

為響應企業智慧化管理巨量資料的需求,協助企業克服同時管理結構化與非結構化資料、並整合異質資料庫的挑戰,微軟於 2019 年底正式推出 SQL Server 2019,以 AI 整合巨量資料叢集全新架構、領先業界的效能與安全,幫助企業與開發人員解決現今資料工作環境的困難。

SQL Sever 2019 透過資料虛擬化 Polybase 能直接查詢外部資料源,協助企業 高效打破資料孤島,解決無法整合異質資料庫的痛點。全新架構巨量資料叢集, 透過 Apache Spark™與 HDFS 的整合,將能輕鬆在 Kubernetes 叢集上部署容器, 同時管理結構與非結構化資料,進而從大量資料中獲得決策先機。

本書完美結合資料庫理論與實務,使用大量實作範例來說明資料庫系統理 論、實體關聯模型和正規化,與資料庫設計,讀者不僅可以實際在資料庫設計工 具繪製專案的實體關聯圖,更可將設計成果建立成 SQL Server 資料庫,來驗證實 體關聯模型的資料庫設計理論。本書完整說明 T-SQL 語法、預存程序、順序物件、 自訂函數、觸發程序、資料指標和交易處理,幫助讀者精通 T-SQL 程式設計,更 有實例說明如何使用 C#和 LINQ 建立用戶端程式。

陳會安老師多年扎實功力在本書中體現,詳盡介紹了T-SQL 語法,相信除了 對學校資料庫相關課程是一本很好的教材之外,對資料庫開發人員來說更是一本 深入淺出的好工具書。我想推薦這本書給您一身處於海量資料快速迭代的現今, 與時俱進地熟稔資料平台已經成為資料庫開發人員職涯上和自我進修不可取代的 技能。歡迎您一起加入 SQL Server 資料庫平台的這個大家庭!

台灣微軟產品行銷資深經理

## 廖育萱 Sherry Liao



# 19-1 認識 SQL Server 機器學習服務

微軟 SQL Server 機器學習服務可以讓 SQL Server 支援資料科學(Data Science)所需的工作,直接讓我們在 SQL Server 訓練和儲存預測模型,在本章是使用 Python 語言來說明 SQL Server 機器學習服務。

#### Memo

本章測試的 SQL 指令是使用【教務系統】資料庫,請執行本書範例「Ch19、 Ch19\_School.sql」的 SQL 指令碼檔案來建立本章測試所需的資料庫、資料表和記錄 資料。

### 認識機器學習服務

SQL Server 從 2017 版開始內建 Python 執行環境,可以讓 SQL Server 執行 Python 程式碼,換句話說,我們可以將相關資料科學運算直接在 SQL Server 中執 行來增進執行效能,而不必將資料從資料庫拉出來,這就是 SQL Server 機器學習 服務(SQL Server Machine Learning Services),如下圖所示:



上述 SQL Server 機器學習服務是 SQL Server 內建功能,提供一個 Python 執行環境(即 Anaconda),能夠使用資料表的關聯式資料來執行 Python 程式 (透過預存程序),我們可以使用此服務來載入和清理資料、進行探索式資料 分析和資料視覺化,然後直接在資料庫訓練機器學習的模型,和儲存訓練結果 的預測模型。

微軟除了將常用 Python 套件安裝至 SQL Server 機器學習服務外,更提供微軟 專屬的 Python 套件,如下表所示:

套件名稱	說明
revoscalepy	提供可攜式、可擴展和分散式 Python 函數,可以用來匯入、轉換和進行資料分析,使用在敘述統計學、廣義線性模型(Generalized Linear Model,GLM)、 Logistic 迴歸、分類、迴歸樹(Regression Trees)和決策森林(Decision Forests)
microsoftml	微軟新增的機器學習演算法,可以建立模型來進行特徵擷取、文字、影像和情 感分析

#### 機器學習服務能作什麼

在 SQL Server 機器學習服務內建眾多機器學習和深度學習模型,我們可以直接在 SQL Server 使用關聯式資料來進行訓練和將模型儲存在資料庫,或將現有模型部署到機器學習服務後,直接使用關聯式資料來進行資料預測。

SQL Server 機器學習服務可以進行的預測類型,如下表所示:

預測類型	說明
分類	將資料進行分類,例如:將客戶意見分為正面與負面;將圖片自動進行分類
迴歸分析	依據連續資料的迴歸分析進行預測,例如:根據地點和面積來預測房屋價格; 年分和哩程數預測二手車車價
異常偵測	預測資料中的異常值,例如:偵測銀行詐騙交易
提供建議	預測資料中的關聯性,例如:根據顧客先前的購買記錄,建議在下一次消費時 想要購買的產品

# 19-2 安裝與啟用 SQL Server 機器學習服務

在 SQL Server 使用機器學習服務之前,我們需要先安裝與啟用 SQL Server 機器學習服務。

## 安裝 SQL Server 的機器學習服務

在第4章安裝 SQL Server 時因為是勾選全部功能,所以預設已經安裝好 SQL Server 機器學習服務,我們可以在安裝程式看到安裝的元件,如下圖所示:

<ul> <li>SQL Server 2019 安装程式</li> <li>特徴選取</li> <li>選取要安装的 Developer</li> </ul>	758世。				-		×
全域規則 產品更新 安裝安裝檔案	● 尋找 Reporting Services 功能(F):	嗎? <u>從網站下載</u>	功能描述	L:			
安裝規則 安裝擁羅型 特徵攝取 功能規則 功能規則 這備安裝 安裝進度 完成	第二回他初期 <th>重延伸模组 重度摄取 ces se 重詞服務 科本源的 JAVA 連接者 &gt;</th> <th>SQL Ser 組懸和作 <u>SQLS</u> 選取之功 磁碟空間</th> <th>ver 執行優體的每一 主藥都與其他 SQL Se Server 執行個體可以; 功能的必要元件(P):</th> <th>国執行 (rver 執行 並存 運 作</th> <th>)體功能的 行個體隔調 (在同一音</th> <th></th>	重延伸模组 重度摄取 ces se 重詞服務 科本源的 JAVA 連接者 >	SQL Ser 組懸和作 <u>SQLS</u> 選取之功 磁碟空間	ver 執行優體的每一 主藥都與其他 SQL Se Server 執行個體可以; 功能的必要元件(P):	国執行 (rver 執行 並存 運 作	)體功能的 行個體隔調 (在同一音	
	全钄(A) 取消全躪(U 執行個體根目錄(R): 共用功能目錄(S): 共用功能目錄(x86)(X):	0) C:\Program Files\1 C:\Program Files\1 C:\Program Files (	Vicrosoft S Vicrosoft S (86)\Micros	QL Server\ QL Server\ soft SQL Server\			
			< 上一	-歩(B) 下一步(N	) >	取浦	

上述圖例如果沒有勾選【機器學習服務和語言延伸模組】下的【Python】功能, 請勾選,按【下一步】鈕來安裝 SQL Server 機器學習服務。

### 啟用 SQL Server 的機器學習服務

在確認已經安裝 SQL Server 機器學習服務後,我們只需啟用 SQL Server 執行 外部 Python 腳本程式碼,就可以在 T-SQL 指令執行 Python 程式碼,其步驟如下 所示:

1 請啟動 Management Studio 連接安裝有機器學習服務的 SQL Server 資料庫 引擎後,開啟和執行 Ch19 2.sql 指令碼檔案,如下所示:

```
EXEC sp_configure 'external scripts enabled', 1
RECONFIGURE WITH OVERRIDE
```

💼 訊息 組態選項 'external scripts enabled' 從 0 變更為 1。請執行 RECONFIGURE 陳述式來安裝。 完成時間: 2020-01-16T13:34:33.6988324+08:00 100 % - 4



**2** 然後啟動設定管理員,在 SQL Server 上執行【右】鍵快顯功能表的【重新 啟動】指令,重新啟動 SOL Server 服務,如下圖所示:

Sql Server Configuration Manager 檔案(F) 動作(A) 檢視(V) 說明(H)				- 🗆 X
<ul> <li>◆●</li> <li>※</li> <li>※</li> <li>※</li> <li>※</li> <li>SQL Server 超整管理員 (本機)</li> <li>⑤</li> <li>⑤</li> <li>SQL Server 服務組建 (32 位元)</li> <li>※</li> <li>SQL Native Client 11.0 組建 (32 位元)</li> </ul>	名編 ⑦SQL Server Integration Services 15.0 ⑤SQL Server (SQLEXPRESS) ⑦SQL Full-text Filter Daemon Launcher (	(MSSOLSERVER)	狀態 執行中 執行中 執行中	啟動模式 自動 自動 手動
> 旦 SQL Server 網路組題 > 曼 SQL Native Client 11.0 組態	<ul> <li>③ SQL Server Launchpad (MSSQLSERVE)</li> <li>⑤ SQL Server (MSSQLSERVER)</li> <li>⑥ SQL Server Agent (SQLEX/PRESS)</li> <li>⑥ SQL Server Agent (SQLEX/PRESS)</li> <li>⑦ SQL Server PolyBase 資料移動 (MSS ⑦ SQL Server PolyBase 引擎 (MSSQLS)</li> <li>⑧ SQL Server Agent (MSSQLS)</li> <li>⑧ SQL Server Agent (MSSQLSERVER)</li> </ul>	R)		- 山 自動 自動 其他 (開機、系統、 其他 (開機、系統、 自動 手動
重新啟動 (停止之後再啟動) 選取的服務。	<	的谷(R) 說明(H)		>

3 在 Management Studio 開啟和執行 Ch19\_2a.sql 指令碼檔案,可以看到 config value 值已經改為1,如下所示:

EXEC sp\_configure 'external scripts enabled'

===	結果 🗊 訊息				
	name	minimum	maximum	config_value	run_value
1	external scripts enabled	0	1	1	1



4 最後,在設定管理員的 SQL Server Launchpad 上,執行【右】鍵快顯功能 表的【重新啟動】指令來重新啟動服務,如下圖所示:

🖀 Sql Server Configuration Manager			– 🗆 X
檔案(F) 動作(A) 檢視(V) 說明(H)			
♦ ♦ 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0			
SQL Server 組態管理員 (本機)	名稱	狀態	啟動模式
SQL Server 服務	SQL Server Integration Services 15.0	執行中	自動
L. SQL Server 網路組態 (32 位元)	SQL Server (SQLEXPRESS)	執行中	自動
SQL Native Client 11.0 組態 (32 恒元)	🐯 SQL Full-text Filter Daemon Launcher (MSSQLSER	WER) 執行中	手動
→ J. SQL Server 納給組態	SQL Server Launchpad (MSSQLSERVER)	** <= ++	自動
⇒ ⊋ SQL Native Client I 1.0 組態	SQL Server (MSSQLSERVER)	啟動(S)	自動
	SQL Server Analysis Services (MSSQLSERVER)	停止(O)	自動
	SQL Server Agent (SQLEXPRESS)	暫停(P)	其他 (開機、系統、
	👸 SQL Server Browser	繼續(E)	其他 (開機、系統、
	nter SQL Server PolyBase 資料移動 (MSSQLSERVER)	重新飲動(T)	自動
	7 SQL Server PolyBase 引擎 (MSSQLSERVER)	E-MIRALDU()	自動
	SQL Server Agent (MSSQLSERVER)	內容(R)	手動
	<	說明(H)	>
重新啟動 (停止之後再啟動) 選取的服務。	·	ſ	

至 SQL Server 測試執行 Python 程式碼: Ch19\_2b.sql

在成功安裝和啟用 SOL Server 機器學習服務後,我們可以建立 SOL 指令來測 試在 SQL Server 執行 Python 程式碼,如下所示:

```
EXEC sp_execute_external_script
@language = N'Python',
@script = N'print("Welcome to Python in SQL Server")'
GO
```

上述 SQL 指令是執行 sp\_execute\_external\_script 系統預存程序,執行的 Python 程式碼字串是呼叫 print()函數來輸出一段文字內容,其執行結果就是參數的訊息文 字,如下圖所示:



## 19-3 在 SQL Server 執行 Python 程式碼

SQL Server 機器學習服務提供二種方式來執行 Python 程式碼,如下所示:

- 使用 sp execute external script 系統預存程序來執行 Python 程式碼。
- 我們也可以使用 Python 開發工具撰寫 Python 程式,然後送至 SQL Server 機器學習服務來執行。

在本章主要說明如何使用 sp\_execute\_external\_script 系統預存程序來執行 Python 程式碼,此預存程序的詳細語法說明請參閱第 19-4-1 節。

## Python 變數與運算式: Ch19\_3.py

SQL Server 機器學習服務可以執行合法的 Python 程式碼字串,首先是 Python 變數與運算式,如下所示:

```
EXEC sp_execute_external_script
@language = N'Python',
@script = N'
cels = 25
fahr = cels * 9 /5 + 32
print(cels, " = ", fahr)
.
```

上述 SQL 指令使用系統預存程序 sp\_execute\_external\_script 執行 Python 程式 碼字串,其參數說明如下所示:

- @language 參數:指定外部腳本程式是使用 Python 語言。
- @script 參數:定義送至機器學習服務的腳本程式碼字串,使用的是 Unicode 字串 N'',其字串內容就是合法的 Python 程式碼。

上述 Python 程式碼計算出攝氏 25 度轉換成的華氏溫度後,使用 print()函數輸出執行結果,如右圖所 示:

■ 訊息		
來自外部指令碼的 SIDOUT 訊息:		*
25 = 77.0		
完成時間: 2020-01-16T14:03:40.9684672+08:0	0	-
100 % 🔹 🖣	•	

www.python 條件判斷:Ch19\_3a.py

Python 條件判斷的程式區塊需要在程式碼字串中縮排,例如:判斷成績變數 grade 值是否大於等於 60 分,如下所示:

```
EXEC sp_execute_external_script
@language = N'Python',
@script = N'
grade = 78
if grade >= 60:
    print("成績及格!", grade)
else:
    print("成績不及格!", grade)
'
```

上述 Python 程式碼使用 if/else 條 件判斷 grade 變數的成績,可以看到 2 個縮排的程式區塊,其執行結果顯示 成績及格和分數,如右圖所示:





在 Python 使用迴圈計算 1 加到 10 的總和,如下所示:

```
EXEC sp_execute_external_script
@language = N'Python',
@script = N'
sum = 0
for i in range(11):
    sum = sum + i
print("總和 = " + str(sum))
'
```

上述 Python 程式碼使用 for/in 迴 圈來計算 1 加到 10 的總和,其執行結 果顯示總和是 55,如右圖所示:





在 Python 程式碼改用 convert to f()函數來轉換攝氏成為華氏溫度,如下所示:

```
EXEC sp_execute_external_script
@language = N'Python',
@script = N'
def convert_to_f(c):
    f = (9.0 * c) / 5.0 + 32.0
    return f

cels = 25
fahr = convert_to_f(cels)
print(cels, " = ", fahr)
'
```

上 述 Python 程 式 碼 建 立 convert\_to\_f()函數後,呼叫函數將攝氏 25 度轉換成華氏溫度後,顯示溫度的轉 換結果,其執行結果如右圖所示:

≧ 訊息		
來自外部指令碼的 STDOUT 訊息:	-	
25 = 77.0		
	- 1	
完成時間: 2020-01-16T14:20:40.1153287+08:00		
	1	۳
100 % - 4	•	



Python 程式可以匯入 sys 模組來檢查 SQL Server 機器學習服務使用的 Python 版本,如下所示:

```
EXEC sp_execute_external_script
@language = N'Python',
@script = N'
import sys
print(sys.version)
'
```

上述 Python 程式碼使用 sys.version 屬性取得 Python 程式的版本 3.7.1 版,如下圖所示:

副 訊息	
來自外部指令碼的 STDOUT 訊息:	
3.7.1 (default, Dec 10 2018, 22:54:23) [MSC v.1915 64 bit (AMD64)]	
完成時間: 2020-01-16T14:25:26.0822250+08:00	-
100 % - 4	•

# 19-4 Python 程式碼的輸入與輸出參數

第 19-3 節我們是使用 Python 標準輸出的 print()函數來輸出執行結果,事實 上,因為是在 SQL Server 執行 Python 程式,預設情況的輸入應該是單一資料表的 SQL 查詢結果,輸出是 Python 的 DataFrame 資料框架物件。

## 19-4-1 輸入與輸出參數

在說明如何使用 Python 輸入與輸出參數前,我們需要先了解 sp\_execute\_external\_script系統預存程序的語法,如下所示:

```
sp_execute_external_script
@language = N'language',
@script = N'script'
[ , @input_data_1 = N'input_data_1' ]
[ , @input_data_1_name = N'input_data_1_name' ]
[ , @output_data_1_name = N'output_data_1_name' ]
[ , @params = N'@parameter_name data_type [ OUT | OUTPUT ] [ ,...n ]' ]
[ , @parameter1 = 'value1' [ OUT | OUTPUT ] [ ,...n ] ]
WITH RESULT SETS((欄位名稱 資料類型, [欄位名稱 資料類型]));
GO
```

上述@language 和@script 參數已經在第 19-3 節說明過,其他參數和子句的說明,如下所示:

- @input\_data\_1 參數:指定 Python 程式碼使用的輸入資料,其資料類型是 nvarchar(max)。
- @input\_data\_1\_name 參數:指定@input\_data\_1 定義輸入資料的變數名稱。Python 輸入變數的預設名稱是 InputDataSet。
- @output\_data\_1\_name 參數:指定完成預存程序呼叫後,回傳給 SQL Server 的資料變數名稱。在 Python 程式碼的輸出資料必須指定給輸出變數,輸出 變數的預設名稱是 OutputDataSet,變數值是 pandas 套件的 DataFrame 資 料框架物件。
- @params 和@parameter1 參數:分別是外部腳本輸入參數名稱和值宣告的 清單。
- WITH RESULT SETS 子句:定義回傳資料表的結構描述,可以將回傳的 DataFrame 物件轉換成 SQL Server 資料表。

## 顯示 InputDataSet 輸入參數的資料: Ch19\_4\_1.sql

Python 預設的輸入參數名稱是 InputDataSet 變數,當我們指定輸入資料的 SELECT 查詢指令後,就可以使用 print()函數顯示輸入參數的資料,這是一個 DataFrame 物件,如下所示:

```
USE 教務系統
GO
EXEC sp_execute_external_script
@language = N'Python',
@script = N'
print(type(InputDataSet))
print(InputDataSet)
',
@input_data 1 = N'SELECT 課程編號, dbo.UnicodeString(名稱) AS 名稱, 學分 FROM 課程'
```

上述 Python 程式碼使用 2 個 print()函數依序使用 type()函數顯示輸入參數的 型態和參數值,@input\_data\_1參數指定輸入資料的 SELECT 指令,如下所示:

SELECT 課程編號,dbo.UnicodeString(名稱) AS 名稱,學分 FROM 課程

上述 SELECT 指令可以查詢課程資料表的記錄資料,因為名稱欄位並不是 Unicode 編碼,在轉換成 Python 輸入資料時會產生錯誤,所以在欄位呼叫 dbo.UnicodeString()純量值函式轉換成 Unicode 編碼,這是我們自行建立的一個 SQL Server 自訂函數,如下圖所示:

```
    ■ 函數
    ● 資料表值函式
    ● 純量值函式
    ● 繰 dbo.fnFactorial
    ● 繰 dbo.fnGetSalary
    ● 繰 dbo.fnPrice
    ● 繰 dbo.fnValidCode
    ● 繰 dbo.fnVolume
    ● 繰 dbo.UnicodeString
```

```
df = pd.DataFrame(s, index=[1])
OutputDataSet = df
'
WITH RESULT SETS(([值] int))
```

上述 Python 程式碼使用 Series 物件建立 DataFrame 物件, 第 2 個參數 index=[1]指定只取出索引值 1(即第 2 個元素),其 執行結果如右圖所示:

	結果	B.	訊息	
	值			
1	29	)		

## 19-5 匯入訓練資料至 SQL Server 資料庫

在使用 SQL Server 訓練機器學習模型前,我們需要先將訓練資料匯入 SQL Server 資料庫。

## 19-5-1 使用 Python 將資料匯入 SQL Server 資料庫

在 Python 的 Scikit-learn 套件內建的 Iris 資料集是鳶尾花的資料,可以讓我們 訓練模型使用花瓣和花萼來分類鳶尾花。我們準備將 Iris 資料集匯入儲存至 SQL Server 資料庫。

#### 步驟一:建立資料庫與資料表

首先,我們準備建立名為 iris\_dataset 資料庫,和新增 iris\_data 資料表來匯入 Iris 資料集的資料(SQL 指令碼檔案: Ch19 5 1.sql),如下所示:

```
USE master
GO
CREATE DATABASE iris_dataset
```

執行上述 SQL 指令可以建立名為 iris\_dataset 的資料庫,然後,我們就可以在 此資料庫新增 iris\_data 資料表來匯入資料(SQL 指令碼檔案: Ch19\_5\_1a.sql), 如下所示:

```
USE iris_dataset

GO

DROP TABLE IF EXISTS iris_data;

GO

CREATE TABLE iris_data (

id INT NOT NULL IDENTITY PRIMARY KEY,

Sepal_Length FLOAT NOT NULL,

Sepal_Width FLOAT NOT NULL,

Petal_Length FLOAT NOT NULL,

Petal_Width FLOAT NOT NULL,

Species VARCHAR(100) NOT NULL,

SpeciesId INT NOT NULL

);
```

上述 SQL 指令在切換至 iris\_dataset 資料庫後,檢查 資料表 iris\_data 是否存在,如果存在就刪除資料表,然 後建立 iris\_data 資料表,執行後,我們可以在 iris\_dataset 資料庫建立 iris data 資料表,如右圖所示:

```
    ■ iris_dataset
    ● 資料庫圖表
    ● 資料表
    ● 資料表
    ● FileTable
    ● FileTable
    ● S統資料表
    ● 目記形資料表
    ● 圖形資料表
    ● 圖形資料表
    ● ■ dbo.iris_data
    ● ● 檢視
    ● ● 外部資源
```

#### 步驟二:將訓練資料載入 SQL Server 資料表

在成功建立 iris\_dataset 資料庫和 iris\_data 資料表後,我們就可以將 Iris 資料 集匯入儲存至 SQL Server 資料庫,首先,我們需要建立預存程序來載入 Iris 資料 集(SQL 指令碼檔案: Ch19 5 1b.sql),如下所示:

```
USE iris_dataset
GO
CREATE PROCEDURE load_iris_dataset
AS
BEGIN
EXEC sp_execute_external_script @language = N'Python',
@script = N'
from sklearn import datasets
iris = datasets.load_iris()
iris_data = pandas.DataFrame(iris.data)
iris_data["Species"] = pandas.Categorical.from_codes(iris.target,
iris.target_names)
iris_data["SpeciesId"] = iris.target
```

```
',
@input_data_1 = N'',
@output_data_1_name = N'iris_data'
WITH RESULT SETS ((Sepal_Length float not null,
    Sepal_Width float not null,
    Petal_Length float not null, Petal_Width float not null,
        Species varchar(100) not null, SpeciesId int not null));
END;
```

上述 SQL 指令可以建立名為 load\_iris\_dataset 的預存程序, Python 程式碼是載入 Scikit-learn 套件 內建的 Iris 資料集,如右圖所示:

```
    □ iris_dataset
    ● 資料庫圖表
    ● 資料表
    ● 檢視
    ● 「● 新部資源
    ● 同義字
    ● 可程式性
    ● 預存程序
    ● 通 条統預存程序
    ● ■ 金數
```

然後,我們可以呼叫預存程序,將 Iris 資料集匯入 iris\_data 資料表(SQL 指 令碼檔案: Ch19 5 1c.sql),如下所示:

```
USE iris_dataset
GO
INSERT INTO iris_data (Sepal_Length, Sepal_Width,
Petal_Length, Petal_Width, Species, SpeciesId)
EXEC dbo.load_iris_dataset;
```

上述 SQL 指令呼叫 load\_iris\_dataset 預存程序來匯入 Iris 資料集,其執行結果可以看到匯入 150 筆記錄,如下圖所示:

■ 訊息		
(150 個資料列受到影響)		
完成時間: 2020-01-17T15:33:35.2457603+08:00		Ļ
100 % - 4	•	

### 步驟三:查詢載入 SQL Server 資料表的訓練資料

在成功匯入 Iris 資料集後,我們可以查詢前 10 筆記錄資料(SQL 指令碼檔案: Ch19\_5\_1d.sql),如下所示:

```
USE iris_dataset
GO
SELECT TOP(10) * FROM iris_data
```

上述 SQL 指令的執行結果可以顯示前 10 筆記錄,如下圖所示:

田 結果		🗊 訊息					
	id	Sepal_Length	Sepal_Width	Petal_Length	Petal_Width	Species	SpeciesId
1	1	5.1	3.5	1.4	0.2	setosa.	0
2	2	4.9	3	1.4	0.2	setosa	0
3	3	4.7	3.2	1.3	0.2	setosa.	0
4	4	4.6	3.1	1.5	0.2	setosa	0
5	5	5	3.6	1.4	0.2	setosa.	0
6	6	5.4	3.9	1.7	0.4	setosa.	0
7	7	4.6	3.4	1.4	0.3	setosa	0
8	8	5	3.4	1.5	0.2	setosa.	0
9	9	4.4	2.9	1.4	0.2	setosa	0
10	1	4.9	3.1	1.5	0.1	setosa.	0

上述資料表的前幾個欄位分別是花萼(Sepal)和花瓣(Petal)的長和寬,單 位是公分,Species 是哪一種鳶尾花的名稱 setosa、versicolor 和 virginica,最後是 種類值:0是 setosa;1是 versicolor;2是 virginica。

## 19-5-2 使用 SQL Server 的匯入和匯出精靈

在第 19-5-1 節我們是使用 Python 程式匯入內建 Iris 資料集,如果擁有外部資料,例如:CSV 檔案,我們可以直接使用 SQL Server 匯入和匯出精靈,將多種資料來源的資料匯入 SQL Server 資料庫。

#### 步驟一:建立資料庫

首先建立 nba\_dataset 資料庫來匯入 HOU\_players\_stats\_2017.csv 的 CSV 檔案 的休士頓火箭隊球員的統計資料(SQL 指令碼檔案: Ch19 5 2.sql),如下所示:

USE master GO CREATE DATABASE nba\_dataset

上述 SQL 指令可以建立名為 nba dataset 資料庫。

## 步驟二:使用 SQL Server 的匯入和匯出精靈匯入 CSV 檔案

在成功建立 nba\_dataset 資料庫後,我們準備將訓練資料的 CSV 檔案: HOU players stats 2017.csv 匯入成資料表,其步驟如下所示:

① 請在 Windows 作業系統執行「開始>SQL Server 2019>SQL Server 2019 匯 入及匯出資料(64 位元)」命令啟動匯入和匯出精靈,在歡迎畫面按【下一步】 鈕,如下圖所示:

🖾 SQL Server 匯入和匯出績靈		-		×
	歡迎使用 SQL Server 匯入和匯出料	情靈		
	此諸靈可以協助您建立簡單封裝,以在各種常見資料格式 料,包括資料庫、試算表和文字檔。此諸靈也可以建立目 入資料的資料表。	之間匯入和 的地資料國	回匯出資 軍及要插	
	若要從一個伺服器執行個體將資料庫及其物件移動或複製 行個體,請取消此凝靈,並改用複製資料庫構靈。您可以 Management Studio 中找到複製資料庫構靈。	到另一個 《在 SQL S	伺服器執 Server	
	□ 不要再顯示此開始頁面(D)。			
說明(H)	<上一步(B) <b>下一步(N) &gt;</b> 完成	(F) >>1	取消	

② 首先選擇資料來源,在上方【資料來源】欄選【一般檔案來源】後,按【瀏覽】鈕選【HOU\_players\_stats\_2017.csv】檔案後,可以看到自動填入的文字檔案格式,請按【下一步】鈕,如下圖所示:

🖳 SQL Server 🗷	入和匯出精靈		-		×
選擇資料來這	<b>原</b> 資料的來源。			Canol I	Y
資料來源(D):	🗟 一般檔案來	源			•
(2) 一統 (1) 資料行 (1) 通知路 (1) 預算	<ul> <li>違取檔案並指定檔案團性和</li> <li>檔案名稱(1):</li> <li>地區設定(L):</li> <li>字碼頁(C):</li> <li>格式(M):</li> <li>文字限定词(0):</li> <li>標頭資料利分歸符號(R):</li> <li>略通的標荫資料利數(S):</li> <li>(&gt; 第一個資料利的資料利</li> </ul>	當案格式。 D:SQLServe(2019/Ch19HOU_players_stats_2) 中文(繁體,台湾) 950 (ANSL/OEM - 繁體中文 Big5) 分隔符號 (無> 「回過調問調 0 百名稱(A)	017.c	瀏覽(W	
▲ 此連接管理管	的資料行未定義。	. L. #200 [ 75 #200. ] =#	//h1	Here's	щ 1
2%99(H)			(r) >>1	μχ;)	3

3. 指定資料列記錄的分隔符號,和分隔資料行欄位的符號後,可以在下方預覽
 記錄資料,然後按【下一步】鈕,如下圖所示:

					4 4
資料來源(D):		🗟 一般檔案來源			
10 er.	指定分隔來;	原檔案的字元:			
る。一般 副 資料行	資料列分開	風符號(0)	(CR)(LF)		
■ 進階 ■ 預覽	2010/2010				
	資料打分的	特付弧(し):	建硫 {;}		
	預覽資料列2	-25:			
	No	Player	Pos	Ht	W
	33	Ryan Anderson	PF	6-10	24
	1	Trevor Ariza	SF	6-8	21
	28	Tarik Black	С	6-9	25
	6	Bobby Brown	PG	6-2	15
	26	Markel Brown	SG	6-3	19
	98	Isaiah Canaan	SG	6-0	20
	15	Clint Capela	С	6-10	24
	10	Eric Gordon	SG	6-4	21
	14	Gerald Green	SG	6-7	20
	13	James Harden	SG	6-5	22
	42	Nene Hilario	С	6-11	25
	<				>
			市好地平	F(R) 雷語溶	彩レティア)

《 接著選擇目的地,在【目的地】欄選【SQL Server Native Client 11.0】後, 依序選 SQL Server 伺服器名稱、Windows 驗證,在【資料庫】欄選 【nba\_dataset】後,按【下一步】鈕,如下圖所示:

🗓 SQL Server 匯入和匯出精靈			-		×
選擇目的地 指定複製資料的目的地。				Cherry 1	Y
目的地(D):	SQL Server Native Client	11.0			•
伺服器名稱(S):	(local)				•
ak ar 使用 Windows 驗證(W) で 使用 SQL Server 驗證(Q) 使用者名稱(U): 密碼(P):					_
資料庫(T):	nba_dataset		<b>▼ ■</b>	新整理(R) 新増(E)	
說明(H)	<上-	-步(B) 下一步(N) >	完成(F) >>	<b>取</b> ()	5

① 在選取【目的地】後,更改成【[dbo].[HOU\_players\_stats]】資料表後,按 【編輯對應】鈕編輯欄位對應資料,如下圖所示:

」SQL Server 匯入和匯出積靈				-
選取來源資料表和檢視 請選擇一個或多個要複製的資料表和檢視。				12-1
資料表和檢視(T):				
▼ 來源:	目的地: (local)			1
D:\SQLServer2019\Ch19\HOU players stats 201	[dbo].[HOU_players_stats]	T		-
		1		
	编辑考虑(E)		預覽(1	²)

④ 在「資料行對應」對話方塊編輯來源和目的地的欄位對應,請將最後4個欄 位的資料類型改為 float 後,按【確定】鈕,然後在精靈步驟按【下一步】 鈕,如下圖所示:

						0.000	-
資料行對應							Ц
來源:		D:\SQLServer	2019\Ch19\H	OU_player:	_stats_2017	.csv	
目的地:		[dbo].[HOU_	players_stats]				
• 建立目的地容 • 刪除目的地容	資料表(R) 資料表中的資料列(	編輯 SQL ()) [] 卸除並重	(\$) 〔新建立目的	也資料表(	D)		
▶將資料列附	加至目的地資料表()	?) 🔽 歇用識別	插入(I)				
來源	目的地	類型	可為	大小	有效	小	1
No	No	varchar	v	50			
Player	Player	varchar	~	50			
Pos	Pos	varchar	~	50			
Ht	Ht	varchar	~	50			
₩t	Wt	varchar		50			
BirthDate	BirthDate	varchar	~	50			
Nationality	Nationality	varchar	~	50			
Ехр	Exp	varchar	~	50			
College	College	varchar	<b>V</b>	50			
PTSG	PTSG	float	~				
AST	AST	float					
TRB	TRB	float					
3P	3P	float	-				
來源資料行:		3P 字串 [D T	_STR] (50)				

2 在檢視資料類型對應步驟顯示欄位資料轉換設定,SQL Server 會自動判決是 否需進行轉換,沒有問題,請按【下一步】鈕,如下圖所示:

l so	QL Server 匯入和	匯出精靈						_		×	
檢問	<b>檢問 資料 類型對應</b> 選取資料表,檢問它的資料類型對應到目的地資料類型的方式,和處理轉換問題的方式。										
資料	料表:										
	來源			Т	目的地					_	
	D:\SQLServer201	9\Ch19\HOU_plays	rs_stats_2017.csv	1	[dbo].[HOU	J_players_s	tats]				
<										>	
資料	斗類型對應:										
	來源資料行	來源類型	目的地資料行	目的	的地類型	轉換	錯誤時	3	交易時	^	
0	No	字串 [DT_STR]	No	Va	rchar					-	
0	Player	字串 [DT_STR]	Player	va	rchar						
0	Pos	字串 [DT_STR]	Pos	va	rchar						
0	Ht	字串 [DT_STR]	Ht	va	rchar						
0	Wt	字串 [DT_STR]	Wt	va	rchar					~	
R					•				)		
若	要檢視轉換細節,	請按兩下含有要	專換的資料行來源	漢頁型	的資料列。						
錯	呉時 (全域)					失見	<u>χ</u>	_	-		
交	易時 (全域)					失調	<b></b>		•		
						1.01					
	1						1	1			
Ĭ	記明(H)		<	Ŀ-	步(B) 下	一步(N) >	完成(F) >	>>	取;)	Ě	
			_								

3 勾選【立即執行】(如需重複執行,請儲存成 SSIS 封裝)後,按【下一步】 鈕,如下圖所示:

🖾 SQL Server 匯入和匯出精靈					_		×
儲存並執行封裝 指出是否儲存 SSIS 封裝。						China I	y y
✓ 立即執行(U)							
┌─ 儲存 SSIS 封裝(S)							
😨 SQL Server(Q)							
C 檔案系統(F)							
封裝保護等級(L):							
機密資料以使用者金鑰加密						-	-
密碼(P):							
再次輸入密碼(R):							
說明(H)	< 上-	-步(B) 下-	-步(N) >	完成(F)	>>	取消	i

⑦ 可以看到選擇的作業內容,請按【完成】鈕執行作業,如下圖所示:

] sq	L Server 匯入和匯出精靈		-		×
12-2	完成精靈 確認在精靈中所作的選擇,再按一一	下 [完成]。			
按-	-下 [完成] 以執行下列動作:				
目的目的	D地位置:(local) )地提供者:SQLNCL111 將資料列従 D/SQLServer2019/Ch19/HOU_play	ers_stats_2017.csv 複製到 [dbo].[H	OU_players_stats]		
	將建立新的目標資料表。				
ŀ	將个曾儲仔封裝。 將立即執行封裝。				
提供	者對應檔案 : C:\Program Files\Microsoft SQL Se:	rver\150\DTS\MappingFiles\SSIS10	foMSSQL.XML		
說	明(H)	<上一步(B) 下一步(Ⅳ) >	完成(F)	取消	í –

① 可以看到正在執行從來源至目的地的匯入和匯出作業,成功執行後,請按【關
 閉】鈕完成操作,如下圖所示:

	成功	11 總計 11 成功	0 錯誤 0 警告
É細	<b>音</b> 料(D):		
	動作	狀態	訊息
9	正在初始化資料流程工作	成功	
9	正在初始化連接	成功	
9	正在設定 SQL 命令	成功	
9	正在設定來源連接	成功	
9	正在設定目的地連接	成功	
9	正在驗證	成功	
9	準備執行	成功	
D	執行前	成功	
D	止在執行	成功	The second se
j)	止任很器到[dbo].[HOU_players_stats]	成功	<u> 出得法 24 間資料列</u>
Ð	執行後	成功	

在 Management Studio 開啟【HOU\_players\_stats】資料表,可以看到成功匯入的 24 筆記錄資料,如下圖所示:

DESK	TOP-JOE.	nbaOU_players_sta	ats ⊹⊨ ×	物件總管書	羊細資料						-
	No	Player	Pos	Ht	Wt	BirthDate	Nationality	Exp	College	PTSG	^
۰.	33	Ryan Anderson	PF	6-10	240	"May 6 1988"	us	9	University of C	9.3	
	1	Trevor Ariza	SF	6-8	215	"June 30 1985"	us	13	"University of	11.7	
	28	Tarik Black	С	6-9	250	"November 22	us	3	"University of	3.5	
	6	Bobby Brown	PG	6-2	175	"September 2	us	3	"California Sta	2.5	
	26	Markel Brown	SG	6-3	190	"January 29 19	us	2	Oklahoma Sta	1.3	
	98	Isaiah Canaan	SG	6-0	201	"May 21 1991"	us	4	Murray State	0	
	15	Clint Capela	С	6-10	240	"May 18 1994"	ch	3		13.9	
	10	Eric Gordon	SG	6-4	215	"December 25	us	9	Indiana Univer	18	
	14	Gerald Green	SG	6-7	205	"January 26 19	us	10		12.1	~
<										3	>
14	∢  1	/24   ▶ ▶∥ ▶									