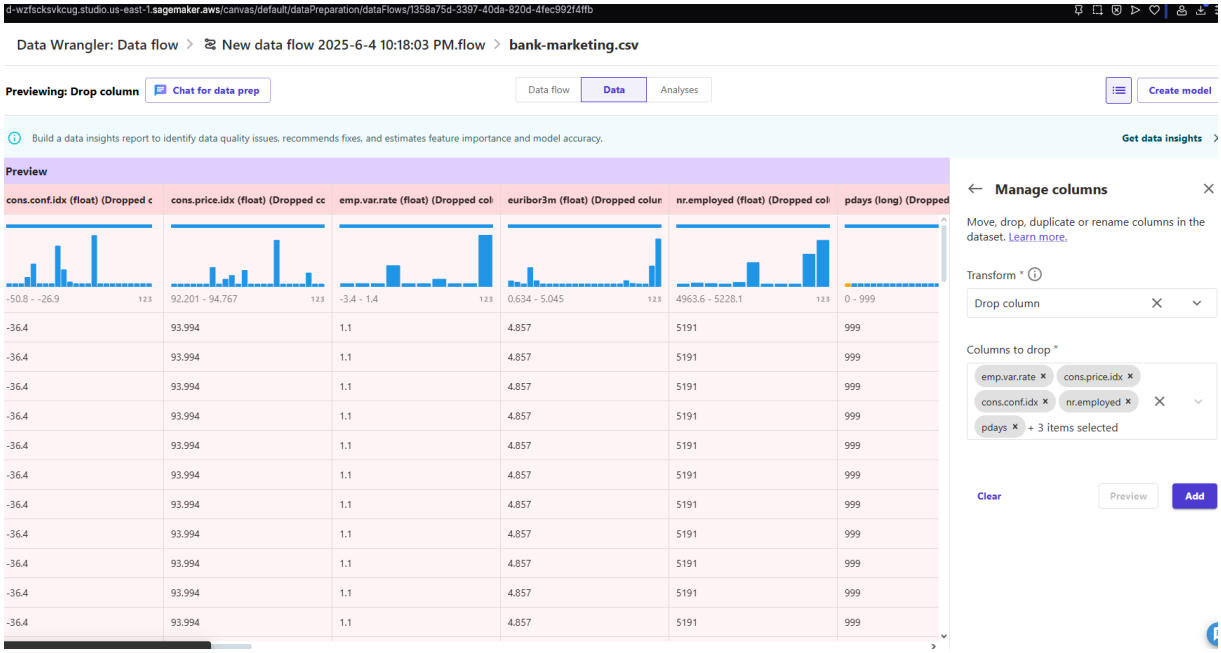
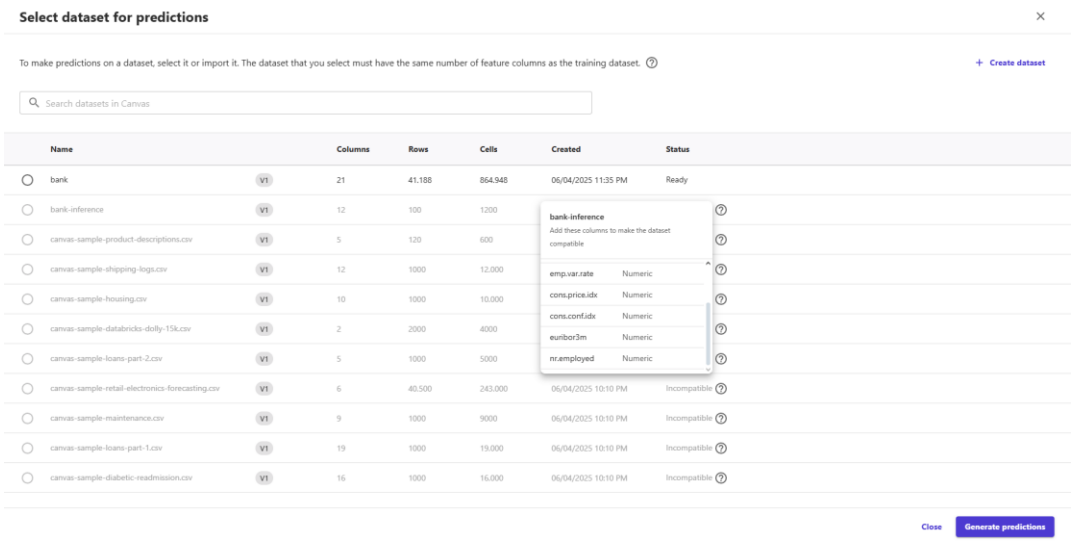
Actividad 9: refuerzo con AWS Sagemaker Autopilot

Antes de empezar con los puntos de la actividad voy a mostrar el proceso que seguí la primera vez que hice la tarea, menos detalladamente y que nos llevó a que nos pasara el mismo problema en el apartado del inference que cuando hicimos el ejemplo en clase el día 20 de mayo.

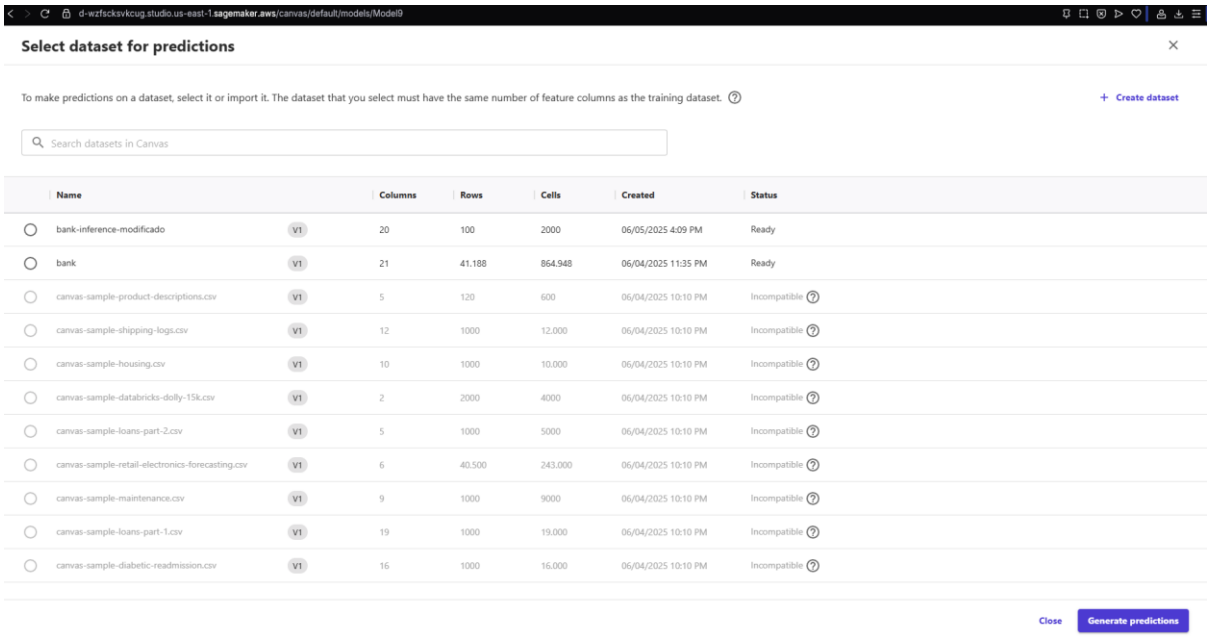
En el proceso de seguir las pautas para agilizar los pasos si hacemos el modelo sobre el dataset inicial sin meter el transform nos saldran estos dos problemas:



Aqui ya nos avisa de la posibilidad de que de probema realizar el transform. Despues en la vez que realizamos el modelo sobre el dataset ya modificado no nos dara este problema.

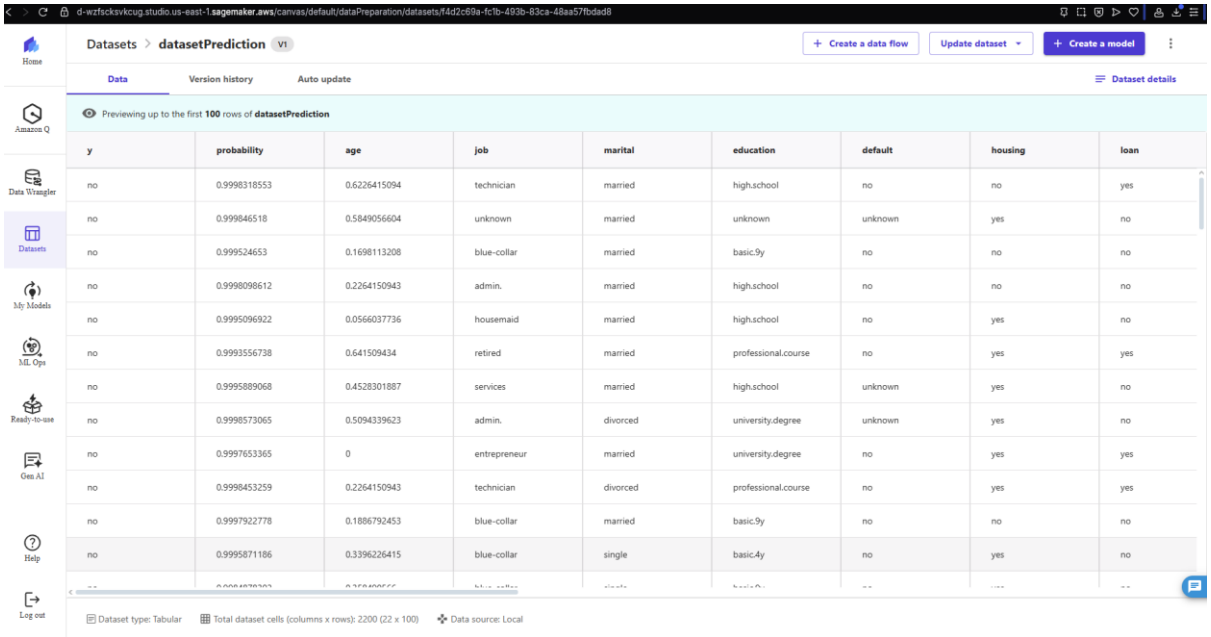


Como podemos observar nos dice que faltan columnas para realizar las predicciones y no nos deja activar el dataset de inference por eso mismo. Con la finalidad de poder hacerlo de esta manera por si no encontraba otra solucion , hice un edit del dataset de inference en pandas y le añadi esas columnas, rellenando sus datos con el valor medio y en el caso de las categoricas con el resultado más comun de la columna.



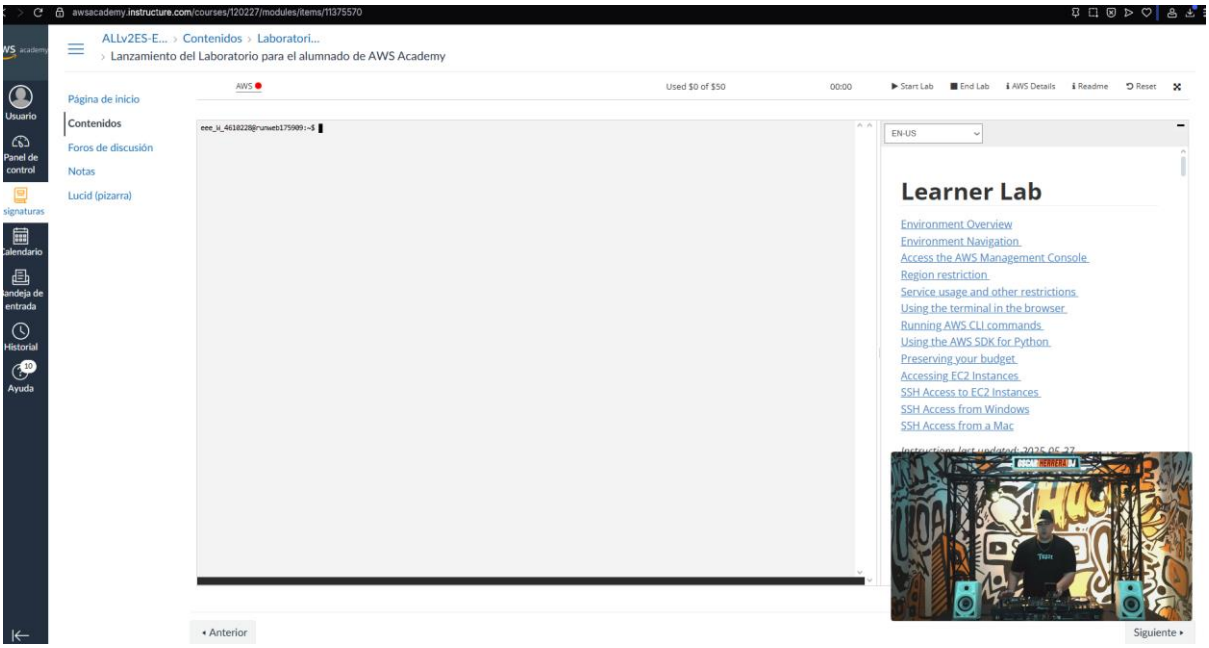
Aquí observamos que ya nos permite seleccionar el dataset de inference.

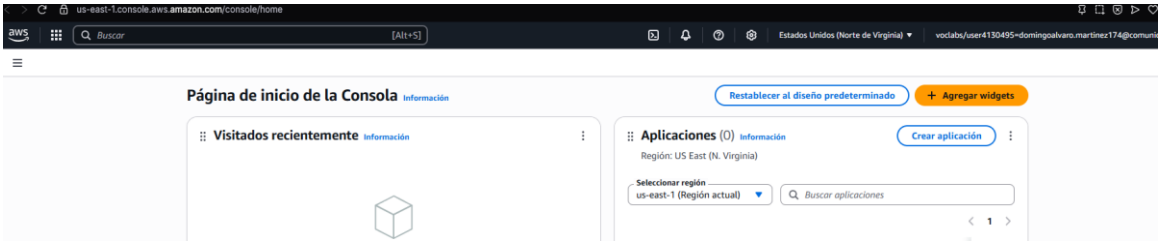
Cargamos ese dataset que hemos generado dentro de la pestaña de canvas de datasets para poder ver que se hizo la predicción correctamente.



Vemos que ha generado bien el dataset de predicción y vamos ya con el ejercicio con su proceso para que el inference no de problemas a la hora de cargar ese dataset.

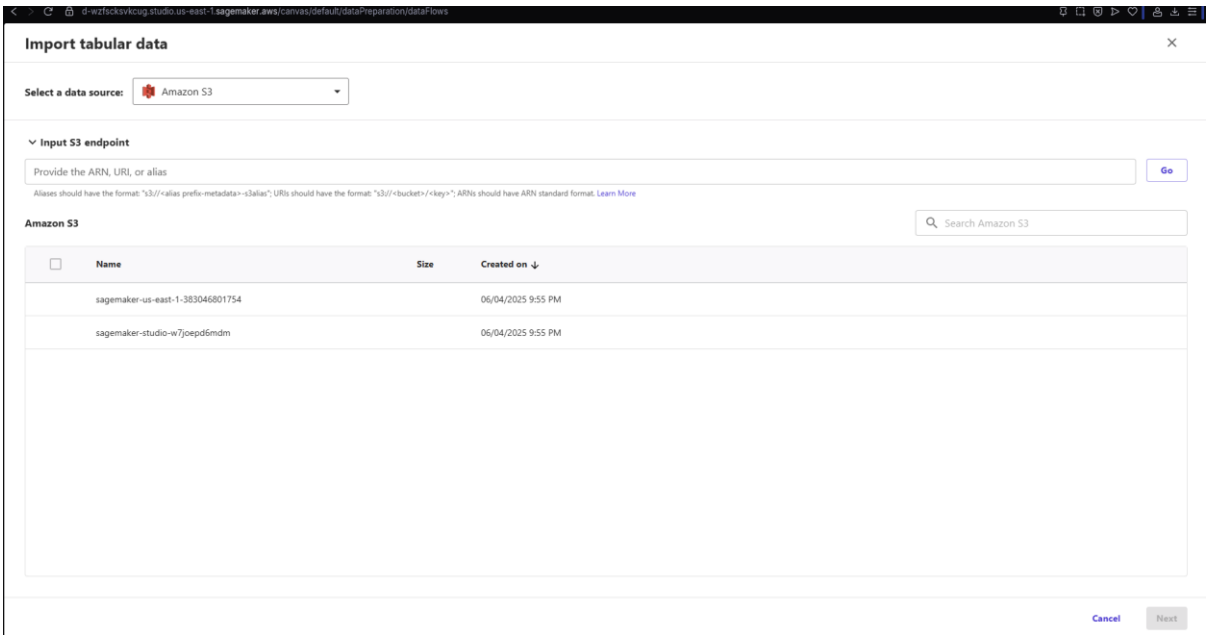
1. Empezamos con la entrada a nuestro laboratorio de AWS.



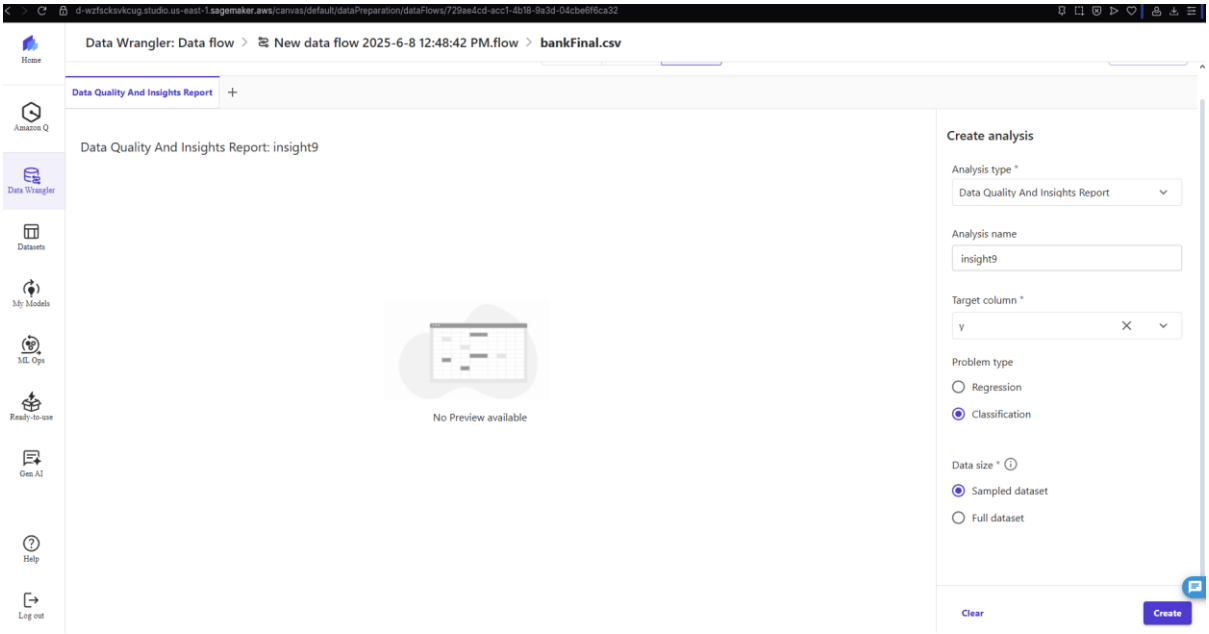


Ya tenemos la cuenta cargada, procedemos a entrar a canvas.

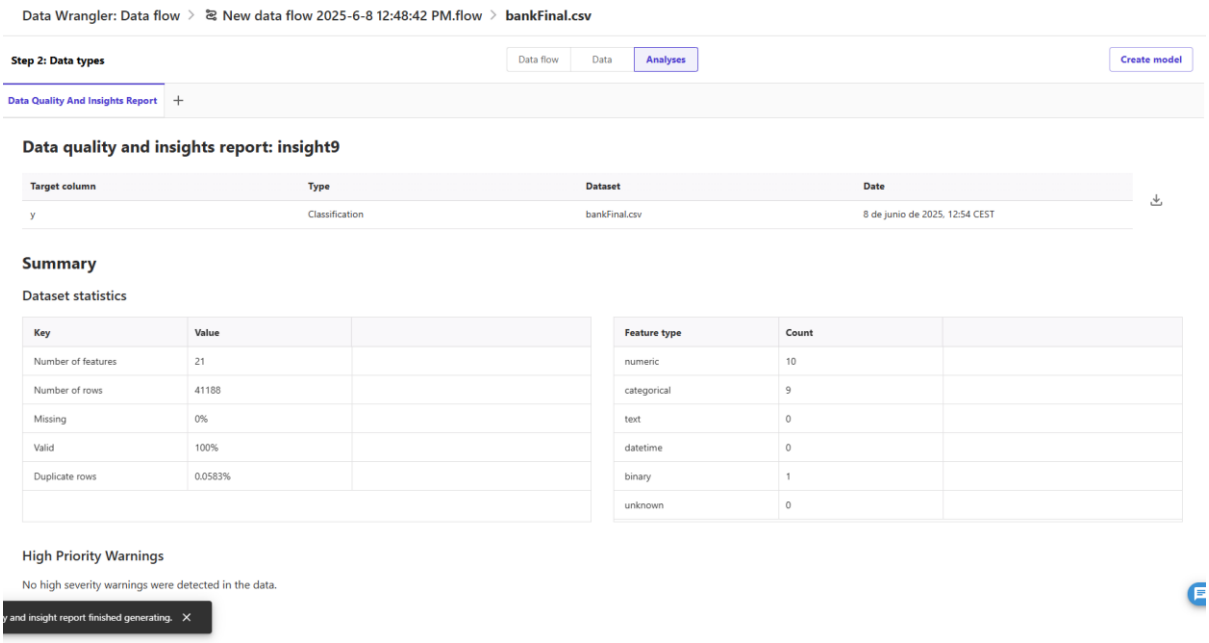
2. Carga del dataset.



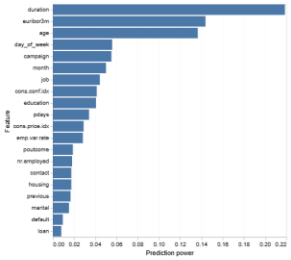
3. Generamos data insight.



4. Analizamos el resultado de estos.



Feature	Prediction power	Type	Valid	Missing	High severity warnings	Medium severity warnings
duration	0.219	numeric	100%	0%	0	0
euribor3m	0.144	numeric	100%	0%	0	0
age	0.137	numeric	100%	0%	0	0
day_of_week	0.056	categorical	100%	0%	0	0
campaign	0.0552	numeric	100%	0%	0	0
month	0.0501	categorical	100%	0%	0	1



Prediction power of the features. Higher prediction power scores, toward 1, indicate columns that are more useful for predicting the target on their own. Lower scores, toward 0 point to columns that contain little useful information for predicting the target on their own.

Canvas detected that 0.0583% of the data are duplicate. Some data sources could include valid duplicates. Other data sources could have duplicates that point to problems in data collection. Duplicate samples that result from faulty data collection could interfere with machine learning processes that rely on splitting the data into independent training and validation folds.

The following are examples of processes that can suffer from duplicated samples:

- Quick model analysis
- Prediction power estimation
- Hyper-parameters optimization

The most common duplicate rows are presented below. The number of occurrences of a row is given in left most column named Duplicate count. You can remove duplicate samples from the dataset using the Drop duplicates transform under Manage rows.

Note: features of type vector are ignored in duplicate rows detection

Duplicate count	age	job	marital	education	default	housing
2	24	services	single	high.school	no	yes
2	27	technician	single	professional.course	no	no
2	32	technician	single	professional.course	no	yes
2	35	admin.	married	university.degree	no	yes
2	36	retired	married	unknown	no	no
2	39	admin.	married	university.degree	no	no

Duplicate rows Medium

We found that 0.0583% of the data are duplicate. Some data sources could include valid duplicates and in other cases these duplicates could point to problems in data collection. Duplicate samples resulting from faulty data collection, could derail machine learning processes that rely on splitting to independent training and validation folds. For example quick model scores, prediction power estimation and automatic hyper parameter tuning. Duplicate samples could be removed from the dataset using the Drop duplicates transform under Manage rows.

Anomalous samples

Canvas detects anomalous samples using the Isolation forest algorithm after basic preprocessing. The isolation forest associates an anomaly score to each sample (row) of the dataset.

Quick model

Quick model provides an estimate of the expected predicted quality of a model that you train on your data.

The data is split into training and validation folds where Canvas uses 80% of the samples for training and 20% of the values for validation. For classification the sample is stratified split. For a stratified split, each data partition has the same ratio of labels. For classification problems, it's important to have the same ratio of labels between the training and classification folds. Canvas trains the XGBoost model with the default hyper-parameters. It applies early stopping on the validation data and performs minimal feature pre-processing.

Metric	Validation scores	Train scores
Accuracy	0.916	0.932
Balanced accuracy	0.753	0.793
ROC-AUC	0.949	0.964
F1	0.594	0.669
Precision	0.656	0.736
Recall	0.543	0.614

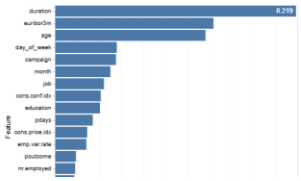
class	precision	recall	f1-score	support
no	0.9432396251673361	0.9638850889192886	0.9534506089309879	7310.0
yes	0.65625	0.5431034482758621	0.5943396226415094	928.0

Feature summary

See a summary of the features ordered by the prediction power. Prediction power is measured by stratified splitting the data into 80%/20% training and validation folds.
We fit a model for each feature separately on the training fold after applying minimal feature pre-processing and measure prediction performance on the validation data.

- The scores are normalized to the range [0,1].
- Higher prediction power scores, toward 1, indicate columns that are more useful for predicting the target on their own.
- Lower scores, toward 0 point to columns that contain little useful information for predicting the target on their own. Although it can happen that a column is uninformative on its own but is useful in predicting the target when used in tandem with other features, a low score usually indicates the feature is redundant.
- A score of 1 implies perfect predictive abilities, which often indicates an error called target leakage. The cause is typically a column that will not be available at prediction time such as a duplicate of the target.

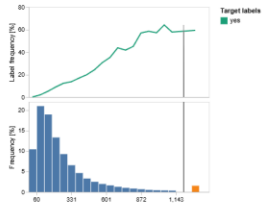
Feature	Prediction power	Type	Valid	Missing	High severity warnings	Medium severity warnings
duration	0.219	numeric	100%	0%	0	0
euribor3m	0.144	numeric	100%	0%	0	0
age	0.137	numeric	100%	0%	0	0
day_of_week	0.056	categorical	100%	0%	0	0
campaign	0.0552	numeric	100%	0%	0	0
month	0.0501	categorical	100%	0%	0	1



Feature details

duration · numeric

key	value
Feature name	duration
Type	numeric
Prediction power	0.219
Valid	100%
Missing	0%
Outliers	1.5%
Min	0
Max	4.92e+03
Mean	258



Histogram of duration with the corresponding target distribution. The lower plot provides the feature distribution and the upper - the corresponding target class frequency. The red bars contain outliers and the value below them is the outliers average.

euribor3m · numeric

key	value
Feature name	euribor3m
Type	numeric
Prediction power	0.144
Valid	100%
Missing	0%

Definitions

Feature types

Numeric: Numeric values, either floats or integers. For example: age, income. The machine learning models assume that numeric values are ordered and a distance is defined over them, that is 3 is closer to 4 than to 10 and 3 < 4 < 10.
Categorical: The column entries belong to a set of unique values which is usually much smaller than the number of entries in the column. For example, a column of length 100 containing the unique values "Dog", "Cat" and "Mouse". The values could be numeric, text of combination of both e.g. "Horse", "House", 8, "Love" and 3.1 are all valid values and can be found in the same categorical column. As opposed to numeric features, the machine learning model does not assume order or distance on the values of a categorical feature (even when all the values are numbers).
Binary: A special case of categorical feature where the cardinality of the set of unique values is 2.
Text: A text column contains many non-numeric unique values. In extreme cases, all the elements of the column are unique i.e., no two entries are the same.
Datetime:This column contains date and/or time information.

Feature Statistics

Prediction power

Outliers (in numeric features and regression target): Outliers are detected using two statistics that are robust to outliers: median and robust standard deviation (RSTD). RSTD is derived by clipping the feature values to the range [5 percentile, 95 percentile] and calculating the standard deviation of the clipped vector. All values larger than median + 5 * RSTD or smaller than median - 5 * RSTD are considered to be outliers.
Skew (in numeric features and regression target): Skew measures the symmetry of the distribution and is defined as the third moment of the distribution divided by the third power of the standard deviation. The skewness of the normal distribution or any other symmetric distribution is zero. Positive values imply that the right tail of the distribution is longer than the left tail and negative values - vice versa. As a thumb rule, a distribution is considered skewed when the absolute value of the skew is larger than 3.
Kurtosis (in numeric features and regression target): Pearson's kurtosis measures the heaviness of the tail of the distribution and is defined as the fourth moment of the distribution divided by the square of the second moment. The kurtosis of the normal distribution is 3. Thus, kurtosis values lower than 3 imply that the distribution is concentrated around the mean and the tails are lighter than the tails of the normal distribution. On the other hand, Kurtosis values higher than 3 imply heavier tails and/or outliers.

Values

Numeric features / regression target: All values that could be casted to finite floats are valid. Missing values are not valid.
Categorical / binary / text features / classification target: All values that are not missing are valid.
Datetime features: All values that could be casted to datetime object are valid. Missing values are not valid.
Invalid values: Either missing or could not be casted to the desired type. See the definition of valid values for more information

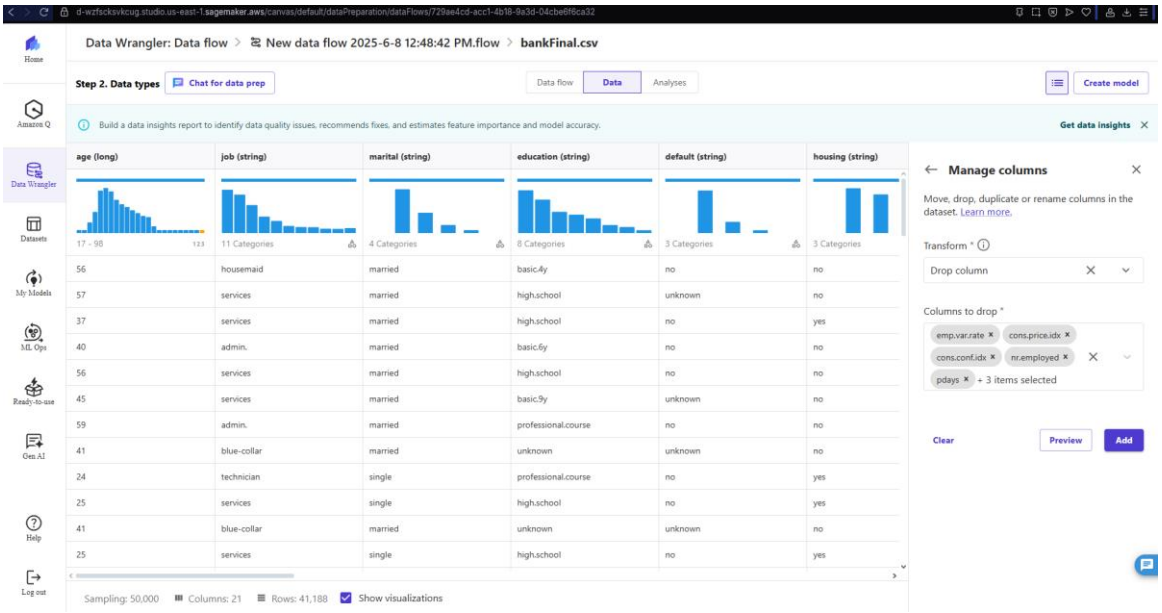
Quick model metrics

Regression

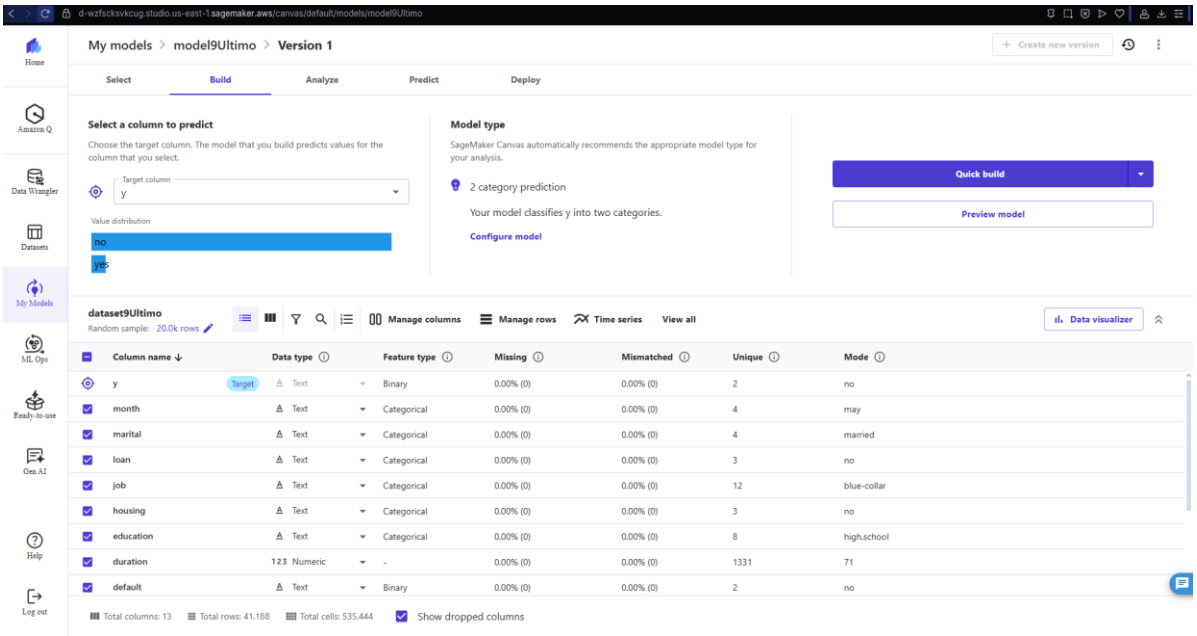
R2 (coefficient of determination): R2 is the proportion of the variation in the target that is predicted by the model. R2 is in the range of [-infty, 1] where 1 is the score of the model that predicts the target perfectly and 0 is the score of the trivial model that always predicts the target mean.
MSE: Mean squared error. MSE is in the range [0, infty] where 0 is the score of the model that predicts the target perfectly
MAE: Mean absolute error. MAE is in the range [0, infty] where 0 is the score of the model that predicts the target perfectly
RMSE: Root mean square error. RMSE is in the range [0, infty] where 0 is the score of the model that predicts the target perfectly
Median absolute error: In the range [0, infty] where 0 is the score of the model that predicts the target perfectly

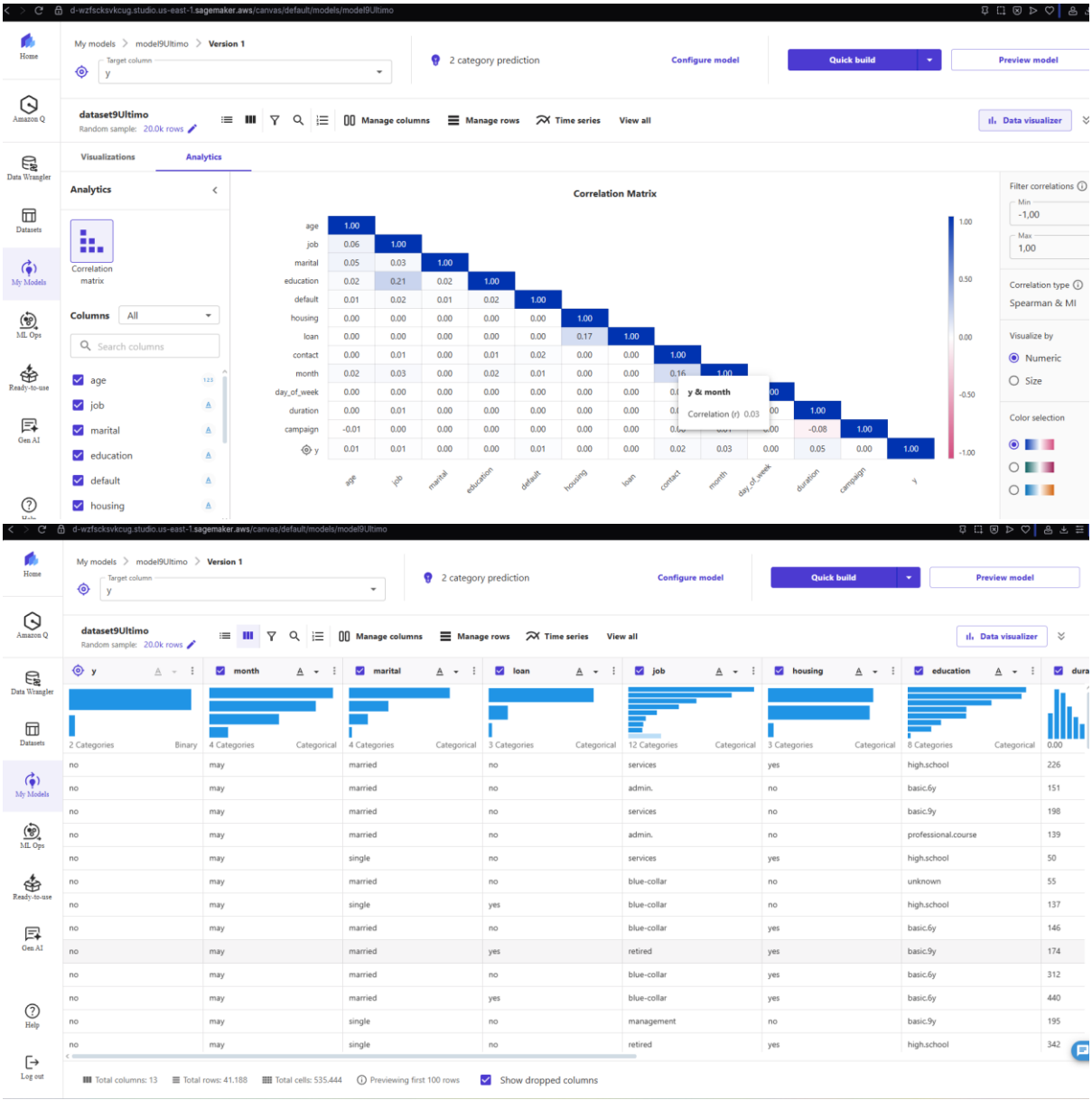
Vemos en el análisis que al final del todo nos da una descripción detallada del dataset. En las primeras capturas lo más reseñable es el porcentaje bajo de datos duplicados. Podemos seguir con una imagen de una estimación que daría un quick model. Seguimos con una captura de las características ordenadas por poder de predicción. Tenemos los datos normalizados de 0 a 1 para un análisis más fácil. En la última captura vemos una explicación de las cosas más importantes relacionadas con el EDA.

5. Aplicación del transform al dataset.



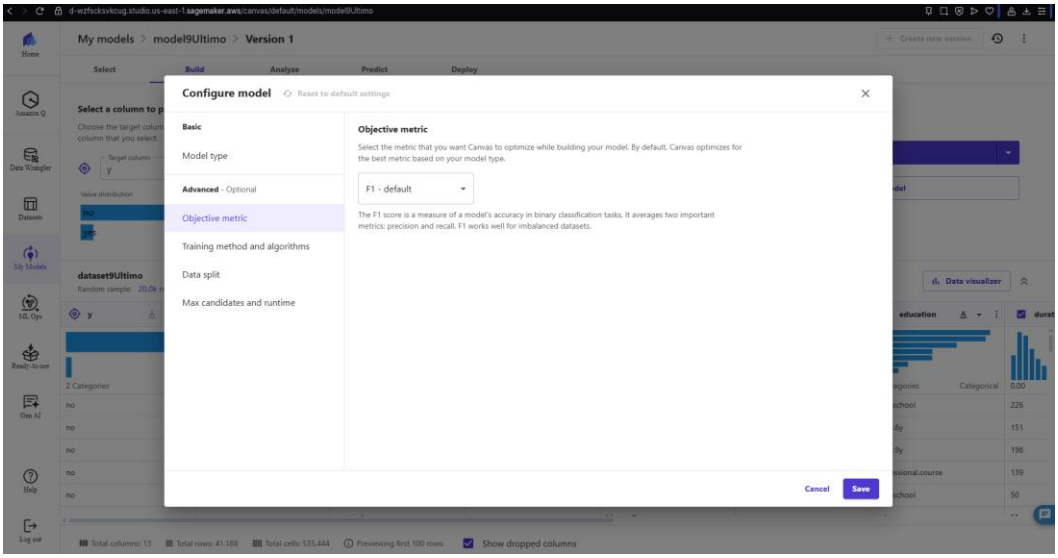
Le eliminamos las columnas que no aportan nada al entrenamiento del modelo.

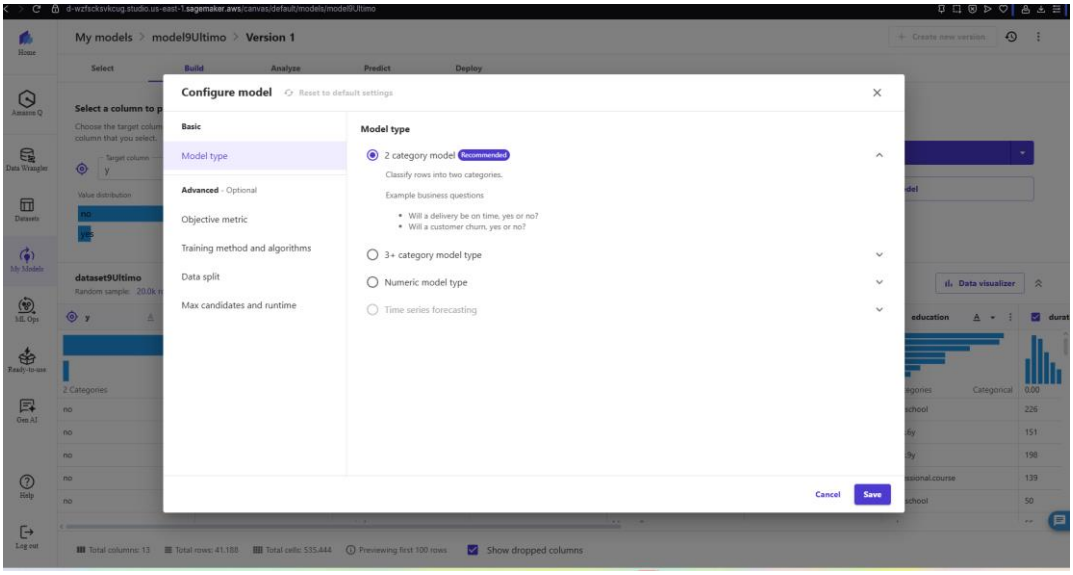


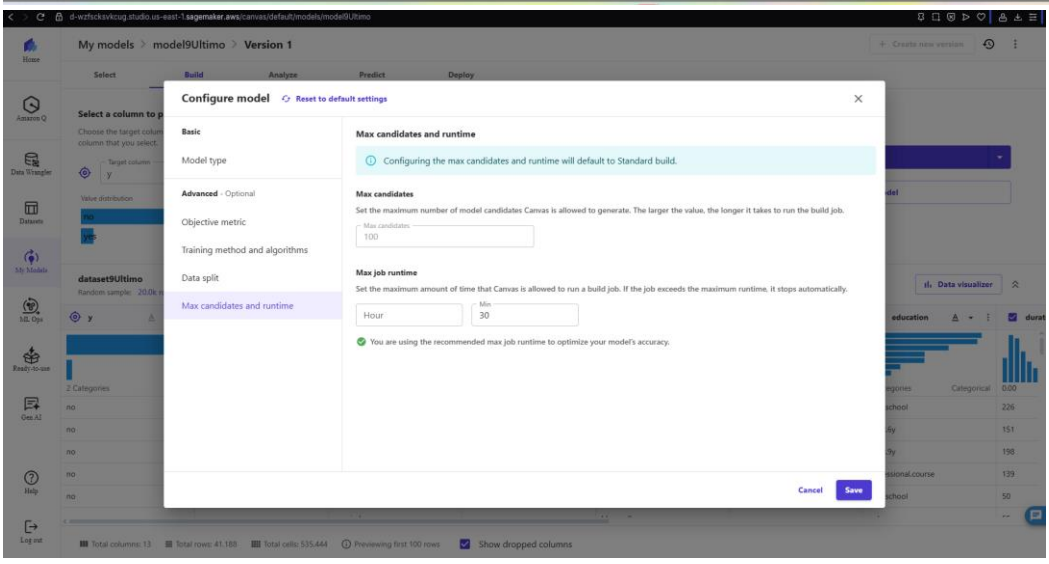


En la pestaña de build del modelo vemos como ha quedado el dataset y algunas graficas relevantes de este.

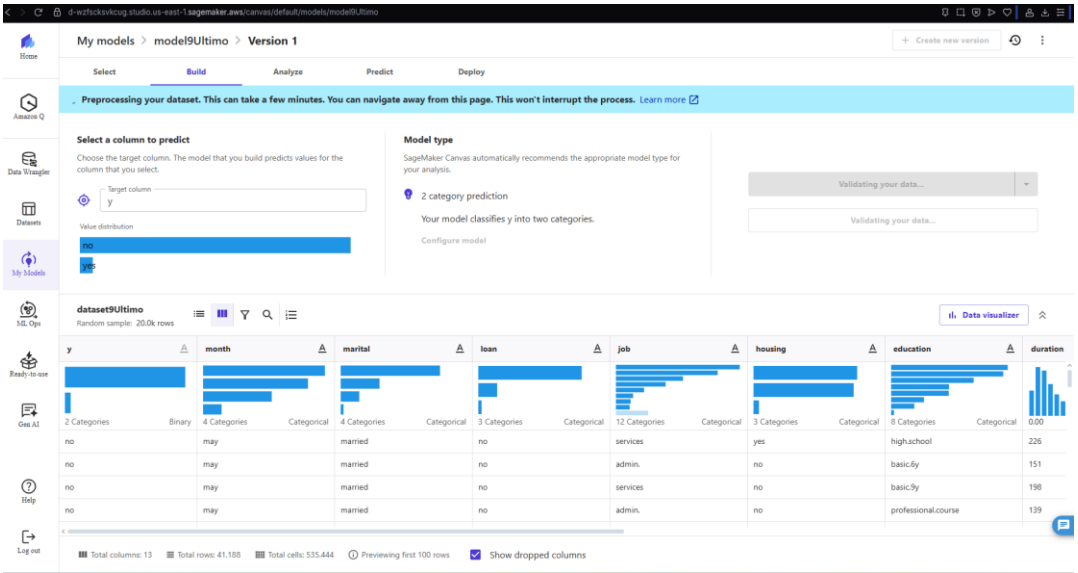
6. Entrenamiento del modelo
   Ahora cambiaremos en una sola ventana al darle a la configuración del modelo.

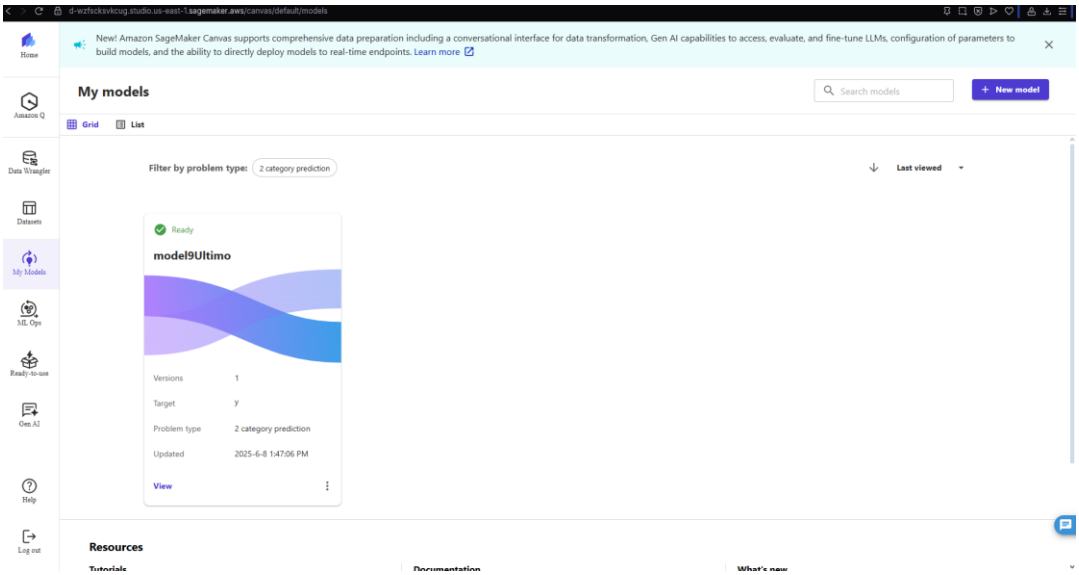


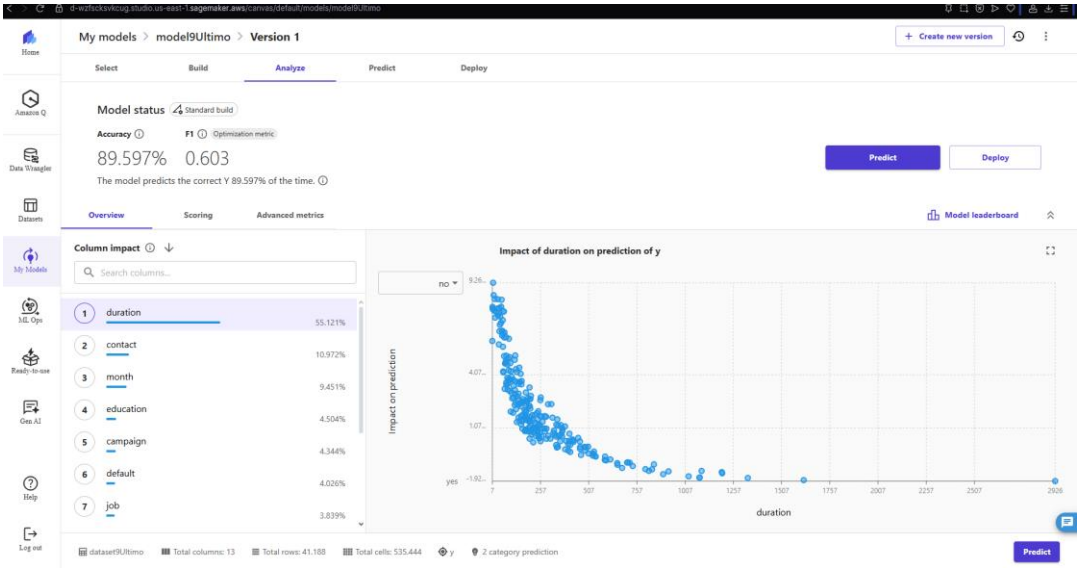


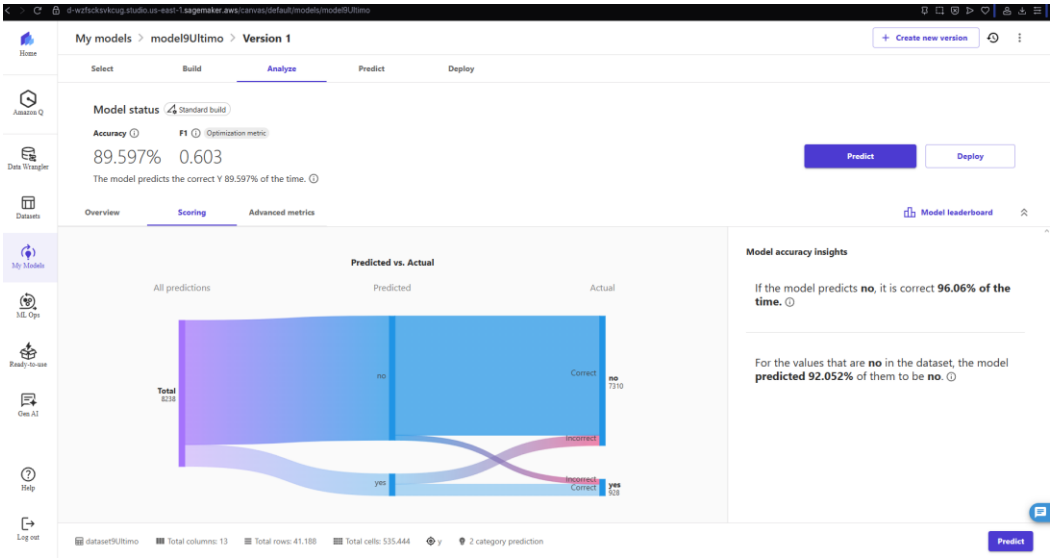


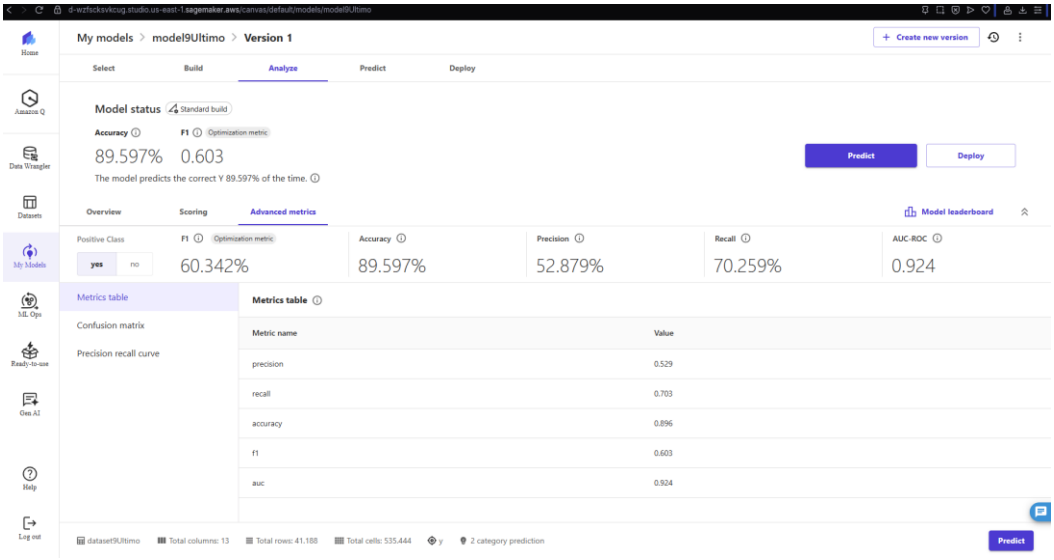
En la casilla min del apartado de max job runtime cambiamos a 30 lo que hará automáticamente que cambie de quick build a standard build. Esto hará que el entrenamiento del modelo sea más largo.

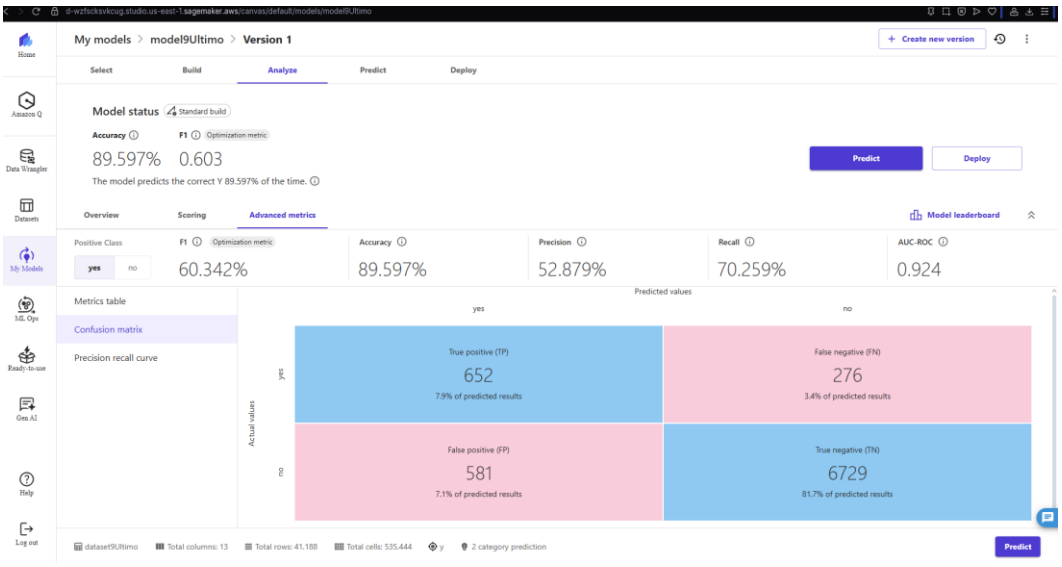

My models > model9Ultimo > Version 1
Create new version
Select
Build
Analyze
Predict
Deploy
Preprocessing your dataset. This can take a few minutes. You can navigate away from this page. This won't interrupt the process. Learn more
Select a column to predict
Choose the target column. The model that you build predicts values for the column that you select.
Target column
y
Value distribution
no
yes
Model type
SageMaker Canvas automatically recommends the appropriate model type for your analysis.
2 category prediction
Your model classifies y into two categories.
Configure model
Validating your data...
Validating your data...
dataset9Ultimo
Random sample: 20.0k rows
Data visualizer
y
month
marital
loan
job
housing
education
duration
2 Categories
Binary
4 Categories
Categorical
4 Categories
Categorical
3 Categories
Categorical
12 Categories
Categorical
3 Categories
Categorical
8 Categories
Categorical
0.00
no may married no services yes high.school 226
no may married no admin. no basic.6y 151
no may married no services no basic.9y 198
no may married no admin. no professional.course 139
Total columns: 13
Total rows: 41,188
Total cells: 535,444
Previewing first 100 rows
Show dropped columns

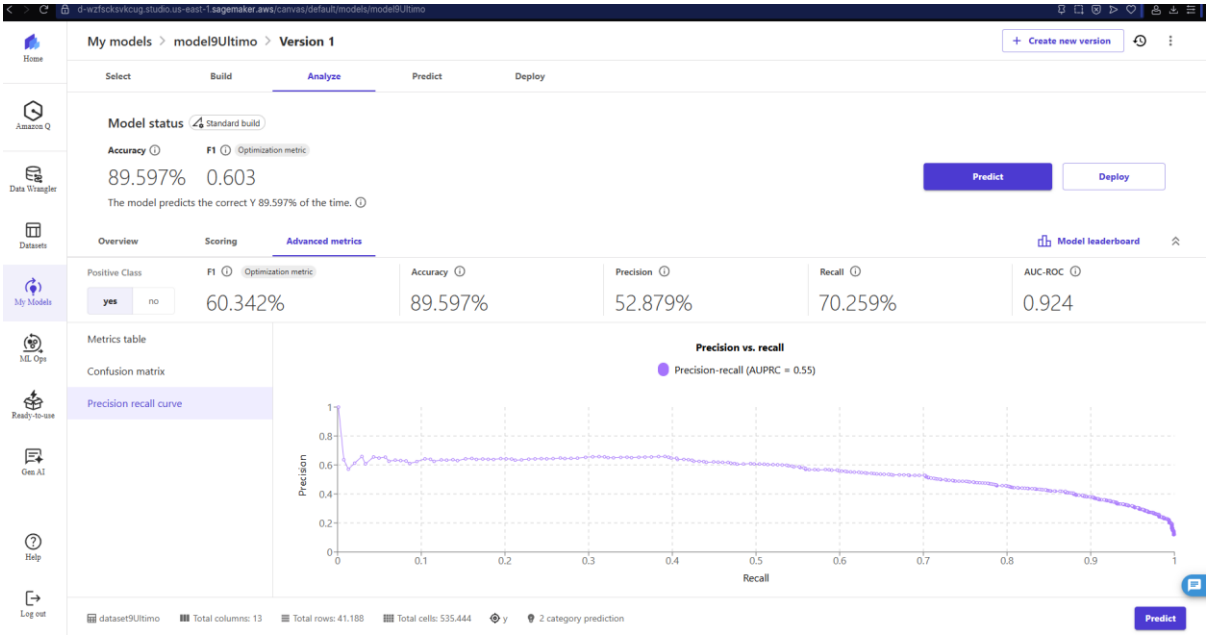

New! Amazon SageMaker Canvas supports comprehensive data preparation including a conversational interface for data transformation, Gen AI capabilities to access, evaluate, and fine-tune LLMs, configuration of parameters to build models, and the ability to directly deploy models to real-time endpoints. Learn more
My models
Search models
New model
Grid
List
Filter by problem type:
2 category prediction
Last viewed
Ready
model9Ultimo
Versions 1
Target y
Problem type 2 category prediction
Updated 2025-6-8 1:47:06 PM
View
Resources


My models > model9Ultimo > Version 1
Create new version
Select
Build
Analyze
Predict
Deploy
Model status
Standard build
Accuracy
F1
Optimization metric
89.597%
0.603
The model predicts the correct Y 89.597% of the time.
Predict
Deploy
Overview
Scoring
Advanced metrics
Model leaderboard
Column impact
Search columns...
1 duration 55.121%
2 contact 10.972%
3 month 9.451%
4 education 4.504%
5 campaign 4.344%
6 default 4.026%
7 job 3.839%
Impact of duration on prediction of y
no
yes
Impact on prediction
duration
dataset9Ultimo
Total columns: 13
Total rows: 41,188
Total cells: 535,444
y
2 category prediction
Predict


My models > model9Ultimo > Version 1
Create new version
Select
Build
Analyze
Predict
Deploy
Model status
Standard build
Accuracy
F1
Optimization metric
89.597%
0.603
The model predicts the correct Y 89.597% of the time.
Predict
Deploy
Overview
Scoring
Advanced metrics
Model leaderboard
Predicted vs. Actual
All predictions
Predicted
Actual
Total
8238
no
yes
Correct
Incorrect
Incorrect
Correct
no
7310
yes
828
Model accuracy insights
If the model predicts no, it is correct 96.06% of the time.
For the values that are no in the dataset, the model predicted 92.052% of them to be no.
dataset9Ultimo
Total columns: 13
Total rows: 41,188
Total cells: 535,444
y
2 category prediction
Predict

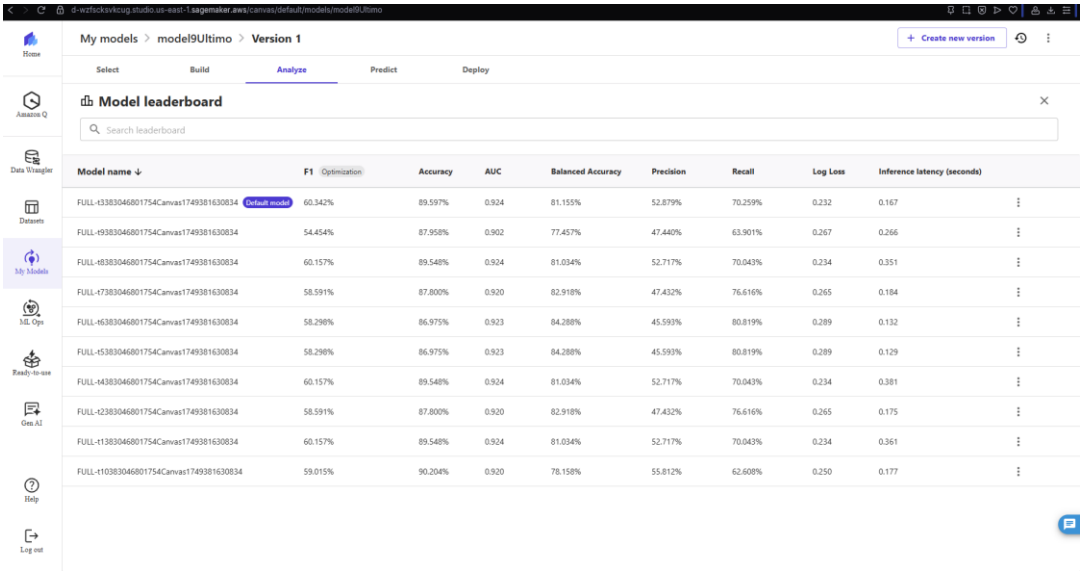

My models > model9Ultimo > Version 1
Create new version
Select
Build
Analyze
Predict
Deploy
Model status
Standard build
Accuracy
F1
Optimization metric
89.597%
0.603
The model predicts the correct Y 89.597% of the time.
Predict
Deploy
Overview
Scoring
Advanced metrics
Model leaderboard
Positive Class
yes
no
F1
60.342%
Accuracy
89.597%
Precision
52.879%
Recall
70.259%
AUC-ROC
0.924
Metrics table
Confusion matrix
Precision recall curve
Metrics table
Metric name
Value
precision
0.529
recall
0.703
accuracy
0.896
f1
0.603
auc
0.924
dataset9Ultimo
Total columns: 13
Total rows: 41,188
Total cells: 535,444
y
2 category prediction
Predict


My models > model9Ultimo > Version 1
Create new version
Select
Build
Analyze
Predict
Deploy
Model status
Standard build
Accuracy
F1
Optimization metric
89.597%
0.603
The model predicts the correct Y 89.597% of the time.
Predict
Deploy
Overview
Scoring
Advanced metrics
Model leaderboard
Positive Class
yes
no
F1
60.342%
Accuracy
89.597%
Precision
52.879%
Recall
70.259%
AUC-ROC
0.924
Metrics table
Confusion matrix
Precision recall curve
Predicted values
yes
no
Actual values
yes
no
True positive (TP)
652
7.9% of predicted results
False negative (FN)
276
3.4% of predicted results
False positive (FP)
581
7.1% of predicted results
True negative (TN)
6729
81.7% of predicted results
dataset9Ultimo
Total columns: 13
Total rows: 41,188
Total cells: 535,444
y
2 category prediction
Predict



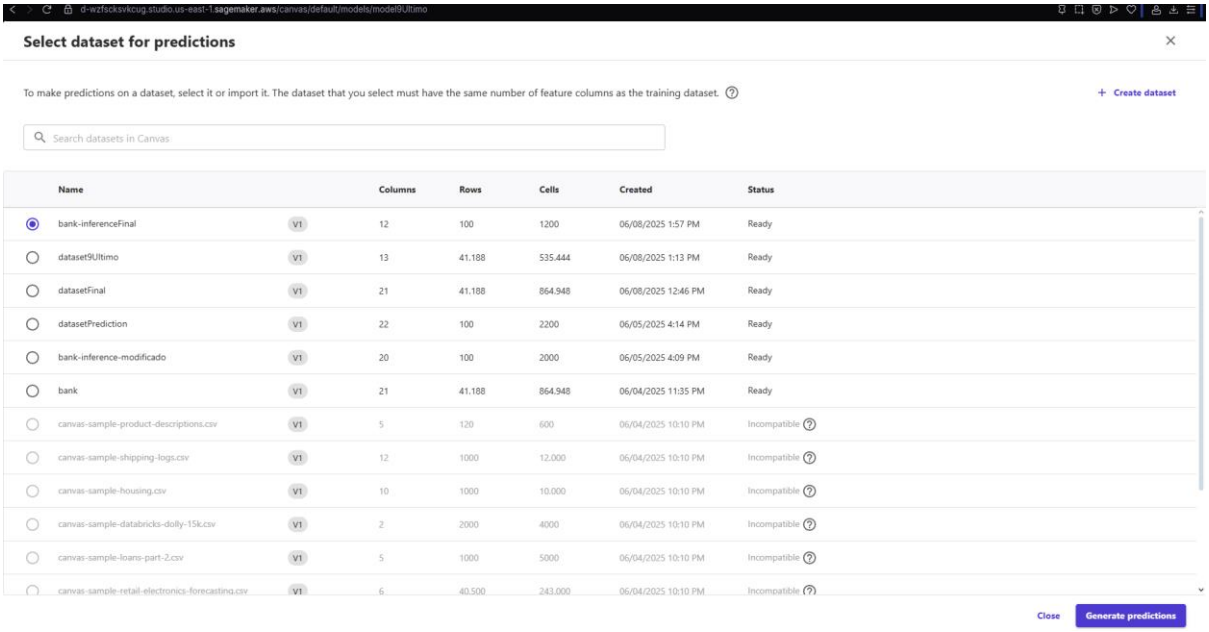
A continuación, vemos la captura del proceso de entrenamiento del modelo y el modelo ya listo. Después podemos observar las métricas más importantes representadas en gráficas y múltiples apartados.

Matriz de confusión, curva de precisión del recall, AUC-ROC , accuracy , f1....

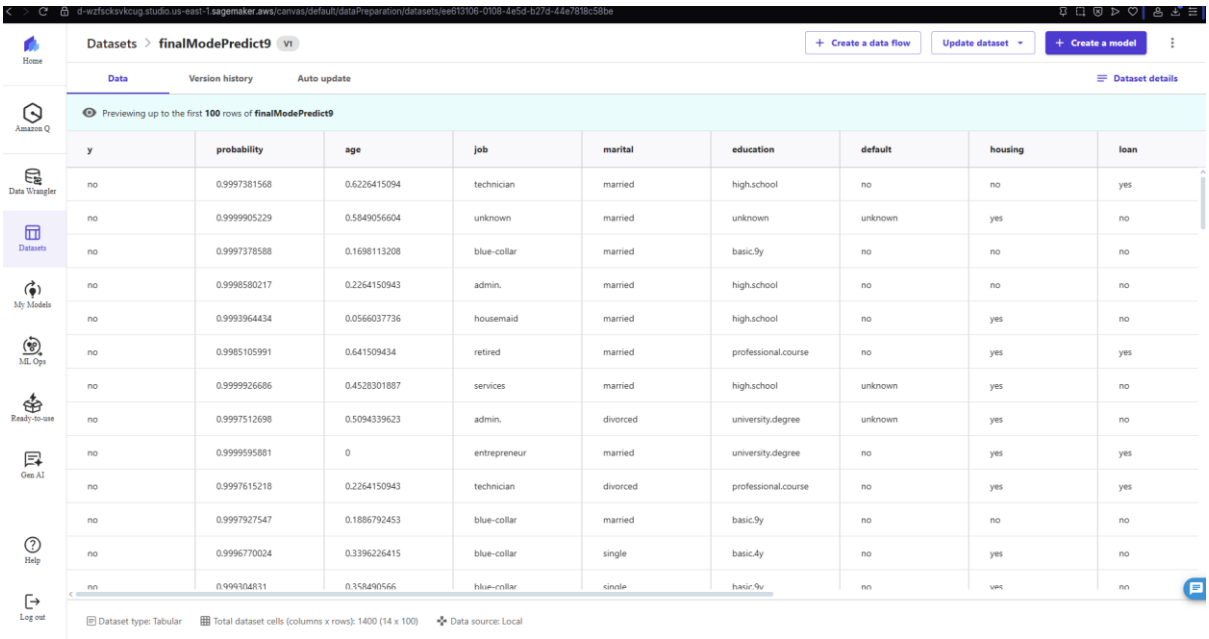


Mostramos la tabla de modelos y sus resultados.

7. Pasamos ya al apartado del inference con el dataset descargado de la página de amazon.

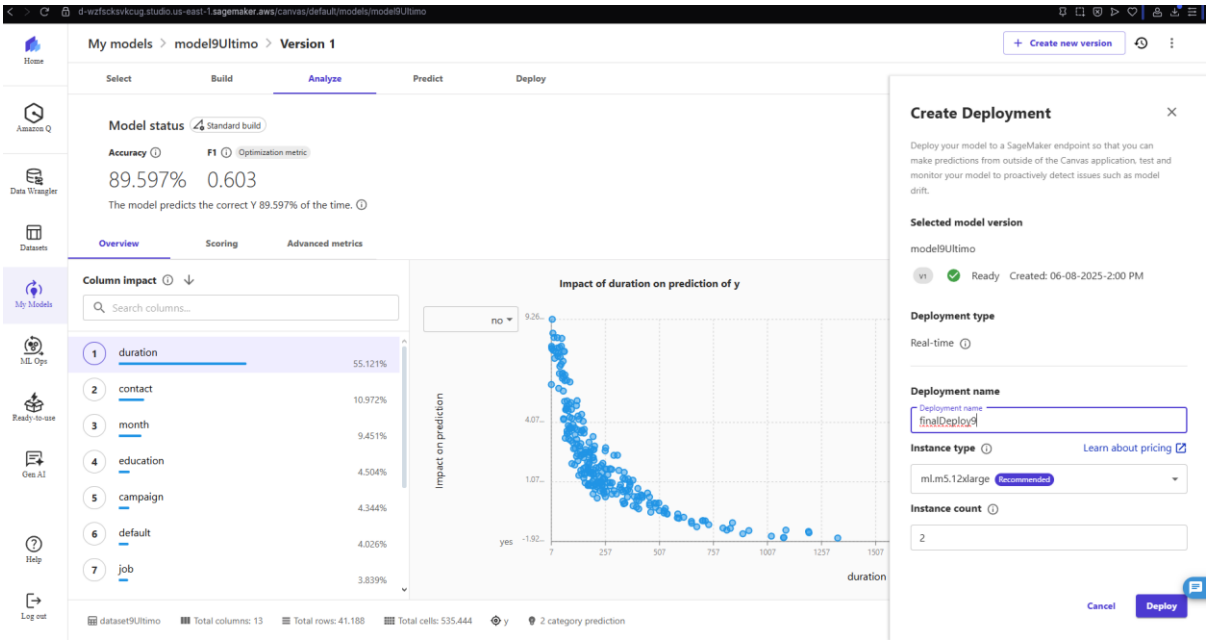


Vemos que el dataset de inference ya no nos muestra ninguna incompatibilidad como en el primer intento.

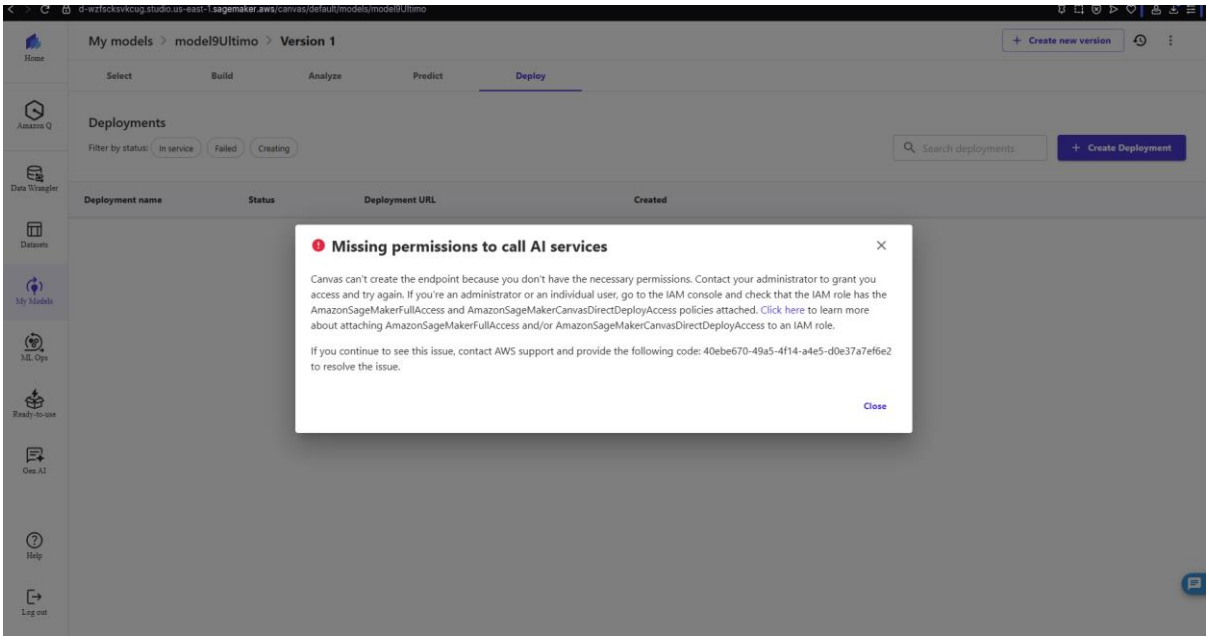


Cargamos el dataset resultante del predict y vemos las 100 primeras columnas.

8. Realizar el deploy del modelo.



Cambiamos el nombre al deploy.



Y vemos la captura que nos da canvas que como bien vimos en la clase del día 20, esta capada esta parte del modelado.