

Vertex AI: Predicting Loan Risk with AutoML

Overview

In this lab, you use [Vertex AI](#) to train and serve a machine learning model to predict loan risk with a tabular dataset.

Objectives

You learn how to:

- Upload a dataset to Vertex AI.
- Train a machine learning model with AutoML.
- Evaluate the model performance.
- Deploy the model to an endpoint.
- Get predictions.

Setup

Before you click the Start Lab button

Note: Read these instructions.

Labs are timed and you cannot pause them. The timer, which starts when you click **Start Lab**, shows how long Google Cloud resources will be made available to you.

This Qwiklabs hands-on lab lets you do the lab activities yourself in a real cloud environment, not in a simulation or demo environment. It does so by giving you new, temporary credentials that you use to sign in and access Google Cloud for the duration of the lab.

What you need

To complete this lab, you need:

- Access to a standard internet browser (Chrome browser recommended).
- Time to complete the lab.


Note: If you already have your own personal Google Cloud account or project, do not use it for this lab.**Note:** If you are using a Pixelbook, open an Incognito window to run this lab.


How to start your lab and sign in to the Console


1. Click the **Start Lab** button. If you need to pay for the lab, a pop-up opens for you to select your payment method. On the left is a panel populated with the temporary credentials that you must use for this lab.

[Open Google Console](#)

Caution: When you are in the console, do not deviate from the lab instructions. Doing so may cause your account to be blocked. [Learn more.](#)

Username
google2727032_student@qwiklabs.n 

Password
k68CZXsxMZ 

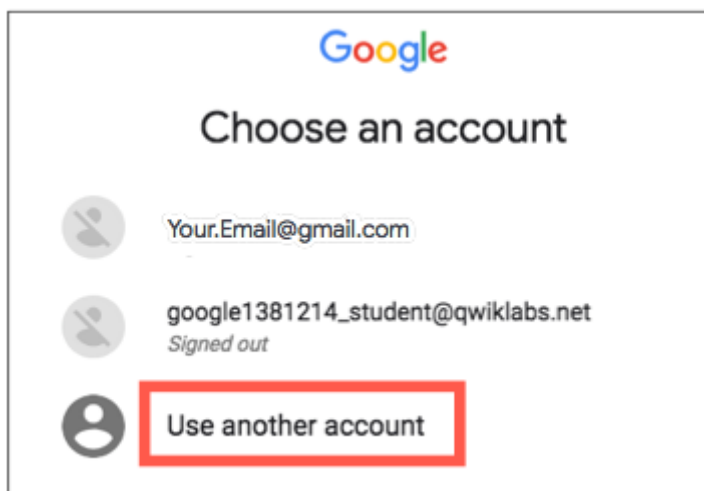
GCP Project ID
qwiklabs-gcp-4fbfecac8667e457 

[New to labs? View our introductory video!](#)

- Copy the username, and then click **Open Google Console**. The lab spins up resources, and then opens another tab that shows the **Choose an account** page.

Note: Open the tabs in separate windows, side-by-side.

- On the Choose an account page, click **Use Another Account**. The Sign in page opens.



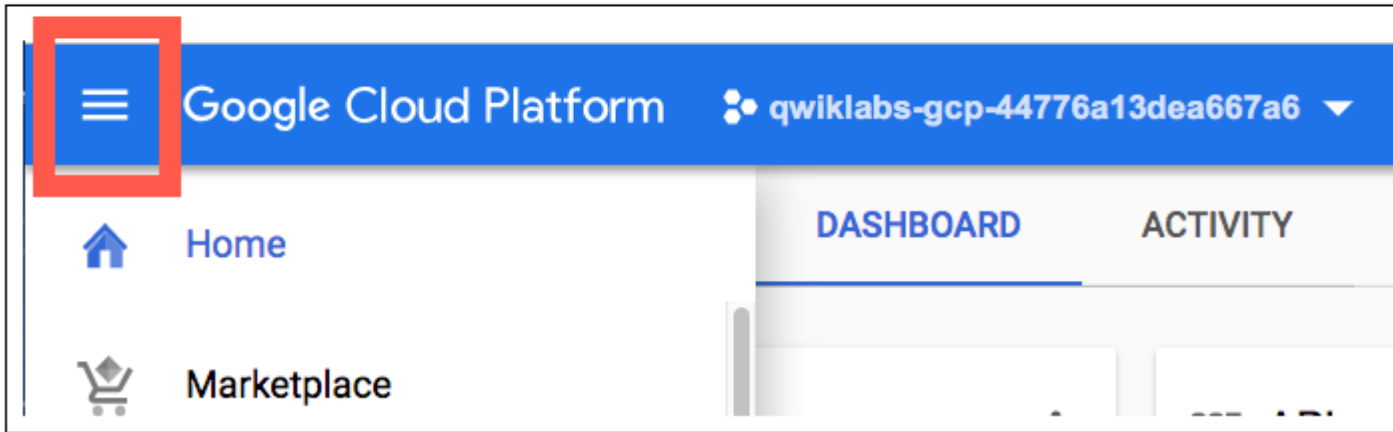
- Paste the username that you copied from the Connection Details panel. Then copy and paste the password.

Note: You must use the credentials from the Connection Details panel. Do not use your Google Cloud Skills Boost credentials. If you have your own Google Cloud account, do not use it for this lab (avoids incurring charges).

- Click through the subsequent pages:
 - Accept the terms and conditions.
 - Do not add recovery options or two-factor authentication (because this is a temporary account).
 - Do not sign up for free trials.

After a few moments, the Cloud console opens in this tab.

Note: You can view the menu with a list of Google Cloud Products and Services by clicking the **Navigation menu** at the top-left.



Introduction to Vertex AI

This lab uses [Vertex AI](#), the unified AI platform on Google Cloud to train and deploy a ML model. Vertex AI offers two options on one platform to build a ML model: a codeless solution with **AutoML** and a code-based solution with **Custom Training** using Vertex **Workbench**. You use **AutoML** in this lab.

In this lab you build a ML model to determine whether a particular customer will repay a loan.

Task 1. Prepare the training data

The initial Vertex AI dashboard illustrates the major stages to train and deploy a ML model: prepare the training data, train the model, and get predictions. Later, the dashboard displays your recent activities, such as the recent datasets, models, predictions, endpoints, and notebook instances.

Create a dataset

1. In the Google Cloud console, on the **Navigation menu**, click **Vertex AI > Datasets**.
2. Click **Create dataset**.
3. On the Datasets page, give the dataset a name.
4. For the data type and objective, click **Tabular**, and then select **Regression/classification**.
5. Click **Create**.

Upload data

There are three options to import data in Vertex AI:

- Upload a local file from your computer.
- Select files from Cloud Storage.
- Select data from BigQuery.

For convenience, the dataset is already uploaded to Cloud Storage.

1. For the data source, select **Select CSV files from Cloud Storage**.
2. For **Import file path**, enter:

spls/cbl455/loan_risk.csv

Copied!

content_copy

3. Click **Continue**.

Note: You can also configure this page by clicking **Datasets** on the left menu and then selecting the dataset name on the Datasets page.

(Optional) Generate statistics

1. To see the descriptive statistics for each column of your dataset, click **Generate statistics** .
Generating the statistics might take a few minutes, especially the first time.
2. When the statistics are ready, click each column name to display analytical charts.

Task 2. Train your model

With a dataset uploaded, you're ready to train a model to predict whether a customer will repay the loan.

- Click **Train new model**.

Training method

The dataset is called *LoanRisk*.

1. For **Objective**, select **Classification**.

You select classification instead of regression because you are predicting a distinct number (whether a customer will repay a loan: 0 for repay, 1 for default/not repay) instead of a continuous number.

2. Click **Continue**.

Model details

Specify the name of the model and the target column.

1. Give the model a name, such as **LoanRisk**.
2. For **Target column**, select **Default** .
3. (Optional) Explore **Advanced options** to determine how to assign the training vs. testing data and specify the encryption.
4. Click **Continue**.

Training options

Specify which columns you want to include in the training model. For example, ClientID might be irrelevant to predict loan risk.

1. Click the minus sign on the **ClientID** row to exclude it from the training model.
2. (Optional) Explore **Advanced options** to select different optimization objectives. For more information about optimization objectives for tabular AutoML models, refer to the [Optimization objectives for tabular AutoML models guide](#).
3. Click **Continue**.

Compute and pricing

1. For **Budget**, which represents the number of node hours for training, enter **1**.
Training your AutoML model for 1 compute hour is typically a good start for understanding whether there is a relationship between the features and label you've selected. From there, you can modify your features and train for more time to improve model performance.
2. Leave early stopping enabled.
3. Click **Start training**.

Depending on the data size and the training method, the training can take from a few minutes to a couple of hours. Normally you would receive an email from Google Cloud when the training job is complete. However, in the Qwiklabs environment, you will not receive an email.

To save the waiting for the model training, you download a pre-trained model in **Task 5** to get predictions in **Task 6**. This pre-trained model is the training result following the same steps from **Task 1** to **Task 2**.

Task 3. Evaluate the model performance (demonstration only)

Vertex AI provides many metrics to evaluate the model performance. You focus on three:

- **Precision/Recall curve**
- **Confusion Matrix**
- **Feature Importance**

Note: If you had a model trained, you could navigate to the **Model Registry** tab in Vertex AI.

1. Navigate to the **Model Registry**.
2. Click on the model you just trained.
3. Browse the **Evaluate** tab.

However in this lab, you can skip this step since you use a pre-trained model.

The precision/recall curve

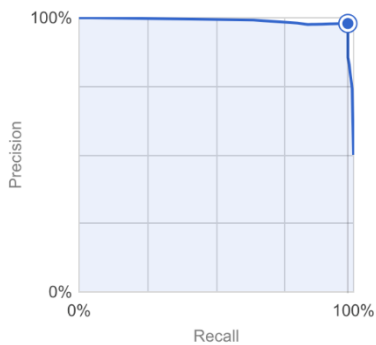
Confidence threshold ? 0.5

All labels

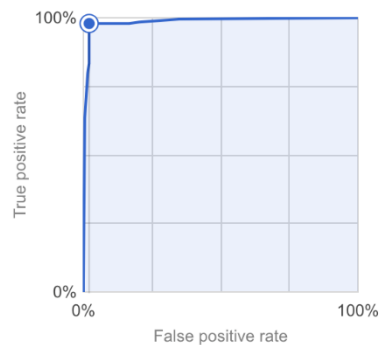
PR AUC ?	0.984
ROC AUC ?	0.986
Log loss ?	0.11
F1 score ?	0.9791667
Precision ?	97.9%
Recall ?	97.9%
Created	Dec 7, 2021, 6:27:33 PM

To evaluate your model, set the **confidence threshold** to see how precision and recall are affected. The best confidence threshold depends on your use case. Read some [example scenarios](#) to learn how evaluation metrics can be used.

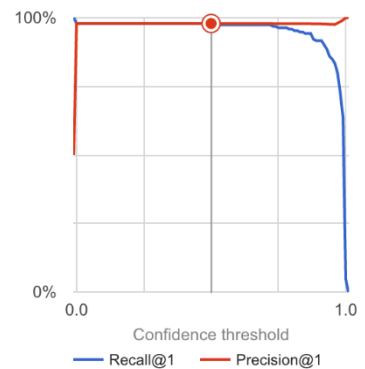
Precision-recall curve ?



ROC curve ?



Precision-recall by threshold ?



The confidence threshold determines how a ML model counts the positive cases. A higher threshold increases the precision, but decreases recall. A lower threshold decreases the precision, but increases recall.

You can manually adjust the threshold to observe its impact on precision and recall and find the best tradeoff point between the two to meet your business needs.

The confusion matrix

A [confusion matrix](#) tells you the percentage of examples from *each class* in your test set that your model predicted correctly.

Confusion matrix

This table shows how often the model classified each label correctly (in blue), and which labels were most often confused for that label (in gray).

True label	Predicted label	
	0	1
0	100%	—
1	13%	87%

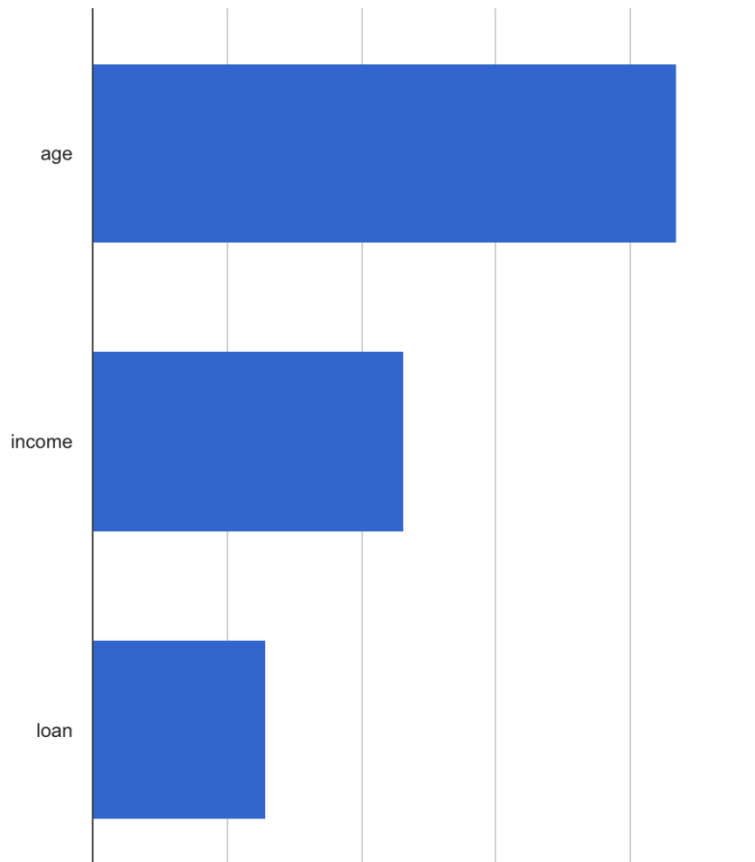
The confusion matrix shows that your initial model is able to predict 100% of the repay examples and 87% of the default examples in your test set correctly, which is not too bad.

You can improve the percentage by adding more examples (more data), engineering new features, and changing the training method, etc.

The feature importance

In Vertex AI, feature importance is displayed through a bar chart to illustrate how each feature contributes to a prediction. The longer the bar, or the larger the numerical value associated with a feature, the more important it is.

Feature Importance



These feature importance values could be used to help you improve your model and have more confidence in its predictions. You might decide to remove the least important features next time you train a model or to combine two of the more significant features into a [feature cross](#) to see if this improves model performance.

Feature importance is just one example of Vertex AI's comprehensive machine learning functionality called **Explainable AI**. Explainable AI is a set of tools and frameworks to help understand and interpret predictions made by machine learning models.

Task 4. Deploy the model (demonstration only)

Note: You will not deploy the model to an endpoint because the model training can take an hour. Here you can review the steps you would perform in a production environment.

Now that you have a trained model, the next step is to create an endpoint in Vertex. A model resource in Vertex can have multiple endpoints associated with it, and you can split traffic between endpoints.

Create and define an endpoint

1. On your model page, on the **Deploy & test** tab, click **Deploy to endpoint**.
2. For **Endpoint name**, enter a name for your endpoint, such as **LoanRisk**
3. Click **Continue**.

Model settings and monitoring

1. Leave the traffic splitting settings as-is.
2. As the machine type for your model deployment, under **Machine type**, select **n1-standard-8, 8 vCPUs, 30 GiB memory**.
3. Click **Continue**.
4. In **Model monitoring**, click **Continue**.
5. In **Model objectives > Training data source**, select **Vertex AI dataset**.
6. Select your dataset from the drop down menu.
7. In **Target column**, enter **Default**
8. Leave the remaining settings as-is and click **Deploy**.

Your endpoint will take a few minutes to deploy. When it is completed, a **green check mark** will appear next to the name.

Now you're ready to get predictions on your deployed model.

Task 5. SML Bearer Token

Retrieve your Bearer Token

To allow the pipeline to authenticate, and be authorized to call the endpoint to get the predictions, you will need to provide your Bearer Token.

Note: Follow the instructions below to get your token. If you have issues getting the Bearer Token, this can be due to cookies in the incognito window. If this is happening to you, try this step in a non-incognito window.

1. Log in to gsp-auth-kjyo252taq-uc.a.run.app.
2. When logging in, use your student email address and password.
3. Click the **Copy** button. This will copy a very long token to your clipboard.

QWIK

15 seconds



Signed in with

a21haWwuY29tliwiZW1haWwiOi

Copy

Note: This token will only be available for about **60 seconds**, so copy and move on to the next steps.**Note:** If you have issues getting the Bearer Token, this can be due to cookies in the incognito window - try in a non-incognito window.

Task 6. Get predictions

In this section, use the Shared Machine Learning (SML) service to work with an existing trained model.

ENVIRONMENT VARIABLE	VALUE
AUTH_TOKEN	Use the value from the previous section
ENDPOINT	https://sml-api-vertex-kjyo252taq-uc.a.run.app/vertex/predict/tabular_classification
INPUT_DATA_FILE	INPUT-JSON

To use the trained model, you will need to create some environment variables.

1. Open a Cloud Shell window.
2. Replace INSERT_SML_BEARER_TOKEN with the bearer token value from the previous section:

```
AUTH_TOKEN="INSERT_SML_BEARER_TOKEN"
```

Copied!

content_copy

3. Download the lab assets:

```
gsutil cp gs://spl/cbl455/cbl455.tar.gz .
```

Copied!

content_copy

4. Extract the lab assets:

```
tar -xvf cbl455.tar.gz
```

Copied!

content_copy

5. Create an ENDPOINT environment variable:

```
ENDPOINT="https://sml-api-vertex-kjyo252taq-uc.a.run.app/vertex/predict/tabular_classification"
```

Copied!

content_copy

6. Create a INPUT_DATA_FILE environment variable:

INPUT_DATA_FILE="INPUT-JSON"

Copied!

content_copy

Note: After the lab assets are extracted, take a moment to review the contents.

The **INPUT-JSON** file is used to provide Vertex AI with the model data required. Alter this file to generate custom predictions.

The smlproxy application is used to communicate with the backend.

The file INPUT-JSON is composed of the following values:

age	ClientID	income	loan
40.77	997	44964.01	3944.22

7. Test the SML Service by passing the parameters specified in the environment variables.

8. Perform a request to the SML service:

```
./smlproxy tabular \  
-a $AUTH_TOKEN \  
-e $ENDPOINT \  
-d $INPUT_DATA_FILE
```

Copied!

content_copy

This query should result in a response similar to this:

SML Tabular HTTP Response:

```
2022/01/10 15:04:45 {"model_class":"0","model_score":0.9999981}
```

9. Alter the INPUT-JSON file to test a new scenario:

age	ClientID	income	loan
30.00	998	50000.00	20000.00

10. Test the SML Service by passing the parameters specified in the environment variables.

11. Edit the file INPUT-JSON and replace the original values.

12. Perform a request to the SML service:

```
./smlproxy tabular \  
-a $AUTH_TOKEN \  
-e $ENDPOINT \  
-d $INPUT_DATA_FILE
```

-e \$ENDPOINT \

-d \$INPUT_DATA_FILE

Copied!

content_copy

In this case, assuming that the person's income is 50,000, age 30, and loan 20,000, the model predicts that this person will repay the loan.

This query should result in a response similar to this:

SML Tabular HTTP Response:

```
2022/01/10 15:04:45 {"model_class":"0","model_score":1.0322887E-5}
```

If you use the Google Cloud console, the following image illustrates how the same action could be performed:

The screenshot displays the Google Cloud console interface for a machine learning model named 'LoanRisk'. The top section shows the model's status as 'Active' with a green checkmark. Below this, the 'Test your model' section is visible, featuring a table with input fields for 'income' (50000), 'age' (30), and 'loan' (20000). To the right of the input fields, the 'Predict label' dropdown is set to '0'. The 'Predict' button is highlighted with a red box. The bottom right corner shows the 'Prediction result' section with the 'Selected label' as '0' and the 'Confidence score' as 0.1017036065459251.

Name	ID	Status	Models	Region	Monitoring	Most recent monitoring job	Most recent alerts	Last updated	API	Notification	Labels	Encryption
LoanRisk	1904890700982386688	Active	0	us-central1	Disabled	—	—	Dec 8, 2021, 12:01:32 AM	Sample request			Google-managed key

Feature column name	Type	Required or optional	Value	Local feature importance
income	Text	Required	50000	0.003289745189249516
age	Text	Required	30	-0.4896751414053142
loan	Text	Required	20000	-0.3588534533046186

Predict label

Prediction result

Selected label: 0

Baseline prediction value: 0.05305759981274605

Confidence score: 0.1017036065459251

PREDICT **RESET**

Congratulations!

You can now use Vertex AI to:

- Upload a dataset.
- Train a model with AutoML.
- Evaluate the model performance.
- Deploy the trained AutoML model to an endpoint.
- Get predictions.

To learn more about different parts of Vertex AI, refer to the [Vertex AI documentation](#).

End your lab

When you have completed your lab, click **End Lab**. Google Cloud Skills Boost removes the resources you've used and cleans the account for you.

You will be given an opportunity to rate the lab experience. Select the applicable number of stars, type a comment, and then click **Submit**.

The number of stars indicates the following:

- 1 star = Very dissatisfied
- 2 stars = Dissatisfied
- 3 stars = Neutral
- 4 stars = Satisfied
- 5 stars = Very satisfied

You can close the dialog box if you don't want to provide feedback.

For feedback, suggestions, or corrections, please use the **Support** tab.

Copyright 2022 Google LLC All rights reserved. Google and the Google logo are trademarks of Google LLC. All other company and product names may be trademarks of the respective companies with which they are associated.