



Introducción a Machine Learning con AWS.

Victor Ascencio | 25 de Octubre 2020

[Linkedin](#)

vascencio@gmail.com

¿QUE ES MACHINE LEARNING?

Es la rama del campo de la inteligencia Artificial, que busca como dotar a las maquinas de capacidad de aprendizaje.

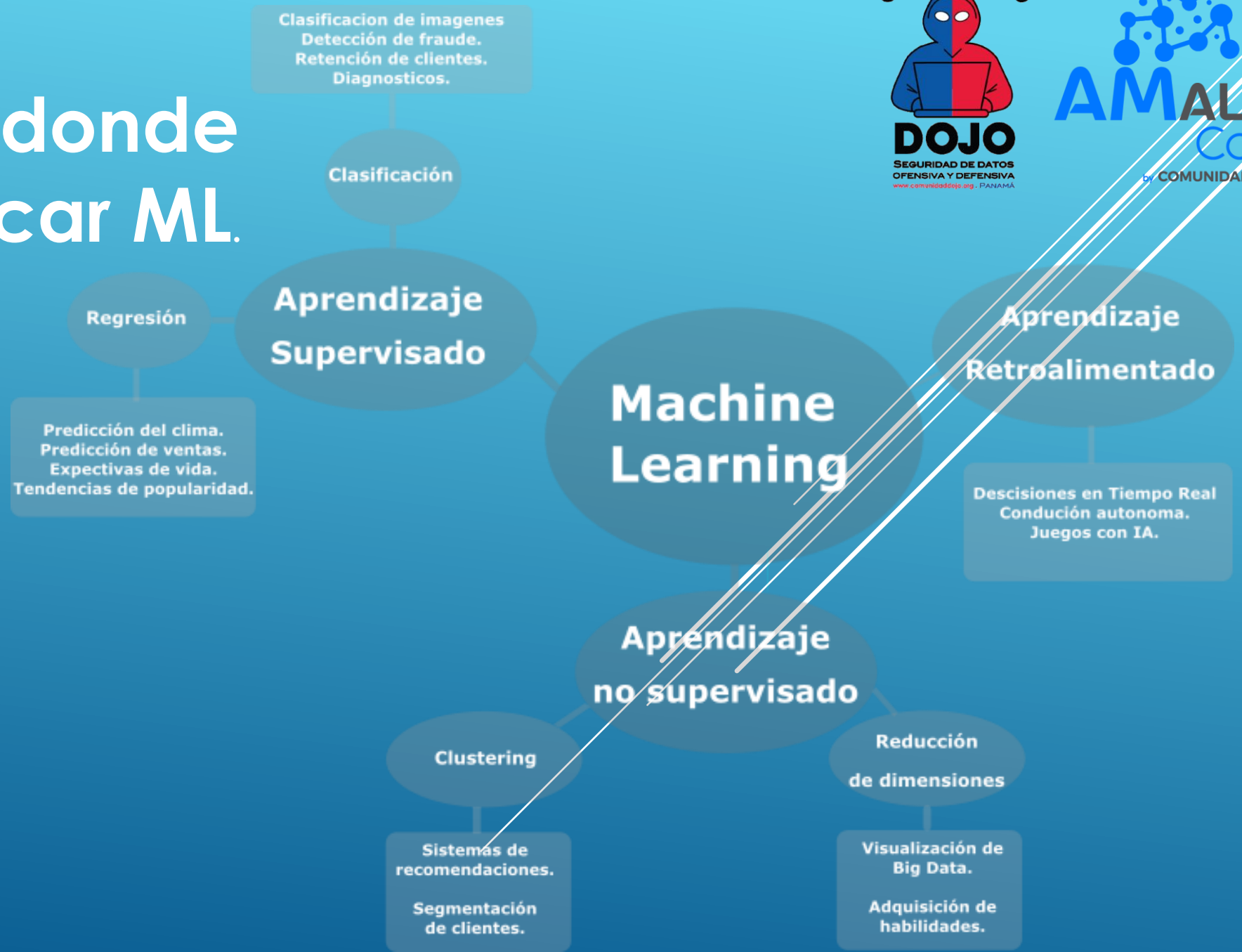


¿CÓMO APRENDEN LAS MAQUINAS?

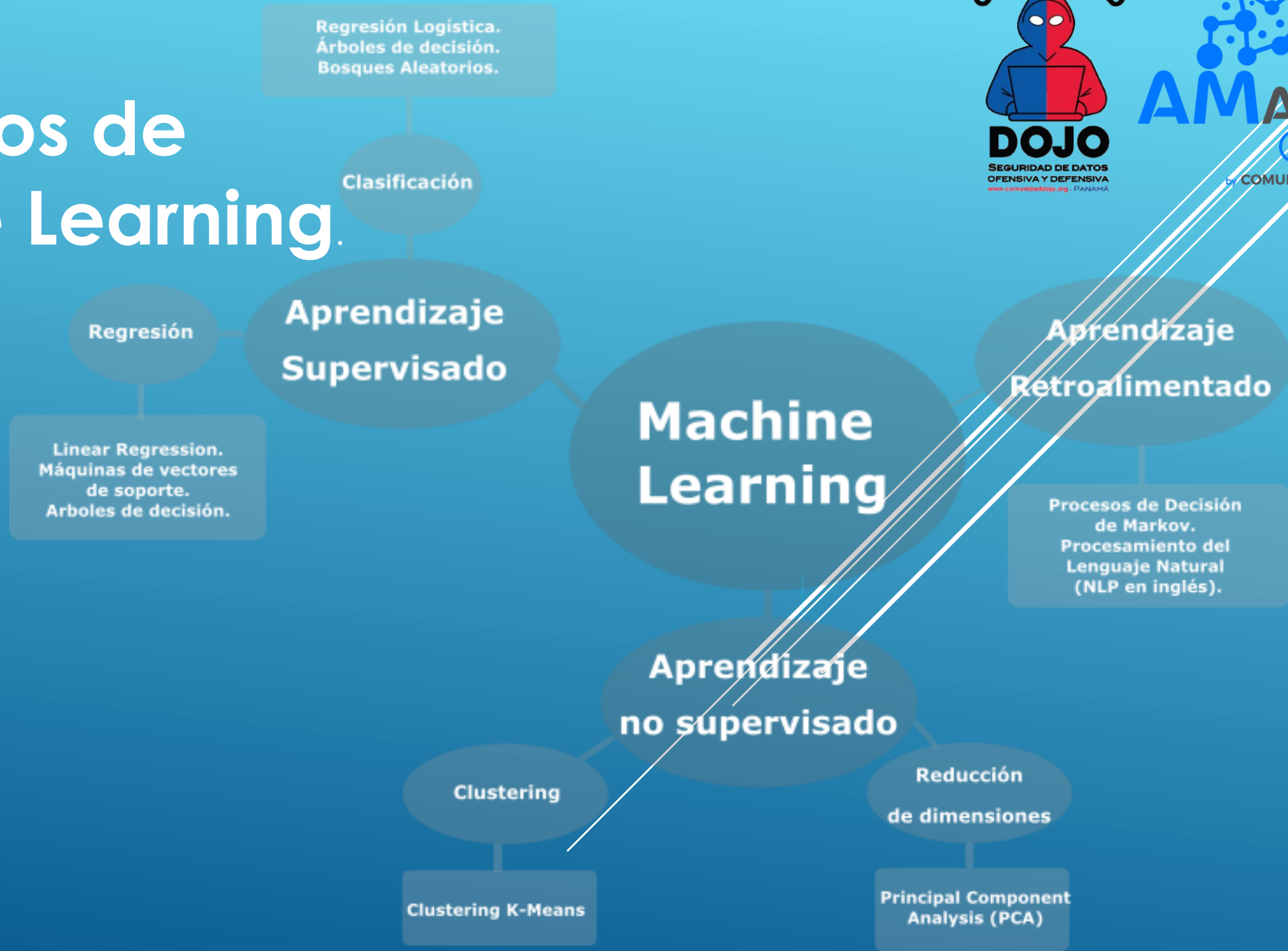


1	<input type="checkbox"/>	3	Lunes	<input type="checkbox"/>	Si
2	<input type="checkbox"/>	6	Domingo	<input type="checkbox"/>	No
3	<input type="checkbox"/>	9	Lunes	<input type="checkbox"/>	Si
4	<input type="checkbox"/>	12	Domingo	<input type="checkbox"/>	No
5	<input type="checkbox"/>	X	Lunes	<input type="checkbox"/>	X

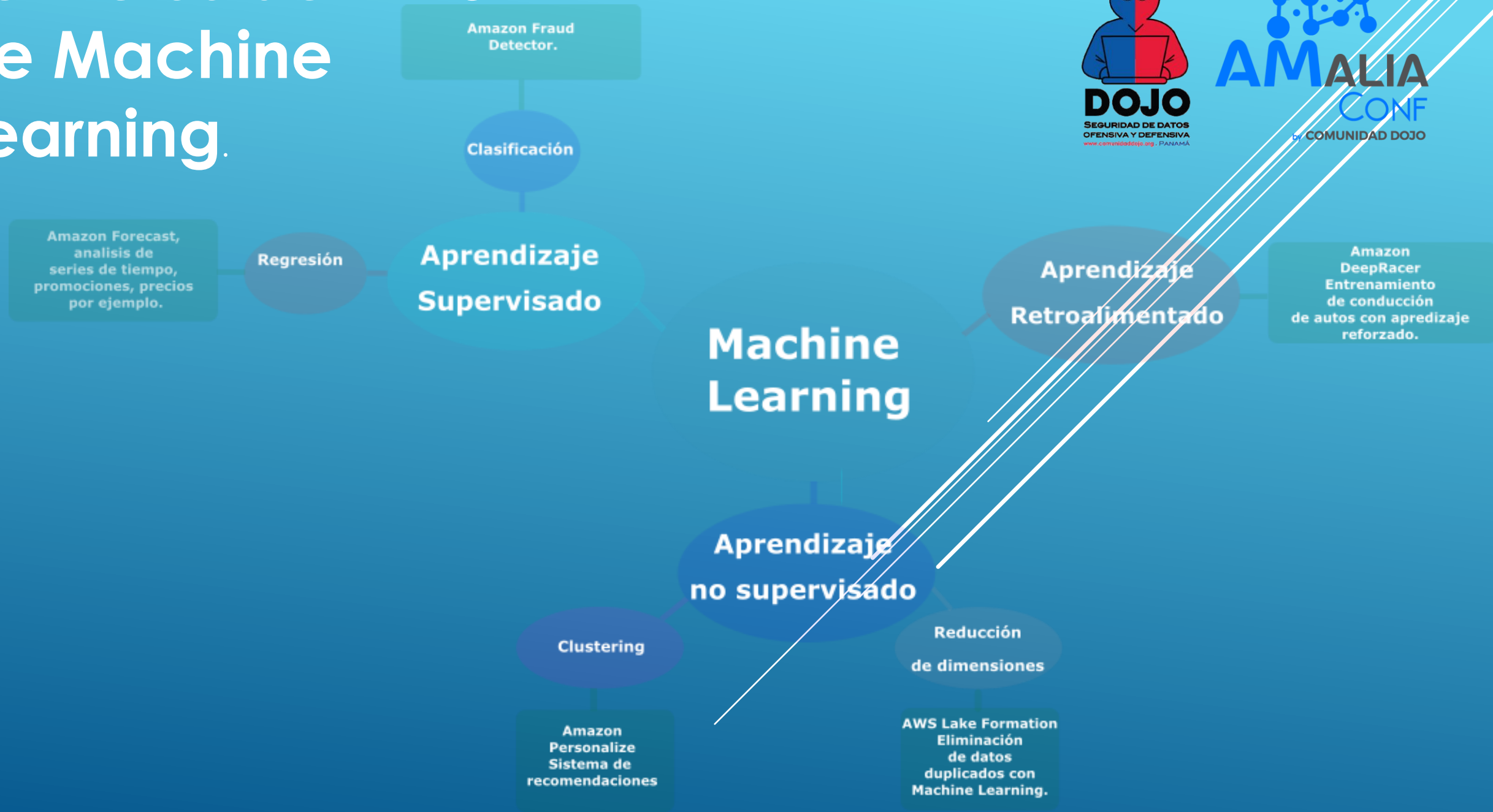
Escenarios donde puedo aplicar ML.



Algoritmos de Machine Learning.



Servicios de AWS de Machine Learning.



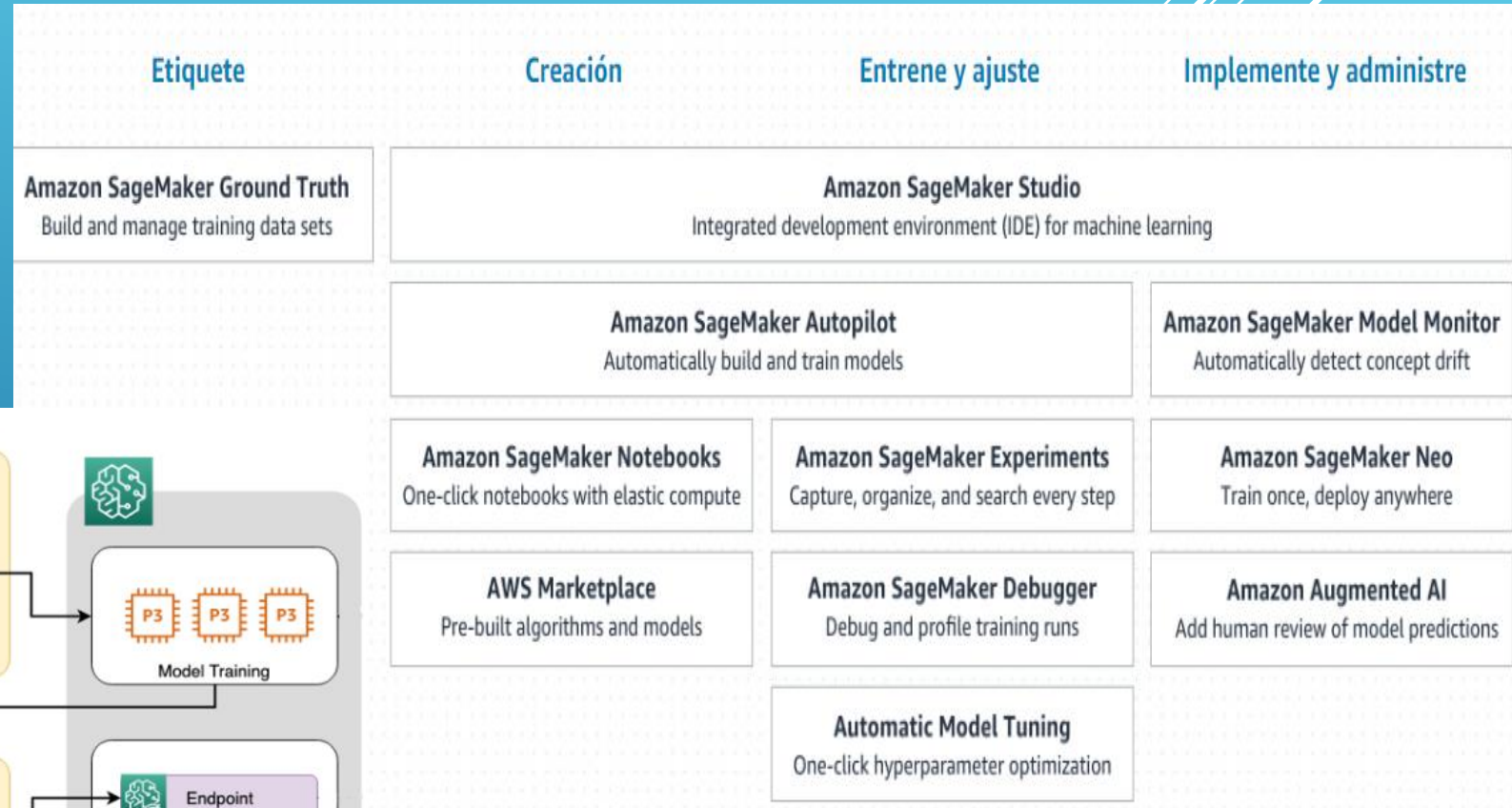
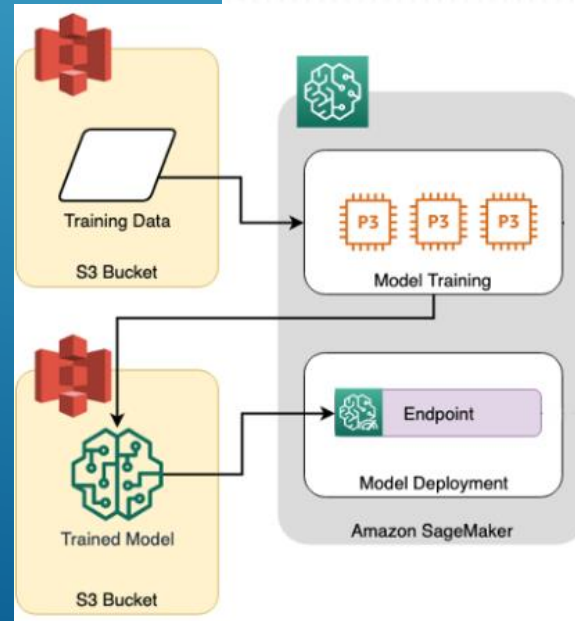
Proceso de despliegue de un Proyecto de Machine Learning



SAGEMAKER


Servicios disponibles:

- Data Labeling.
- Model Building.
- Training
- Tuning and Deployment.



SAGEMAKER



**Services** ^ **Resource Groups** v

History

Amazon Forecast

Amazon Fraud Detector

Amazon Rekognition

Amazon Comprehend

Console Home

S3

Find a service by name or feature (for example, EC2, S3 or VM, storage).


API Gateway

Direct Connect

AWS App Mesh

AWS Cloud Map

Global Accelerator ↗

 **Machine Learning**

Amazon SageMaker

Amazon Augmented AI

Amazon CodeGuru

Amazon Comprehend

Amazon Forecast

Amazon Fraud Detector

Amazon Kendra

Amazon Lex

Amazon Personalize

Amazon Polly

Amazon Rekognition


Amazon Textract

Amazon Transcribe

Amazon Translate

AWS DeepComposer

AWS DeepLens

 **Developer Tools**

CodeStar

CodeCommit

CodeArtifact

CodeBuild

CodeDeploy

CodePipeline

Cloud9

X-Ray

Amazon SageMaker

Amazon SageMaker Studio

Dashboard

Search

► Ground Truth


► Notebook

► Processing

► Training

SAGEMAKER




 Amazon SageMaker Studio

FileEditViewRunKernelGitTabsSettingsHelp

Launcher × Amazon SageMaker Studio ×


Welcome to Amazon SageMaker Studio



Build and train

Spin up sharable Jupyter Notebooks in seconds to build ML models and launch new experiments. Easily organize, track and compare experiments using SageMaker Experiments. Run distributed training, and troubleshoot models with SageMaker Debugger.


Create a notebook



Deploy and monitor

Deploy your models with auto scaling, and automatically monitor for drift in production using SageMaker Model Monitor.

Deploy and monitor models



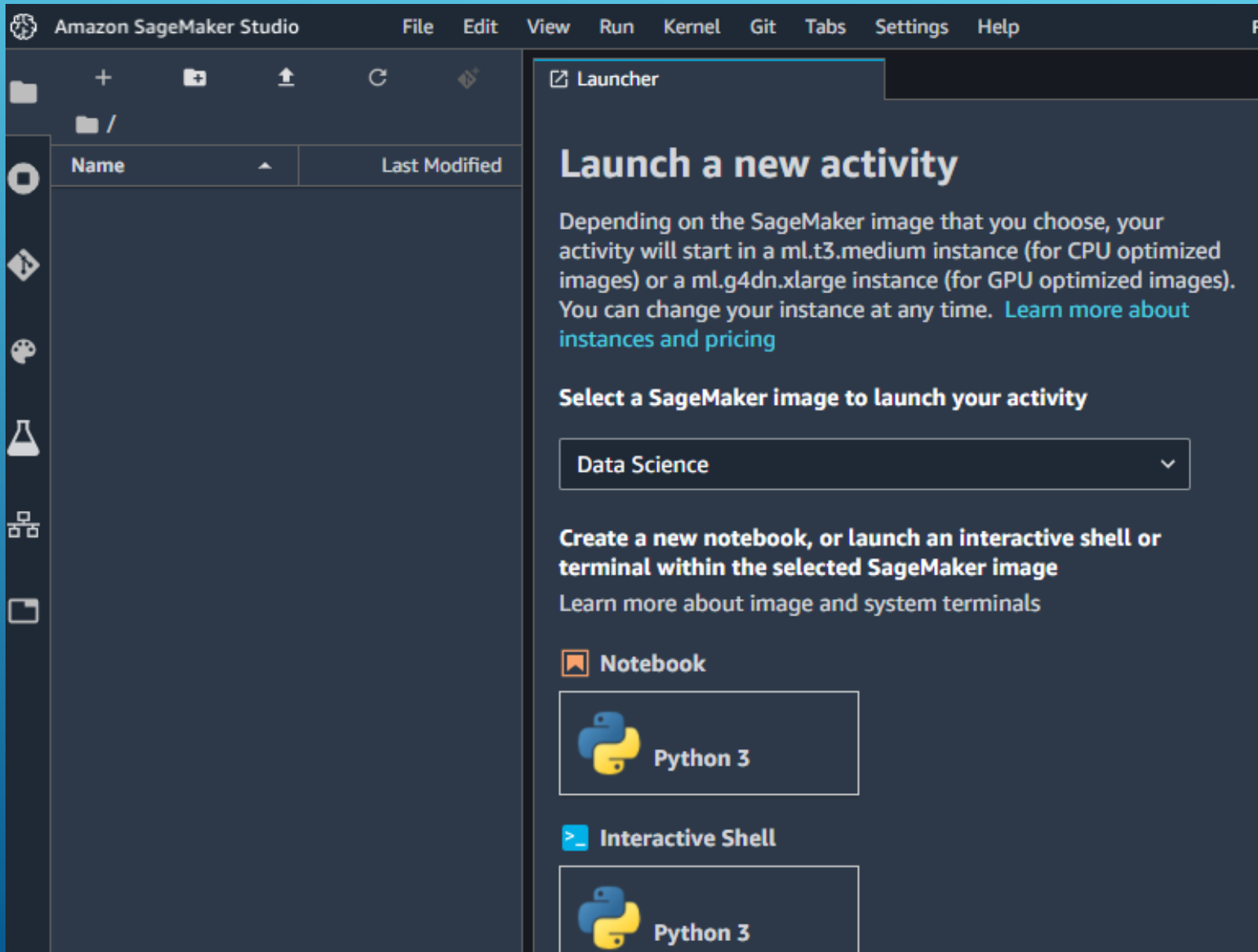
Build models automatically

Automatically build, train, and tune models with full visibility and control, using SageMaker Autopilot.

Create Autopilot experiment

Version: 1.24.3

SAGEMAKER



The screenshot displays the Amazon SageMaker Studio web interface. The top navigation bar includes 'Amazon SageMaker Studio' and menu items: 'File', 'Edit', 'View', 'Run', 'Kernel', 'Git', 'Tabs', 'Settings', and 'Help'. On the left, a sidebar contains icons for file management, a notebook, a terminal, a palette, a flask, a grid, and a folder. The main content area is titled 'Launcher' and features the heading 'Launch a new activity'. Below this, a paragraph explains that the activity will start in a specific SageMaker instance (ml.t3.medium for CPU or ml.g4dn.xlarge for GPU) and provides a link to 'Learn more about instances and pricing'. A section titled 'Select a SageMaker image to launch your activity' contains a dropdown menu currently set to 'Data Science'. Further down, instructions prompt the user to 'Create a new notebook, or launch an interactive shell or terminal within the selected SageMaker image' and provide another link to 'Learn more about image and system terminals'. Two options are listed: 'Notebook' and 'Interactive Shell', each with a 'Python 3' button featuring the Python logo.

Amazon SageMaker Studio

File Edit View Run Kernel Git Tabs Settings Help

Launcher

Launch a new activity

Depending on the SageMaker image that you choose, your activity will start in a ml.t3.medium instance (for CPU optimized images) or a ml.g4dn.xlarge instance (for GPU optimized images). You can change your instance at any time. [Learn more about instances and pricing](#)

Select a SageMaker image to launch your activity

Data Science

Create a new notebook, or launch an interactive shell or terminal within the selected SageMaker image

[Learn more about image and system terminals](#)

Notebook

Python 3

Interactive Shell

Python 3



SAGEMAKER

Importación de las librerías.

```
[61]: #Preparación de datos
import os
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import OrdinalEncoder
import sagemaker

# Modelado
from sklearn.svm import SVC
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier, AdaBoostClassifier
from sklearn.neighbors import KNeighborsClassifier

#Prueba Inferencia.
import boto3
import pandas as pd
```



SAGEMAKER

Lectura de datos.



Descarga y Lectura del dataset

```
[38]: %%sh
#wget -N -P data http://archive.ics.uci.edu/ml/machine-learning-databases/00222/bank.zip
#unzip data/bank.zip -d data
```

Verificamos la ruta de trabajo para los datos.

```
[39]: pwd
```

```
[39]: '/root/sagemaker-studio-demo'
```

Hacemos una inspección de los datos.

```
[119]: DATA_PATH = '/root/sagemaker-studio-demo/bank/data/'
df = pd.read_csv(DATA_PATH+'bank-full.csv', sep=';')
df
```

```
[119]:
```

	age	job	marital	education	default	balance	housing	loan	contact	day	month	duration	campaign	pdays	previous	poutcome	y
0	58	management	married	tertiary	no	2143	yes	no	unknown	5	may	261	1	-1	0	unknown	no
1	44	technician	single	secondary	no	29	yes	no	unknown	5	may	151	1	-1	0	unknown	no

SAGEMAKER

Lectura de datos.

```
[65]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 45211 entries, 0 to 45210
Data columns (total 17 columns):
#   Column      Non-Null Count  Dtype
---  -
0   age         45211 non-null  int64
1   job         45211 non-null  object
2   marital     45211 non-null  object
3   education   45211 non-null  object
4   default     45211 non-null  object
5   balance     45211 non-null  int64
6   housing     45211 non-null  object
7   loan        45211 non-null  object
8   contact     45211 non-null  object
9   day         45211 non-null  int64
10  month       45211 non-null  object
11  duration    45211 non-null  int64
12  campaign    45211 non-null  int64
13  pdays       45211 non-null  int64
14  previous    45211 non-null  int64
15  poutcome    45211 non-null  object
16  y           45211 non-null  object
dtypes: int64(7), object(10)
memory usage: 5.9+ MB
```



SAGEMAKER

Adecuación de los datos.



Separamos el dataset en el grupo de entrenamiento y test

```
[25]: train, test = train_test_split(data, test_size=0.2)
```

Guardamos los datos en csv para el entrenamiento y validación posterior.

```
[27]: train_file = DATA_PATH+'bank-train.csv'
pd.DataFrame.from_records(train).to_csv(train_file, index=False, header=True, sep=',')

test_file = DATA_PATH+'bank-test.csv'
pd.DataFrame.from_records(test).to_csv(test_file, index=False, header=True, sep=',')
```

Subimos el dataset para entrenamiento al bucket de S3

```
[30]: session = sagemaker.Session()
uri = session.upload_data(path=train_file, key_prefix='bank')
```

SAGEMAKER

Creamos el experimento con autopilot.

```
[29]: print(uri)
```

```
s3://sagemaker-us-east-1-682428711111/bank/bank-train.csv
```

JOB SETTINGS

Experiment Name

Maximum of 63 alphanumeric characters. Can include hyphens (-), but not spaces. Must be unique within your account in an AWS Region.

Input data location (S3 bucket)

Enter the location in S3 where your training data is stored. You can point to a single data file, an S3 object key prefix that contains only data files, or a manifest file that contains the location of your input data. See more in the [AWS Docs](#) 

☒ Find S3 bucket ☐ Enter S3 bucket location

Note: The S3 bucket must be in the same AWS Region where you're running SageMaker Studio because SageMaker doesn't allow cross-region requests.

S3 bucket name



SAGEMAKER

Creamos el experimento con autopilot.

```
[65]: df.info()
```

```
16 y          45211 non-null object
```

For more information on the format of a manifest file, please see the [AWS Docs](#)

Target attribute name

The target attribute is the attribute in your dataset that you want Amazon SageMaker Autopilot to make predictions for.

The attribute name is case-sensitive and must match exactly the name in your input dataset

Output data location (S3 bucket)

Enter the location in S3 where you want to store the output.

☒ Find S3 bucket ☐ Enter S3 bucket location

Note: The S3 bucket must be in the same AWS Region where you're running SageMaker Studio because SageMaker doesn't allow cross-region requests.

S3 bucket name



SAGEMAKER

- Seleccionamos el tipo de entrenamiento.

Select the machine learning problem type

- ☒ Auto
- ☐ Binary classification
- ☐ Regression
- ☐ Multiclass classification

Do you want to run a complete experiment?

- ☒ Yes
- ☐ No, run a pilot to create a notebook with candidate definitions



SAGEMAKER

. Esperamos a que se autoejecuten las etapas del entrenamiento.



less than 10 seconds ago

EXPERIMENT: AMALIADemo251020

Analyzing Data

Feature Engineering

Model Tuning

Completed

If experiment is taking too long to run, you can [stop the experiment](#)

You can always return to this page later by choosing this experiment on the Experiments tab in the navigation panel.

Trials

Job profile

SAGEMAKER

. Esperamos a que se autoejecuten las etapas del entrenamiento.



half a minute ago

EXPERIMENT: AMALIADemo251020

Open candidate generation notebook

Open data exploration notebook

✓ Analyzing Data

⌚ Feature Engineering

⌚ Model Tuning

⌚ Completed

Amazon SageMaker Autopilot is extracting features from your dataset.

If experiment is taking too long to run, you can [stop the experiment](#)

You can always return to this page later by choosing this experiment on the Experiments tab in the navigation panel.

Trial Job profile

SAGEMAKER

Notebook generado.

Open data exploration notebook

1. Understand how the job analyzed features to select the candidate pipelines.
2. Modify and improve the generated AutoML pipelines using knowledge that you have about the dataset.

We read **36168** rows from the training dataset. The dataset has **17** columns and the column named **16** is used as the target column. This is identified as a **BinaryClassification** problem. Here are 2 examples of labels: **['1.0', '0.0']**.

💡 Suggested Action Items

- Verify the number of unique values of a feature is expected with respect to domain knowledge. If it differs, one explanation could be multiple encodings of a value. For example **US** and **U.S.** will count as two different words. You could correct the error at the data source or pre-process your dataset in your S3 bucket.



SAGEMAKER

Esperamos a que se autoejecuten las etapas del entrenamiento.



less than 10 seconds ago

EXPERIMENT: AMALIADemo251020

Open candidate generation notebook

Open data exploration notebook

✓ Analyzing Data

✓ Feature Engineering

⌚ Model Tuning

⌚ Completed

Amazon SageMaker Autopilot is tuning the model.

If experiment is taking too long to run, you can [stop the experiment](#)

You can always return to this page later by choosing this experiment on the Experiments tab in the navigation panel.

Trial name	Status	Start time	Objective
tuning-job-1-a1d004f8e15b4d6381-069...	In Progress	13 seconds ago	
tuning-job-1-a1d004f8e15b4d6381-071...	In Progress	13 seconds ago	
tuning-job-1-a1d004f8e15b4d6381-070...	In Progress	13 seconds ago	
tuning-job-1-a1d004f8e15b4d6381-068...	In Progress	1 minute ago	
tuning-job-1-a1d004f8e15b4d6381-067...	In Progress	1 minute ago	

SAGEMAKER

. Análisis de modelo.



EXPERIMENT: AMALIADemo251020

Open candidate generation notebook

Open data exploration notebook

✓ Analyzing Data

✓ Feature Engineering

🕒 Model Tuning

🕒 Completed

Amazon SageMaker Autopilot is tuning the model.

If experiment is taking too long to run, you can [stop the experiment](#)

You can always return to this page later by choosing this experiment on the Experiments tab in the navigation panel.

Trial

Job profile

TRIALS

1 row selected

Deploy model

Trial name	Status	Start time	Objective: F1
tuning-job-1-a1d004f8e15b4d6381-041...	Completed	5 minutes ago	0.7587800025939941
★ Best: tuning-job-1-a1d004f8e15b4d6...	Completed	10 minutes ago	0.7573999762535095
tuning-job-1-a1d004f8e15b4d6381-005...	Completed	14 minutes ago	0.7544000148773193
tuning-job-1-a1d004f8e15b4d6381-025...	Completed	10 minutes ago	0.7540799975395203

SAGEMAKER

Notebook open candidate .



The AutoML recommendation job has recommended the following hyperparameters, objectives and accuracy metrics for the algorithm and problem type:

```
[ ]: ALGORITHM_OBJECTIVE_METRICS = {
    'xgboost': 'validation:f1',
    'linear-learner': 'validation:binary_f_beta',
}

STATIC_HYPERPARAMETERS = {
    'xgboost': {
        'objective': 'binary:logistic',
        'save_model_on_termination': 'true',
        'scale_pos_weight': 7.524157435776573,
    },
    'linear-learner': {
        'predictor_type': 'binary_classifier',
        'loss': 'logistic',
        'mini_batch_size': 800,
        'binary_classifier_model_selection_criteria': 'loss_function',
        'num_models': 1,
        'positive_example_weight_mult': 7.524157435776573,
    },
}
```

SAGEMAKER

Notebook open candidate .



```
[ ]: from sagemaker.parameter import CategoricalParameter, ContinuousParameter, IntegerParameter
```

```
ALGORITHM_TUNABLE_HYPERPARAMETER_RANGES = {
    'xgboost': {
        'num_round': IntegerParameter(2, 1024, scaling_type='Logarithmic'),
        'max_depth': IntegerParameter(2, 8, scaling_type='Logarithmic'),
        'eta': ContinuousParameter(1e-3, 1.0, scaling_type='Logarithmic'),
        'gamma': ContinuousParameter(1e-6, 64.0, scaling_type='Logarithmic'),
        'min_child_weight': ContinuousParameter(1e-6, 32.0, scaling_type='Logarithmic'),
        'subsample': ContinuousParameter(0.5, 1.0, scaling_type='Linear'),
        'colsample_bytree': ContinuousParameter(0.3, 1.0, scaling_type='Linear'),
        'lambda': ContinuousParameter(1e-6, 2.0, scaling_type='Logarithmic'),
        'alpha': ContinuousParameter(1e-6, 2.0, scaling_type='Logarithmic'),
    },
    'linear-learner': {
        'wd': ContinuousParameter(1e-7, 1.0, scaling_type='Logarithmic'),
        'l1': ContinuousParameter(1e-7, 1.0, scaling_type='Logarithmic'),
        'learning_rate': ContinuousParameter(1e-5, 1.0, scaling_type='Logarithmic'),
    },
}
```

SAGEMAKER

Notebook open candidate .



Finally, deploy the model to SageMaker to make it functional.

```
pipeline_model.deploy(initial_instance_count=1,  
                       instance_type='ml.m5.2xlarge',  
                       endpoint_name=pipeline_model.name,  
                       wait=True)
```

Congratulations! Now you could visit the sagemaker [endpoint console page](#) to find the deployed endpoint (it'll take a few minutes to be in service).

SAGEMAKER

. Hacemos el despliegue de end point.



Deploy model

REQUIRED SETTINGS

Endpoint name

Sagemaker07082020

Maximum of 63 alphanumeric characters. Can include hyphens (-), but not spaces. Must be unique within your account in an AWS Region.

Instance type

mLm5.xlarge



Instance count

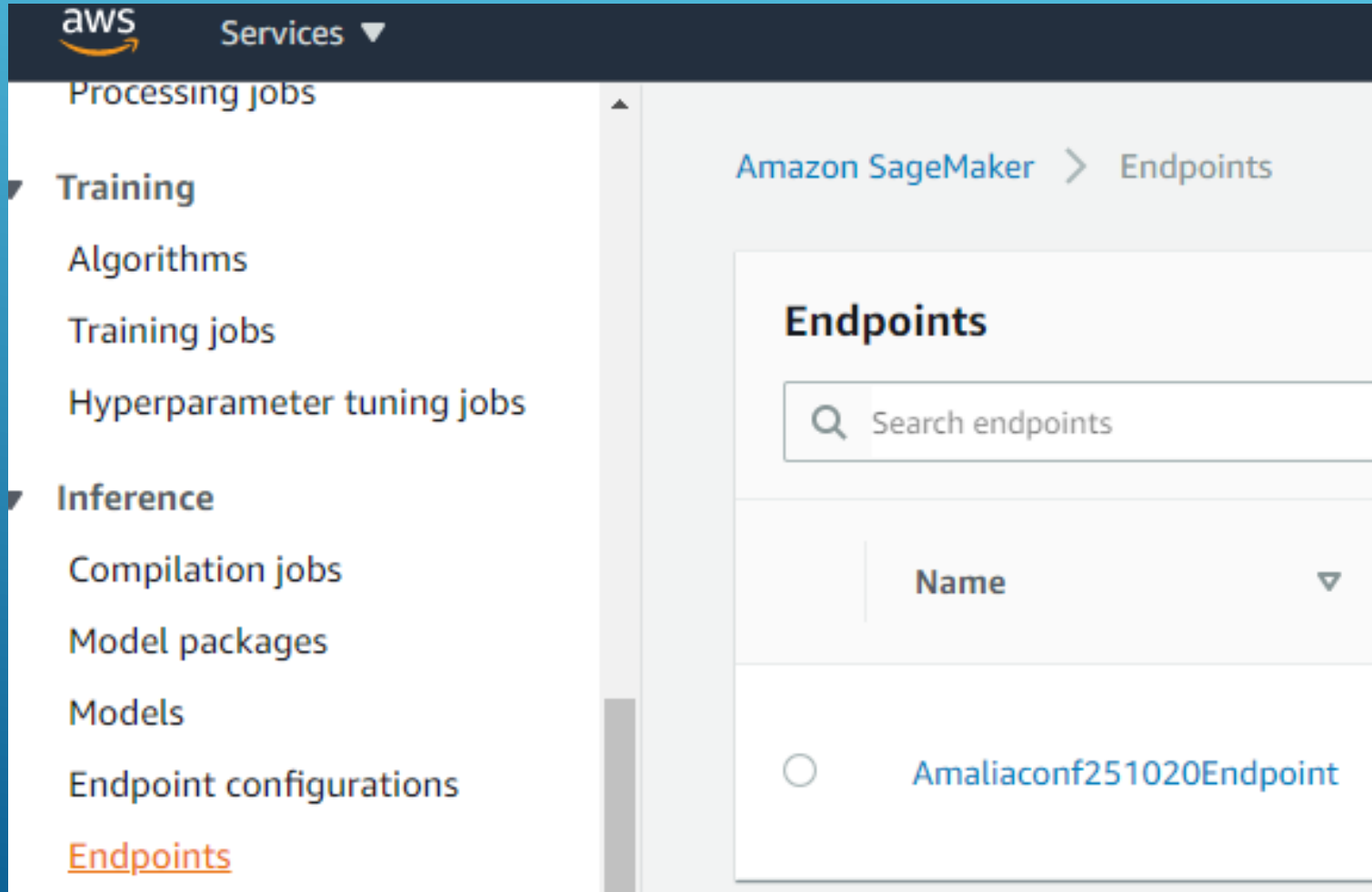
1

Data capture

SageMaker Studio will save prediction requests and responses from the endpoint to an Amazon S3 location specified below

SAGEMAKER

Verificamos el despliegue del endpoint.



SAGEMAKER

Hacemos la prueba de inferencia.



Prueba de Inferencia.

```
[13]: sm = boto3.Session().client('runtime.sagemaker')
```

```
[14]: def infer(endpoint, input_data):  
      response = sm.invoke_endpoint(EndpointName=endpoint, ContentType='text/csv', Accept='text/csv', Body=input_data)  
      prediction = response['Body'].read().decode("utf-8")  
      return prediction
```

```
[19]: endpoint = 'Amaliaconf251020Endpoint'  
      data = '17.0,1.0,1.0,1.0,0.0,916.0,1.0,0.0,0.0,8.0,5.0,279.0,0.0,0.0,0.0,3.0'  
  
      prediction = infer(endpoint, data)  
      print(prediction[0])
```

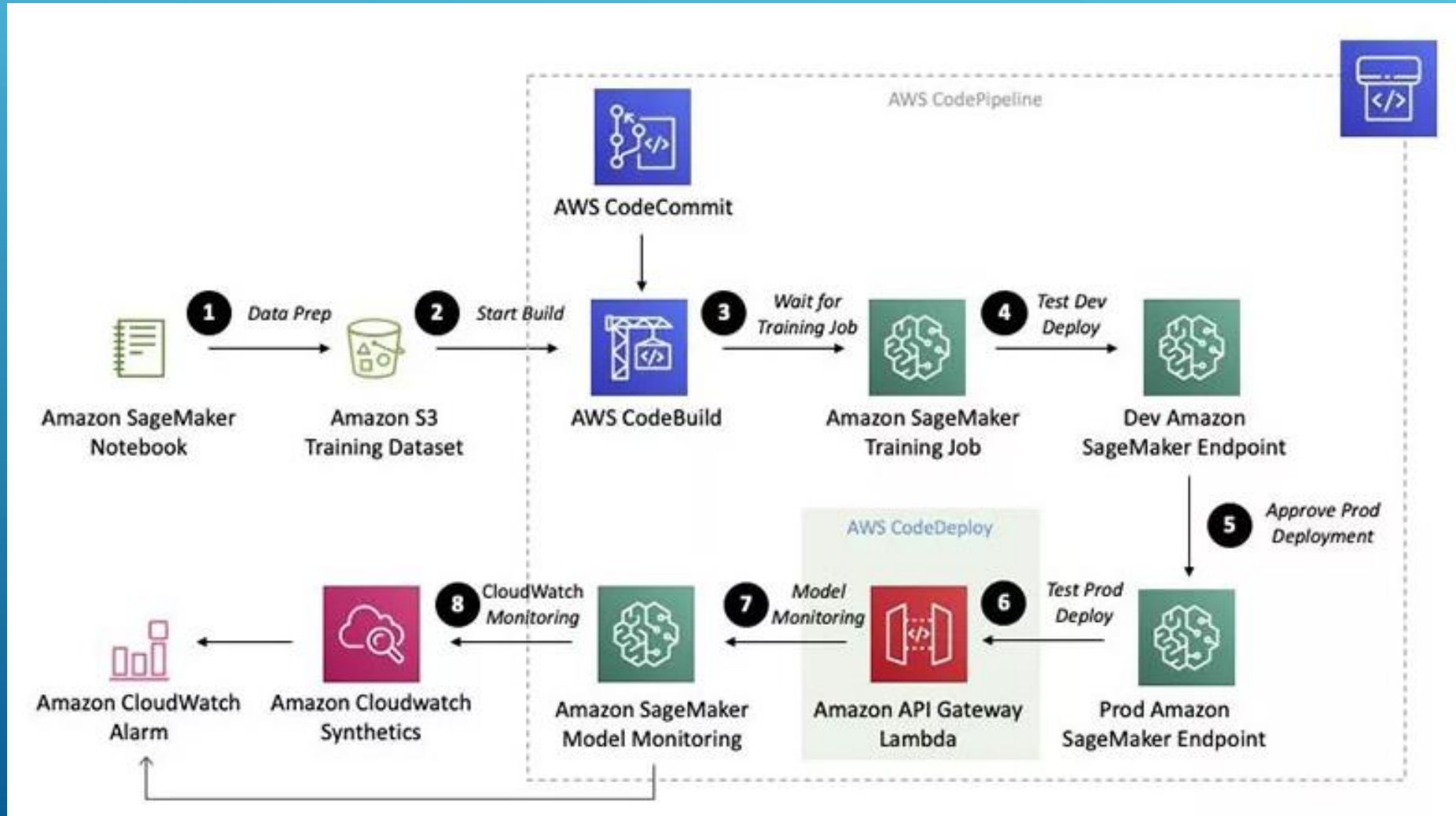
```
0
```

```
[20]: if prediction[0] == '0':  
      print('No se recomienda ofertarle el servicio')  
      else:  
      print('Se recomienda ofertarle el servicio')
```

```
No se recomienda ofertarle el servicio
```

SAGEMAKER

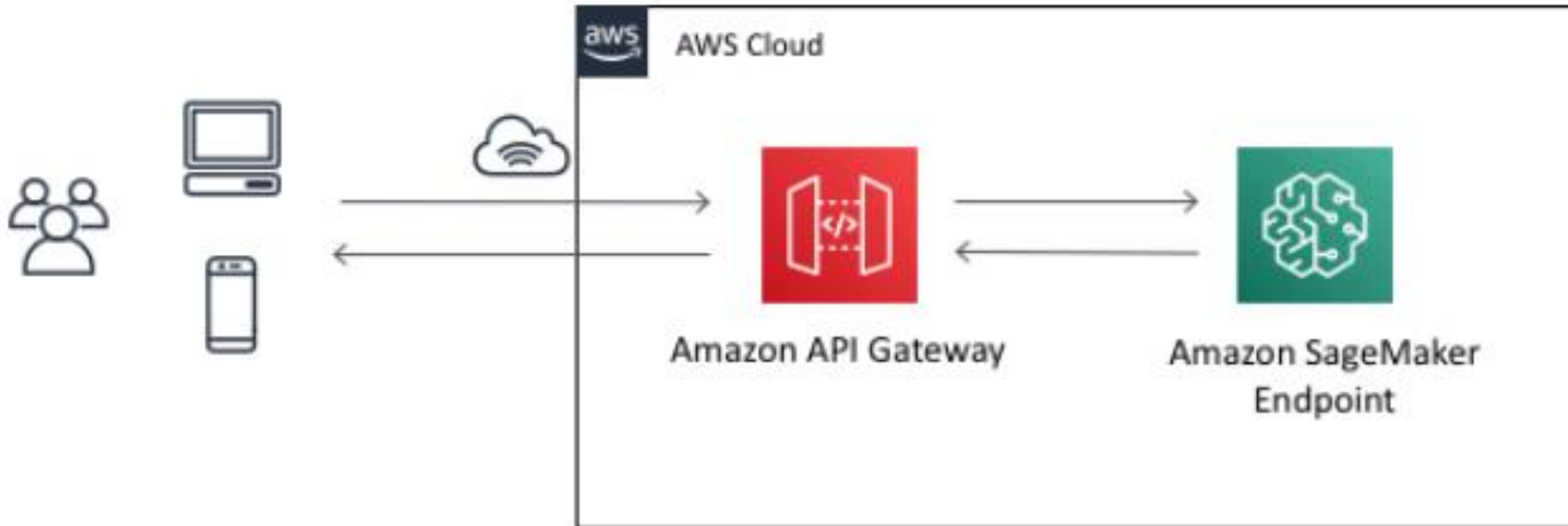
. Siguiendo el siguiente paso el monitoreo del endpoint.



SAGEMAKER



Como ultimo paso es interactuar con el endpoint y nuestras aplicaciones.



SAGEMAKER

. Documentación

Ejemplos AWS Sagemaker

<https://aws.amazon.com/es/sagemaker/getting-started/>

<https://github.com/vascencio/amaliaconf>



GRACIAS.

Victor Ascencio | 25 de Octubre 2020

[Linkedin](#)

vascencio@gmail.com

