



CURSO DE ESPECIALIZACIÓN EN INTELIGENCIA ARTIFICIAL Y BIGDATA

PROGRAMACIÓN DE INTELIGENCIA ARTIFICIAL

TAREA EVALUABLE 4.1

Autor: Carlos Sánchez Recio.
11 / 01 / 2025

Índice

Apartado 1	1
1.1 Crear el recurso	1
1.2 Crear el nuevo trabajo automatizado de <i>machine learning</i>	4
1.3 Lanzar el servicio a internet	14
1.4 Consumir el servicio web	16
➤ Cuaderno de Jupyter	17
➤ Extensión de VSCode de RapidAPI	18
1.5 Limpiar los recursos utilizados	19

Apartado 1

Volveremos a trabajar con un clasificador sobre el conjunto de datos de los pingüinos del archipiélago Palmer. Recuerde que puede encontrar los datos originales en [Kaggle](#) (archivo `penguins_size.csv`) y una [copia en GitHub del curso](#).

Recordemos que estos son los datos de cada individuo:

- **species** (variable objetivo, la especie de pingüino: Chinstrap, Adélie o Gentoo).
- **island** (isla: Dream, Torgersen o Biscoe) (isla: Dream, Torgersen o Biscoe).
- **culmen_length_mm** (longitud del pico en mm).
- **culmen_depth_mm** (profundidad del pico en mm).
- **flipper_length_mm** (longitud de la aleta en mm).
- **body_mass_g** (masa corporal en gramos).
- **sex** (sexo: Male o Female)

Para ello, debe trabajar con Azure Machine Learning con Azure Machine Learning Studio y las funcionalidades de Machine Learning automatizado, de forma similar a como lo hemos hecho en el caso práctico 1 de los apuntes. Sin embargo, ahora se trata de un problema de clasificación, no de regresión.

Para realizar el trabajo de ML automatizado debe utilizar la exactitud (*accuracy*). Y debe considerar los siguientes modelos que hemos visto en el curso: `LogisticRegression` , `SVM` , `KNN` , `DecisionTree` y `RandomForest` . Pon los mismos límites en el entrenamiento de los modelos que hemos empleado en el caso práctico.

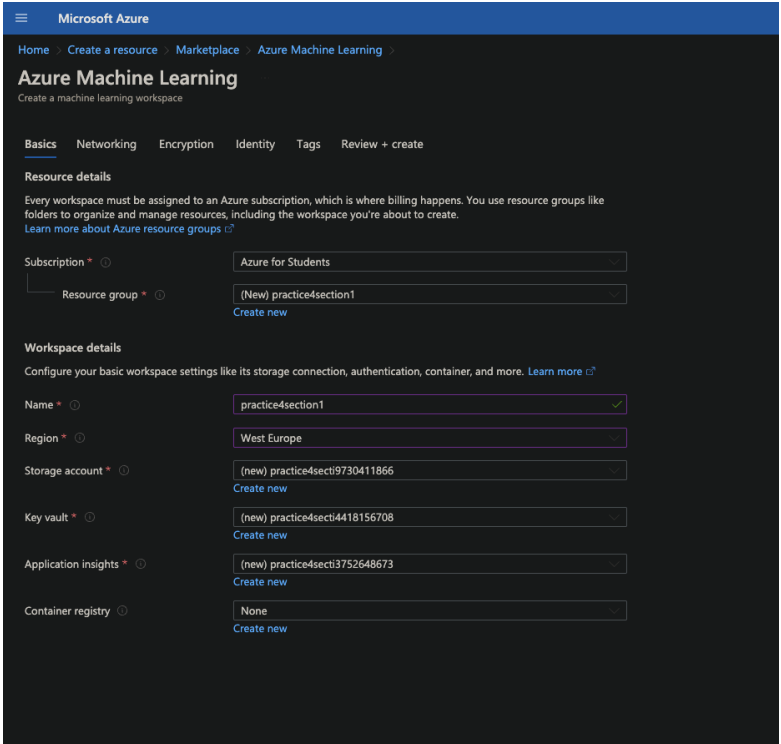
Documenta, incluyendo explicaciones y capturas de pantalla, todo el proceso, pasa a paso. Explica cuál es el modelo que se ha seleccionado y muestra sus detalles, incluyendo los hiperparámetros y todas las métricas (con gráficos). Muestra también la matriz de confusión y coméntala.

Una vez tengas el modelo, desplégalo en un servicio web y envíale al menos dos peticiones de prueba, que den como resultado flores diferentes. Incluye todo esto también en tu documentación.

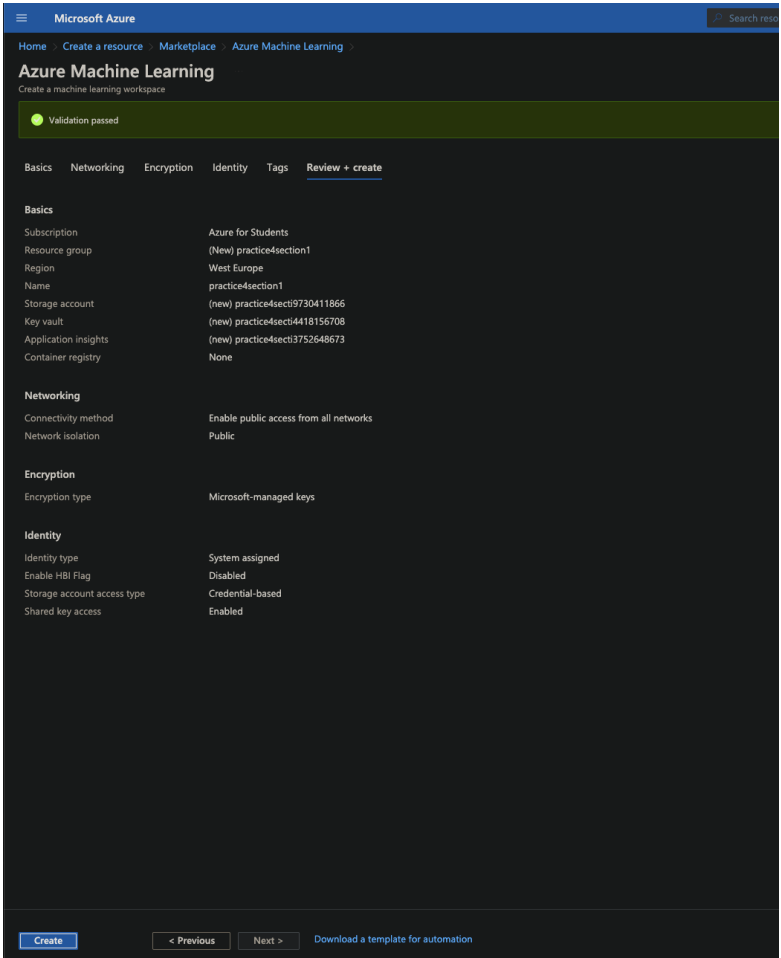
Por último, cuando lo hayas terminado todo, elimina todos los recursos. Documenta también ese proceso de limpieza.

1.1 Crear el recurso

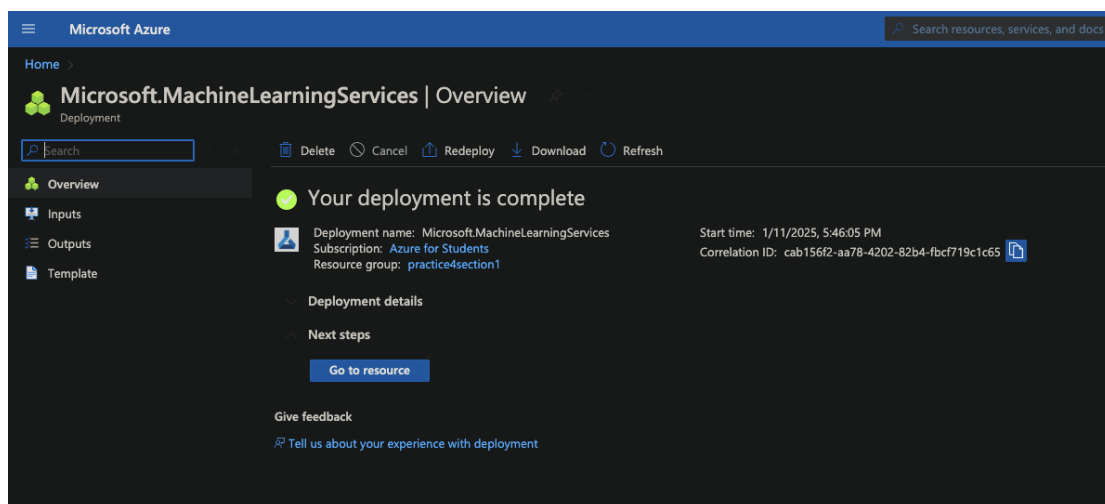
El primer paso a realizar para conseguir el objetivo final de esta tarea es crear el grupo de recursos desde el [portal de Azure](#) como se ha explicado en los apuntes de este bloque. Para esta práctica los datos o parámetros utilizados para el recurso han sido los que se muestran en las siguientes imágenes.



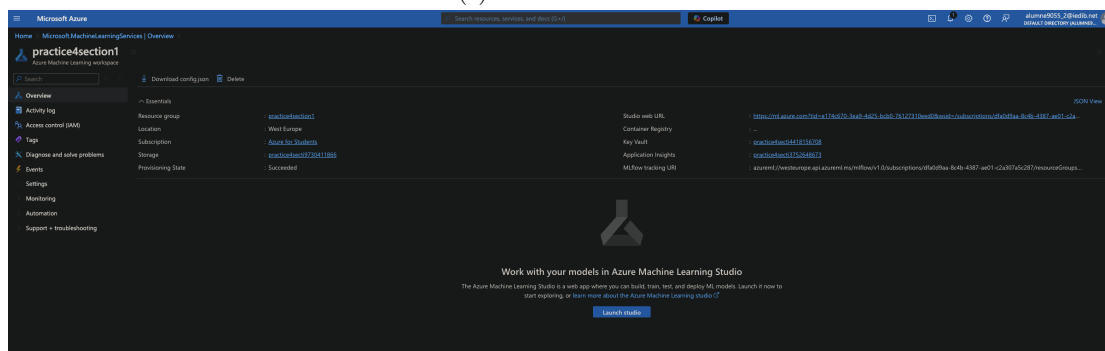
(a) Nombre del recurso nuevo y región.



(b) Revisión del nuevo recurso.



(c) Recurso creado.



(d) Panel del recurso.

Figura 1 1: Creación de un nuevo recurso para *machine learning* en Azure.

Una vez se ha hecho *deploy* correctamente se puede pulsar en el botón '*Launch Studio*' para acceder al panel del recurso creado en el cual podremos crear y entrenar el nuevo trabajo automatizado de *machine learning*.

1.2 Crear el nuevo trabajo automatizado de *machine learning*

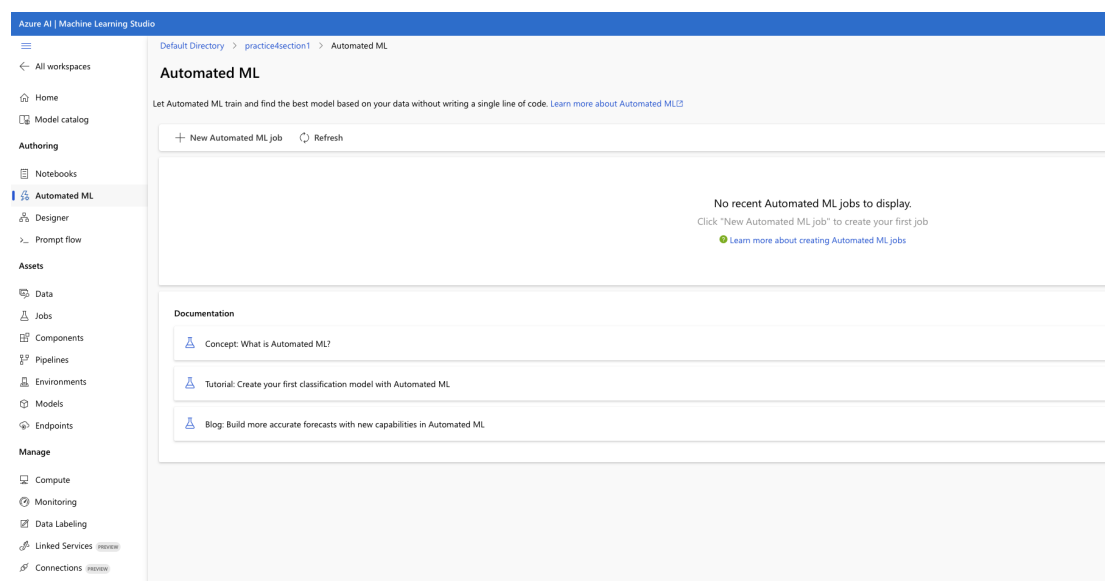


Figura 1 2: *Machine Learning Studio* de Azure.

Desde el panel de *Machine Learning Studio* de Azure, en la parte izquierda tenemos que seleccionar la opción '*Automated ML*' para crear el nuevo trabajo automatizado. En las siguientes imágenes se muestran los pasos a realizar y los valores de los parámetros que se requieren para crear el nuevo trabajo automatizado.

Submit an Automated ML job

✓ Training method

2 Basic settings

3 Task type & data

4 Task settings

5 Compute

6 Review

Basic settings

Let's start with some basic information about your training job.

Job name * ①

penguins-practice4

Experiment name *

☐ Select existing ☒ Create new

New experiment name *

01penguins

Description

ML job for penguins classification

Tags

Name

(a) Nombre del trabajo.

Submit an Automated ML job

Training method
Basic settings
Task type & data
Task settings
Compute
Review

Task type & data

Choose the type of task that you would like your model to perform and the data to use for training.

Select task type * ⓘ

Classification

- Classification**
To predict one of several categories yes/no, blue, red, green.
- Regression**
To predict continuous numeric values.
- Time series forecasting**
To predict values based on time.
- Natural language processing**
To predict based on text-only data types using multi-class or multi-label classification or named entity recognition.
- Computer vision**
To predict using multi-class or multi-label image classification, object detection, and instance segmentation.

(b) Tipo de trabajo.

Create data asset

1 Data type
2 Data source

Set the name and type for your data asset

Name *

penguins-data

Description

Penguins data set for classification experiment

Type * ⓘ

Tabular

(c) Tipo de datos.

Create data asset

2 Data source
3 Web URL
4 Settings
5 Schema
6 Review

Choose a source for your data asset

Choose the data source you want to create your asset from. A data source can be from a local storage location or a cloud storage service.

From Azure storage
Create a data asset from registered data storage services including Azure Blob Storage, Azure file share, and Azure Data Lake.

From web files
Create a data asset from a single file located at a public web URL.

From Fror
Create a data asset from a file stored in a cloud storage service.

(d) Fuente de los datos.

Create data asset

✓ Data type

✓ Data source

3 Web URL

1 Settings

2 Schema

3 Review

Enter a web URL

Specify the URL of a public web page you want your data retrieved from.

Web URL *

https://raw.githubusercontent.com/tnavarrete-iedib/bigdata-24-25/refs/heads/main/penguins_size.csv

Skip data validation

If you choose to skip validation, we will not validate your data path, or try to access your data for preview and schema.

☐ Skip data validation

(e) URL al archivo con los datos.

Create data asset

2 Data type

2 Data source

3 Web URL

1 Settings

2 Schema

3 Review

Settings

These settings determine how the data is parsed. The initial settings are automatically detected, you can change them as needed to reparse the data.

File format

Delimited

Delimiter

Comma

Example

Field1;Field2;Field3

Encoding

UTF-8

Column headers

All files have same headers

Skip rows

None

☐ Dataset contains multi-line data

Note: Processing tabular files with multi-line data is slower because multiple CPU cores cannot be used to ingest the data in parallel. Checking this option may result in slower processing times.

Data preview

species	island	culmen_length_mm	culmen_depth_mm	flipper_length_mm	body_mass_g	sex
Adelie	Torgersen	39.1	18.7	181	3750	MALE
Adelie	Torgersen	39.5	17.4	186	3800	FEMALE
Adelie	Torgersen	40.3	18	195	3250	FEMALE
Adelie	Torgersen	NA	NA	NA	NA	NA
Adelie	Torgersen	36.7	19.3	193	3450	FEMALE
Adelie	Torgersen	39.3	20.6	190	3650	MALE
Adelie	Torgersen	38.9	17.8	181	3625	FEMALE
Adelie	Torgersen	39.2	19.6	195	4675	MALE
Adelie	Torgersen	34.1	18.1	193	3475	NA
Adelie	Torgersen	42	20.2	190	4250	NA
Adelie	Torgersen	37.8	17.1	186	3300	NA
Adelie	Torgersen	37.8	17.3	180	3700	NA
Adelie	Torgersen	41.1	17.6	182	3200	FEMALE
Adelie	Torgersen	38.6	21.2	191	3800	MALE
Adelie	Torgersen	34.6	21.1	198	4400	MALE
Adelie	Torgersen	36.6	17.8	185	3700	FEMALE
Adelie	Torgersen	38.7	19	195	3450	FEMALE

(f) Formato de los datos y previsualización.

Create data asset

2 Data type

2 Data source

3 Web URL

1 Settings

2 Schema

3 Review

Schema

Column types are auto-detected based on the initial subset of the data and can be updated here. Values not aligning with the specified column type will fail conversion and would be either null-filled or repli blocking and you can proceed.

Search column name

Include

Column name

Type

Example values

Date format

Properties

<input type="checkbox"/>	Path	String		Not applicable to selected type	Not applicable to selected type
<input checked="" type="checkbox"/>	species	String	Adelie, Adelie, Adelie	Not applicable to selected type	Not applicable to selected type
<input checked="" type="checkbox"/>	island	String	Torgersen, Torgersen, Torgersen	Not applicable to selected type	Not applicable to selected type
<input checked="" type="checkbox"/>	culmen_length_mm	String	39.1, 39.5, 40.3	Not applicable to selected type	Not applicable to selected type
<input checked="" type="checkbox"/>	culmen_depth_mm	String	18.7, 17.4, 18	Not applicable to selected type	Not applicable to selected type
<input checked="" type="checkbox"/>	flipper_length_mm	String	181, 186, 195	Not applicable to selected type	Not applicable to selected type
<input checked="" type="checkbox"/>	body_mass_g	String	3750, 3800, 3250	Not applicable to selected type	Not applicable to selected type
<input checked="" type="checkbox"/>	sex	String	MALE, FEMALE, FEMALE	Not applicable to selected type	Not applicable to selected type

(g) Selección de los campos de los datos.

Create data asset

- Data type
- Data source
- Web URL
- Settings
- Schema
- Review

Review
Review the settings for your data asset and make any changes as needed.

Data type	
Name	penguins-data
Type	tabular
Description	Penguins data set for classification experiment

Data source	
Type	WebURL
Web URL	https://raw.githubusercontent.com/tnavarrete-iedlib/bigdata-24-25/refs/heads/main/penguins_size.csv
Skip data validation	false

Settings	
Delimiter	Comma
Column headers	All files have same headers
Encoding	UTF-8
Number of rows to skip	None
File format	Dataset contains multi-line data
Delimited	false

Schema	
species	String
island	String
culmen_length_mm	String
culmen_depth_mm	String
flipper_length_mm	String

(showing 5 of 8 columns)

(h) Revisión de la configuración de los datos.

Submit an Automated ML job

- Training method
- Basic settings
- Task type & data
- Task settings
- Compute
- Review

Task type & data

Success: penguins-data data asset created successfully. It may take a few seconds for lists to be updated. [Click here to go to this data asset](#)

Choose the type of task that you would like your model to perform and the data to use for training. [Learn more](#)

Select task type *

Classification

Select data

Make sure your data is preprocessed into a supported format.

+ Create Refresh Show supported data assets only Reset view

Name	Type	Created on ↓	Modified on
penguins-data	Table	Jan 11, 2025 6:13 PM	Jan 11, 2025 6:13 PM

Page 1 of 1 25/Page

(i) Tipo de trabajo.

Submit an Automated ML job

- Training method
- Basic settings
- Task type & data
- Task settings
- Compute
- Review

Task settings

Task type: Classification

Data: penguins-data (View data)

Target column: species (String)

Classification settings

☒ Enable deep learning

A GPU is required for deep learning tasks.

[View additional configuration settings](#) [View featureization settings](#)

Limits

Validate and test

You can choose a validation type and select test data as an optional step.

Validation type: Select a validation type

Test data: None

Additional configuration

Primary metric: Accuracy

☐ Explain best model

☐ Enable ensemble stacking

☐ Use all supported models

Allowed models:

- ☒ LogisticRegression
- ☐ SGD
- ☐ MultinomialNaiveBayes
- ☐ BernoulliNaiveBayes
- ☒ SVM
- ☒ KNN
- ☒ DecisionTree
- ☒ RandomForest
- ☐ ExtremeRandomTrees
- ☐ LightGBM
- ☐ GradientBoosting

(j) Selección de algoritmos y métrica.

Submit an Automated ML job

- ✓ Training method
- ✓ Basic settings
- ✓ Task type & data
- 4 Task settings**
- 5 Compute
- 6 Review

species (String)

Classification settings

☒ Enable deep learning ⓘ
 A GPU is required for deep learning tasks.
[View additional configuration settings](#) [View featurization settings](#)

Limits

Max trials ⓘ
3

Max concurrent trials ⓘ
3

Max nodes ⓘ
3

Metric score threshold ⓘ
0.1

Experiment timeout (minutes) ⓘ
15

Iteration timeout (minutes) ⓘ
15

☒ Enable early termination ⓘ

Validate and test

You can choose a validation type and select test data as an optional step.

Validation type ⓘ
Train-validation split

Percentage validation of data * ⓘ
10
 Automated ML recommends that between 10 and 30 percent of data is held out for validation

Test data ⓘ
None

[Back](#) [Next](#)

(k) Configuración del trabajo.

Submit an Automated ML job

- ✓ Training method
- ✓ Basic settings
- ✓ Task type & data
- ✓ Task settings
- 5 Compute**
- 6 Review

Compute

Select and configure the compute resource for executing your training job.

Select compute type
Serverless

Virtual machine type ⓘ
☒ CPU ☐ GPU

Virtual machine tier ⓘ
☒ Dedicated ☐ Low priority

Virtual machine size
Standard_DS3_v2 (4 core(s), 14GB RAM, 28GB storage, \$0.27/hr)

Number of instances
1

(l) Configuración de la computación.

Default Directory > practice4section1 > training job

Submit an Automated ML job

✓ Training method

✓ Basic settings

✓ Task type & data

✓ Task settings

✓ Compute

6 Review

Review

Review or make changes to your job before submission.

Basic settings

Name
penguins-practice4

Experiment name
01penguins

Timeout (hours)
--

Compute settings

Compute type
Azure ML serverless compute

Virtual machine size
Standard_DS3_v2

Instance count
1

Task type & data

Task type
Classification

Data
penguins-data

Task settings

Target column
species

Limits
Max trials: 3
Max concurrent trials: 3
Max nodes: 3
Metric score threshold: 0.1
Experiment timeout (minutes): 15
Iteration timeout (minutes): 15

Enable deep learning
No

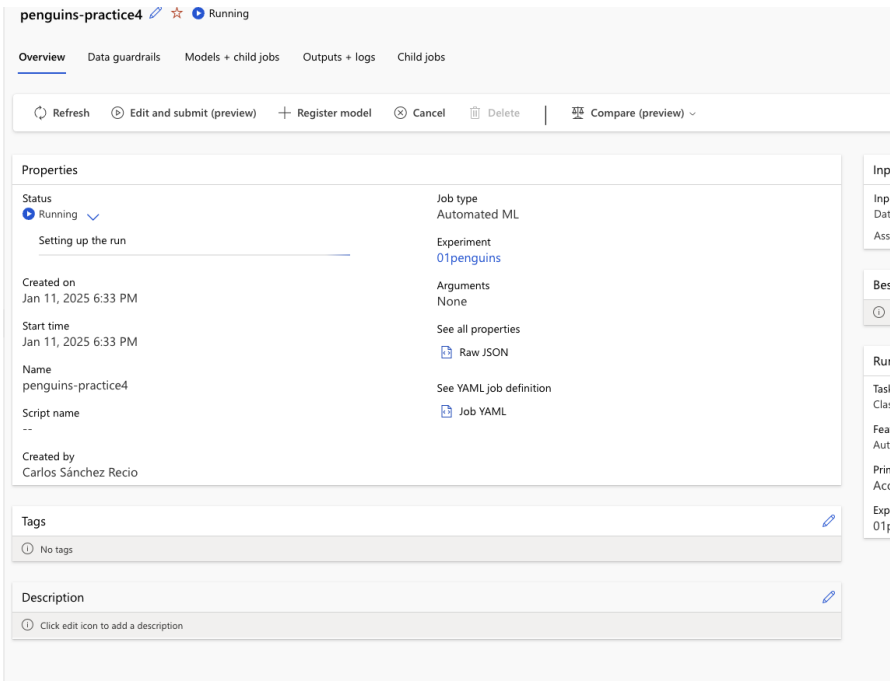
Validate type
Train-validation split

Percentage validation of data
10

Back

Submit training job

(m) Revisión de la configuración del trabajo automatizado.



(n) Trabajo automatizado en ejecución.

Figura 1 3: Creación de un trabajo automatizado de *Machine Learning* en Azure.

Tras todos estos pasos, resta esperar unos minutos hasta que el trabajo aparece como com-
pletado.

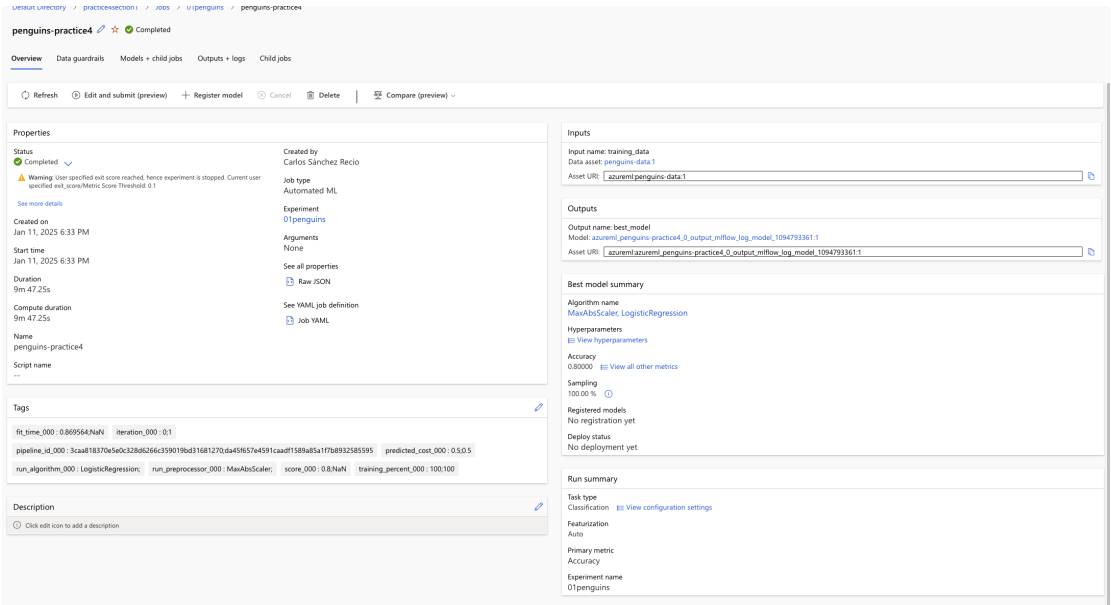


Figura 1 4: Trabajo automatizado de *Machine Learning* lanzado.

Dentro del panel cuando el modelo ya ha sido entrenado se podrá ver el algoritmo finalmente
utilizado (**Logistic Regression**) entre otros muchos datos, no solamente dentro del panel de la
imagen anterior, si no dentro de diferentes paneles que pueden ser accedidos desde éste.

The screenshot displays the MLOps console interface for a model named 'joyful_chayote_nv16qx43'. The left sidebar contains a 'Model summary' section with the following details:

- Algorithm name: MaxAbsScaler, LogisticRegression
- Hyperparameters: [View hyperparameters](#)
- Accuracy: 0.80000 [View all other metrics](#)
- Sampling: 100.00%
- Registered models: No registration yet
- Deploy status: No deployment yet

The main content area shows the 'Hyperparameters' section with the following JSON configuration:

```
{
  "class_name": "MaxAbsScaler",
  "module": "sklearn.preprocessing",
  "param_args": {},
  "param_kwargs": {},
  "prepared_kwargs": {},
  "spec_class": "preproc"
}
```

Below the hyperparameters, the 'Training algorithm' section shows the following JSON configuration:

```
{
  "class_name": "LogisticRegression",
  "module": "sklearn.linear_model",
  "param_args": {},
  "param_kwargs": {
    "C": 1.7575386248547884,
    "class_weight": null,
    "multi_class": "multinomial",
    "penalty": "l2",
    "solver": "lbfgs"
  },
  "prepared_kwargs": {},
  "spec_class": "sklearn"
}
```

Figura 1 5: Hiperparámetros utilizados en el entrenamiento.

Run Metrics

✕

Accuracy	0.80000
AUC macro	0.93940
AUC micro	0.94898
AUC weighted	0.93262
Average precision score macro	0.88267
Average precision score micro	0.90827
Average precision score weighted	0.89747
Balanced accuracy	0.76825
F1 score macro	0.77395
F1 score micro	0.80000
F1 score weighted	0.79146
Log loss	0.41287
Matthews correlation	0.68906
Norm macro recall	0.65238
Precision score macro	0.79940
Precision score micro	0.80000
Precision score weighted	0.79852
Recall score macro	0.76825
Recall score micro	0.80000
Recall score weighted	0.80000
Weighted accuracy	0.81716

Figura 1 6: Otras medidas utilizadas en el entrenamiento.

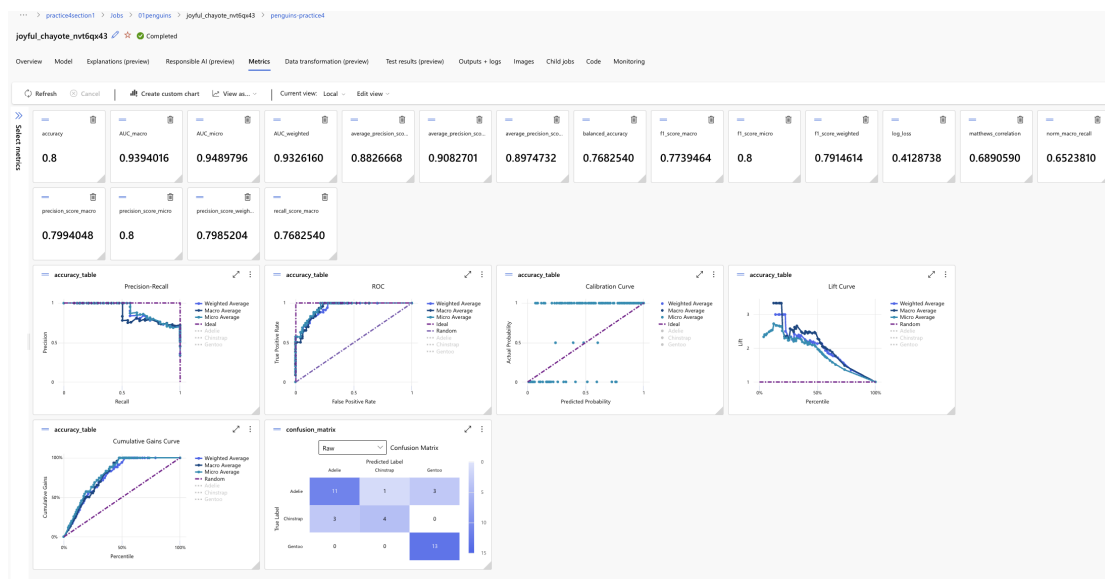


Figura 1 7: Página con las métricas.

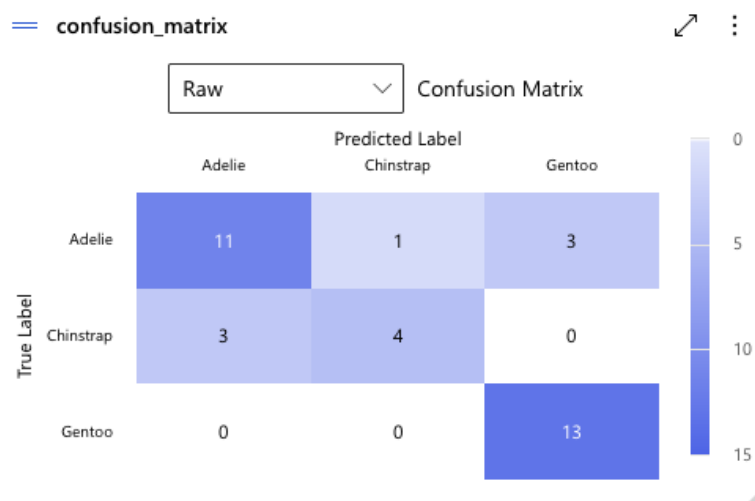


Figura 1 8: Matriz de confusión.

En esta última imagen se puede ver la matriz de confusión. Como su nombre indica es una matriz que muestra las discrepancias entre los datos reales y los que son previstos, dando así una visión de qué cantidad de error podría haber cuando se realicen predicciones. Se puede ver que es especialmente preocupante el caso Chinstrap-Chinstrap ya que la proporción se acerca al 1:1.

1.3 Lanzar el servicio a internet

Una vez el modelo está entrenado, éste puede ser lanzado a internet para ser accedido por aplicaciones externas. Para ello, desde la pestaña de modelos, seleccionamos la opción 'Deploy-Web Service' y seguimos los pasos que se muestran en las imágenes siguientes.

joyful_chayote_nvt6qx43 [🔗](#) [★](#) [✅ Completed](#)

Overview **Model** Explanations (preview) Responsible AI (preview) Metrics Data transform

Refresh Deploy Download Explain model # View generated code

Model summary

Algorithm name: MaxAbsScaler

Hyperparameter: [View hyperparameters](#)

Accuracy: 0.80000 [View](#)

Sampling: 100.00 % [ⓘ](#)

Registered models: No registration yet

Deploy status: No deployment yet

Real-time endpoint
Deploy the model using the real-time endpoint wizard

Batch endpoint
Deploy the model using the batch endpoint wizard

Web service
Deploy the model to a web service

Deploy a model [✕](#)

Name * [ⓘ](#) [👁](#)

predict-penguins

Description [ⓘ](#)

Penguins classification service for practice 4 section 1

Compute type * [ⓘ](#)

Azure Container Instance [⌵](#) *

Models: penguinspractic0

Enable authentication

☒

[ⓘ](#) Keys can be found on the endpoint details page.

This model supports [no-code deployment](#). You may **optionally** override the default environment and driver file.

Use custom deployment assets

☐ Use custom deployment assets

[> Advanced](#)

[Deploy](#) [Cancel](#)

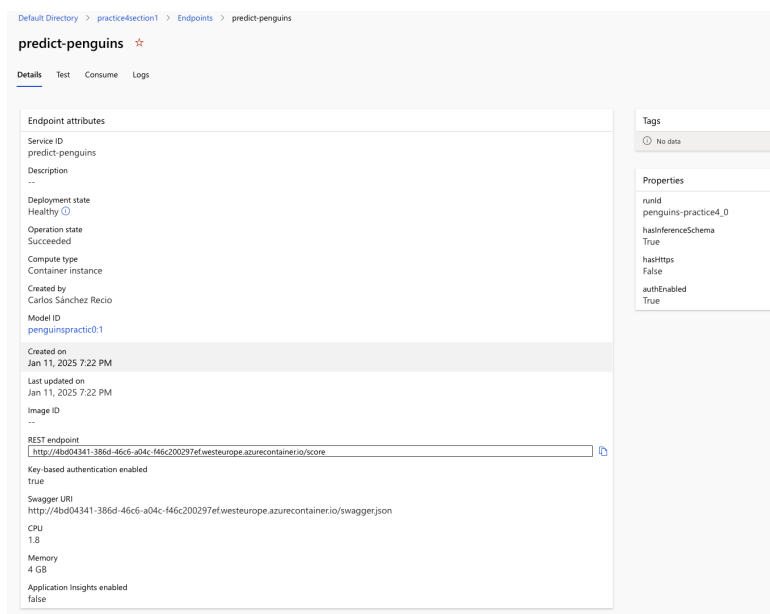


Figura 1 9: Lanzar el servicio web para el modelo.

Tras finalizar esto, se puede ver que se ha generado un *endpoint* el cual podremos utilizar para lanzar nuestras consultas desde aplicaciones externas y obtener las predicciones.

1.4 Consumir el servicio web

Para consumir el servicio web es tan sencillo como ir a la pestaña '*Consume*' en la cual aparecerá lo siguiente:

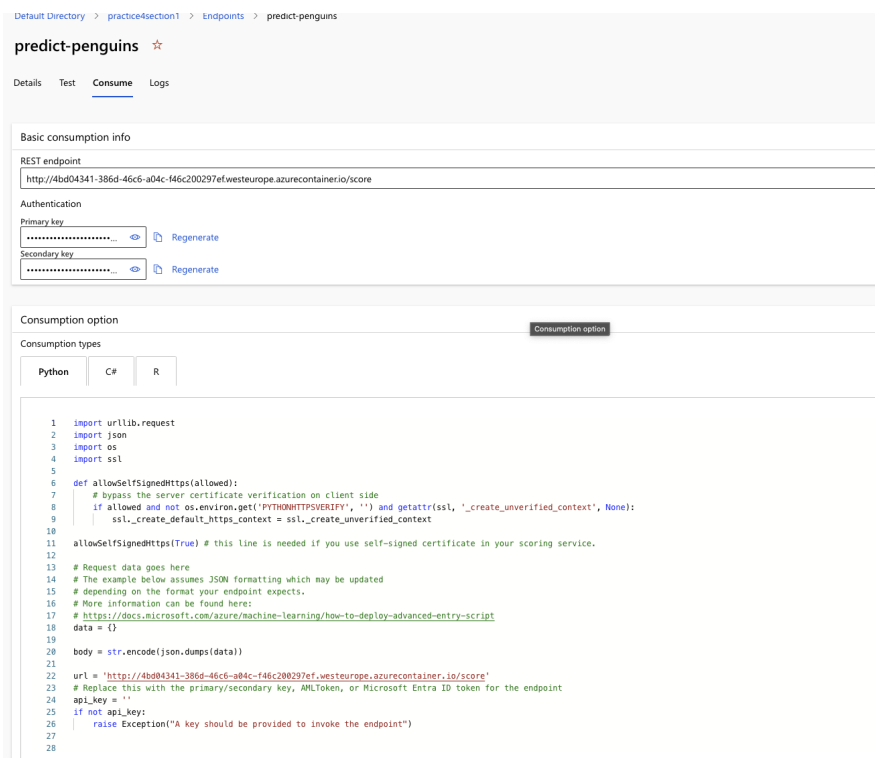


Figura 1 10: Pestaña para consumir el servicio web.

Como se puede ver se ofrecen 3 posibilidades de lenguajes de programación ya pre-configuradas para poder consumir este servicio. Usaré Python como es de esperar. Es importante no solamente copiar el código proporcionado, si no también el *endpoint* y, en caso de haberse configurado la autenticación como es el caso, la clave de autenticación de la API.

Para esta tarea he realizado dos llamadas a la API con diferentes herramientas y diferentes datos.

Recordar que para el momento que se lea este documento, el servicio ya no estará disponible.

► Cuaderno de Jupyter

[enlace al cuaderno completo en mi repositorio personal](#)

```
# https://docs.microsoft.com/azure/machine-learning/how-to-deploy-advanced-environment

data = {
    'Inputs': {
        'data': [{
            'island': 'Biscoe',
            'culmen_length_mm': 39.1,
            'culmen_depth_mm': 18.7,
            'flipper_length_mm': 181,
            'body_mass_g': 3750,
            'sex': 'Male'
        }, {
            'island': 'Dream',
            'culmen_length_mm': 48.7,
            'culmen_depth_mm': 17.4,
            'flipper_length_mm': 195,
            'body_mass_g': 3500,
            'sex': 'Female'
        }, {
            'island': 'Torgersen',
            'culmen_length_mm': 45.2,
            'culmen_depth_mm': 14.5,
            'flipper_length_mm': 210,
            'body_mass_g': 5000,
            'sex': 'Male'
        }, {
            'island': 'Dream',
            'culmen_length_mm': 38.6,
            'culmen_depth_mm': 17.2,
            'flipper_length_mm': 180,
            'body_mass_g': 3400,
            'sex': 'Female'
        }, {
            'island': 'Biscoe',
            'culmen_length_mm': 50.0,
            'culmen_depth_mm': 19.5,
            'flipper_length_mm': 196,
            'body_mass_g': 3900,
            'sex': 'Male'
        }
    ]
}

body = str.encode(json.dumps(data))

url = 'http://4bd04341-386d-46c6-a04c-f46c200297ef.westeurope.azurecontainer.io/sc
# Replace this with the primary/secondary key, AMLToken, or Microsoft Entra ID tok
api_key = 'GZaEMTINUnqrba1q5kcIWBh6rA44hcI'
if not api_key:
    raise Exception("A key should be provided to invoke the endpoint")

headers = {'Content-Type': 'application/json', 'Authorization': ('Bearer ' + api_key)}

req = urllib.request.Request(url, body, headers)

try:
    response = urllib.request.urlopen(req)
    result = response.read()
    print(result)
except urllib.error.HTTPError as error:
    print("The request failed with status code: " + str(error.code))

    # Print the headers - they include the request ID and the timestamp, which are
    print(error.info())
    print(error.read().decode("utf8", 'ignore'))

✓ 0.1s

{"Results": ["Adelie", "Chinstrap", "Adelie", "Adelie", "Adelie"]}
```

► Extensión de VSCode de RapidAPI

The screenshot shows the VS Code interface with the RapidAPI extension. The top bar indicates the current endpoint is `http://4bd04341-386d-46c6-a04c-f46` and the method is `POST`. The status bar shows a `200 OK` response with a `189 ms` response time.

The left pane displays the request body in JSON format, which is a list of five penguin input records. The right pane displays the response in JSON format, which is a list of results.

Request Body (JSON):

```
1 {
2   "Inputs": {
3     "data": [
4       {
5         "island": "Dream",
6         "culmen_length_mm": 41,
7         "culmen_depth_mm": 18,
8         "flipper_length_mm": 190,
9         "body_mass_g": 3800,
10        "sex": "Female"
11      },
12      {
13        "island": "Torgersen",
14        "culmen_length_mm": 49.3,
15        "culmen_depth_mm": 16.8,
16        "flipper_length_mm": 200,
17        "body_mass_g": 3550,
18        "sex": "Male"
19      },
20      {
21        "island": "Biscoe",
22        "culmen_length_mm": 44,
23        "culmen_depth_mm": 15,
24        "flipper_length_mm": 215,
25        "body_mass_g": 5050,
26        "sex": "Female"
27      },
28      {
29        "island": "Torgersen",
30        "culmen_length_mm": 39,
31        "culmen_depth_mm": 17.5,
32        "flipper_length_mm": 182,
33        "body_mass_g": 3450,
34        "sex": "Male"
35      },
36      {
37        "island": "Dream",
38        "culmen_length_mm": 51.5,
39        "culmen_depth_mm": 20,
40        "flipper_length_mm": 198,
41        "body_mass_g": 4000,
42        "sex": "Female"
43      }
44    ]
45  }
46 }
```

Response Body (JSON):

```
1 {
2   "Results": [
3     "Chinstrap",
4     "Adelie",
5     "Gentoo",
6     "Adelie",
7     "Chinstrap"
8   ]
9 }
```

1.5 Limpiar los recursos utilizados

Para limpiar los recursos utilizados primero es necesario eliminar el *endpoint* del servicio que se ha creado.

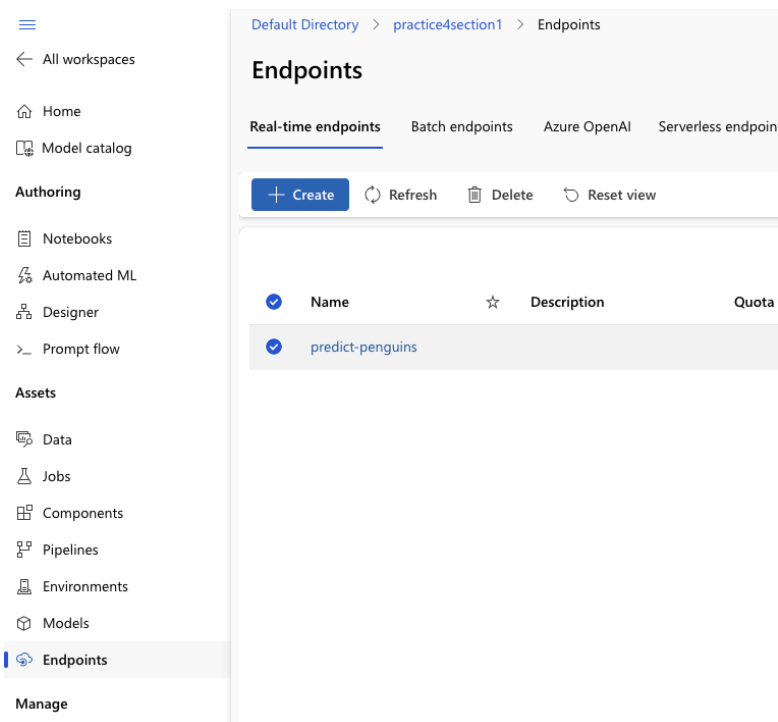
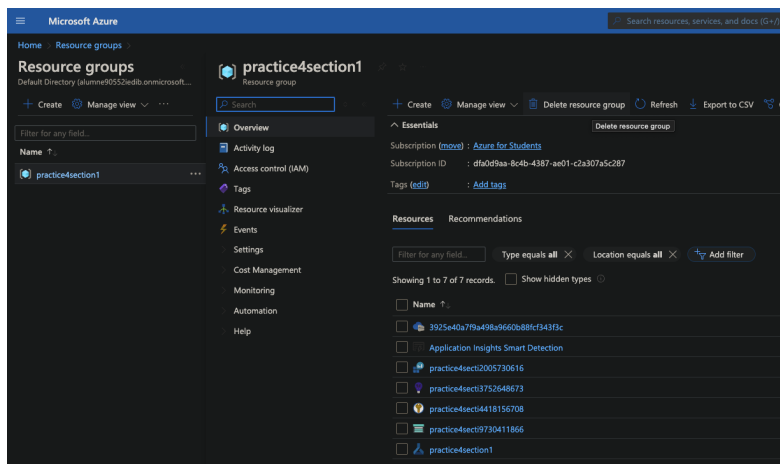


Figura 1 11: Eliminar el *endpoint* del servicio web.

Luego queda volver al portal de azure y eliminar el grupo de recursos tal y como se indica en los apuntes del bloque.



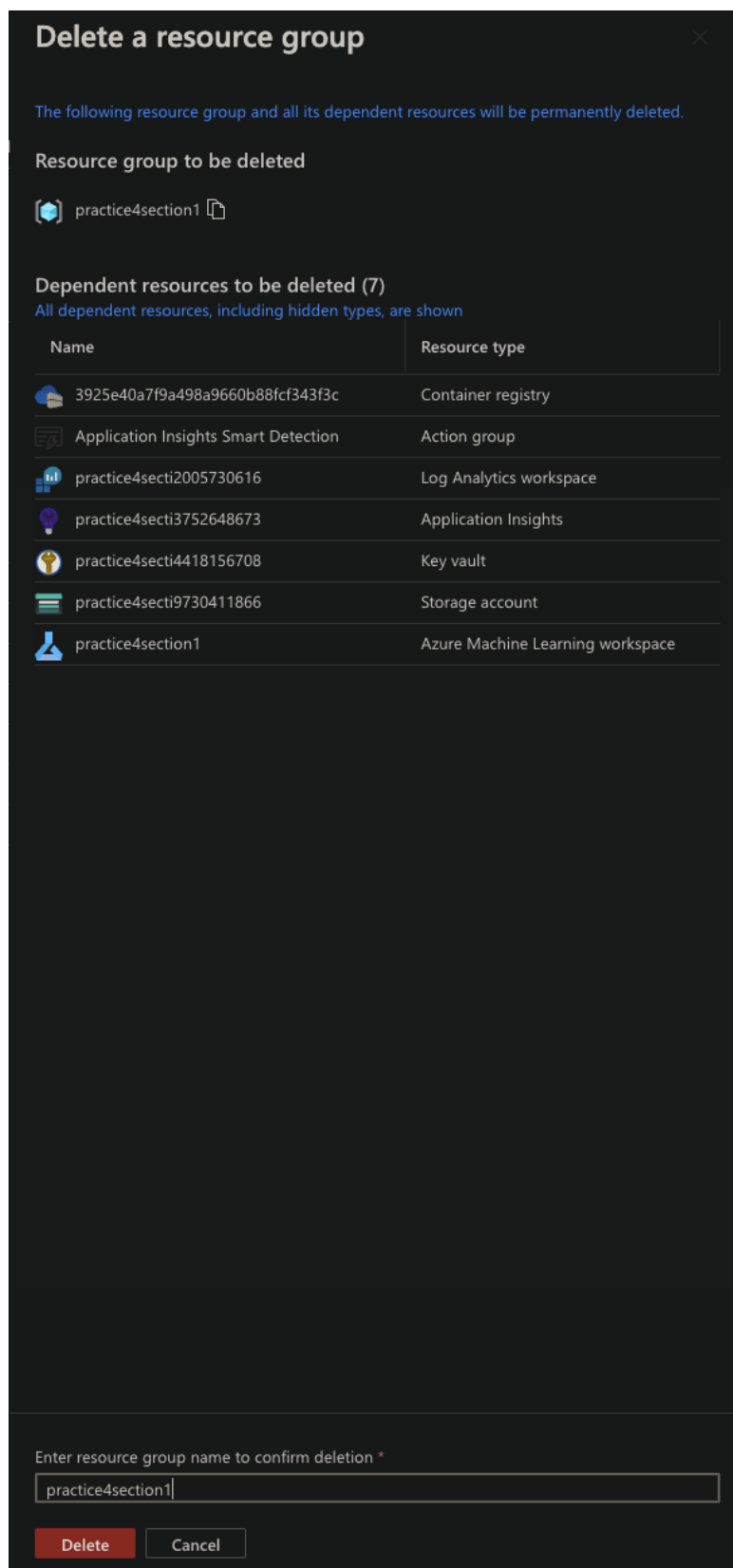


Figura 1 12: Eliminar el grupo de recursos para el trabajo automatizado de *Machine Learning*.

Se escribe el nombre del grupo y se selecciona 'Delete'.