

Analitica & Big Data

Wuilson Estacio

**INFORMACIÓN PRELIMINAR**

|  |  |
| --- | --- |
| NOMBRE DEL CLIENTE | G |
| CÓDIGO Y NOMBRE DEL PROYECTO | 750001\_MODELO\_PREDICION\_VINOS |
| NOMBRE DEL REQUERIMIENTO | Modelos MLOPS Fase 1 – Evaluación de procesos actuales y estándares en G para Modelos |
| RESPONSABLE GLOBAL G | **﻿Susana Cardona Tangarife** |
| CARGO | Analista de Selección |
| NOMBRE DEL CONTACTO DEL CLIENTE | Susana Cardo Tangarife |
| CARGO | Analista de Selección |
| CORREO ELECTRÓNICO DEL CONTACTO | [**susana.cardona@g**](mailto:susana.cardona@g)**.....com** |

NFORMACIÓN GENERAL DEL REQUERIMIENTO

|  |  |  |  |
| --- | --- | --- | --- |
| TIPO DE REQUERIMIENTO |  |  | |
| Creación |  | |
|  |  | |
|  |  | |
| TIPO DE USUARIO |  |  |  |
| Técnico |  |  |
| PRODUCTO PARA REALIZAR |  |  |  |
|  |  |  |
|  |  |  |
| Generacion Modelo Vinos  con implementación Metodología MLOps en Arquitectura Azure  **Azure AI | Machine Learning Studio** |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

[1. INTRODUCCIÓN 4](#_Toc168649605)

[2. OBJETIVOS DEL DOCUMENTO 5](#_Toc168649606)

[3. MODELO NBO Vinos 6](#_Toc168649607)

[3.1 Objetivos del modelo 6](#_Toc168649608)

[3.2 Proceso de ejecución actual 6](#_Toc168649609)

1. INTRODUCCIÓN

Se desea implementar un modelo que pueda clasificar los vinos en sus distintos tipos de uvas donde existen 3 tiipos de uvas.

1. uvas tipo A
2. uvas tipo B
3. uvas tipo C

para ello supopniendo que el cliente maneja millones de datos, se le propone al cliente realizar el tratamiento de los datos en Pyspark Y en Azure Databrick y/o Machine Learning Studio, indicandole que esto mejorara los tiempos de procesamiento y la clasificacion de los vinos. Dado el problema del cliente y la estructura de los datos se le proponen al cliente realizar el modelo con tres modelos, **Logistic Regression**, **Random Forest Classifier**, y **Gradient Boosted Trees Classifier (GBTClassifier),** esto dadas las siguientes ventajas que tienen para este tipo de problema.

**1. Logistic Regression**

**La regresión logística multiclase amplía la versión binaria utilizando estrategias como "One-vs-Rest (OvR)" o "Softmax (Multinomial)", lo que permite manejar múltiples clases.**

**Por qué es adecuada:**

* **Probabilidad interpretativa: Estima probabilidades para cada clase, lo que permite una interpretación directa de los resultados.**
* **Linealidad: Es efectiva si las clases son separables de manera aproximadamente lineal en el espacio de características.**
* **Eficiencia: Es computacionalmente eficiente para problemas con un número razonable de clases.**

**Cómo maneja múltiples clases:**

* **OvR: Entrena un clasificador para cada clase frente a las demás. Cada modelo devuelve una probabilidad, y la clase con la mayor probabilidad es la predicha.**
* **Multinomial: Calcula directamente la probabilidad para todas las clases usando la función softmax. Es más eficiente y preciso en datasets balanceados.**

**2. Random Forest Classifier**

**Random Forest es un algoritmo basado en árboles de decisión y es naturalmente adecuado para clasificaciones multiclase.**

**Por qué es adecuado:**

* **Naturaleza intrínseca multiclase: Cada árbol en el bosque genera una predicción de clase. El modelo final toma una decisión mediante votación mayoritaria.**
* **Capacidad no lineal: Random Forest puede capturar relaciones complejas y no lineales entre características.**
* **Robustez frente al ruido: Es menos propenso al sobreajuste, especialmente con datos de alta dimensión.**
* **Manejo de clases desbalanceadas: Puede ajustarse mediante pesos para equilibrar clases minoritarias o desbalanceadas.**

**3. Gradient Boosted Trees Classifier (GBTClassifier)**

**GBTClassifier es un modelo basado en árboles que utiliza boosting para mejorar iterativamente el rendimiento.**

**Por qué es adecuado:**

* **Optimización de la función objetivo multiclase: Implementa técnicas como softmax o multinomial log-loss para adaptar su función de pérdida a problemas multiclase.**
* **Predicciones más precisas: Cada árbol corrige los errores del modelo anterior, mejorando continuamente la clasificación.**
* **Flexibilidad: Como Random Forest, puede manejar relaciones no lineales y datos de alta dimensionalidad.**
* **Control de sobreajuste: Los parámetros como la tasa de aprendizaje y la profundidad de los árboles permiten un control efectivo del sobreajuste.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Comparación de los tres modelos:** |  |  |  |
| **Aspecto** | **Logistic Regression** | **Random Forest** | **GBTClassifier** |
| **Capacidad no lineal** | Limitada (modelo lineal) | Alta (captura interacciones no lineales) | Muy alta |
| **Manejo de multiclase** | Softmax u OvR | Votación mayoritaria | Boosting multinomial |
| **Robustez al ruido** | Moderada | Alta | Alta |
| **Eficiencia computacional** | Alta (simple y rápida) | Moderada | Baja (entrenamiento más largo) |
| **Interpretabilidad** | Alta (coeficientes claros) | Moderada (difícil interpretar árboles) | Baja (modelo complejo) |

Y ademas se le propone poner a entrenar los modelos en serie o en paralelo bajo el mismo costo y compararlos luego para decidir cual fue el mejor realizando la clasificacion de los vinos de acuerdo a las caracteristicas de las uvas.

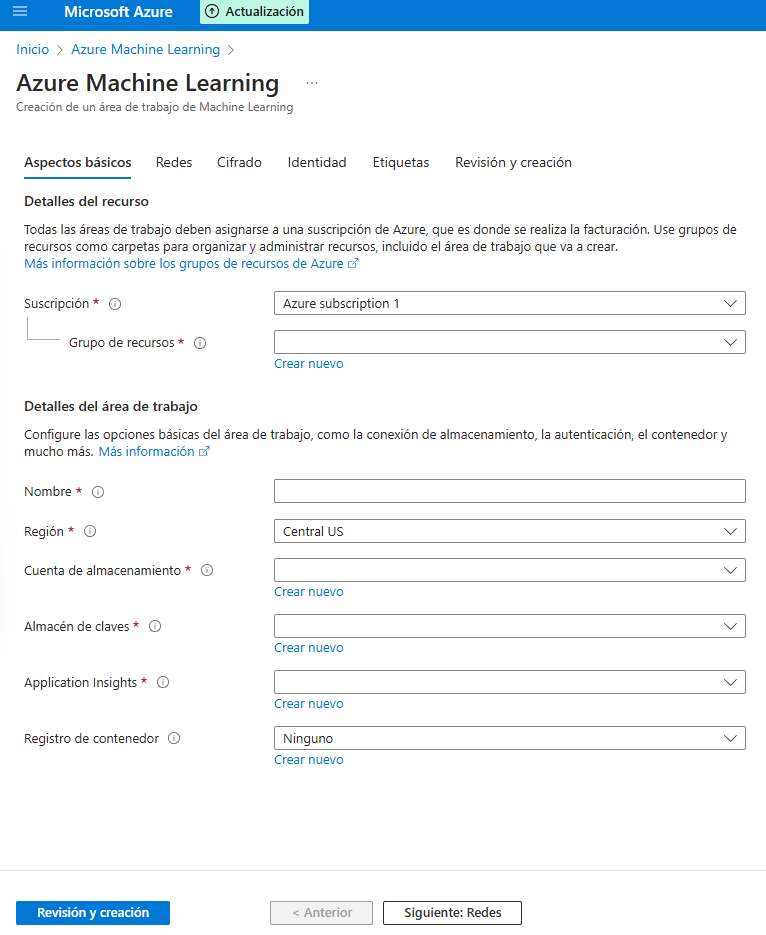
1. OBJETIVOS DEL DOCUMENTO

* Implementacion de modelos en MLflow y Mlops
* Identificar mejoras en MLops y Modelo de Vinos
* Explicacion de la generacion de modelo de Vinos

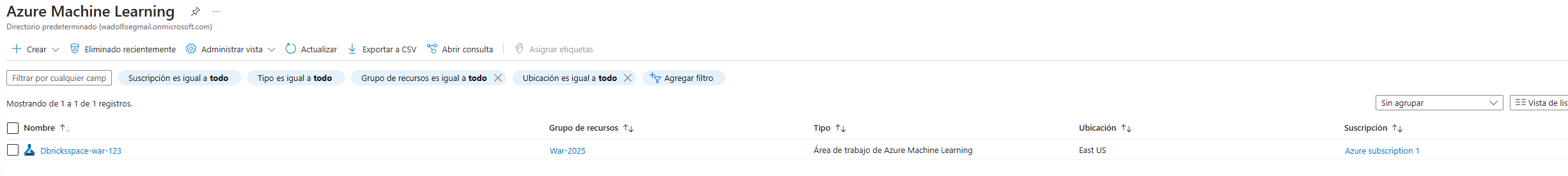
1. MODELO de vinos

inicio de modelo se debe crear un ambien de te Databrick o Azure Machine learning

dado a que la cuenta desponible en azure para Databricks presento problemas en el encendido de los cluster por ser gratuita se debio generar otra en Azure Machine Learning como muestra la imagen 1

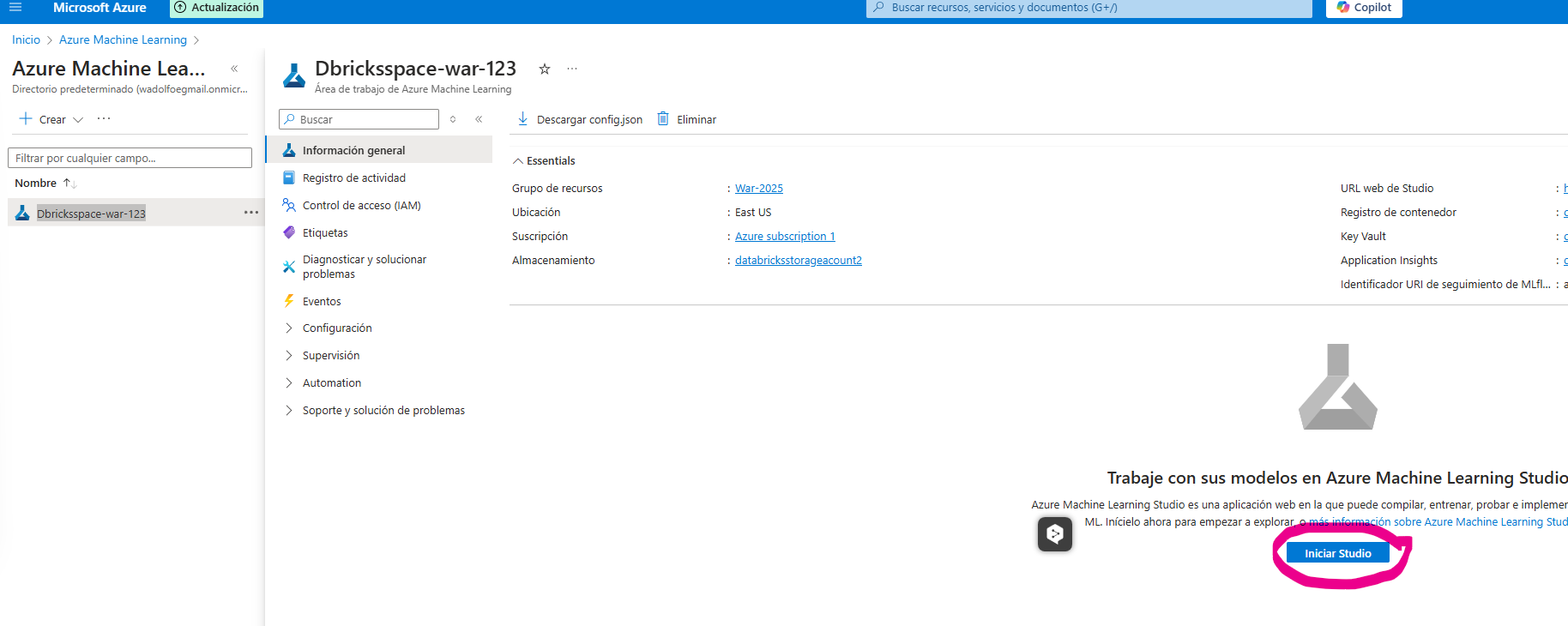
Imagen 1  


luego de tener su cuenta configurada en Azure Machine Learnign debe aparecer algo asi como muestra la imagen 2:

Imagen 2  


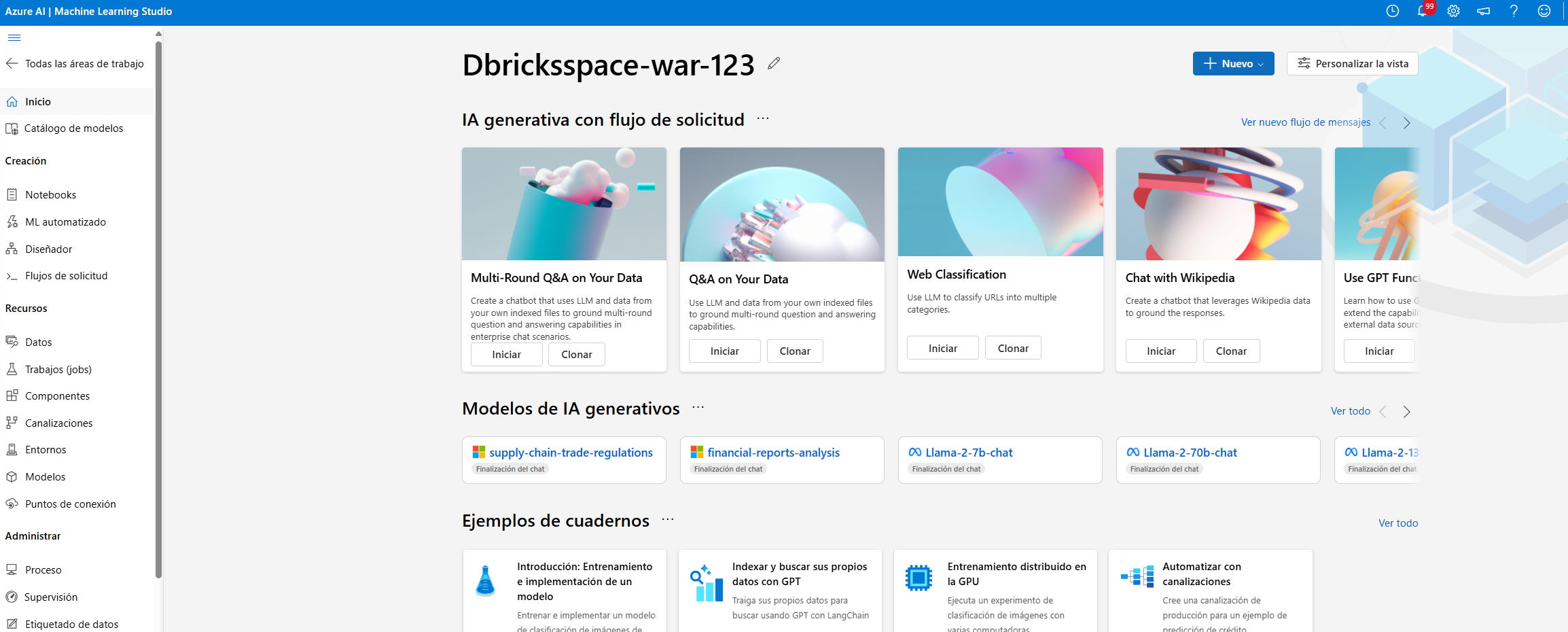
Luego al darle clip sobre [Dbricksspace-war-123](https://portal.azure.com/#@wadolfoegmail.onmicrosoft.com/resource/subscriptions/e2fce082-d0f5-42e6-892d-078f098b0d47/resourceGroups/War-2025/providers/Microsoft.MachineLearningServices/workspaces/Dbricksspace-war-123) te debra aparecer algo asi cmo muestra la imagen3, y alli deberar dar click en Iniciar estudio.

Imagen 3

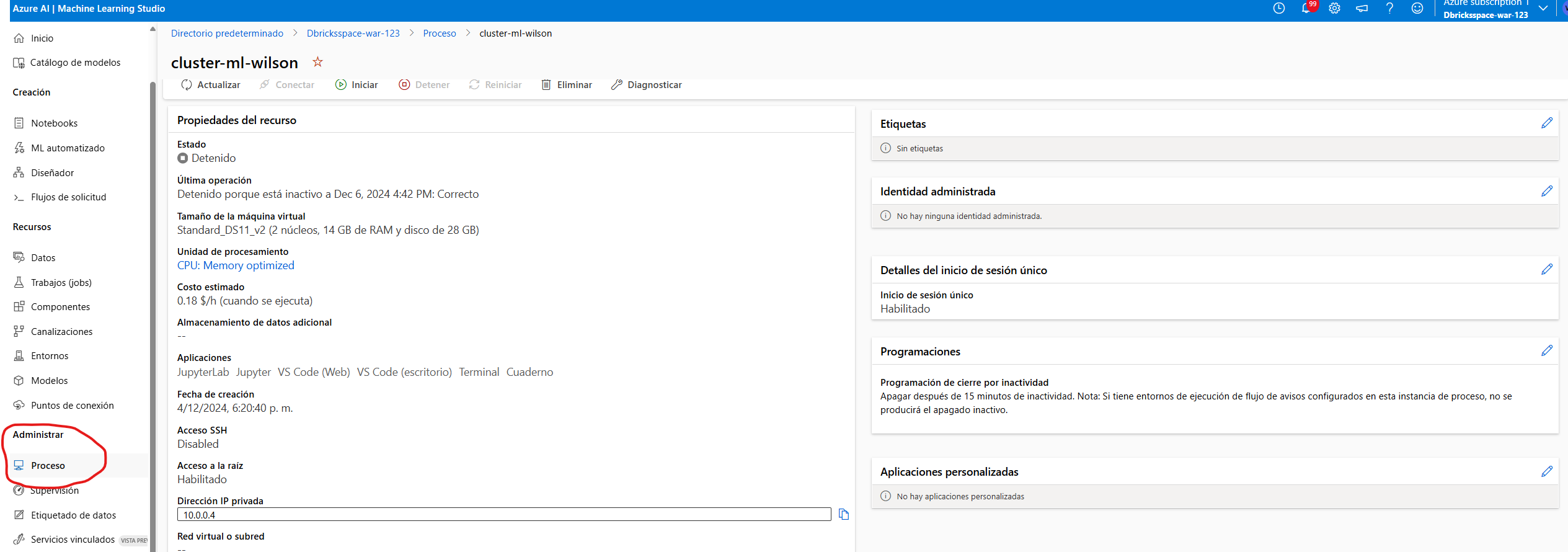


Alli te aparecera como muestra la imagen 4, esto porque fue una creacion estandar, la cual consume menos recuros:

Imagen 4



Lo primero que debemos hacer al estar alli es crear un cluster., eso se crea en Administrar y procesos asi como muestra la imagen 5, y alli luego debes configurar el cluster de acuerdo a tus necesidades. En eest caso se escogio uno bastante basico dado los requerimientos y cuyo consumo es de 0.18 dolares la hora.

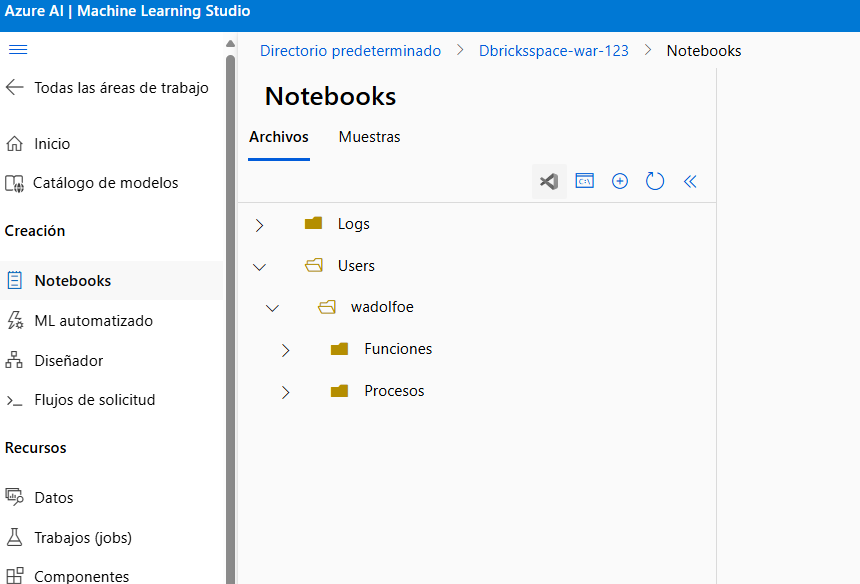
Imagen5  


Luego de ello dado que aquí no existe como tal un catalos de datos como en databricks, puedes subir la data en la parte de recuros y Dattos como aparece en la imagen 5 o anterior

Alli se te ofreceran varias opciones para cargar tus datos como desde un Storage de Azure, archivos locales o Web. Ya queda a tu preferencia.

Luego de ello deberas ir a la parte de Notebooks y crear la organzacion de tus carpetas un ejemplo es el siguiente de la imagen 7:

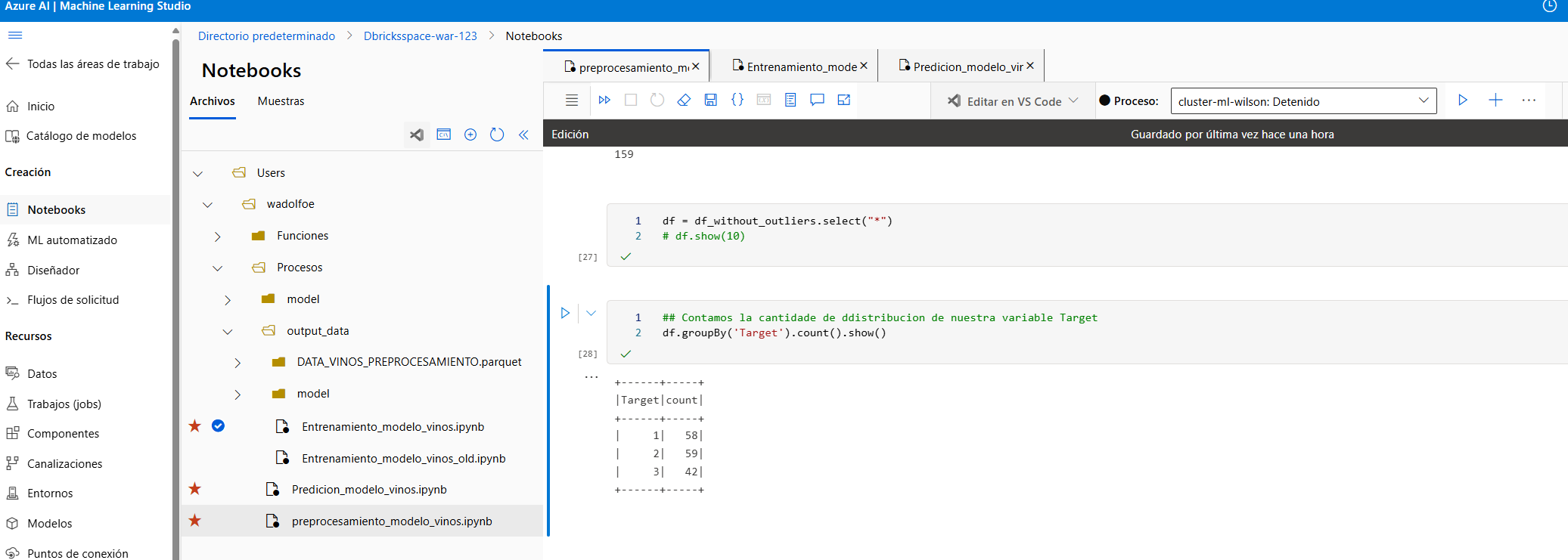
Imagen7



En la imagen 7 podemos la distribucion de las carpetas para el modelo, en la carpeta de funciones se crearan las funciones que utilizara el modelo,

En la carpeta de procesos se crearan dos carpetas mas una para algunos archivos que se generen del procesos y otra donde estaran los modelo. Asi como muestra la imagen 8

Imagen8



Alli se crearon 3 scrip en python y con lenguaje pyspark

* 1. preprocesamiento\_modelo\_vinos.ipynb.
  2. Entrenamiento\_modelo\_vinos.ipynb.
  3. Predicion\_modelo\_vinos.ipynb

La parte 1 de preprocesamiento\_modelo\_vinos en ella va toda la limpieza y entendimiento de la data, eliminacion de outliers, escalamiento de datos, vectorizacion y demas correspondiente al tratamiento de datos.

La parte 2 Entrenamiento\_modelo\_vinos, en ella se debe ya tener la data lista y liimpia con las columnas ya selecionadas desde el scrip anterior, para poder iniciar el entrenamiento de los modelos. Y demas seleciones respectivas a la parte de entrenamiento de modelos.

La parte 3 Predicion\_modelo\_vinos, en ella se realiza la prediccion de las nuevas variables teneindo encuenta el mejor modelo generado en la seccion de Entrenamiento.

* 1. preprocesamiento\_modelo\_vinos.ipynb.

este escrip tiene el insumo inicial quue es el datased de Vinos.csv

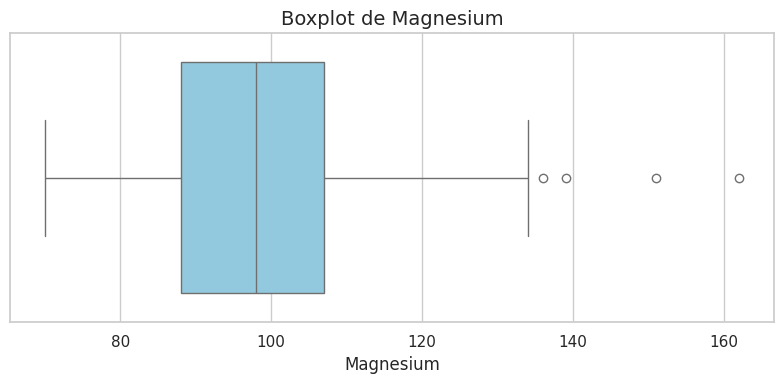
este contenia las siguiente para la descripcion del tipo de vino

|  |  |  |
| --- | --- | --- |
|  | Tabla de Vinos | |
|  | Campo | Descripcion |
| 1 | Alcohol | Contenido alcohólico del vino |
| 2 | Malic acid | Acidez málica del vino |
| 3 | Ash | Cenizas totales del vino |
| 4 | Alcalinity of ash | Alcalinidad de las cenizas |
| 5 | Magnesium | Concentración de magnesio |
| 6 | Total phenols | Fenoles totales presentes |
| 7 | Flavanoids | Compuestos flavonoides antioxidantes |
| 8 | Nonflavanoid phenols | Fenoles no flavonoides presentes |
| 9 | Proanthocyanins | Pigmentos precursores del color |
| 10 | Color intensity | Intensidad del color del vino |
| 11 | Hue | Tono del color |
| 12 | OD280/OD315 of diluted wines | Relación de absorción UV |
| 13 | Proline | Aminoácido relacionado con dulzura |

En ella se opservo como viene el datased original, se hace limpiesa y selecion de variables de importancia, se veriifica el tipo de datos que contiene, se detectan anomalias, se transforman los datos al tipo que se desea, se identifican los datos faltantes por columnas y se realiza su respectivo tratamieto, se identifican valores a tipicos y se corriguen

por ejemplo la columna Magnesion tienie outliers ver imagen 9

imagen 9



De de esta columna se corriguen estos outlier, y asi con cada una de las variables, luego de ello se escalan las variables de las columnas desea das y en seguida se puede o no realizar un MCA o PCA de acuedo a como valla dando el modelo, esto se realizaria para mejorar las prediciones del modelo, en esta caso no fue necesario hacer MCA o PCA.

Luego de ello es necesario hacer una matris de correlacion con correlacion de pearson, para ver que variables eestan mas correlacionadas con nuestra variable objetivo o Target, en este caso se eliminaron las columnas que aportaban menos del 1% al entendimiento del modelo y tambien se realiza una distribucion de cada una de las variables para mirar su comportamiento y deacuerdo a ello hay si desidir si se elimina o no, en este caso fue necesario eliminar la vairiable Ash. Ver imágenes 10 y 11

Imagen 10

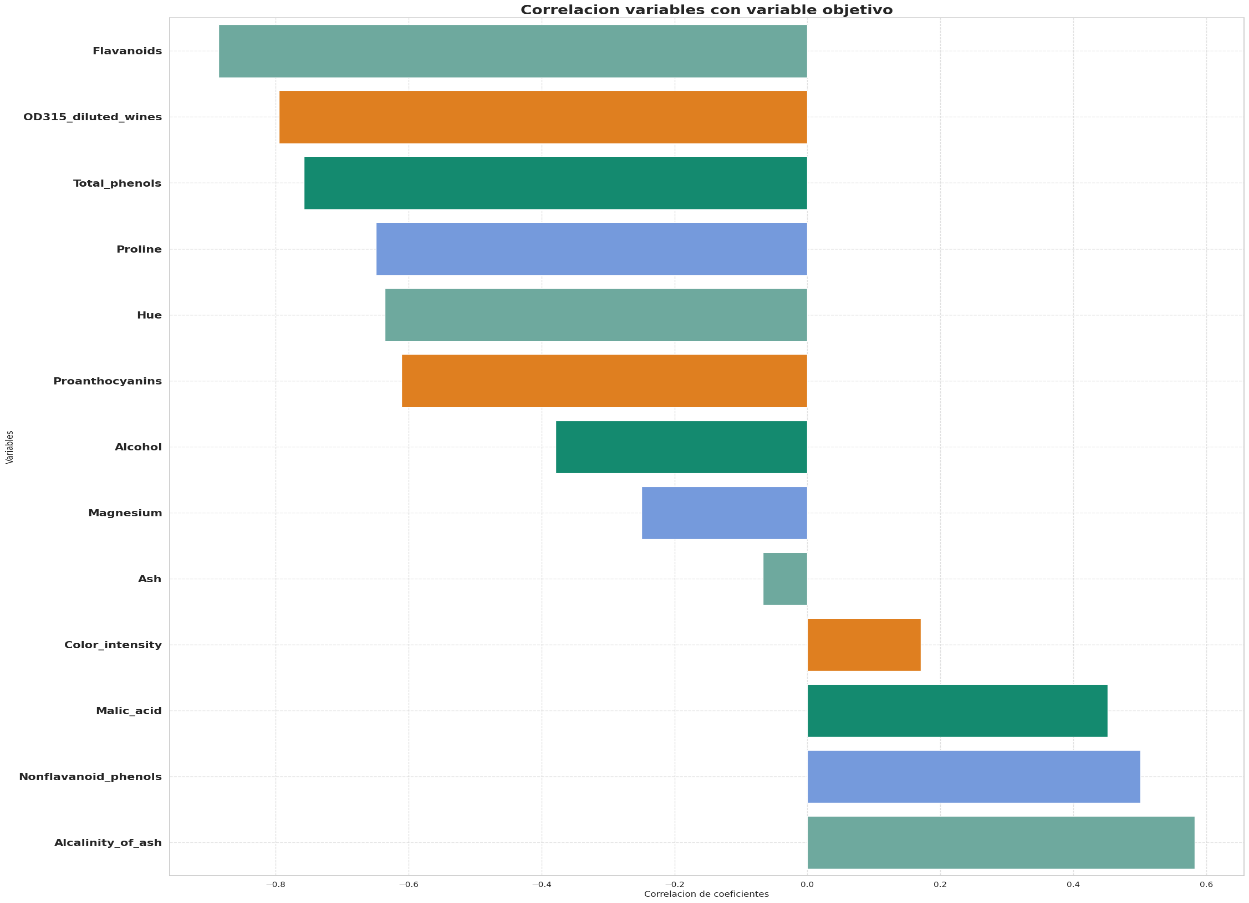
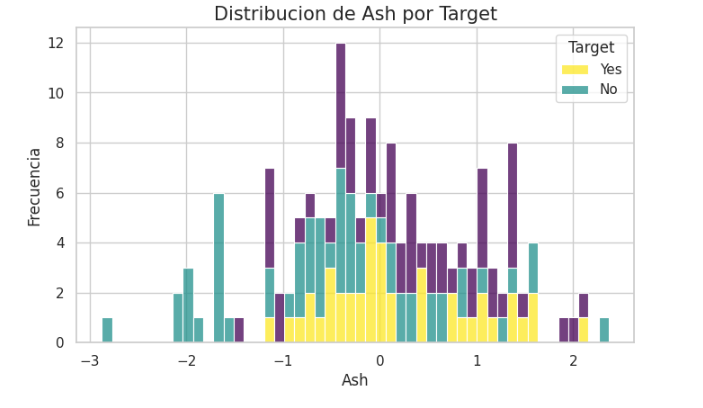


Imagen 11



Luuego como se esta trabajando en pyspark y se requiere hacer el moddelo en pyspark vectorizamos con from pyspark.ml.feature import VectorAssembler

Porque en pyspark se reciben vactores y no columnas.

Y luuego como parte final se guarda estte escript como

DATA\_VINOS\_PREPROCESAMIENTO.parquet, se guarda comom parque porque su guardado para grandes volumens de dattos es mucho mas rapido y eficiente. A demas de que ocupan menos espacio en memoria.



* 1. Entrenamiento\_modelo\_vinos.ipynb

En este scrip se carga la data ya organizada osea la data DATA\_VINOS\_PREPROCESAMIENTO.parquet, con la cual se va a trabajar los modelos.

En este escrip primero se instalasn las librerias necesarias, luego se hace un llamado a las librerias que se van a utilizar y sus fuunciones.

Aquí en esta parte se cargan 3 modelos para hacer la comparacion de cual fuue el mejor los modelos fueron LogisticRegression, RandomForestClassifier, GBTClassifier.

Ya anteriormente explicamos porque se decidio escoger estos modelos. Estos modelos se trabajaron con Mlflow de Azure est porque tiene varias ventajas debido a la integración nativa con los servicios de Azure y la infraestructura empresarial que ofrece. Un de ellas es:

Azure Machine Learning ofrece una integración nativa con MLflow, lo que significa que puedes aprovechar todas las capacidades de MLflow dentro del ecosistema de Azure:

* **Registro automático de experimentos y modelos:** Los modelos entrenados y las métricas generadas se registran automáticamente en MLflow.
* **Pipeline ML de extremo a extremo:** Desde la experimentación hasta el despliegue, todo puede gestionarse dentro de Azure Machine Learning con soporte de MLflow.
* **Seguimiento de experimentos:** Puedes rastrear métricas, hiperparámetros y artefactos (como modelos) de forma centralizada.

De esta forma se Registran las metricas

# Registrar métricas por clase

        mlflow.log\_metric(f"{model\_name}\_class\_{class\_label}\_precision", precision)

        mlflow.log\_metric(f"{model\_name}\_class\_{class\_label}\_recall", recall)

        mlflow.log\_metric(f"{model\_name}\_class\_{class\_label}\_f1", f1\_class)

    # Guardar el modelo en MLflow

    artifact\_path = "model\_logistic\_regression"

    mlflow.spark.log\_model(bestModel, artifact\_path=artifact\_path)

y de esta forma se registra el modelo  
  
    # Registrar el modelo en el Model Registry

    run\_id = run.info.run\_id

    model\_uri = f"runs:/{run\_id}/{artifact\_path}"

    model\_registry\_name = "LogisticRegression\_Model"

    registered\_model = mlflow.register\_model(model\_uri=model\_uri, name=model\_registry\_name)

luego de ello se realiza una validacion de que el modelo fue guardado correctamente, esto se puede hhacer con:

from mlflow.tracking import MlflowClient

client = MlflowClient()

# Listar versiones del modelo

model\_versions = client.search\_model\_versions(f"name='GBT\_Model\_OvR'")

for version in model\_versions:

print(f"Version: {version.version}, Run ID: {version.run\_id}, Status: {version.current\_stage}")

luego se hace una comparacion por las metricas de cual fue el mejor modelo como muestra la imagen 12 y 13, donde se opserva que el mejor modelo fue el modelo RandomForest\_Model V8

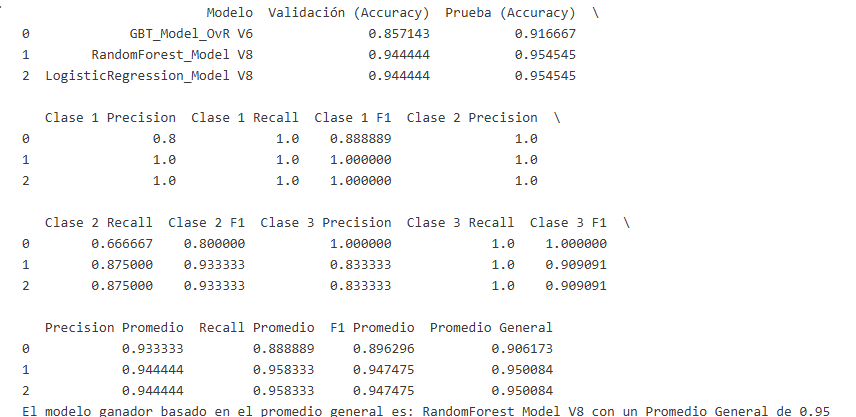
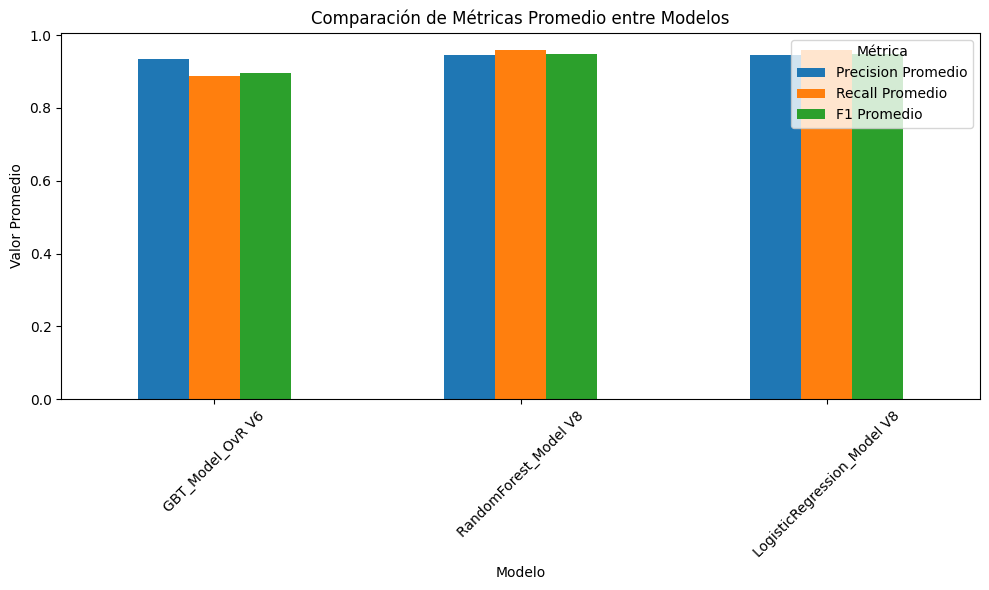
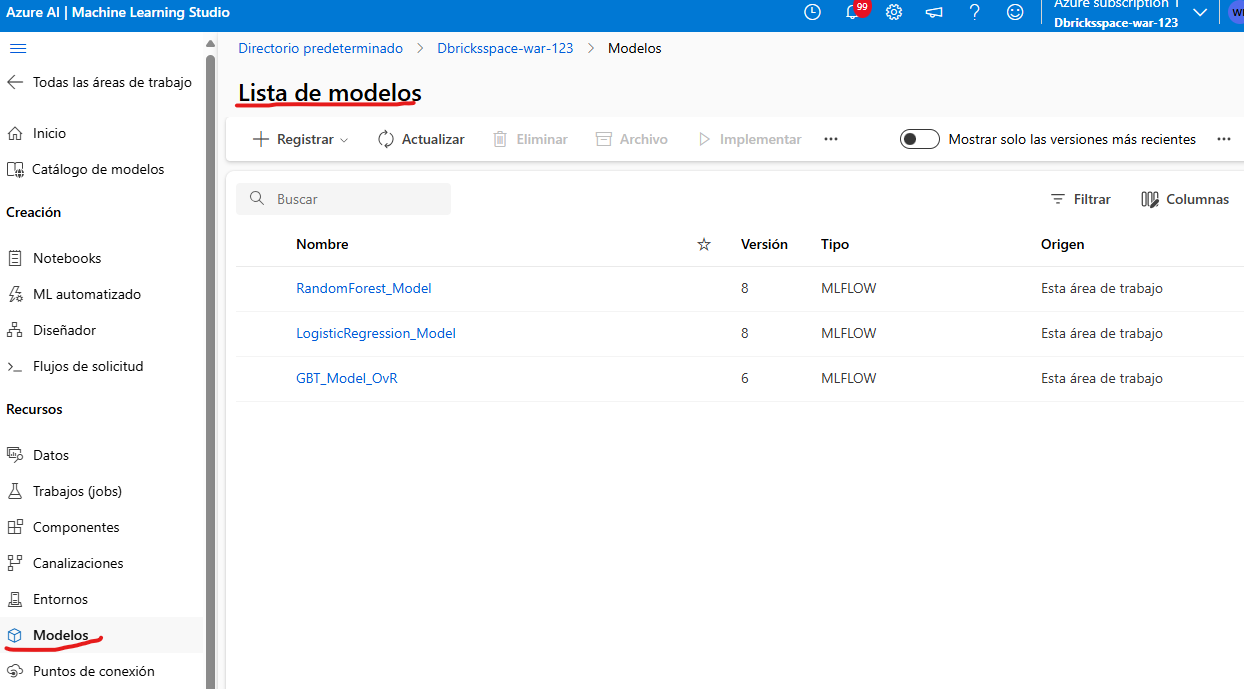
imagen 12

imagen13



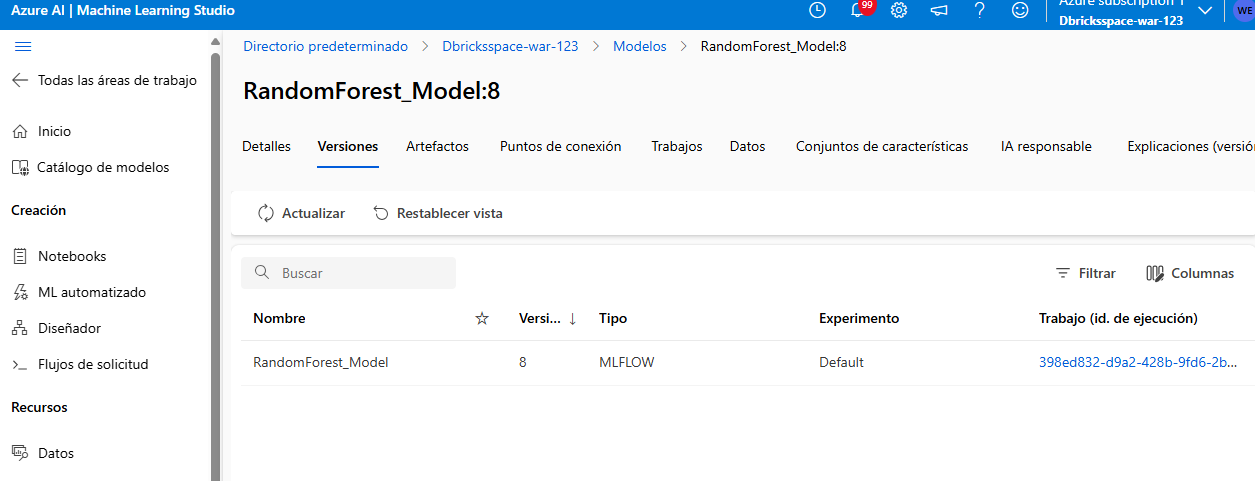
Los modelos quuedan registrados en la parte de Modelos, asi como muestra la imagen 14

Imagen 14



Al ingresar a cada uno de ellos, por ejemplo en **RandomForest\_Model:8, podremos ver los detalles, Versiones, artefactos, puntos de conexión, metricas y demas. Ver imagen 15**

**Imagen 15**



Si vamos a las parte de Trrabajo, alli podremos encontrar las metricas, imagens, fechas de creacion, quien lo modifico, cuando y demas variables, ver imagen 15

Imagen 16

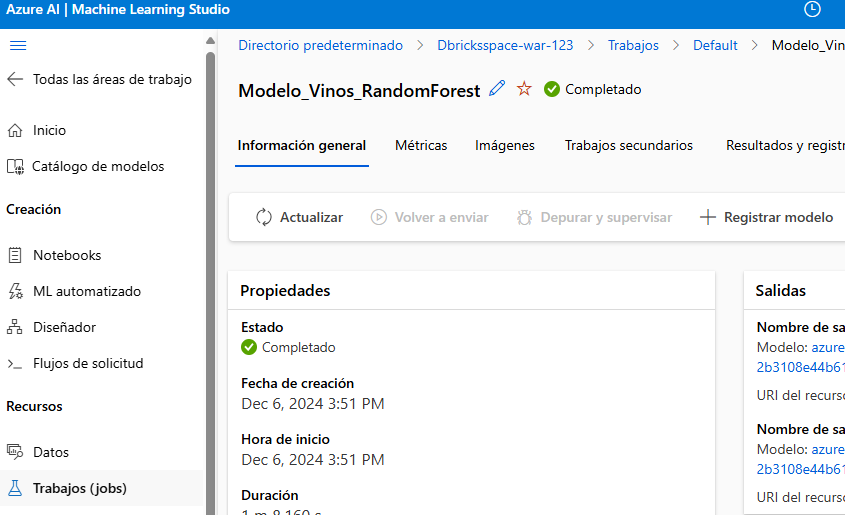


Imagen 17

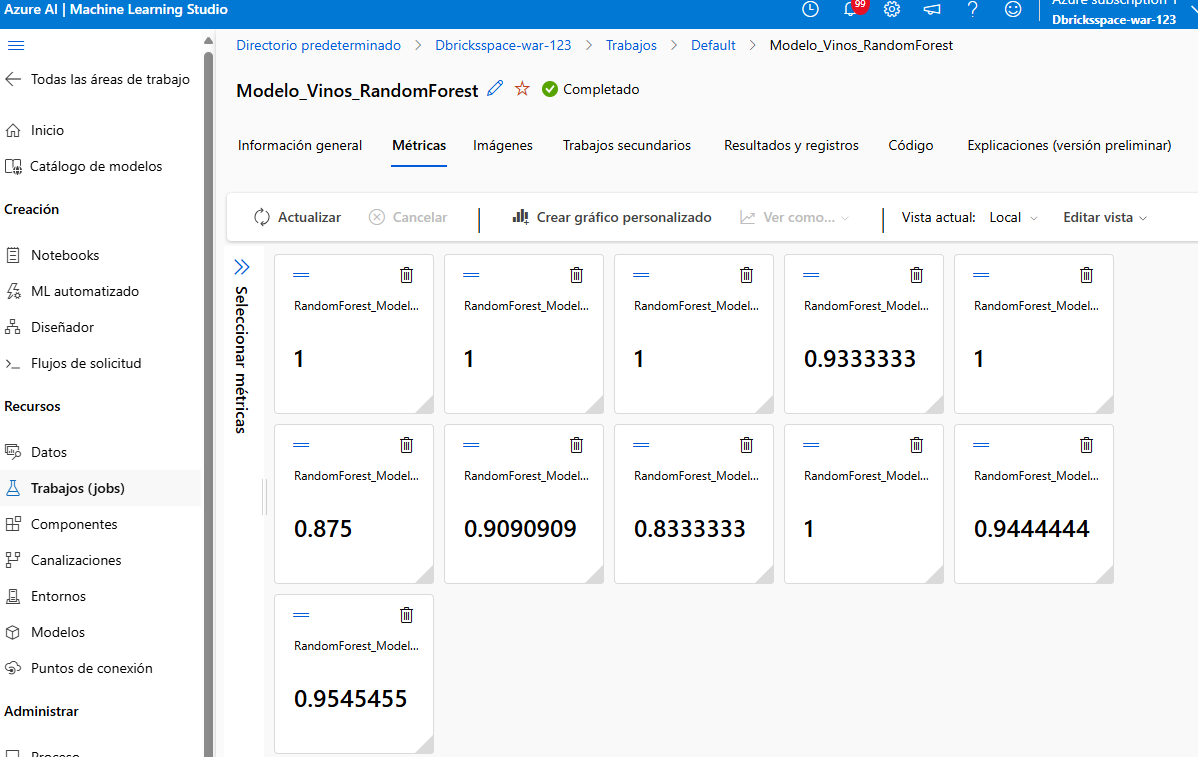
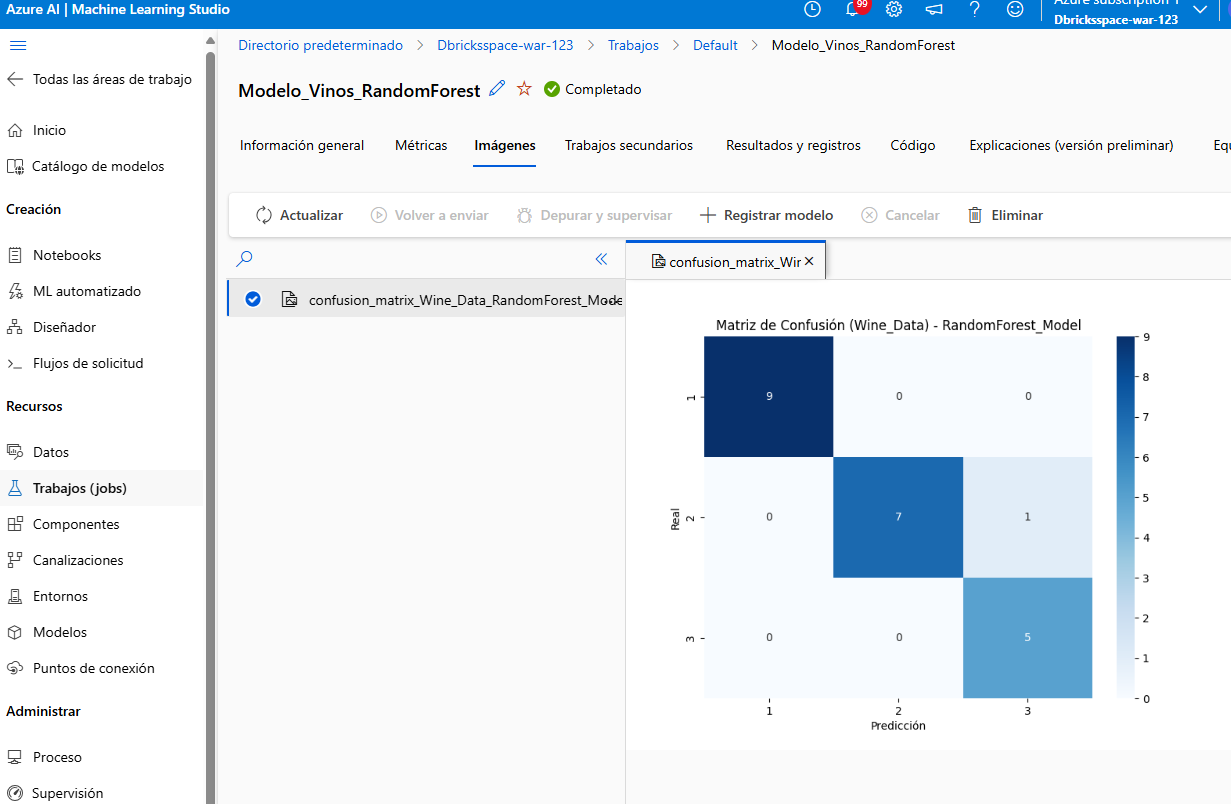


Imagen 18



Como podemos opservar en la imagen 17 podemos ver las metricas registradas para ese modelo y en la imagen 18 podemos ver como dio la matriz de confucion. Lla cual nos indica que tanto estamos hacertando en los Verdaderos positivos.



* 1. Predicion\_modelo\_vinos.ipynb

En esta parte se cargan o se carga el mejor modelo, se instalan y cargan las librerias correspondientes

El modelo se carga de la siguiente forma:

from azureml.core import Workspace, Model

# Conectar al workspace

workspace\_name = "Dbricksspace-war-123"  # Cambia al nombre de tu workspace

subscription\_id = "e2fce082-d0f5-42e6-892d-078f098b0d47"

resource\_group = "War-2025"

ws = Workspace.get(

    name=workspace\_name,

    subscription\_id=subscription\_id,

    resource\_group=resource\_group,

)

# Cargar el modelo desde el registro

model\_name = "RandomForest\_Model"

model\_version = 8

model = Model(ws, name=model\_name, version=model\_version)

# Descargar los artefactos del modelo

model.download(target\_dir="./model", exist\_ok=True)

print(f"Modelo {model\_name} descargado a './model'")

tambien se llaman los nuevos datos y se realizan las predicciones de la siguiente forma

from joblib import load

predicciones = spark\_model.transform(df\_escalado)

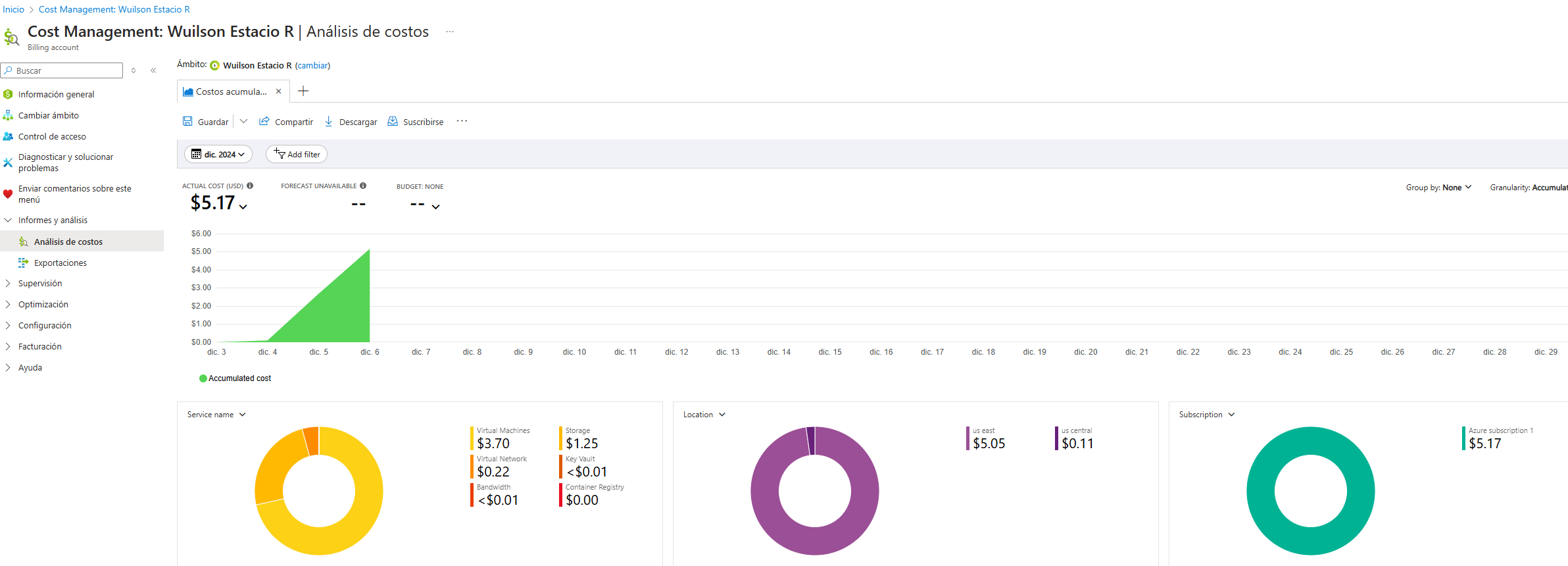
# Mostrar las predicciones

predicciones.show()

y se pueden realizar los ajustes necesarios.



Como parte final tenemos un resumen de costos totales de uso, desde el inicio de su configuracion y usos de cluster, el cual muestra que hasta la fecha el costo de generar ese modelo fueron de 5.17 dolares, tal cual como muestra la imagen.



1. CONCLUSIONES

Como conclusión, dado el contexto actual y los desafíos que representa para todas las industrias la implementación de infraestructuras y procesos para el aprovechamiento de oportunidades de negocio y toma de decisiones más acertadas basadas en datos, la implementación de procesos de Machine Learning Operations (MLOps) representa una oportunidad estratégica para G. La integración de modelos y procesos de datos dentro de un flujo lógico y automatizado en el marco de MLOps ofrece una gran oportunidad para mejorar la eficiencia, seguridad y efectividad de los modelos en su aplicación a los casos de negocio de cada cliente.

En este documento se llevo a cabo una descripción de los procesos que se debieron llevar a cabo para cumplir con la generación de un modelo MLFlow

1. ANEXOS

|  |  |
| --- | --- |
| **NOMBRE** | **ANEXO** |
|  |  |