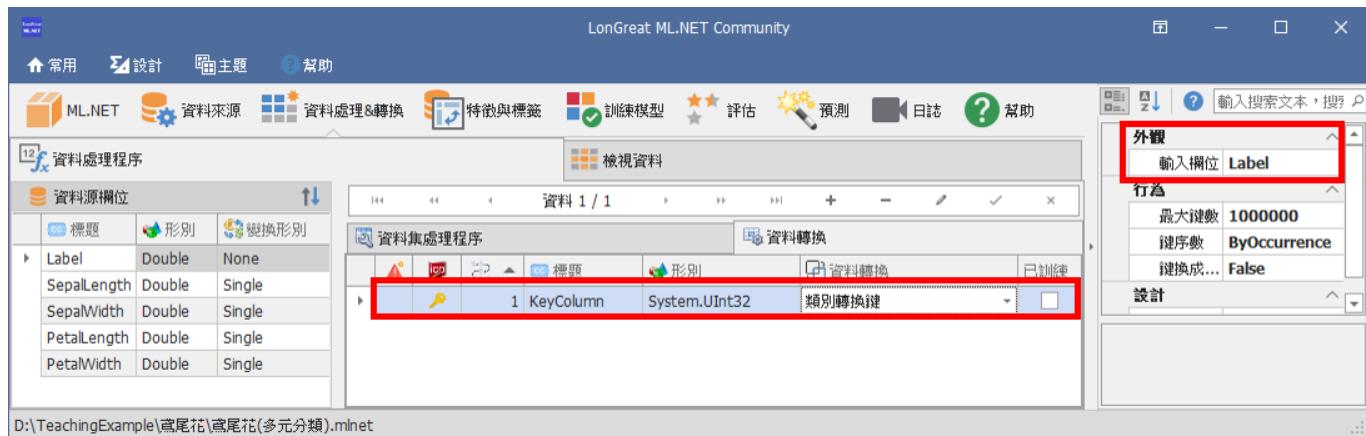
## 检视资料

	Sepal Length	Sepal Width	Petal Length	Petal Width	Label
1	5.1	3.5	1.4	0.2	Iris-setosa
2	4.9	3	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	3.6	1.4	0.2	Iris-setosa
6	5.4	3.9	1.7	0.4	Iris-setosa
7	4.6	3.4	1.4	0.3	Iris-setosa
8	5	3.4	1.5	0.2	Iris-setosa
9	4.4	2.9	1.4	0.2	Iris-setosa
10	4.9	3.1	1.5	0.1	Iris-setosa
11	5.4	3.7	1.5	0.2	Iris-setosa
12	4.8	3.4	1.6	0.2	Iris-setosa
13	4.8	3	1.4	0.1	Iris-setosa
14	4.3	3	1.1	0.1	Iris-setosa
15	5.8	4	1.2	0.2	Iris-setosa
16	5.7	4.4	1.5	0.4	Iris-setosa
17	5.4	3.0	1.3	0.4	Iris-setosa

## 数据转换

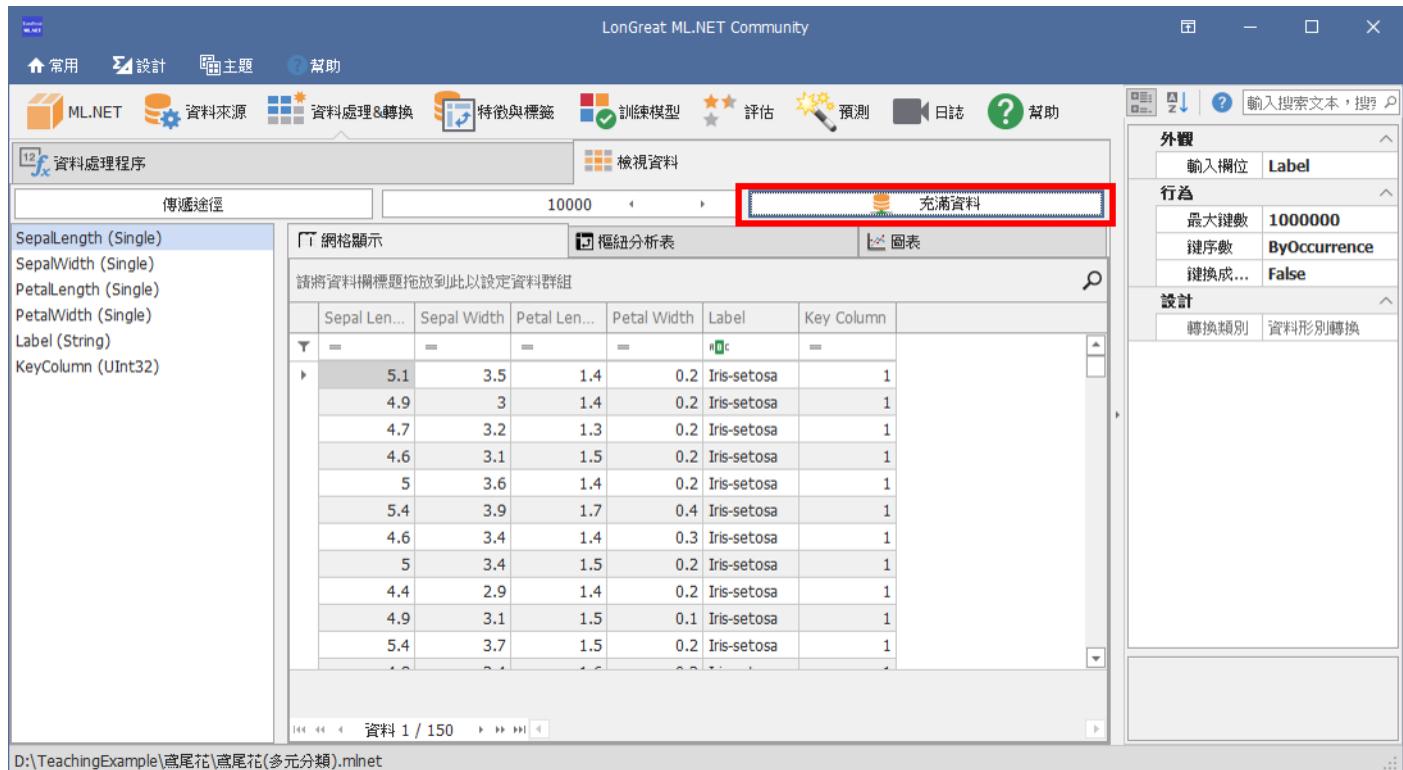
### 新增数据转换

#### 1. 类别转换键 => 将 Label 转换为键值



## 结果检视

- 至检视资料页签
- 选择显示笔数
- 按充满资料



## 特征与卷标

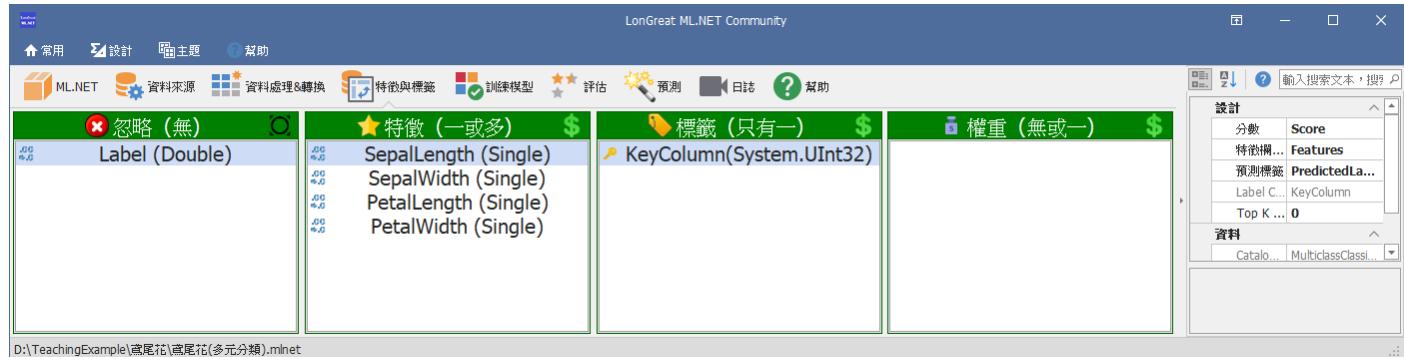
拖拉 SepalLength=>特征

拖拉 SepalWidth=>特征

拖拉 PetalLength=>特征

拖拉 PetalWidth=>特征

拖拉 KeyColumn=>标签



## 训练模型

### 训练器

按+新增

选择多元 L-BFGS 最大熵模型 (LbfgsMaximumEntropy)



## 执行

1. 选择训练模型
2. 要显示最多笔数
3. 执行

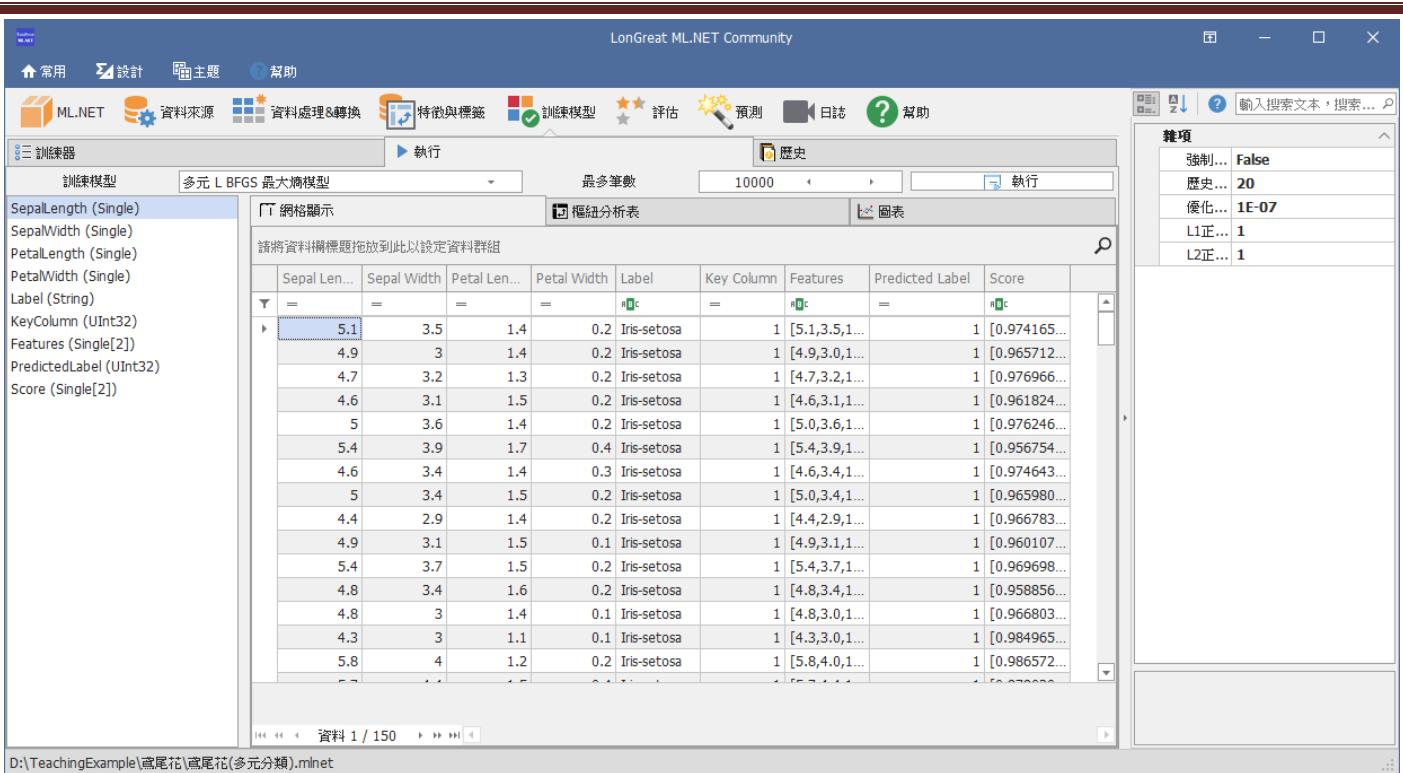
结果多三个字段

特征 Features

预测标签 PredictedLabel

分数 Score

# LonGreat ML.NET Community



## 评估

- A. 选取以训练完合身的模型
- B. 选取设定数据须与原来相同
- C. 点击预测按钮=>产生出批次预测结果
- D. 点击评估按钮=>评估结果

資料	
宏觀準確率	<b>0.9733333333333333</b>
每類對數損失	(Collection)
所有 K 的 Top-K 準確度	(Collection)
混淆矩陣	MindCore.MlNet.Catalogs.ReturnConfusionMatrix
分類數	3
每類召回	(Collection)
每類精度	(Collection)
筆數	(Collection)
微觀準確率	<b>0.973333333333334</b>
對數損失	<b>0.15254354312265206</b>
對數損失減少	<b>0.8611488832811196</b>
Top-K準確率	0
Top-K預測計數	0

### 宏觀準確率

宏觀平均值是類級別的平均準確度。 計算每個類的準確度，宏觀準確度是這些準確度的平均值。宏觀平均指標為每個類賦予相同的權重，無論數據集中包含該類的多少實例。

# LonGreat ML.NET Community

The screenshot shows the LonGreat ML.NET Community application interface. The main window displays a grid of data for the Iris dataset, with columns including Sepal Length, Sepal Width, Petal Length, Petal Width, Label, Key Column, Features, Predicted, and Score. A sidebar on the right provides detailed performance metrics, such as Macro Average Accuracy (0.973333333333333) and various collection counts (Collection, MindCore.MLNet.Cat...).

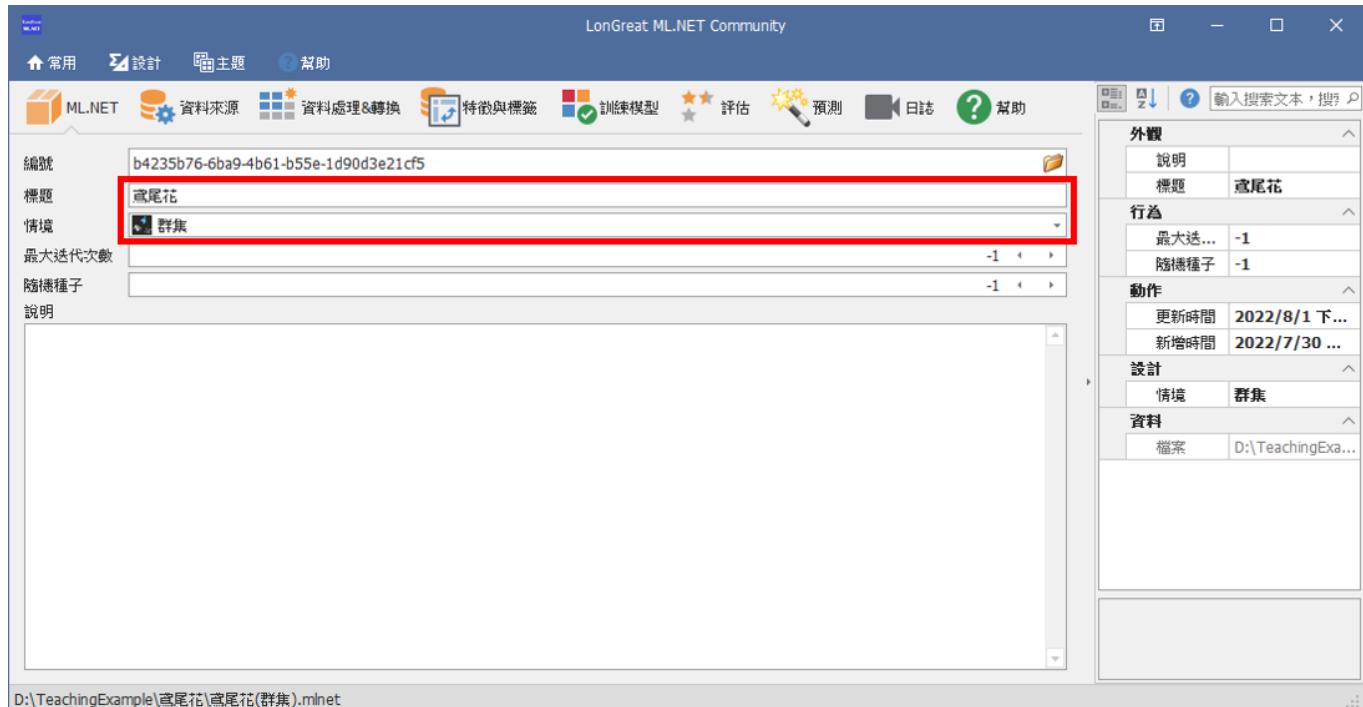
Sepal Length (Single)	Sepal Width (Single)	Petal Length (Single)	Petal Width (Single)	Label	Key Column	Features	Predicted	Score
5.1	3.5	1.4	0.2	Iris-setosa	1	[5.1,3.5,1...	1	[0.974165...
4.9	3	1.4	0.2	Iris-setosa	1	[4.9,3.0,1...	1	[0.965712...
4.7	3.2	1.3	0.2	Iris-setosa	1	[4.7,3.2,1...	1	[0.976966...
4.6	3.1	1.5	0.2	Iris-setosa	1	[4.6,3.1,1...	1	[0.961824...
5	3.6	1.4	0.2	Iris-setosa	1	[5.0,3.6,1...	1	[0.976246...
5.4	3.9	1.7	0.4	Iris-setosa	1	[5.4,3.9,1...	1	[0.956754...
4.6	3.4	1.4	0.3	Iris-setosa	1	[4.6,3.4,1...	1	[0.974643...
5	3.4	1.5	0.2	Iris-setosa	1	[5.0,3.4,1...	1	[0.965980...
4.4	2.9	1.4	0.2	Iris-setosa	1	[4.4,2.9,1...	1	[0.966783...
4.9	3.1	1.5	0.1	Iris-setosa	1	[4.9,3.1,1...	1	[0.960107...
5.4	3.7	1.5	0.2	Iris-setosa	1	[5.4,3.7,1...	1	[0.969698...
4.8	3.4	1.6	0.2	Iris-setosa	1	[4.8,3.4,1...	1	[0.958856...
4.8	3	1.4	0.1	Iris-setosa	1	[4.8,3.0,1...	1	[0.966803...
4.3	3	1.1	0.1	Iris-setosa	1	[4.3,3.0,1...	1	[0.984965...
5.8	4	1.2	0.2	Iris-setosa	1	[5.8,4.0,1...	1	[0.986572...
5.7	4.4	1.5	0.1	Iris-setosa	1	[5.7,4.4,1...	1	[0.970020...

D:\TeachingExample\鳶尾花\鳶尾花(多元分類).mlnet

## 鸢尾花(群集)

### 建立项目

1. 输入标题 => 鸢尾花
2. 选择情境 => 群集

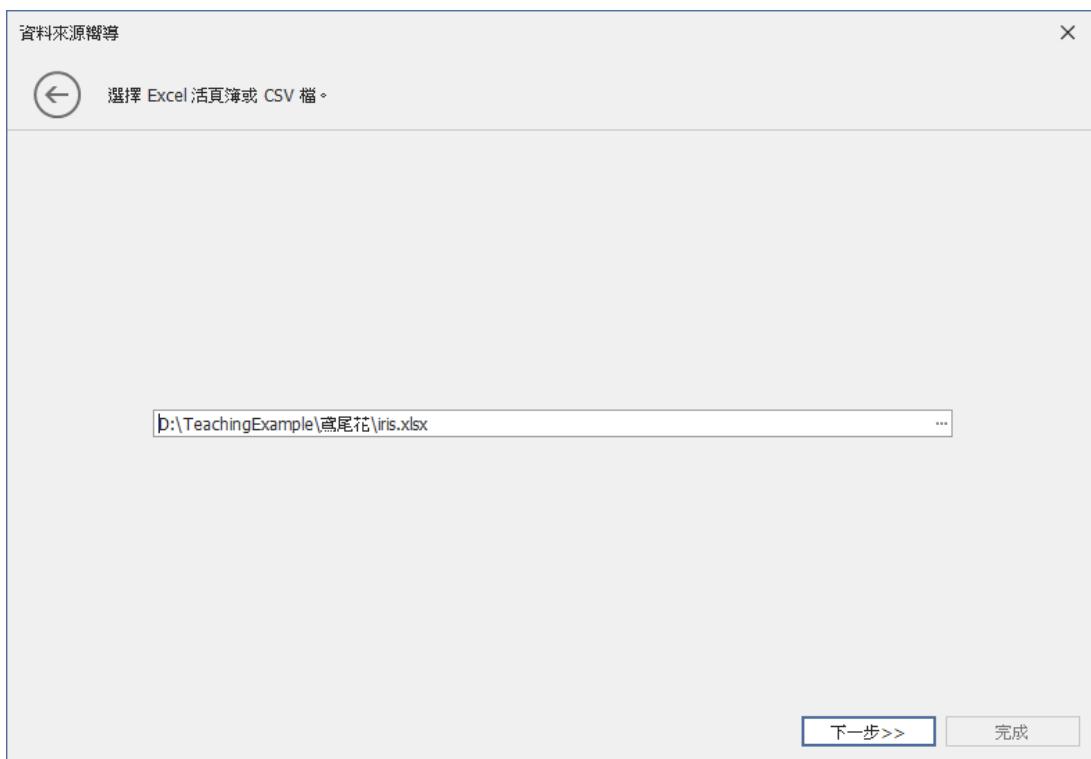


### 加载数据

数据源->增加->Excel 档



选取 iris.xlsx

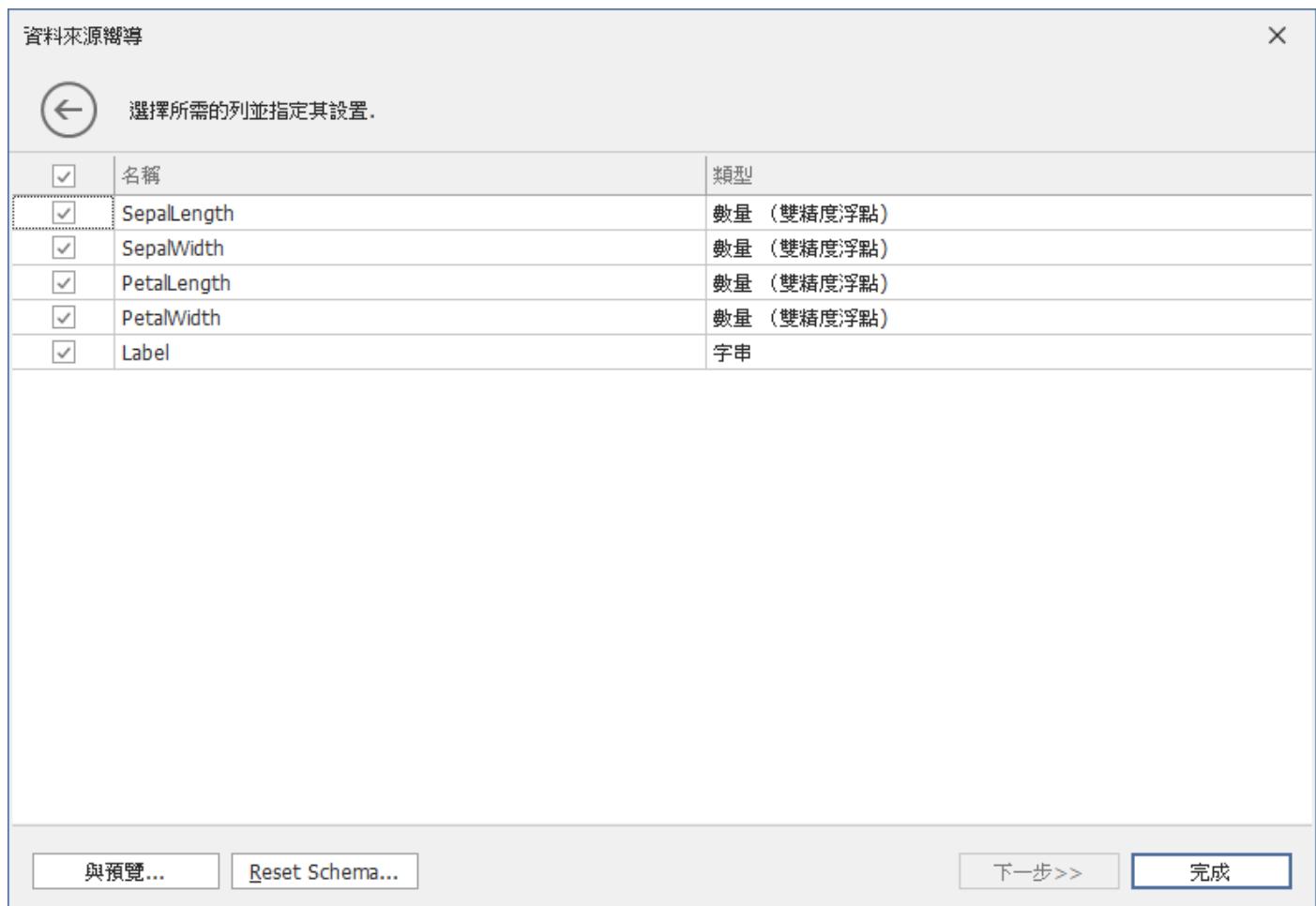


This screenshot shows the second and third steps of the 'Data Source Wizard'.  
Step 2 (Import Settings): The title bar says '資料來源嚮導' (Data Source Wizard). The main area contains several checked import options:

- 使用 first 行的值為欄位名稱 (Use the value of the first row as column names)
- Skip empty rows (Skip empty rows)
- Skip hidden rows (Skip hidden rows)
- Skip hidden columns (Skip hidden columns)

Below this is another 'Data Source Wizard' window titled '選擇所需的工作表、表或定義的名稱，指定的指定範圍內。' (Select the required worksheet, table or defined name, specify the specified range). It shows a list with 'iris' selected.  
Step 3 (Finish): The title bar says '資料來源嚮導' (Data Source Wizard). It contains the same import settings as Step 2 and the 'iris' sheet selection window. At the bottom are the '下一步>>' (Next >) and '完成' (Finish) buttons.

## 按完成



## 检视资料

	Sepal Length	Sepal Width	Petal Length	Petal Width	Label
=	5.1	3.5	1.4	0.2	Iris-setosa
=	4.9	3	1.4	0.2	Iris-setosa
=	4.7	3.2	1.3	0.2	Iris-setosa
=	4.6	3.1	1.5	0.2	Iris-setosa
=	5	3.6	1.4	0.2	Iris-setosa
=	5.4	3.9	1.7	0.4	Iris-setosa
=	4.6	3.4	1.4	0.3	Iris-setosa
=	5	3.4	1.5	0.2	Iris-setosa
=	4.4	2.9	1.4	0.2	Iris-setosa
=	4.9	3.1	1.5	0.1	Iris-setosa
=	5.4	3.7	1.5	0.2	Iris-setosa
=	4.8	3.4	1.6	0.2	Iris-setosa
=	4.8	3	1.4	0.1	Iris-setosa
=	4.3	3	1.1	0.1	Iris-setosa
=	5.8	4	1.2	0.2	Iris-setosa
=	5.7	4.4	1.5	0.4	Iris-setosa
=	5.4	2.0	1.3	0.1	Iris-setosa
					150

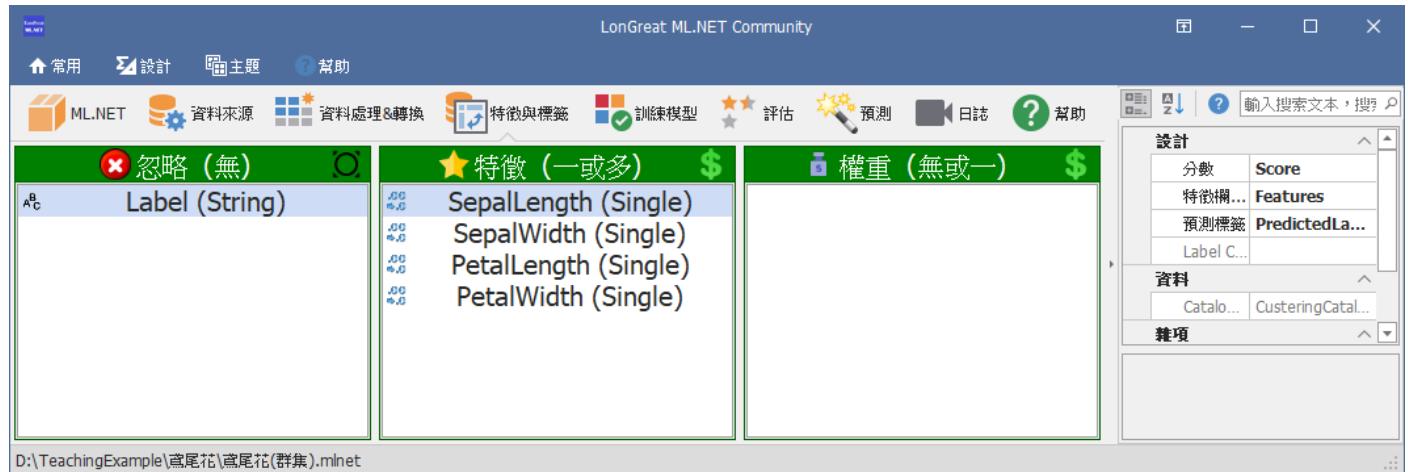
## 特征与卷标

拖拉 SepalLength=>特征

拖拉 SepalWidth=>特征

拖拉 PetalLength=>特征

拖拉 PetalWidth=>特征



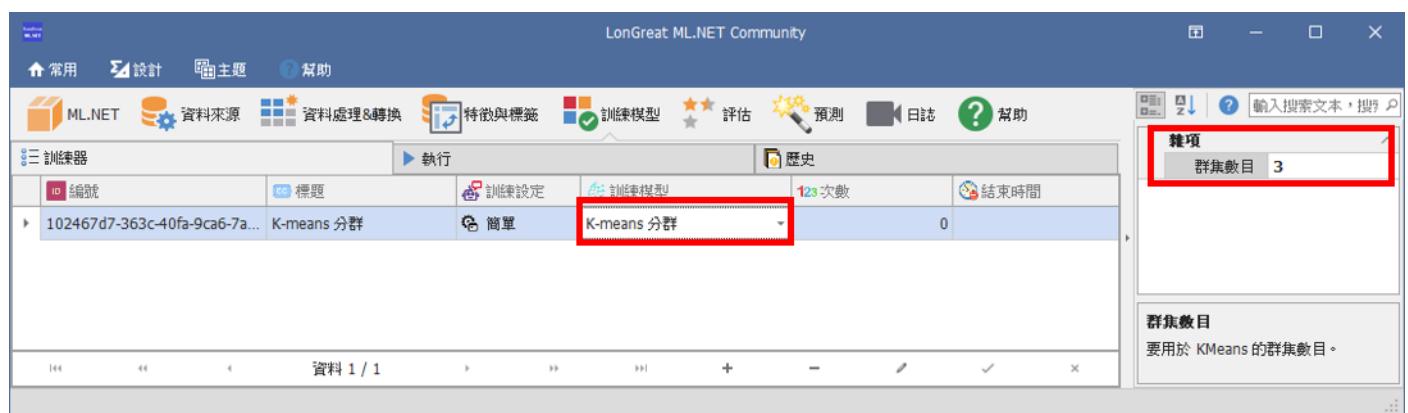
## 训练模型

### 训练器

按+新增

选择 K-means 分群(KMeans)

群集数目改为 3



## 执行

1. 选择训练模型
2. 要显示最多笔数
3. 执行

结果多三个字段

- 特征 Features
- 预测标签 PredictedLabel
- 分数 Score

Sepal Length	Sepal Width	Petal Length	Petal Width	Label	Features	Predicted Label	Score
5.1	3.5	1.4	0.2	Iris-setosa	[5.1,3.5,1...]	3	[11.64523...]
4.9	3	1.4	0.2	Iris-setosa	[4.9,3.0,1...]	3	[11.48966...]
4.7	3.2	1.3	0.2	Iris-setosa	[4.7,3.2,1...]	3	[12.67441...]
4.6	3.1	1.5	0.2	Iris-setosa	[4.6,3.1,1...]	3	[11.64392...]
5	3.6	1.4	0.2	Iris-setosa	[5.0,3.6,1...]	3	[11.97375...]
5.4	3.9	1.7	0.4	Iris-setosa	[5.4,3.9,1...]	3	[9.875401...]
4.6	3.4	1.4	0.3	Iris-setosa	[4.6,3.4,1...]	3	[12.30015...]
5	3.4	1.5	0.2	Iris-setosa	[5.0,3.4,1...]	3	[11.08245...]
4.4	2.9	1.4	0.2	Iris-setosa	[4.4,2.9,1...]	3	[12.68146...]
4.9	3.1	1.5	0.1	Iris-setosa	[4.9,3.1,1...]	3	[11.22063...]
5.4	3.7	1.5	0.2	Iris-setosa	[5.4,3.7,1...]	3	[11.02097...]

## 评估

- A. 选取以训练完合身的模型
- B. 选取设定数据须与原来相同
- C. 点击预测按钮=>产生出批次预测结果
- D. 点击评估按钮=>评估结果

資料	
平均距離	0.5263009389241536
標準化互資訊	非數值
戴維斯布爾丁指數	0.6663912498680068

## 平均距離

平均分。對於 K-Means 算法，'score' 是從質心到示例的距離。因此，平均分數是示例與聚類質心的接近程度的量度。換句話說，它是“聚類緊密度”的量度。但是請注意，該指標僅在集群數量增加時才會減少，並且在極端情況下（每個不同的示例都是其自己的集群），它將等於零。

The screenshot shows the ML.NET Studio interface with the 'Evaluation' tab selected. The main workspace displays a grid of data points from the Iris dataset, categorized into three clusters (Iris-setosa). The columns represent Sepal Length, Sepal Width, Petal Length, Petal Width, Label (Iris-setosa), Features, Predicted Label, and Score. The sidebar on the left lists the features used in the model: SepalLength (Single), SepalWidth (Single), PetalLength (Single), PetalWidth (Single), Label (String), Features (Single[2]), PredictedLabel (UInt32), and Score (Single[2]). The top-right panel shows the following summary statistics:

平均距離	0.5263009389241536
標準化互資訊	非數值
戴維斯布爾丁指數	0.6663912498680068

## 推荐的电影(矩阵分解模型)

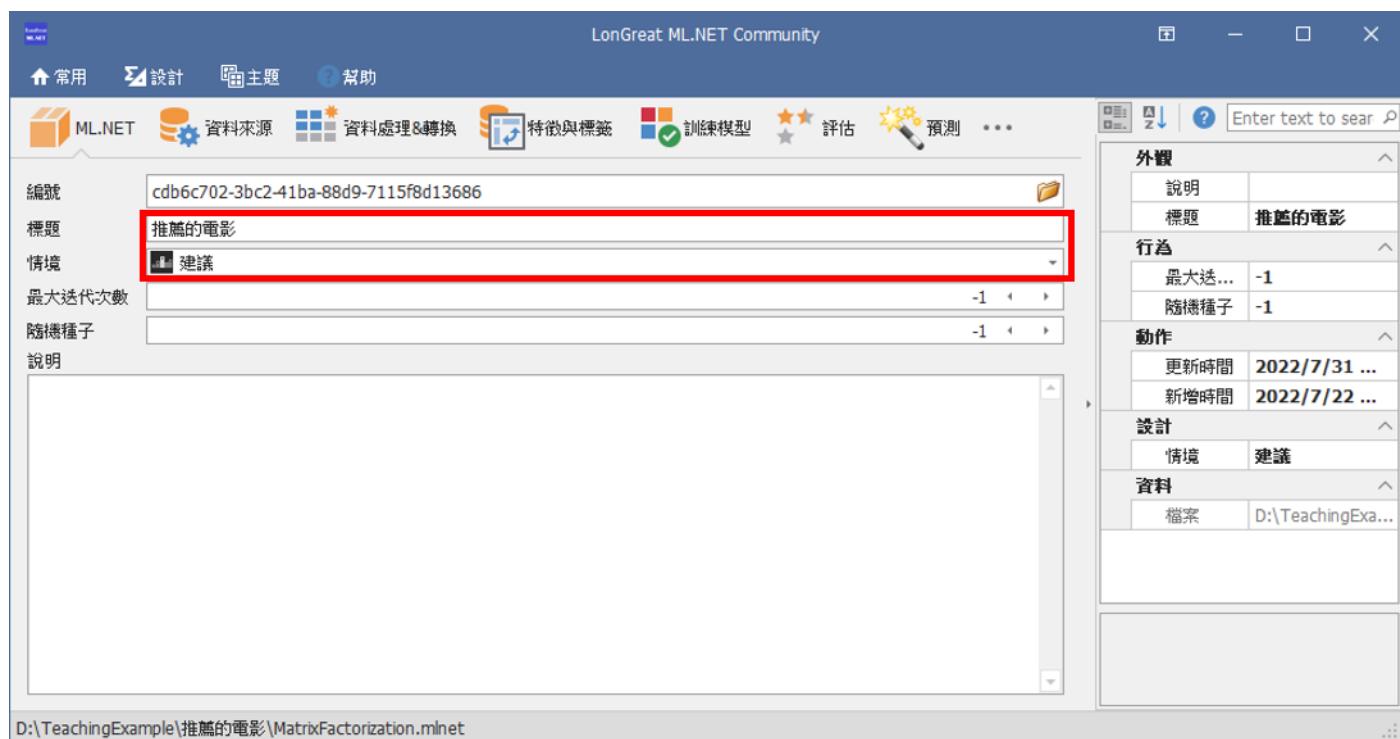
根据先前使用者的评分产生电影建议

### 选取适当的机器学习工作

有数种方式可以解决推荐问题，例如推荐电影列表或推荐相关产品列表；但在此情况下，您将预测使用者会给予该电影的评等 (1-5)，如果特定电影的预测评等高于所定义阈值，即推荐该电影（评等愈高，使用者喜欢特定电影的可能性愈高）。

### 建立项目

1. 输入标题 => 推荐的电影
2. 选择情境 => 建议



## 加载您的数据

以下是 \*.csv 档案中数据的预览:

userId	movieId	rating	timestamp
1	1	4	964982703
1	3	4	964981247
1	6	4	964982224
1	47	5	964983815
1	50	5	964982931
1	70	3	964982400
1	101	5	964980868
1	110	4	964982176
1	151	5	964984041

在 \*.csv 档案中，有四个数据行:

- userId
- movieId
- rating
- timestamp

在机器学习服务中，用来进行预测的数据行称为[特征](#)，而传回预测的资料行称为[标签](#)。

您希望预测电影评等，因此评等数据行是 Label。 其他三个数据行 userId、movieId 和 timestamp 都是 Features，用来预测 Label。

### 功能

userId

movieId

timestamp

### 标签

rating

由您决定使用哪些 Features 来预测 Label。 您也可以使用 [排列特征重要性](#) 等方法，协助选取最佳 Features。

在此情况下，您应该排除 timestamp 数据行为 Feature，因为时间戳并不会实际影响使用者对特定影片的评分方式，因此无法提供更精确的预测：

## 功能

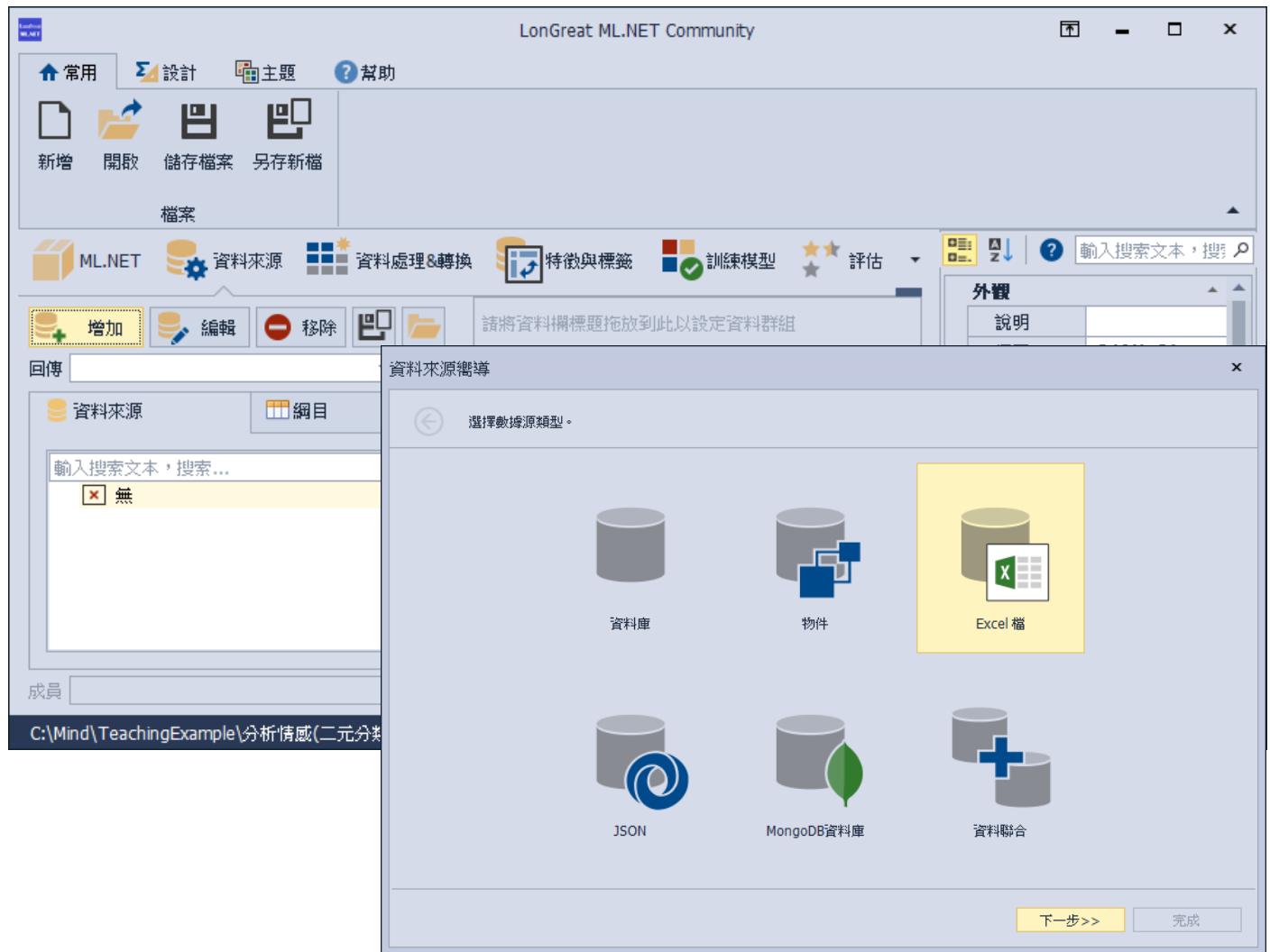
userId

movieId

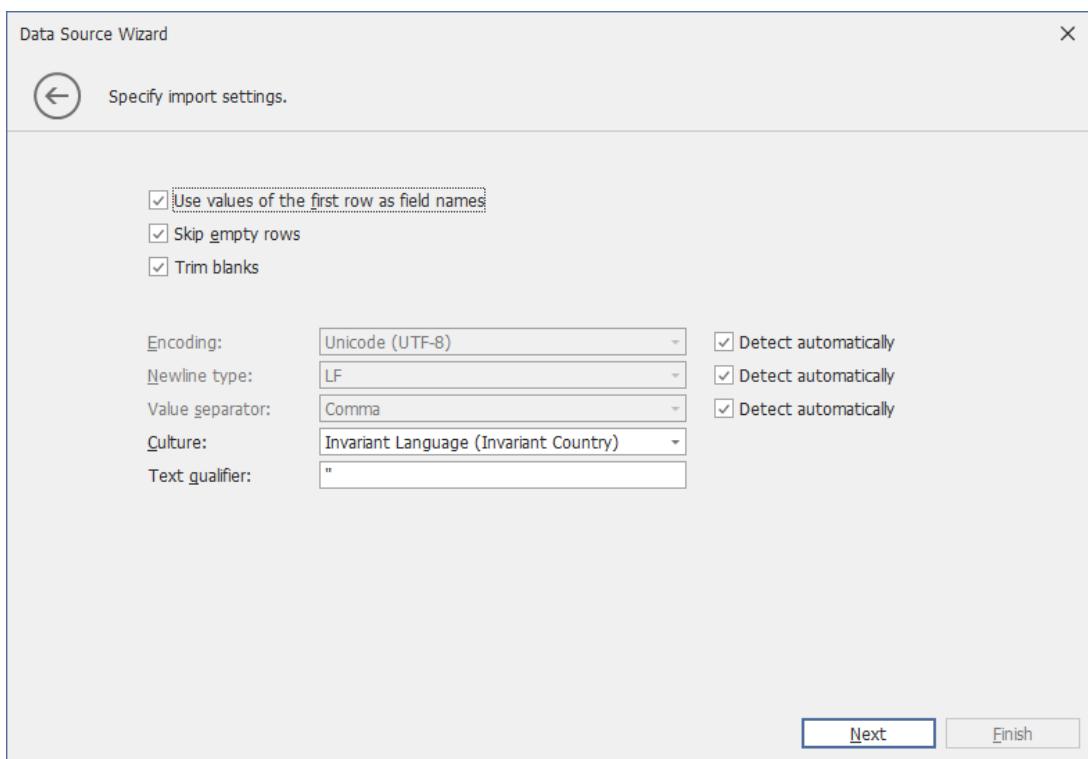
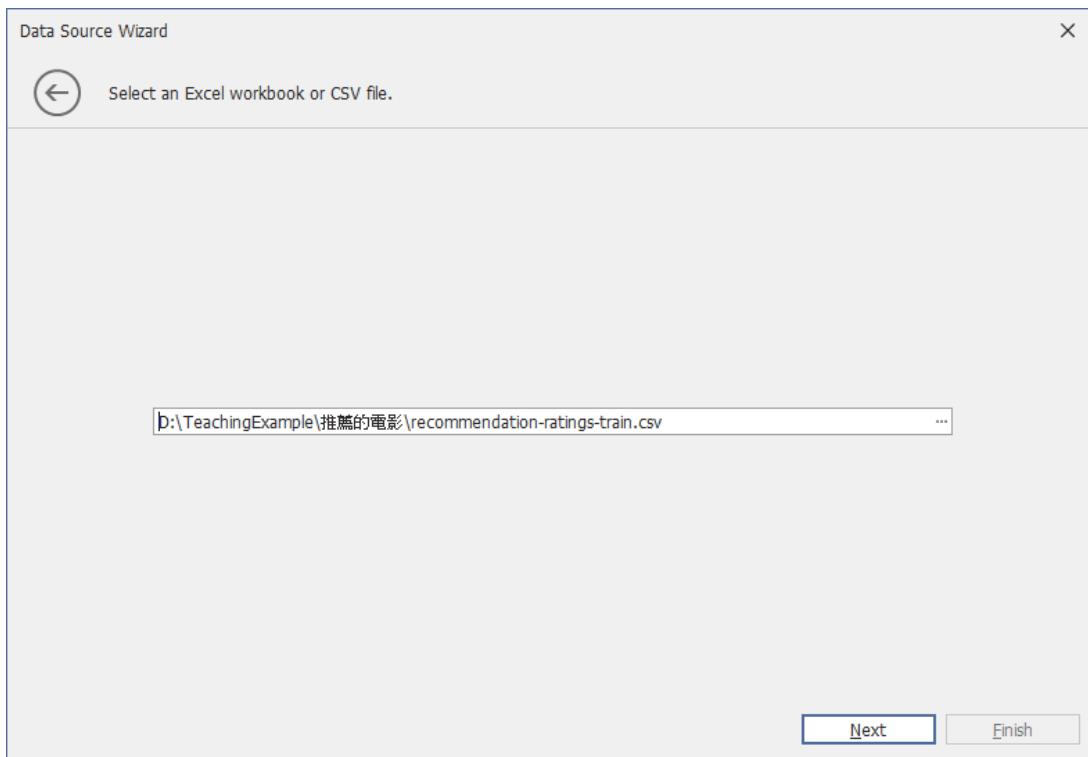
## 标签

rating

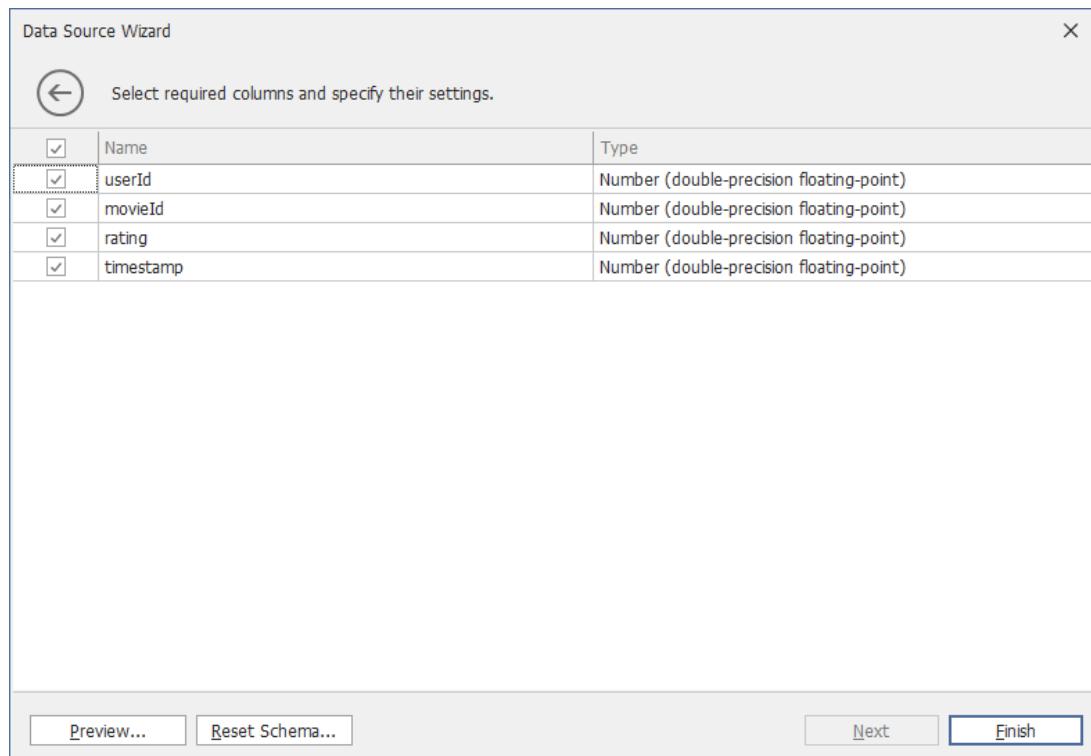
数据源->增加->Excel 檔



选取 recommendation-ratings-train.csv



## 按完成



## 检视资料

The screenshot shows the ML.NET Studio interface. On the left, there's a sidebar with 'ML.NET' tools like '增加' (Add), '編輯' (Edit), '移除' (Delete), and a dropdown for 'excelDataSource1'. A red box highlights the eye icon next to the dropdown. The main area is a data grid titled 'Drag a column header here to group by that column' with columns 'user Id', 'movie Id', 'rating', and 'timestamp'. The data grid contains numerous rows of movie ratings. To the right, there's a detailed sidebar for a specific movie entry:

- 外觀**: 說明 (Description: 推薦的電影), 標題 (Title: 推薦的電影)
- 行為**: 最大值 (-1), 隨機種子 (-1)
- 動作**: 更新時間 (Last Update: 2022/7/31 ...), 新增時間 (Created: 2022/7/22 ...)
- 設計**: 情境 (Scenario: 建議), 資料 (Data: D:\TeachingExa...)

The status bar at the bottom shows the path 'D:\TeachingExample\推薦的電影\MatrixFactorization.mlnet'.

## 数据转换

由于 userId 和 movieId 代表使用者与电影标题，而非真正的值，所以您会使用 [MapValueToKey\(\)](#) 方法来将每个 userId 和每个 movieId 转换成数值索引键类型 Feature 数据行（推荐算法所接受的格式），并将其新增为新的数据集资料行：

userId	movieId	标签	userIdEncoded	movieIdEncoded
1	1	4	userKey1	movieKey1
1	3	4	userKey1	movieKey2
1	6	4	userKey1	movieKey3

### 新增数据转换

#### 1. 类别转换建 => 将 userId 转换为键值

D:\TeachingExample\推薦的電影\MatrixFactorization.mlnet

#### 2. 类别转换建 => 将 movieId 转换为键值

D:\TeachingExample\推薦的電影\MatrixFactorization.mlnet

## 结果检视

#### I. 至检视资料页签

## II. 选择显示笔数

## III. 按充满资料

## 特征与卷标

拖拉 rating=>标签

拖拉 userIdEncoded=>栏列

拖拉 movieIdEncoded=>笔行

## 训练模型

### 训练器

矩阵分解模型是您的推荐定型算法。 当您拥有用户过去如何评等产品的数据时，[矩阵分解](#)是推荐的常见方法，此亦为本教学课程数据集的情况。 当您有不同的可用数据时，也有其他推荐算法（请参阅[其他推荐算法](#)一节以深入了解）。

在此案例中，Matrix Factorization 算法使用的方法称为「共同筛选」，此方法假设如果使用者 1 与使用者 2 对特定问题具有相同的意见，则使用者 1 对其他问题的想法较可能与使用者 2 相同。

比方说，如果使用者 1 对电影的评分与使用者 2 类似，则使用者 2 较可能享受使用者 1 已观看并给予高度评分的电影：

	Incredibles 2 (2018)	The Avengers (2012)	Guardians of the Galaxy (2014)
使用者 1	已观看及已按赞的电影	已观看及已按赞的电影	已观看及已按赞的电影
使用者 2	已观看及已按赞的电影	已观看及已按赞的电影	尚未观看 -- 推荐电影

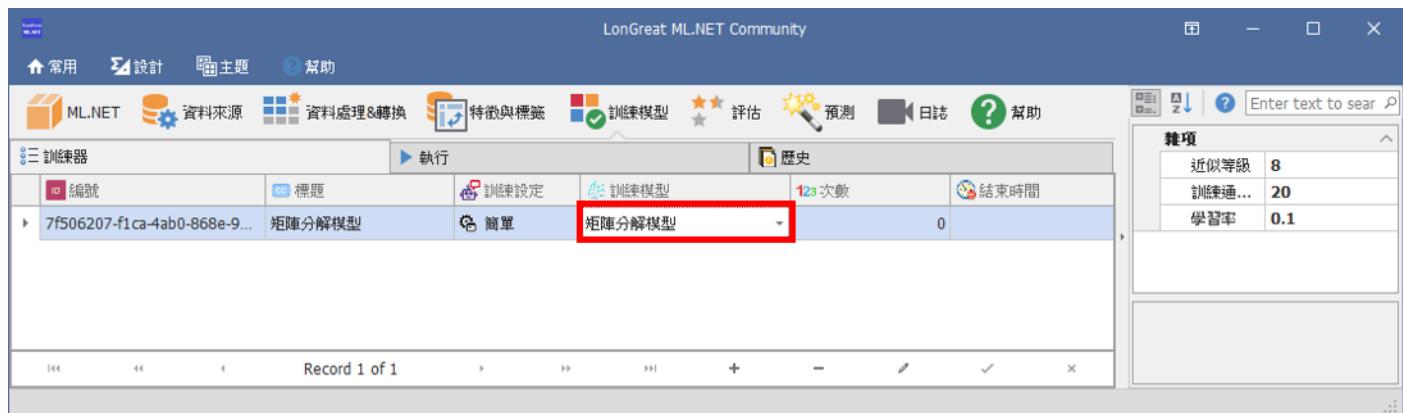
您可以在以下算法参数

NumberOfIterations = 20,

ApproximationRank = 100

按+新增

选择矩阵分解模型(MatrixFactorization)



执行

1. 选择训练模型
2. 要显示最多笔数

## 3. 执行

结果多一个字段

分数 Score

The screenshot shows the ML.NET Studio interface with the following details:

- Top Bar:** Includes tabs for 常用 (常用), 設計 (Design), 主題 (Theme), 幫助 (Help), and several ML.NET specific tabs: 資料來源 (Data Source), 資料處理&轉換 (Data Processing & Transformation), 特徵與標籤 (Features & Labels), 訓練模型 (Train Model), 評估 (Evaluate), 預測 (Predict), 日誌 (Logs), and a search bar.
- Left Sidebar:** Shows the "訓練模型" (Training Model) section with "矩陣分解模型" (Matrix Factorization) selected. It lists columns: userId (Single), movieId (Single), rating (Single), timestamp (Single), userIdEncoded (UInt32), movieIdEncoded (UInt32), and Score (Single).
- Central Grid:** Displays a table with the following columns: user Id, movie Id, rating, timestamp, user Id En..., movie Id ..., and Score. The data shows 10000 records, with the first few rows being:
 

user Id	movie Id	rating	timestamp	user Id En...	movie Id ...	Score
1	1	4	964982700	1	1	4.6904664
1	3	4	964981250	1	2	3.840612
1	6	4	964982200	1	3	4.5491705
1	47	5	964983800	1	4	4.7377234
1	50	5	964982900	1	5	4.978429
1	70	3	964982400	1	6	3.981197
1	101	5	964980860	1	7	4.4167533
1	110	4	964982140	1	8	4.7335863
1	151	5	964984060	1	9	4.29742
1	157	5	964984100	1	10	3.833571
1	163	5	964983700	1	11	4.2958946
1	216	5	964981200	1	12	4.0183554
1	223	3	964981000	1	13	4.456947
1	231	5	964981200	1	14	3.6249235
1	235	4	964980900	1	15	4.1606374
1	260	5	964981700	1	16	5.101451
1	296	3	964983000	1	17	4.914312
- Right Panel:** Shows configuration settings for "近似等級" (Approximation Level) set to 100, "訓練通..." (Training Iterations) set to 20, and "學習率" (Learning Rate) set to 0.1.
- Bottom Status:** Shows the path D:\TeachingExample\推薦的電影\MatrixFactorization.mlnet and the message "Record 1 of 10000".

## 其他建议算法

具备共同筛选的矩阵分解算法，仅为执行电影推荐的其中一种方法。在许多情况下，您可能会没有可用的评等数据，并只有用户的电影观看记录。而在其他情况下，您拥有的数据可能不只是用户评等数据。

算法	案例
单一类别矩阵分解	当您只需要 userId 和 movieId 时，请使用此选项。此推荐类型乃根据共同采购案例或经常同时购买的产品，也就是会根据客户自己的采购订单记录向客户推荐一组产品。
字段感知分解机器	当您所拥有的功能多于 userId、productId 和评等（如产品描述或产品价格）时，请使用此选项来进行推荐。此方法也会使用共同作业筛选方法。

## 评估

- A. 选取以训练完合身的模型
- B. 选取设定数据须与原来相同
- C. 点击预测按钮=>产生出批次预测结果
- D. 点击评估按钮=>评估结果

**資料**

平均絕對誤差(MAE)	0.4829922852118623
判定係數	0.6410528044739426
均方誤差(MSE)	0.3913956413984396
根均方誤差(RMSE)	0.6256162093475837
損失函數	0.3913956413638045

**平均絕對誤差(MAE)**

絕對損失定義為  $L1 = \frac{1}{m} \sum_{i=1}^m |y_i - \hat{y}_i|$ , 其中  $m$  是測試集中的實例數,  $\hat{y}_i$  是每個實例的預測標籤, 和  $y_i$  是每個實例的正確標籤。L1 損失是一個非負的遞減度量。較小的值表明該指標的模型更好。

LonGreat ML.NET Community

評估

訓練合身 2022-07-31-21-40-58 矩陣分解模型

網格顯示

網格顯示 標籤分析表 圖表

Drag a column header here to group by that column

user Id	movie Id	rating	timestamp	user Id En...	movie Id ...	Score
1	1	4	964982700	1	1	4.6904664
1	3	4	964981250	1	2	3.840612
1	6	4	964982200	1	3	4.5491705
1	47	5	964983800	1	4	4.7377234
1	50	5	964982900	1	5	4.978429
1	70	3	964982400	1	6	3.981197
1	101	5	964980860	1	7	4.4167533
1	110	4	964982140	1	8	4.7335863
1	151	5	964984060	1	9	4.29742
1	157	5	964984100	1	10	3.833571
1	163	5	964983700	1	11	4.2958946
1	216	5	964981200	1	12	4.0183554
1	223	3	964981000	1	13	4.456947
1	231	5	964981200	1	14	3.6249235

Record 1 of 99979

**資料**

平均絕對誤差(MAE)	0.4829922852118623
判定係數	0.6410528044739426
均方誤差(MSE)	0.3913956413984396
根均方誤差(RMSE)	0.6256162093475837
損失函數	0.3913956413638045

**平均絕對誤差(MAE)**

絕對損失定義為  $L1 = \frac{1}{m} \sum_{i=1}^m |y_i - \hat{y}_i|$ , 其中  $m$  是測試集中的實例數,  $\hat{y}_i$  是每個實例的預測標籤, 和  $y_i$  是每個實例的正確標籤。L1 損失是一個非負的遞減度量。較小的值表明該指標的模型更好。

## 花瓣识别(影像分类)

示范如何使用定型现有的 TensorFlow 模型，以建立自定义影像分类器。

### 影像分类传输学习范例概观

可使用预先定型的深度学习 TensorFlow 模型来分类影像。

### 了解问题

影像分类是计算机视觉问题。 影像分类会采用影像作为输入，并将其分类为指定的类别。 影像分类模型通常会使用深度学习和神经网络来定型。 如需详细信息，[请参阅深度学习与机器学习](#)。

影像分类很有用的一些案例包括：

- 脸部辨识
- 情绪侦测
- 医疗诊断
- 地标侦测

本教学课程会训练自定义影像分类模型，以执行花品总辨识的自动化视觉检查，以识别因破解而损毁的结构。

### ML.NET 影像分类 API

ML.NET 提供各种执行影像分类的方式。 本教学课程会使用影像分类 API 来套用传输学习。 影像分类 API 会使用 [TensorFlow.NET](#)，这是提供 TensorFlow C++ API C# 低阶链接库。

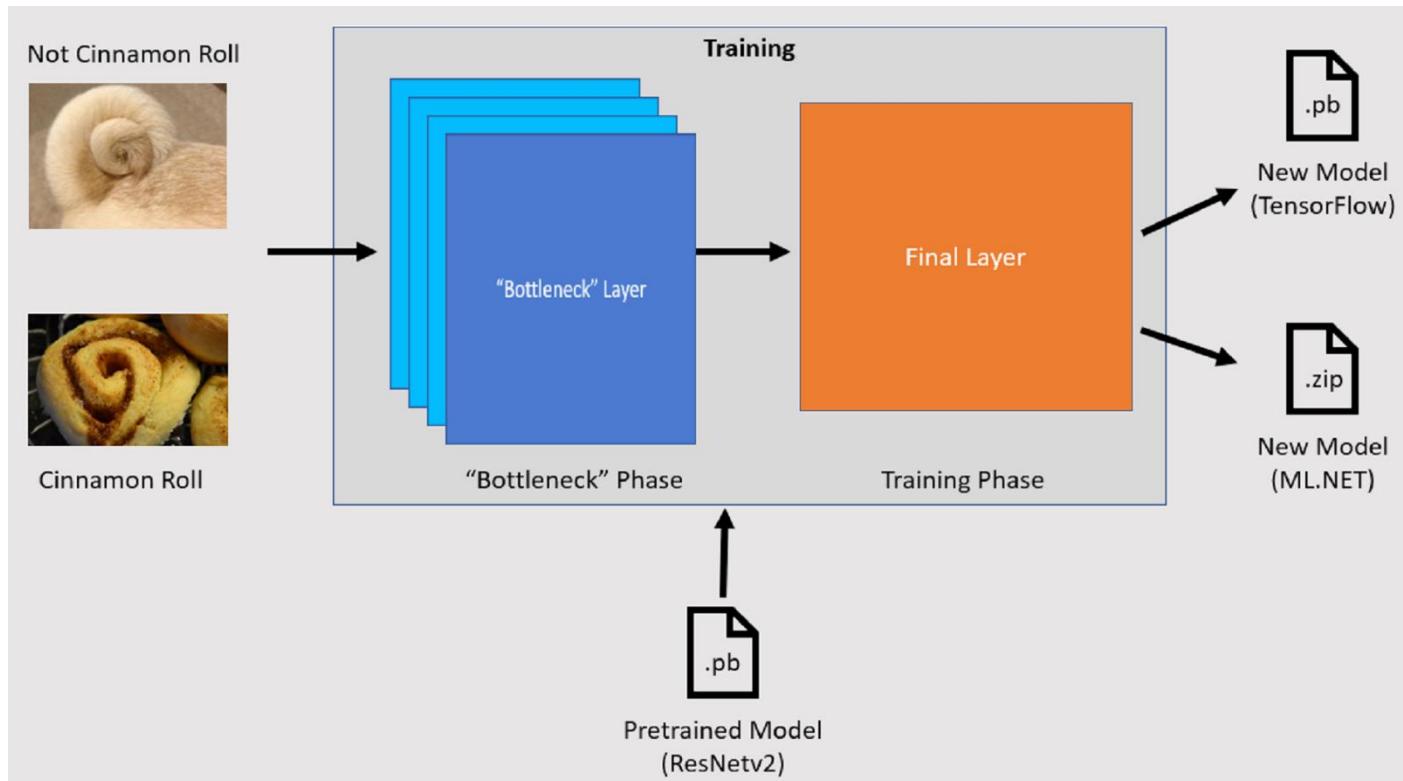
### 什么是传输学习？

转移学习会套用从解决一个问题到另一个相关问题所得到的知识。从头开始定型深度学习模型需要设定数个参数、大量加上卷标的定型数据，以及大量的计算资源，（数百个 GPU 时数）。 使用预先定型的模型以及传输学习可让您将定型程序快捷方式。

## 训练程序

影像分类 API 会加载预先定型的 TensorFlow 模型，以启动定型程序。 定型程序包含两个步骤：

1. 瓶颈阶段
2. 训练阶段



### 瓶颈阶段

在瓶颈阶段，会加载定型影像集，并将像素值当做预先定型模型的冻结层输入或特征使用。 冻结层包含神经网络中的所有层，最多到第二层，非正式称为瓶颈层。 这些层称为冻结，因为这些层上不会发生任何定型，而且作业是传递的。 其位于这些冻结层中，可协助模型区分不同类别的较低层级模式。 层数目愈大，此步骤的计算密集程度就越高。 幸运的是，由于这是一次性计算，因此可以在试验不同的参数时快取结果，并在稍后执行时使用。

### 训练阶段

一旦计算瓶颈阶段的输出值，就会用来作为输入来重新定型模型的最后一层。 此程序是反复的，会针对模型参数所指定的次数执行。 在每次执行期间，都会评估遗失和精确度。 然后，会进行适当的

调整，以改善模型，目标是将损失降至最低，并将精确度最大化。定型完成后，会输出两种模型格式。其中一个是 .pb 模型的版本，另一个是 .zip 模型 ML.NET 串行化版本。在 ML.NET 支持的环境中工作时，建议使用 .zip 模型的版本。不过，在不支援 ML.NET 的环境中，您可以选择使用 .pb 版本。

## 了解预先定型的模型

本教学课程中使用的预先定型模型是剩余网络 (ResNet) v2 模型的 101 层变体。原始模型已定型，以将影像分类成千个类别。模型会接受大小为 224 x 224 的影像做为输入，并输出其定型的每个类别的类别机率。此模型的一部分是用来使用自定义影像来定型新的模型，以在两个类别之间进行预测。

## 准备并了解资料



向日葵



玫瑰



蒲公英



雏菊



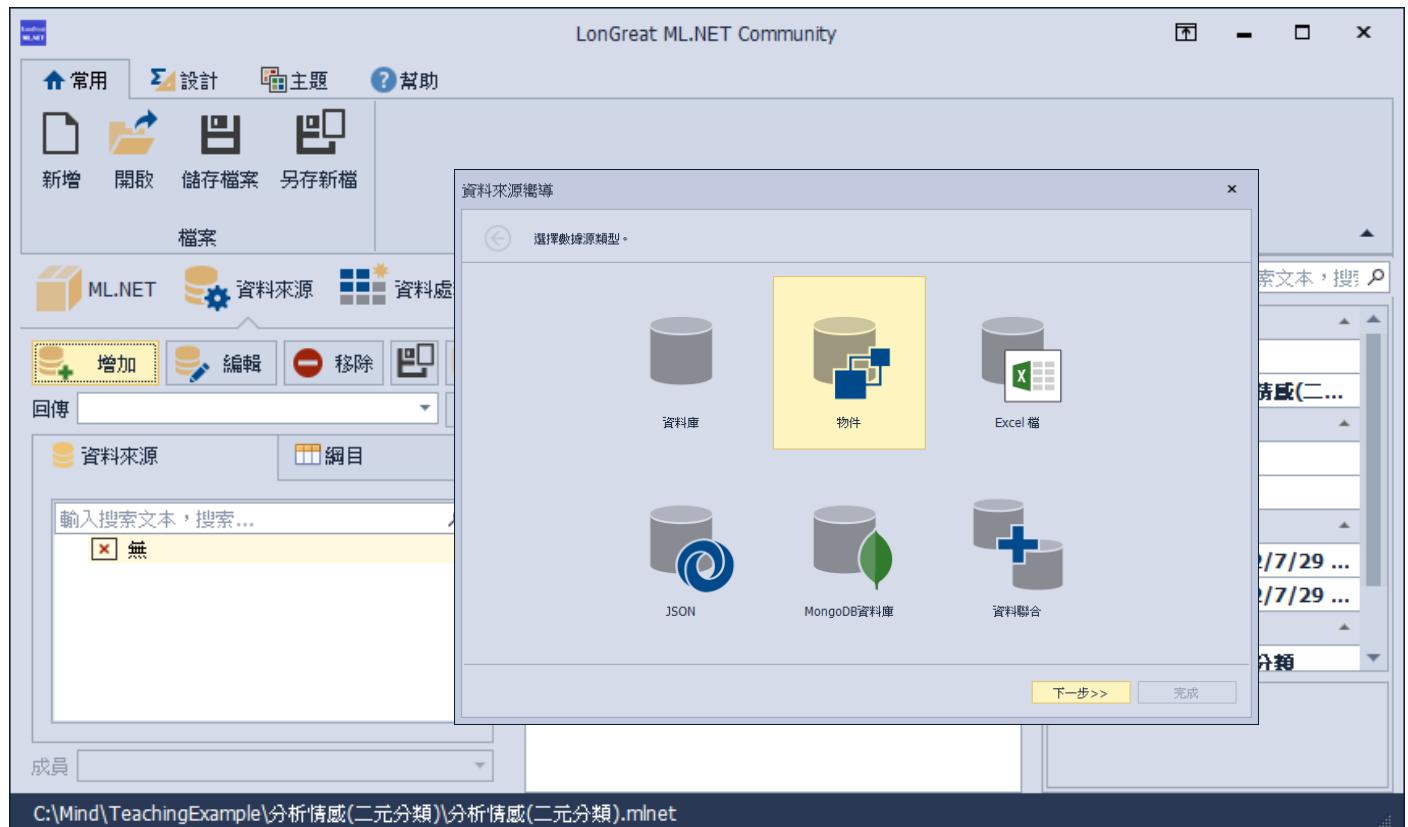
害金香

40411019\_526f  
3fc8d9\_m.jpg40411100\_7fbe  
10ec0f\_n.jpg44079668\_34df  
ee3da1\_n.jpg45045003\_30bb  
d0a142\_m.jpg45045005\_5735  
4ee844.jpg50987813\_7484  
bfbcdf.jpg58636535\_bc53  
ef0a21\_m.jpg127192624\_afab  
3d9cb84.jpg145303599\_262  
7e23815\_n.jpg147804446\_ef9  
244c8ce\_m.jpg151898652\_b5f  
1c70b98\_n.jpg164668737\_aea  
b0cb55e\_n.jpg164670176\_9f5  
b9c7965.jpg164670455\_29d  
8e02bbd\_n.jpg164671753\_ab3  
6d9cbb7\_n.jpg164672339\_f2b  
5b164f6.jpg175638423\_058  
c07afb9.jpg184682095\_46f  
8607278.jpg

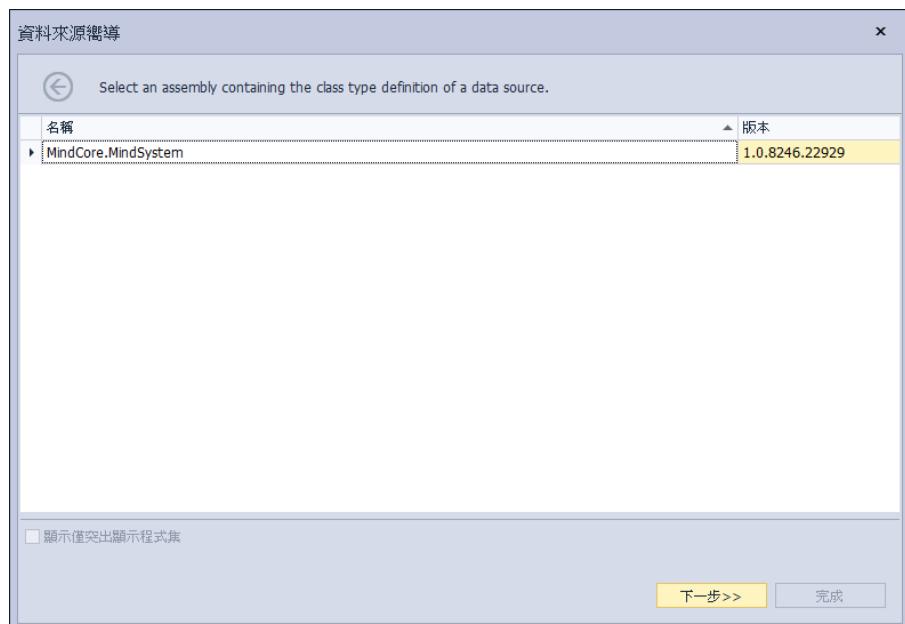


## 加载数据

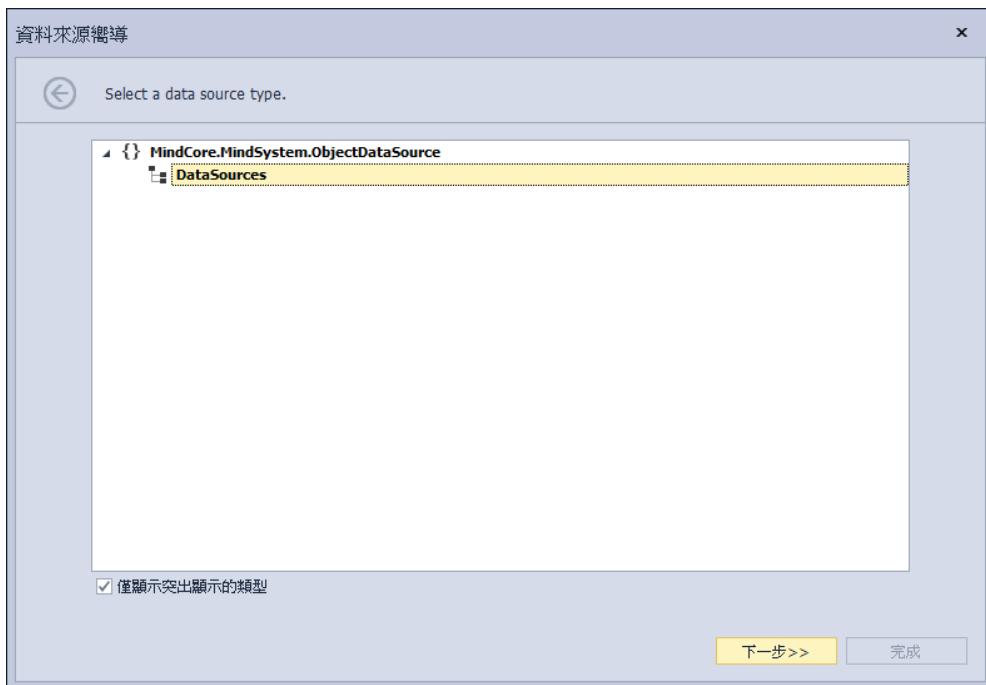
数据源->增加->对象->下一步



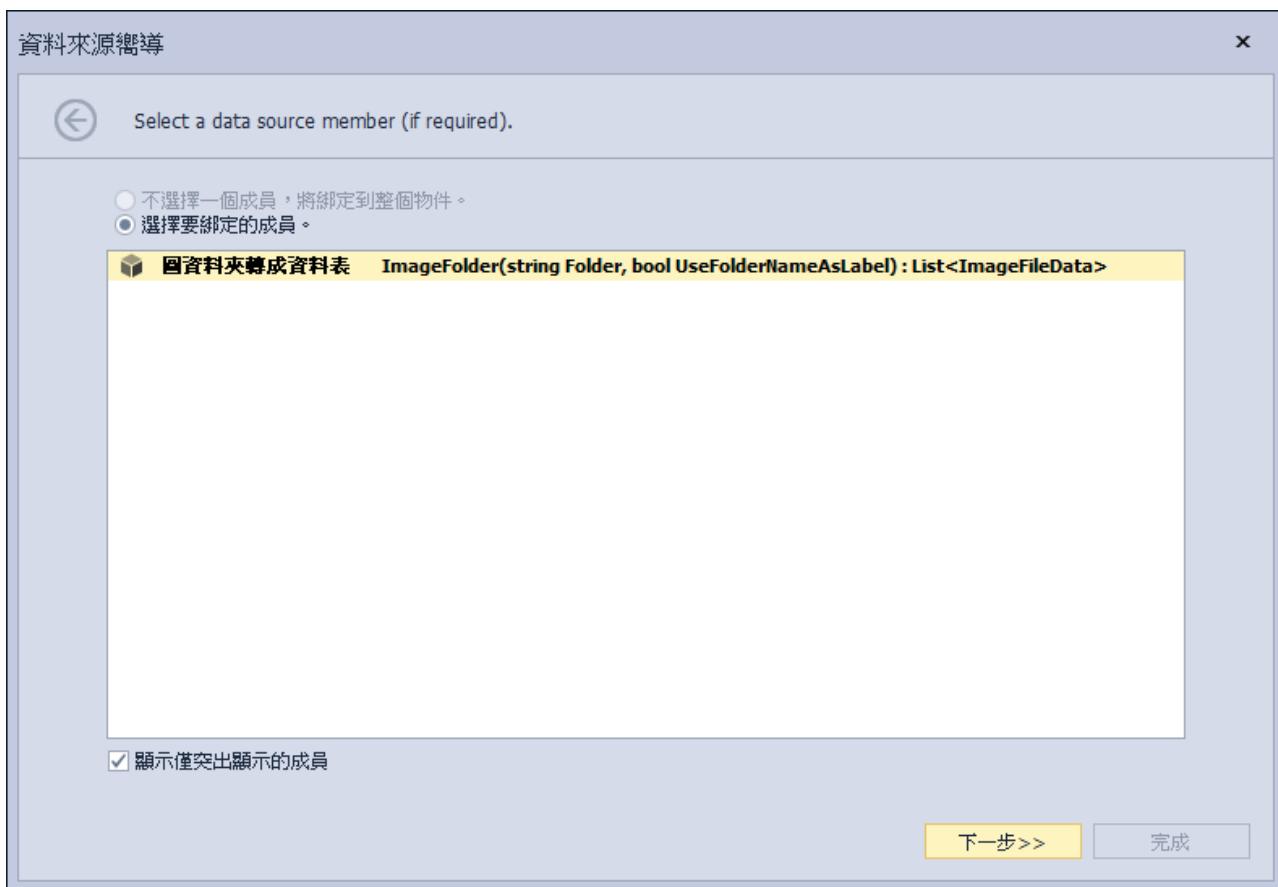
选取 MindCore.MindSystem -> 下一步



选取 DataSources -> 下一步



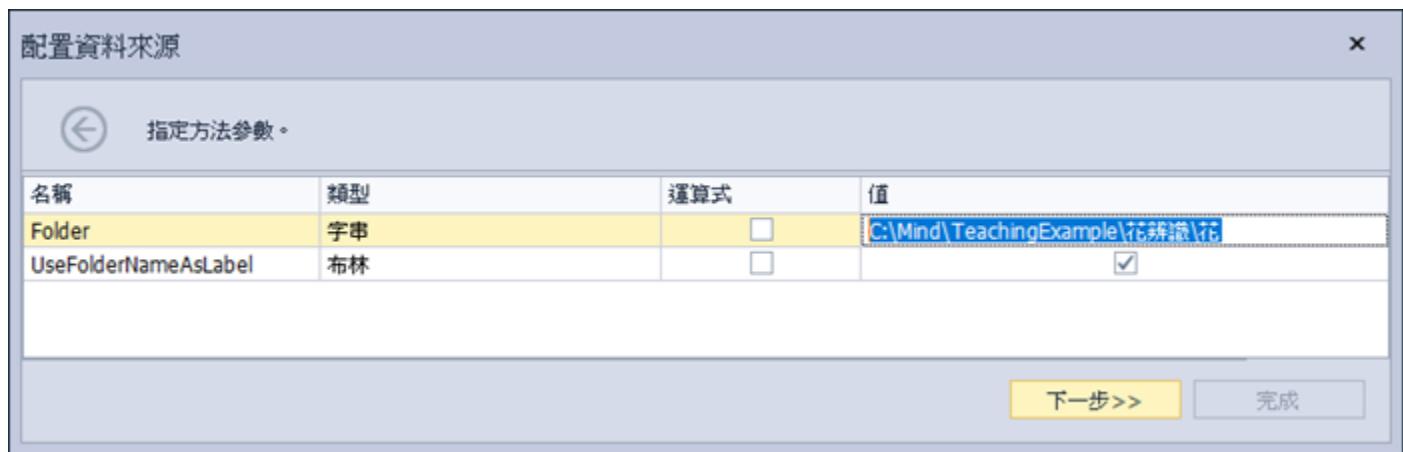
选取图文件夹转成数据表 -> 下一步



文件夹(Folder)值 = C:\Mind\TeachingExample\花辨识\花

使用活页夹名称作为标签(UseFolderNameAsLabel) = True

下一步 → 按完成



按检视 → 可看出产生的数据表

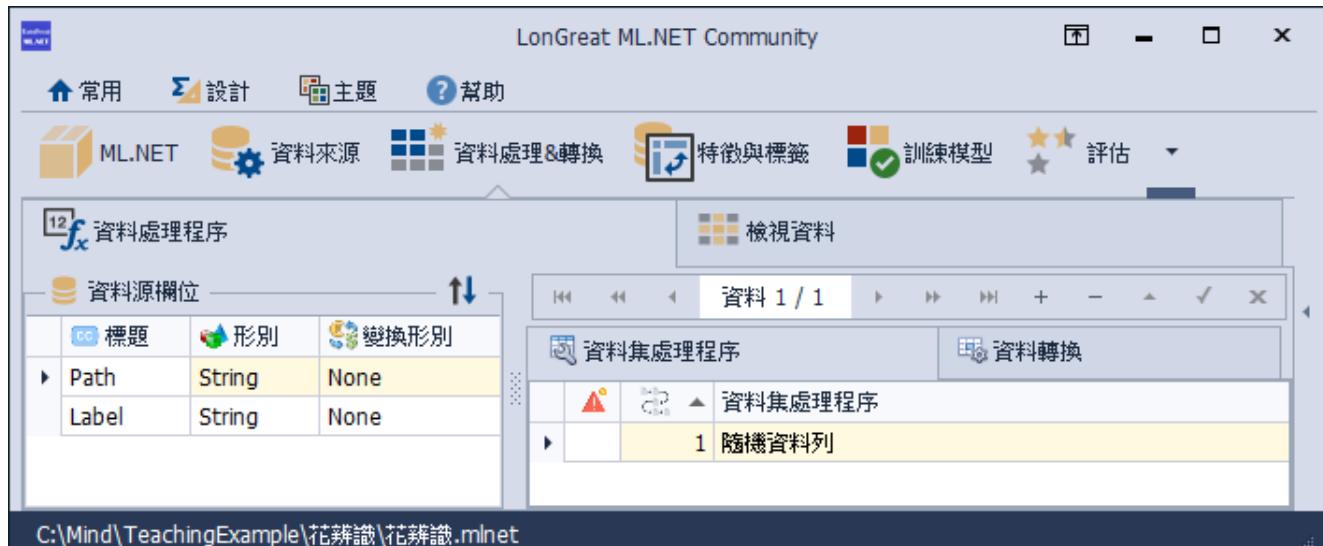
字段 = 路径与卷标

路徑	標籤
C:\Mind\TeachingExample\花辨識\花\向日葵\127192624_afab3d9cb84.jpg	向日葵
C:\Mind\TeachingExample\花辨識\花\向日葵\145303599_262f7e23815_n.jpg	向日葵
C:\Mind\TeachingExample\花辨識\花\向日葵\147804446_ef9244c8ce_m.jpg	向日葵
C:\Mind\TeachingExample\花辨識\花\向日葵\151898652_b5f1c70b98_n.jpg	向日葵
C:\Mind\TeachingExample\花辨識\花\向日葵\164668737_aeab0cb55e_n.jpg	向日葵
C:\Mind\TeachingExample\花辨識\花\向日葵\164670176_9f5b9c7965.jpg	向日葵
C:\Mind\TeachingExample\花辨識\花\向日葵\164670455_29d8e02bbd_n.jpg	向日葵
C:\Mind\TeachingExample\花辨識\花\向日葵\164671753_ab36d9ccb7_n.jpg	向日葵
C:\Mind\TeachingExample\花辨識\花\向日葵\164672339_f2b5b164f6.jpg	向日葵
C:\Mind\TeachingExample\花辨識\花\向日葵\175638423_058c07afb9.jpg	向日葵
C:\Mind\TeachingExample\花辨識\花\向日葵\184682095_46f8607278.jpg	向日葵
C:\Mind\TeachingExample\花辨識\花\向日葵\184682320_73cf74710.jpg	向日葵
C:\Mind\TeachingExample\花辨識\花\向日葵\184682506_8a9b8c662d.jpg	向日葵
C:\Mind\TeachingExample\花辨識\花\向日葵\184682652_c927a49226_m.jpg	向日葵
C:\Mind\TeachingExample\花辨識\花\向日葵\184682920_97ae41ce60_m.jpg	向日葵
C:\Mind\TeachingExample\花辨識\花\向日葵\184683023_737fec5b18.jpg	向日葵
C:\Mind\TeachingExample\花辨識\花\向日葵\193874852_fb633d8d00_n.jpg	向日葵

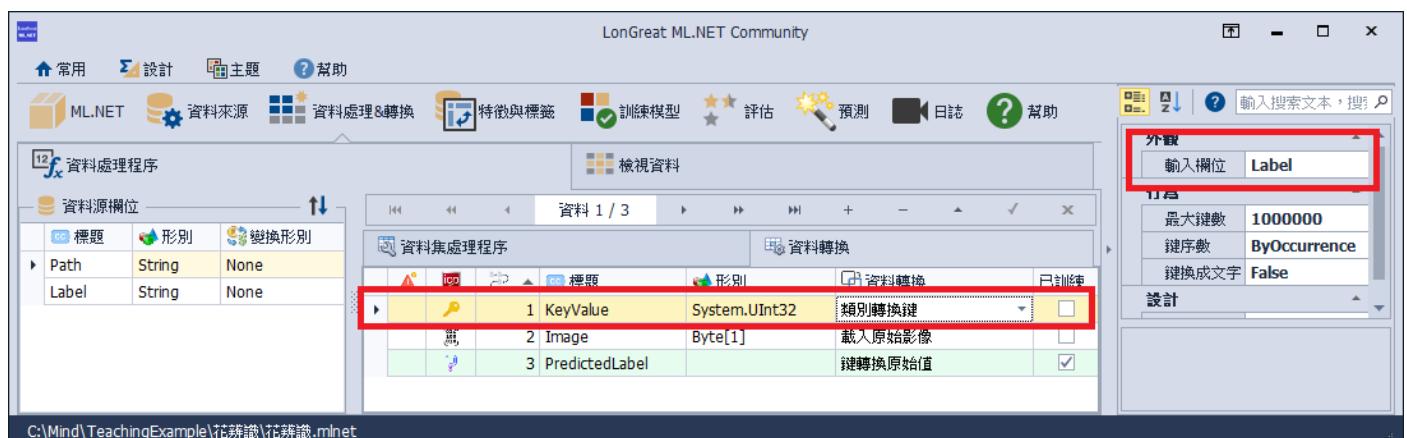
## 数据转换

### 随机资料列

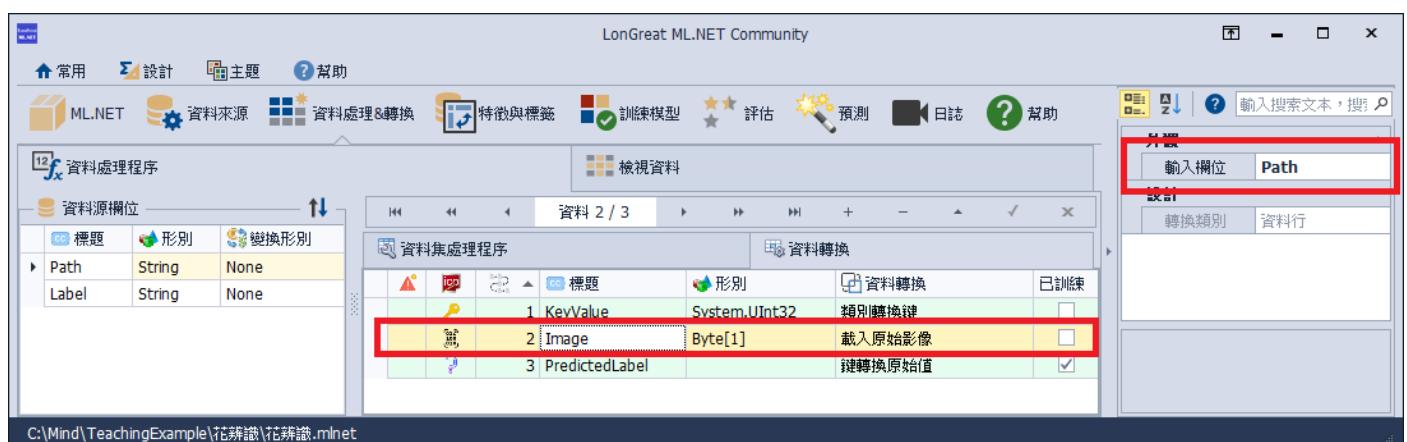
#### 1. 将资料列随机打散



#### 2. 类别转换键 => 将 Label 转换为键值



#### 3. 加载原始影像=>将 Path 字段影像加载到 Image 字段



## 4. 键转换原始值 => 预测标签(PredictedLabel)转换回来原始值

### 结果检视

- I. 至检视资料页签
- II. 选择显示笔数
- III. 按充满资料

Screenshot of the LonGreat ML.NET Community application interface showing the '檢視資料' (View Data) tab.

The interface includes a toolbar with various icons for ML.NET, data sources, processing, labeling, training models, evaluation, prediction, and help. A red box highlights the '充满資料' (Fill Data) button in the top right corner of the toolbar.

The main area displays a grid of flower images with their corresponding labels and keys. A specific row for a rose image is selected, and a detailed view dialog is open over the grid, showing the image and its details.

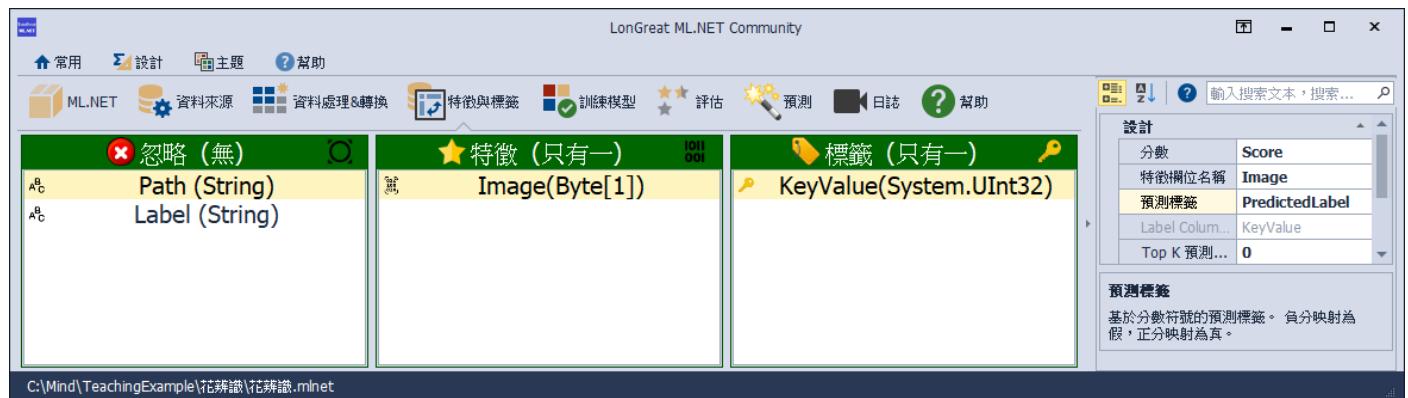
Path	Label	Key Val...	Image
C:\Mind\TeachingExample\花辨識\花\向日葵\193878348_43571127b9_n.jpg	向日葵	1	
C:\Mind\TeachingExample\花辨識\花\雛菊\99306615_739eb94b9e_m.jpg	雛菊	2	
C:\Mind\TeachingExample\花辨識\花\玫瑰\218630974_5646dafc63_m.jpg	玫瑰	3	
C:\Mind\TeachingExample\花辨識\花\蒲公英\129019877_8eea2978ca_m.jpg	蒲公英	4	
C:\Mind\TeachingExample\花辨識\花\雛菊\171972704_389cf7a953.jpg	雛菊	2	
C:\Mind\TeachingExample\花辨識\花\鬱金香\10791227_7168491604.jpg	鬱金香	5	
C:\Mind\TeachingExample\花辨識\花\向日葵\184682095_46f8607278.jpg	向日葵	1	
C:\Mind\TeachingExample\花辨識\花\鬱金香\11746452_5bc1749a36.jpg	鬱金香	5	
C:\Mind\TeachingExample\花辨識\花\玫瑰\123128873_546b8b7355_n.jpg	玫瑰	3	
C:\Mind\TeachingExample\花辨識\花\向日葵\184682506_8a9b8c662d.jpg	向日葵	1	
C:\Mind\TeachingExample\花辨識\花\玫瑰\180613732_3a7aba0b80_n.jpg	玫瑰	3	
C:\Mind\TeachingExample\花辨識\花\蒲公英\61242541_a04395e6bc.jpg	蒲公英	4	
C:\Mind\TeachingExample\花辨識\花\蒲公英\16041975_2f6c1596e5.jpg	蒲公英	4	
C:\Mind\TeachingExample\花辨識\花\玫瑰\165985535_7178ce6350.jpg	玫瑰	3	
C:\Mind\TeachingExample\花辨識\花\鬱金香\112650879_82adc2cc04_n.jpg	鬱金香	5	
C:\Mind\TeachingExample\花辨識\花\雛菊\163978992_8128b49d3e_n.jpg	雛菊	2	
C:\Mind\TeachingExample\花辨識\花\玫瑰\505517255_cfb6f6394.jpg	玫瑰	3	
C:\Mind\TeachingExample\花辨識\花\雛菊\144099102_bf63a41e4f_n.jpg	雛菊	2	
C:\Mind\TeachingExample\花辨識\花\鬱金香\141479422_5a6fa1fd1b_m.jpg	鬱金香	5	
C:\Mind\TeachingExample\花辨識\花\鬱金香\38287568_627de6ca20.jpg	鬱金香	5	

At the bottom left, the status bar shows the path: C:\Mind\TeachingExample\花辨識\花辨識.mlnet. At the bottom right, there are buttons for '確定(O)' (Confirm) and '取消(C)' (Cancel).

## 特征与卷标

拖拉 Image=>特征

拖拉 KeyValue=>标签



## 训练模型

### 训练器

按+新增

选择 DNN 影像分类模型 (ImageClassification)



## 执行

1. 选择训练模型
2. 要显示最多笔数
3. 执行

结果多二个字段

- 预测标签 PredictedLabel
- 分数 Score

The screenshot shows the LonGreat ML.NET Community application interface. The main window title is "LonGreat ML.NET Community". The top menu bar includes "常用" (Common), "設計" (Design), "主題" (Theme), and "幫助" (Help). Below the menu is a toolbar with icons for "新增" (New), "開啟" (Open), "儲存檔案" (Save File), "另存新檔" (Save As), "檔案" (File), "ML.NET", "資料來源" (Data Source), "資料處理&轉換" (Data Processing & Transformation), "特徵與標籤" (Features & Labels), "訓練模型" (Train Model), "評估" (Evaluate), "預測" (Predict), "日誌" (Log), and "歷史" (History). A status bar at the bottom displays the path "C:\Mind\TeachingExample\花辨識\花辨識.mlnet" and the text "資料 1 / 200".

The central area contains a search results grid titled "DNN影像分類模型". The grid has columns: Path, Label, Key Value, Image, Predicted Label, and Score. The "Score" column is highlighted in yellow. The results show various flower classifications with their predicted labels and scores. For example, one entry is "蒲公英" (Puffball) with a score of "[0.9997749,0...]".

Path	Label	Key Value	Image	Predicted Label	Score
C:\Mind\TeachingExample\花辨識\花\蒲公英\15987457_49dc11...	蒲公英	1		蒲公英	[0.9997749,0...]
C:\Mind\TeachingExample\花辨識\花\向日葵\184682506_8a9b8...	向日葵	2		向日葵	[0.01333415...]
C:\Mind\TeachingExample\花辨識\花\197011740_21825...	向日葵	2		向日葵	[0.02435277...]
C:\Mind\TeachingExample\花辨識\花\玫瑰\118974357_0faa23c...	玫瑰	3		玫瑰	[3.597713E-0...]
C:\Mind\TeachingExample\花辨識\花\雛菊\175106495_53ebdef...	雛菊	4		雛菊	[0.00097999...]
C:\Mind\TeachingExample\花辨識\花\向日葵\193878348_43571...	向日葵	2		向日葵	[0.04289465...]
C:\Mind\TeachingExample\花辨識\花\蒲公英\140951103_69847...	蒲公英	1		蒲公英	[0.937225,0...]
C:\Mind\TeachingExample\花辨識\花\雛菊\173350276_02817aa...	雛菊	4		雛菊	[0.02567177...]
C:\Mind\TeachingExample\花辨識\花\向日葵\127192624_afad3...	向日葵	2		向日葵	[0.00162402...]
C:\Mind\TeachingExample\花辨識\花\玫瑰\295257304_de893fc...	玫瑰	3		玫瑰	[0.00025254...]
C:\Mind\TeachingExample\花辨識\花\鬱金香\132538273_33524...	鬱金香	5		鬱金香	[0.01251243...]
C:\Mind\TeachingExample\花辨識\花\玫瑰\410425647_458666...	玫瑰	3		玫瑰	[6.996205E-0...]
C:\Mind\TeachingExample\花辨識\花\向日葵\200011914_q3f57	向日葵	2		向日葵	[0.00578804]

## 评估

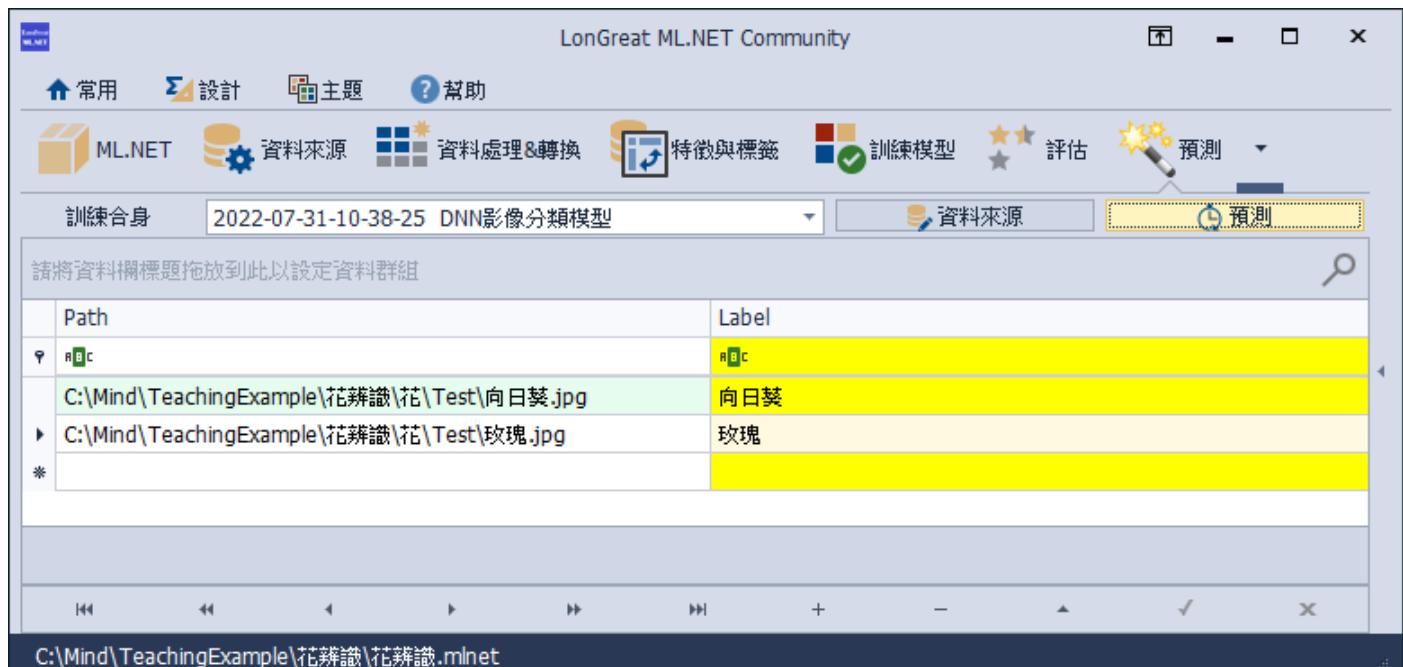
- A. 选取以训练完合身的模型
- B. 选取设定数据须与原来相同
- C. 点击预测按钮=>产生出批次预测结果
- D. 点击评估按钮=>评估结果

Path	Label	Key Value	Image	Predicted Label	Score
C:\Mind\TeachingExample\花辨識\花\玫瑰\466486216_ab13b55763.jpg	玫瑰	3		玫瑰	[0.000106...]
C:\Mind\TeachingExample\花辨識\花\玫瑰\475936554_a2b38aa8e.jpg	玫瑰	3		玫瑰	[0.01106...]
C:\Mind\TeachingExample\花辨識\花\玫瑰\475947979_554062a608_m.jpg	玫瑰	3		玫瑰	[0.000449...]
C:\Mind\TeachingExample\花辨識\花\玫瑰\483444865_65962ce07_m.jpg	玫瑰	3		玫瑰	[2.129267...]
C:\Mind\TeachingExample\花辨識\花\玫瑰\488849503_63a290a8c2_m.jpg	玫瑰	3		玫瑰	[0.060416...]
C:\Mind\TeachingExample\花辨識\花\玫瑰\494803274_f84f21d53a.jpg	玫瑰	3		玫瑰	[3.327059...]
C:\Mind\TeachingExample\花辨識\花\玫瑰\505517255_cfbef6f394.jpg	玫瑰	3		玫瑰	[0.001512...]
C:\Mind\TeachingExample\花辨識\花\玫瑰\509239741_28e2fce492_m.jpg	玫瑰	3		玫瑰	[9.789431...]
C:\Mind\TeachingExample\花辨識\花\玫瑰\512578026_f6e6f2ad26.jpg	玫瑰	3		玫瑰	[0.000684...]
C:\Mind\TeachingExample\花辨識\花\玫瑰\512694812_48ba9c0b49_n.jpg	玫瑰	3		玫瑰	[1.485577...]
C:\Mind\TeachingExample\花辨識\花\玫瑰\99383371_37a5ac12a3_n.jpg	玫瑰	3		鬱金香	[0.314170...]
C:\Mind\TeachingExample\花辨識\花\蒲公英\10443973_aeb97513fc_m.jpg	蒲公英	1		蒲公英	[0.986101...]
C:\Mind\TeachingExample\花辨識\花\蒲公英\10683189_bd6e371b97.jpg	蒲公英	1		蒲公英	[0.999980...]
C:\Mind\TeachingExample\花辨識\花\蒲公英\10919961_0af657c4e8.jpg	蒲公英	1		蒲公英	[0.999438...]
C:\Mind\TeachingExample\花辨識\花\蒲公英\11405573_24a8a838cc_n.jpg	蒲公英	1		蒲公英	[0.969518...]
C:\Mind\TeachingExample\花辨識\花\蒲公英\11545123_50a340b473_m...	蒲公英	1		蒲公英	[0.996686...]
C:\Mind\TeachingExample\花辨識\花\蒲公英\126012913_edf771c564_n...	蒲公英	1		蒲公英	[0.969953...]
C:\Mind\TeachingExample\花辨識\花\蒲公英\129019877_Beea2978ca_m...	蒲公英	1		蒲公英	[0.945920...]
C:\Mind\TeachingExample\花辨識\花\蒲公英\130733200_fbe28eea19.jpg	蒲公英	1		蒲公英	[0.965220...]

資料 0 / 202

## 预测

1. 选取以训练完合身的模型
2. 点击数据源按钮=>产生数据表
3. 输入数据表 Path 字段
4. 点击预测按钮
5. 产生出 Label 预测结果



## 相关连结

 GitHub	<a href="https://github.longgreat.net">github.longgreat.net</a>
 gitee	<a href="https://gitee.longgreat.net">gitee.longgreat.net</a>