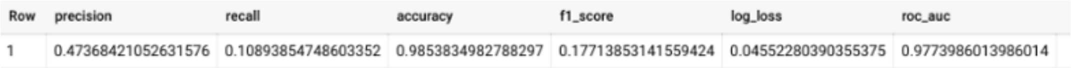
You should see a table similar to this:



Row	precision	recall	accuracy	f1_score	log_loss	roc_auc
1	0.47368421052631576	0.10893854748603352	0.9853834982788297	0.17713853141559424	0.04552280390355375	0.9773986013986014

Test completed task

Click **Check my progress** to verify your performed task. If you have completed the task successfully you will be granted an assessment score.

✓

Evaluate the Model

Check my progress

Assessment Completed!

Task 5. Test your understanding

Below are multiple choice questions to reinforce your understanding of this lab's concepts. Answer them to the best of your abilities.

✓

BigQuery is a fully-managed enterprise data warehouse that enables super-fast SQL queries.

✓ True

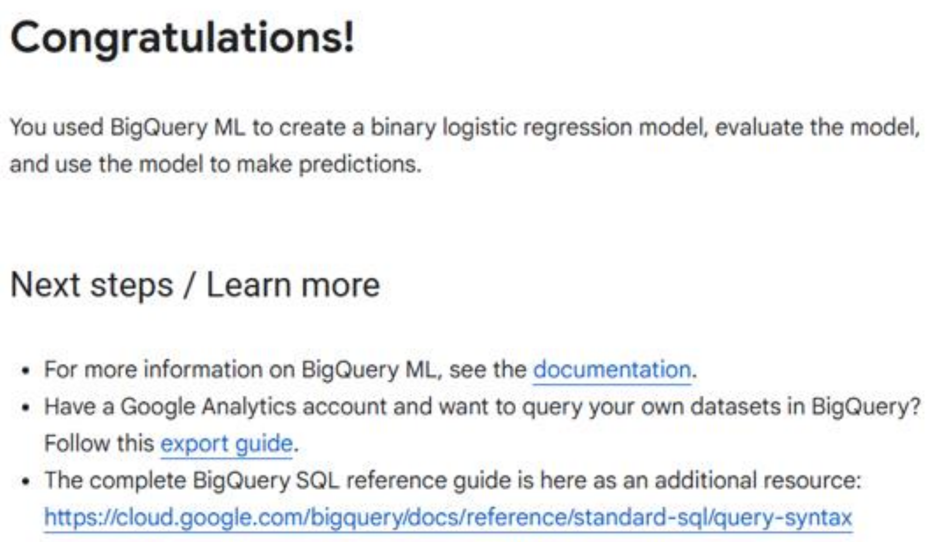
○ False

Which option best describes what BigQuery ML does?

○ Creates machine learning models using Python or Java in BigQuery, then executes the model using standard SQL queries.

✓ Creates and executes machine learning models in BigQuery using standard SQL queries.

✓

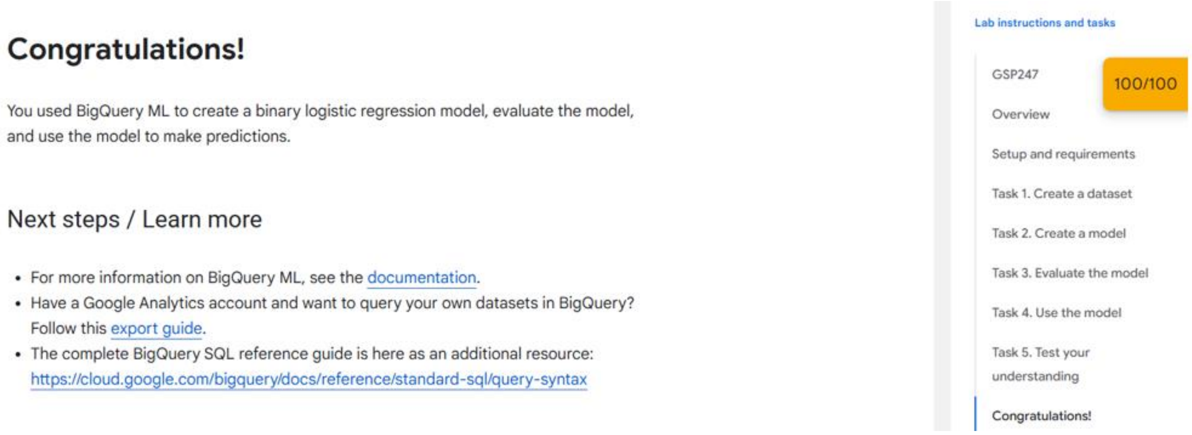


Congratulations!

You used BigQuery ML to create a binary logistic regression model, evaluate the model, and use the model to make predictions.

Next steps / Learn more

- For more information on BigQuery ML, see the documentation.
- Have a Google Analytics account and want to query your own datasets in BigQuery? Follow this export guide.
- The complete BigQuery SQL reference guide is here as an additional resource: https://cloud.google.com/bigquery/docs/reference/standard-sql/query-syntax

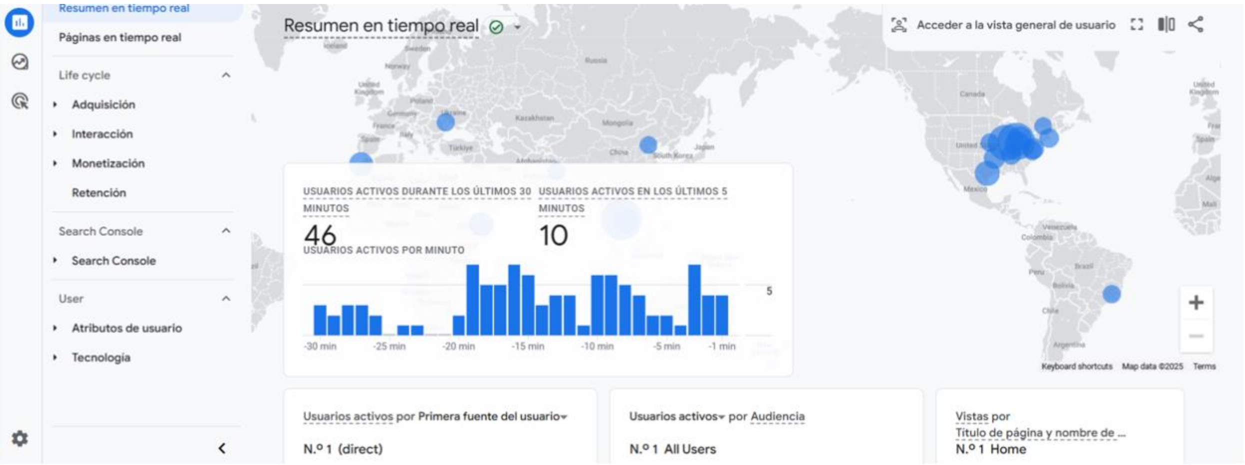


Congratulations!

You used BigQuery ML to create a binary logistic regression model, evaluate the model, and use the model to make predictions.

Next steps / Learn more

- For more information on BigQuery ML, see the documentation.
- Have a Google Analytics account and want to query your own datasets in BigQuery? Follow this export guide.
- The complete BigQuery SQL reference guide is here as an additional resource: https://cloud.google.com/bigquery/docs/reference/standard-sql/query-syntax



Resumen en tiempo real

USUARIOS ACTIVOS DURANTE LOS ÚLTIMOS 30 MINUTOS
46

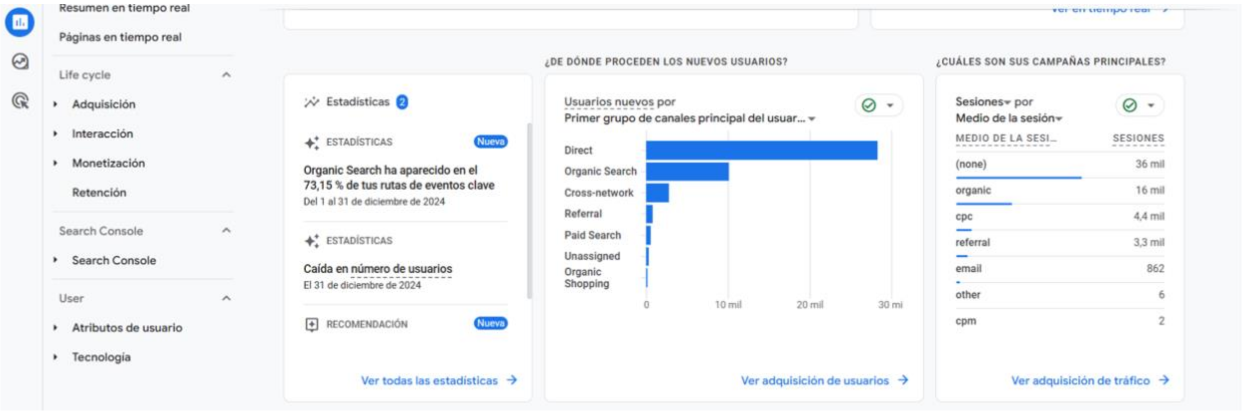
USUARIOS ACTIVOS EN LOS ÚLTIMOS 5 MINUTOS
10

USUARIOS ACTIVOS POR MINUTO

Usuarios activos por Primera fuente del usuario
N.º 1 (direct)

Usuarios activos por Audiencia
N.º 1 All Users

Vistas por Título de página y nombre de ...
N.º 1 Home



Estadísticas 2

ESTADÍSTICAS Nueva

Organic Search ha aparecido en el 73,15 % de tus rutas de eventos clave
Del 1 al 31 de diciembre de 2024

ESTADÍSTICAS

Caída en número de usuarios
El 31 de diciembre de 2024

RECOMENDACIÓN Nueva

Ver todas las estadísticas →

¿DE DÓNDE PROCEDEN LOS NUEVOS USUARIOS?

Usuarios nuevos por Primer grupo de canales principal del usuar...

Direct
Organic Search
Cross-network
Referral
Paid Search
Unassigned
Organic Shopping

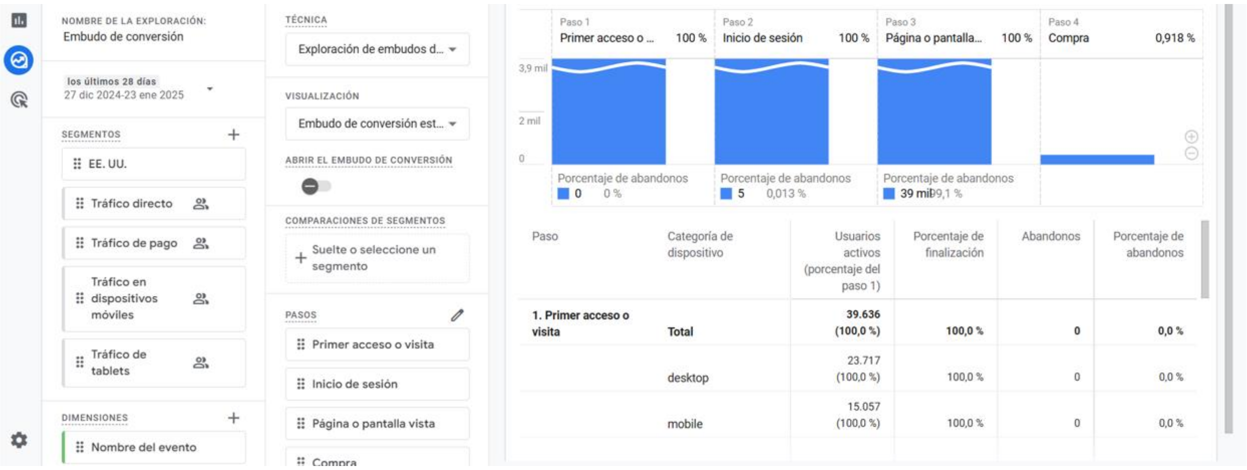
Ver adquisición de usuarios →

¿CUÁLES SON SUS CAMPAÑAS PRINCIPALES?

Sesiones por Medio de la sesión

MEDIO DE LA SESI...	SESIONES
(none)	36 mil
organic	16 mil
cpc	4,4 mil
referral	3,3 mil
email	862
other	6
cpm	2

Ver adquisición de tráfico →



NOMBRE DE LA EXPLORACIÓN:
Embudo de conversión

los últimos 28 días
27 dic 2024-23 ene 2025

SEGMENTOS
- EE. UU.
- Tráfico directo
- Tráfico de pago
- Tráfico en dispositivos móviles
- Tráfico de tablets

DIMENSIONES
- Nombre del evento

TÉCNICA
Exploración de embudos d...

VISUALIZACIÓN
Embudo de conversión est...

ABRIR EL EMBUDO DE CONVERSIÓN

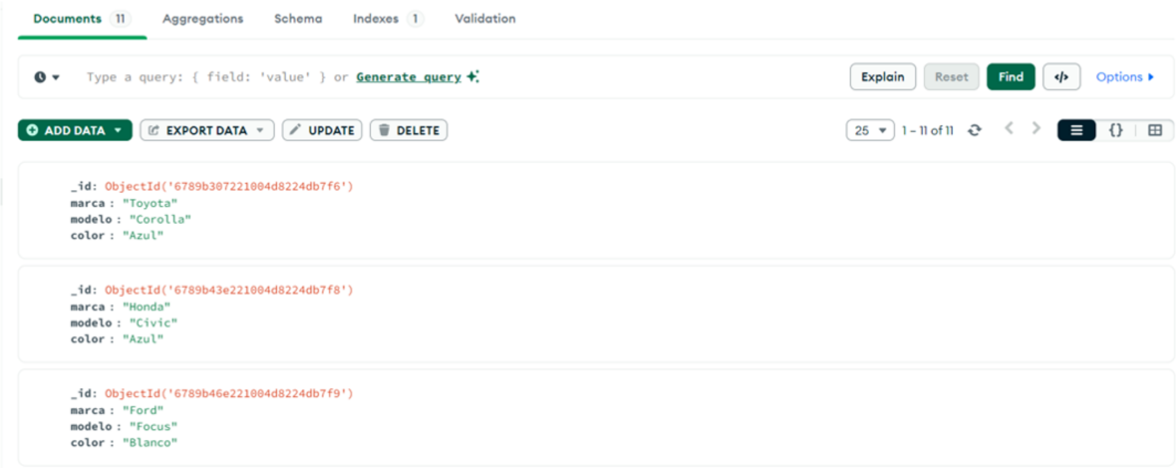
COMPARACIONES DE SEGMENTOS
+ Suelte o seleccione un segmento

PASOS
- Primer acceso o visita
- Inicio de sesión
- Página o pantalla vista
- Compra

Paso 1	Paso 2	Paso 3	Paso 4
Primer acceso o ... 100 %	Inicio de sesión 100 %	Página o pantalla... 100 %	Compra 0,918 %
Porcentaje de abandonos 0 0 %	Porcentaje de abandonos 5 0,013 %	Porcentaje de abandonos 39 mil99,1 %	

Paso	Categoría de dispositivo	Usuarios activos (porcentaje del paso 1)	Porcentaje de finalización	Abandonos	Porcentaje de abandonos
1. Primer acceso o visita	**Total**	**39.636 (100,0 %)**	**100,0 %**	**0**	**0,0 %**
	desktop	23.717 (100,0 %)	100,0 %	0	0,0 %
	mobile	15.057 (100,0 %)	100,0 %	0	0,0 %

1. **Creación BD automóviles**
2. **Creación colección Vehículos**
3. **Insertar data en colección Vehículos**



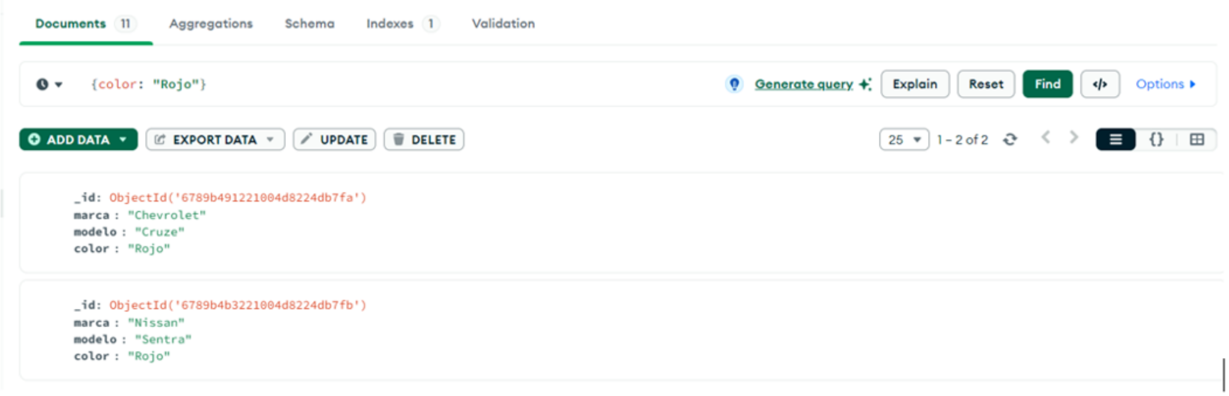
```
_id: ObjectId('6789b307221004d8224db7f6')
marca : "Toyota"
modelo : "Corolla"
color : "Azul"
```

```
_id: ObjectId('6789b43e221004d8224db7f8')
marca : "Honda"
modelo : "Civic"
color : "Azul"
```

```
_id: ObjectId('6789b46e221004d8224db7f9')
marca : "Ford"
modelo : "Focus"
color : "Blanco"
```

4. **Realizar Filtros**

Filtrar vehículos color Rojo

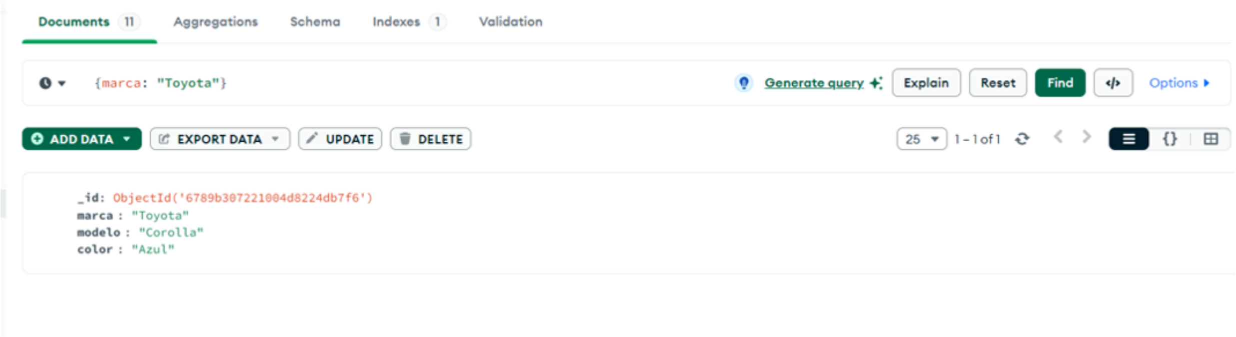


```
{color: "Rojo"}
```

```
_id: ObjectId('6789b491221004d8224db7fa')
marca : "Chevrolet"
modelo : "Cruze"
color : "Rojo"
```

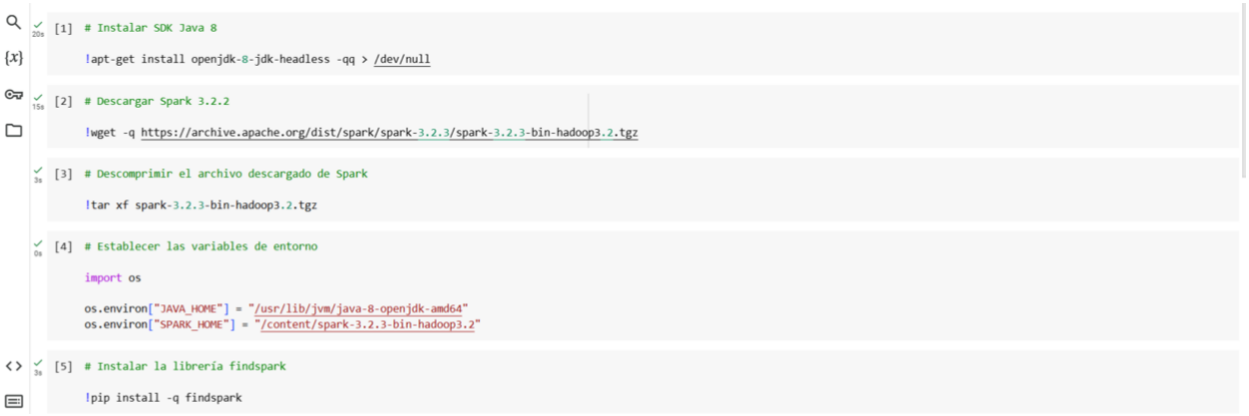
```
_id: ObjectId('6789b4b3221004d8224db7fb')
marca : "Nissan"
modelo : "Sentra"
color : "Rojo"
```

Filtrar vehículos marca Toyota



```
{marca: "Toyota"}
```

```
_id: ObjectId('6789b307221004d8224db7f6')
marca : "Toyota"
modelo : "Corolla"
color : "Azul"
```



```
[1] # Instalar SDK Java 8

    !apt-get install openjdk-8-jdk-headless -qq > /dev/null
```

```
[2] # Descargar Spark 3.2.2

    !wget -q https://archive.apache.org/dist/spark/spark-3.2.3/spark-3.2.3-bin-hadoop3.2.tgz
```

```
[3] # Descomprimir el archivo descargado de Spark

    !tar xf spark-3.2.3-bin-hadoop3.2.tgz
```

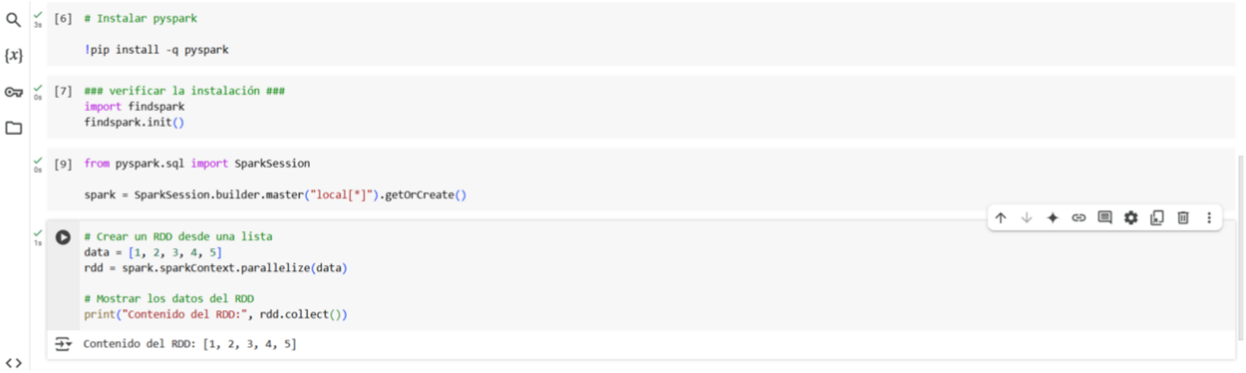
```
[4] # Establecer las variables de entorno

    import os

    os.environ["JAVA_HOME"] = "/usr/lib/jvm/java-8-openjdk-amd64"
    os.environ["SPARK_HOME"] = "/content/spark-3.2.3-bin-hadoop3.2"
```

```
[5] # Instalar la librería findspark

    !pip install -q findspark
```



```
[6] # Instalar pyspark

    !pip install -q pyspark
```

```
[7] ### verificar la instalación ###
    import findspark
    findspark.init()
```

```
[9] from pyspark.sql import SparkSession

    spark = SparkSession.builder.master("local[*]").getOrCreate()
```

```
# Crear un RDD desde una lista
data = [1, 2, 3, 4, 5]
rdd = spark.sparkContext.parallelize(data)

# Mostrar los datos del RDD
print("Contenido del RDD:", rdd.collect())
```

```
Contenido del RDD: [1, 2, 3, 4, 5]
```