

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

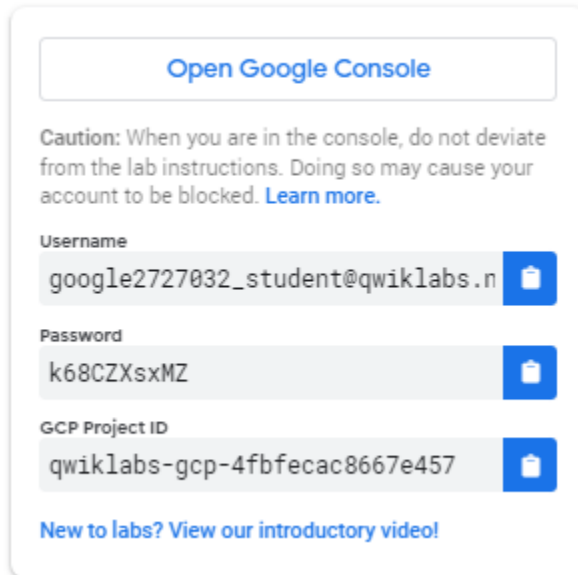
To complete this lab, you need:

- 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.

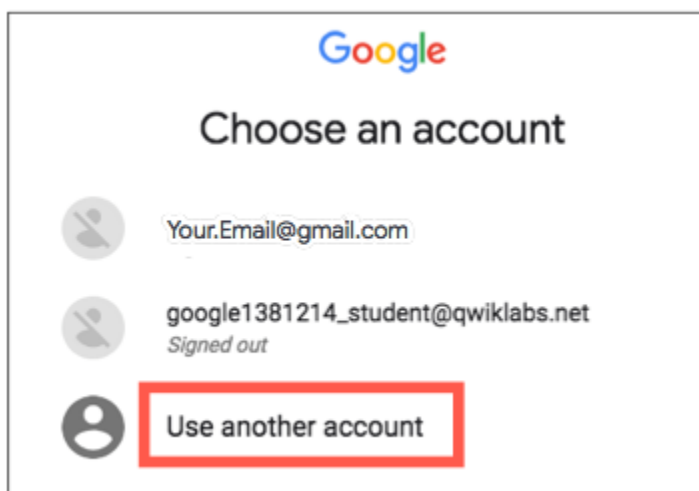


A panel with a white background and a thin grey border. At the top is a button labeled "Open Google Console" in blue text. Below it is a caution message: "Caution: When you are in the console, do not deviate from the lab instructions. Doing so may cause your account to be blocked. [Learn more.](#)". Underneath are three input fields, each with a label and a value, and a blue button with a clipboard icon to the right of each value. The first field is labeled "Username" and contains "google2727032_student@qwiklabs.n". The second is labeled "Password" and contains "k68CZXsxMZ". The third is labeled "GCP Project ID" and contains "qwiklabs-gcp-4fbfecac8667e457". At the bottom is a link: "New to labs? View our introductory video!"

2. 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.

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



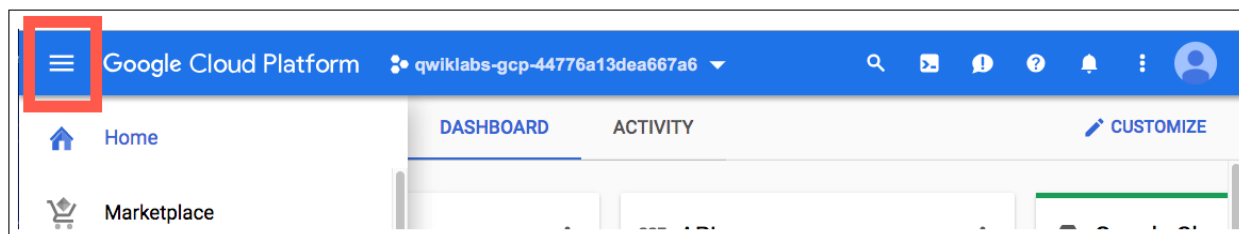
4. Paste the username that you copied from the Connection Details panel. Then copy and paste the password.

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).

5. 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. Give dataset a name **LoanRisk**.
4. For the data type and objective, click **Tabular**, and then select **Regression/classification**.
5. Click **Create**.

Google Cloud

qwiklabs-gcp-01-48ac4628ad22

Search (/) for resources, docs, products, and more

Search

S

Vertex AI

Dashboard

LEARN

Vision

Speech

DATA

Feature Store

Datasets

Labeling tasks

MODEL DEVELOPMENT

Training

Experiments

Metadata

DEPLOY AND USE

Model Registry

Online prediction

Marketplace

Get started with Vertex AI

Vertex AI empowers machine learning developers, data scientists, and data engineers to take their projects from ideation to deployment, quickly and cost-effectively. [Learn more about Vertex AI](#)

ENABLE ALL RECOMMENDED APIS

SHOW API LIST

Colab Enterprise

Model Garden

Tutorials

Try an interactive tutorial to learn how to train, evaluate, and deploy a Vertex AI AutoML or custom-trained model.

VIEW TUTORIALS

Vertex AI

Create dataset

Dataset name *

LoanRisk

Can use up to 128 characters.

Select a data type and objective

First select the type of data your dataset will contain. Then select an objective, which is the outcome that you want to achieve with the trained model.

IMAGE

TABULAR

TEXT

VIDEO

Regression/classification

Predict a target column's value. Supports tables with hundreds of columns and millions of rows.

Forecasting

Predict the likelihood of certain events or demand.

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`

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.

Vertex AI

Vision

Speech

DATA

Feature Store

Datasets

Labeling tasks

MODEL DEVELOPMENT

Training

Experiments

Metadata

DEPLOY AND USE

Model Registry

Online prediction

Marketplace

← LoanRisk

SOURCEANALYZE

Add data to your dataset

Before you begin, review the data guide to make sure your data is formatted correctly and optimized for the best results.

[VIEW DATA GUIDE](#)

Select a data source

- **CSV file:** Can be uploaded from your computer or on Cloud Storage. [Learn more](#)
- **BigQuery:** Select a table or view from BigQuery. [Learn more](#)

☐ Upload CSV files from your computer

☒ Select CSV files from Cloud Storage

☐ Select a table or view from BigQuery


Select CSV files from Cloud Storage

Enter the Cloud Storage path to one or more CSV files. Data from multiple files will be referenced as one dataset.


Import file path *

BROWSE

Summary



\$625,000



\$975,000

You can build two model types with tabular data. The model type is automatically chosen based on the data type of your target column.

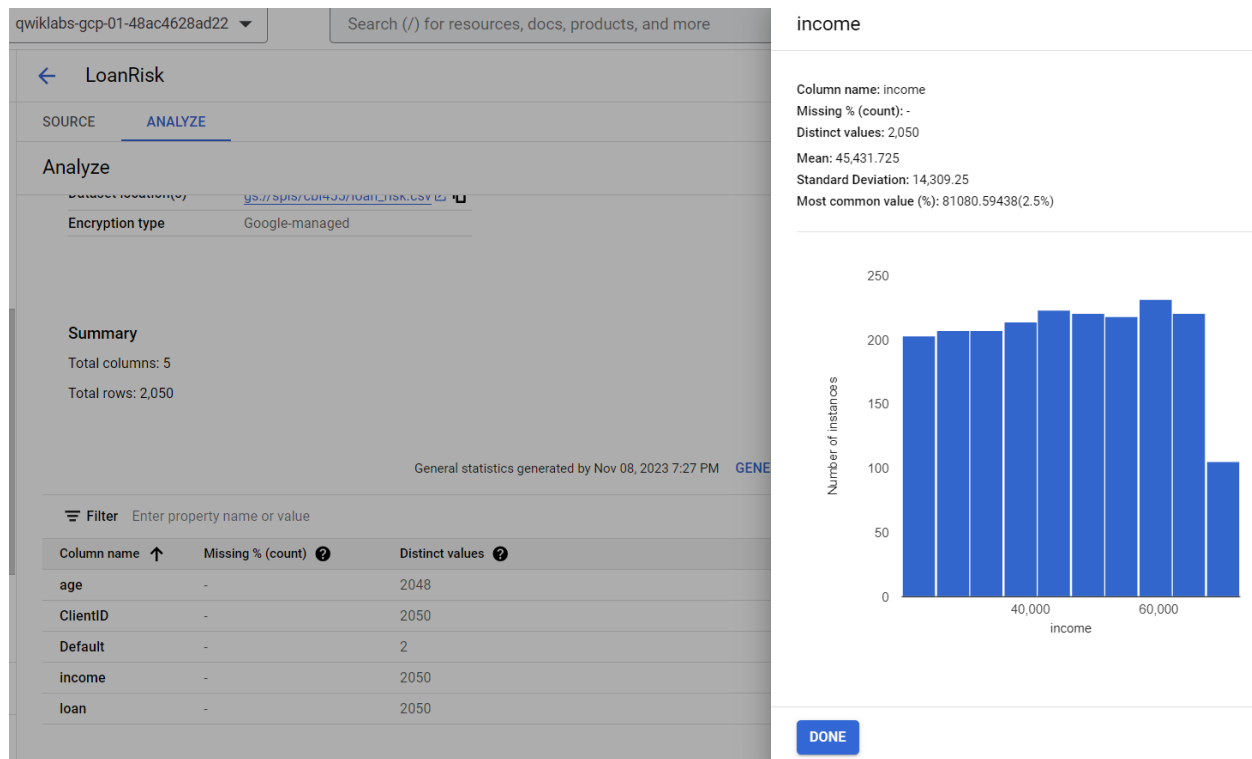
- **Regression models** predict a numeric value. For example, predicting home prices or consumer spending.
- **Classification models** predict a category from a fixed number of categories. Examples include predicting whether an email is spam or not, or classes a student might be interested in attending.

The screenshot shows the Vertex AI console interface for a dataset named 'LoanRisk'. The left sidebar contains navigation links for Vision, Speech, DATA, and MODEL DEVELOPMENT. The 'DATA' section is expanded, showing 'Datasets' as the selected option. The main panel is titled 'LoanRisk' and has tabs for 'SOURCE' and 'ANALYZE'. The 'ANALYZE' tab is active, displaying the 'Analyze' view. This view includes a 'Properties' section with details like 'Created' (Nov 08, 2023 7:16 PM), 'Dataset format' (CSV), 'Dataset location(s)' (gs://spls/cbl455/loan_risk.csv), and 'Encryption type' (Google-managed). Below this is a 'Summary' section showing 'Total columns: 5' and 'Total rows: -'. A status indicator at the bottom right of the main panel says 'GENERATING STATISTICS...'. At the bottom, there is a table with columns 'Column name', 'Missing % (count)', and 'Distinct values'. The table has one row for the 'age' column, showing 0% missing and 1 distinct value.

Column name	Missing % (count)	Distinct values
age	0	1

(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.

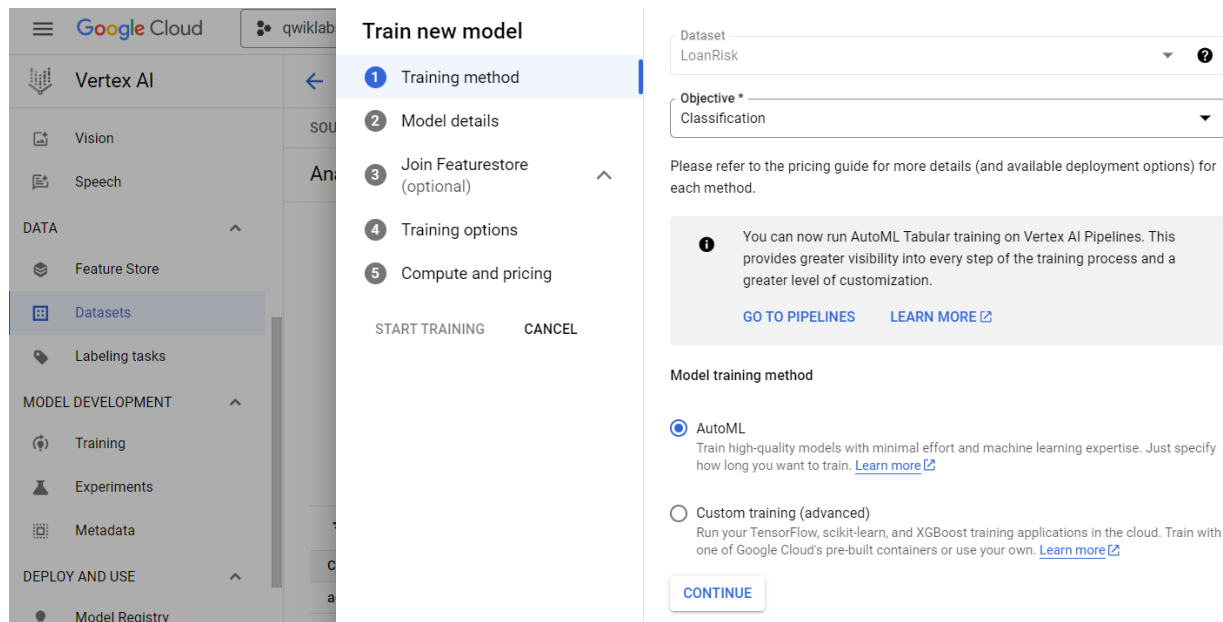
- Click **Train new model** and select **Other**.

Training method

1. The dataset is already named **LoanRisk**.
2. 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.

3. 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**.
5. For Add features, click **Continue**.

Train new model

☒ Training method

2 Model details

3 Join Featurestore
(optional) ^

4 Training options

5 Compute and pricing

START TRAINING

CANCEL

☒ Train new model

Creates a new model group and assigns the trained model as version 1

☐ Train new version

Trains model as a version of an existing model

Name *

LoanRisk

Description

Target column *

Default



☐ Export test dataset to BigQuery

Data split

☒ Random

80% of your data is randomly assigned for training, 10% for validation, and 10% for testing

● Training: 80%

● Validation: 10%

● Test: 10%



☐ Manual

You assign each data row for training, validation, and testing. [Learn more](#)

☐ Chronological

The earliest 80% of your data is assigned to training, the next 10% for validation and the latest 10% for testing. This option requires a Time column in your dataset. [Learn more](#)



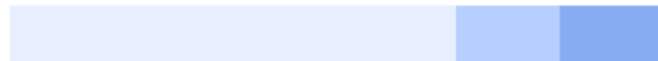
Training 80%



Validation 10%



Testing 10%



Train new model

☒ Training method

☒ Model details

3 Join Featurestore
(optional) ^

4 Training options

5 Compute and pricing

START TRAINING

CANCEL

Add features **EXPERIMENTAL**

Choosing featurestore and entity type will generate a table of features that you can join to your AutoML table.

[ADD JOIN](#)

[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**.

Train new model

☒ Training method

☒ Model details

☒ Join Featurestore (optional)

☒ Training options

☒ Compute and pricing

START TRAINING

CANCEL

Filter Enter property name or value

<input type="checkbox"/>	Column name ↑	Transformation	Missing % (count) ?	Distinct values ?	Correlation w/ target ?	
<input type="checkbox"/>	age	Numeric ▼	-	2048	-	⊖
<input type="checkbox"/>	ClientID	Numeric ▼	-	2050	-	⊕
<input type="checkbox"/>	Default		-	2	-	
						Target
<input type="checkbox"/>	income	Numeric ▼	-	2050	-	⊖
<input type="checkbox"/>	loan	Numeric ▼	-	2050	-	⊖

Total 4 feature columns are included in the training

Weight column

Select a column to specify how to weight each row of the training data. By default, each row of your training data is weighted equally. ?

Optimization objective *

☒ AUC ROC

Distinguish between classes

☐ Log loss

Keeps prediction probabilities as accurate as possible

☐ AUC PRC

Maximize precision-recall for the less common class

☐ Precision at recall

Maximize precision for the less common class

☐ Recall at precision

Maximize recall for the less common class

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**.

Train new model

- ✓ Training method
- ✓ Model details
- ✓ Join Featurestore (optional)
- ✓ Training options
- 5 Compute and pricing**

START TRAINING

CANCEL

Enter the **maximum** number of node hours you want to spend training your model.

You can train for as little as 1 node hour. You may also be eligible to train with free node hours. [Pricing guide](#)

Budget *

1

Maximum node hours



Estimated completion: Nov 8, 2023 9 PM GMT-8

☒ Enable early stopping

Ends model training when no more improvements can be made and refunds leftover training budget. If early stopping is disabled, training continues until the budget is exhausted.

Vertex AI

Vision

Speech

DATA

Feature Store

Datasets

Labeling tasks

MODEL DEVELOPMENT

Training

Experiments

Metadata

DEPLOY AND USE

Training

TRAIN NEW MODEL

REFRESH

LEARN

TRAINING PIPELINES

CUSTOM JOBS

HYPERPARAMETER TUNING JOBS

NAS JOBS

Training pipelines are the primary model training workflow in Vertex AI. You can use training pipelines to create an AutoML-trained model or a custom-trained model. For custom-trained models, training pipelines orchestrate custom training jobs and hyperparameter tuning with additional steps like adding a dataset or uploading the model to Vertex AI for prediction serving. [Learn More](#)

Region
us-central1 (Iowa)

Filter

Enter a property name

Name	ID	Status	Job type	Model type	Duration	Last updated	Created	
LoanRisk	7125659775550881792	Training	Training pipeline	Tabular classification	1 hr 9 min	Nov 8, 2023, 7:37:12 PM	Nov 8, 2023, 7:36:49 PM	

Training

+

TRAIN NEW MODEL

REFRESH

LEARN

TRAINING PIPELINES

CUSTOM JOBS

HYPERPARAMETER TUNING JOBS

NAS JOBS

Training pipelines are the primary model training workflow in Vertex AI. You can use training pipelines to create an AutoML-trained model or a custom-trained model. For custom-trained models, training pipelines orchestrate custom training jobs and hyperparameter tuning with additional steps like adding a dataset or uploading the model to Vertex AI for prediction serving. [Learn More](#)

Region

us-west1 (Oregon)

?

Filter

Enter a property name

?

⋮

Name	ID	Status	Job type	Model type	Duration ?	Last updated ↓	Created	
LoanRisk	6677053740018565120	✔ Finished	Training pipeline	Tabular classification	1 hr 45 min	Nov 8, 2023, 10:47:30 PM	Nov 8, 2023, 9:01:31 PM	⋮

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.

Vertex AI

DATA

Feature Store

Datasets

Labeling tasks

MODEL DEVELOPMENT

Training

Experiments

Metadata

DEPLOY AND USE

Model Registry

Online prediction

Batch predictions

Vector Search

Marketplace

LoanRisk

Version 1

VIEW DATASET

EXPORT

EVALUATE

DEPLOY & TEST

BATCH PREDICT

VERSION DETAILS

untitled_5822696443242270621

COMPARE

CREATE EVALUATION

Labels

Filter

All labels0.991

10.937

00.994

Evaluation details

Confidence threshold0.5

All labels

PR AUC0.991

ROC AUC0.991

Log loss0.1

F1 score0.9739583

Micro-F10.9739583

Macro-F10.9484951

Precision97.4%

Recall97.4%

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.

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

The precision/recall curve

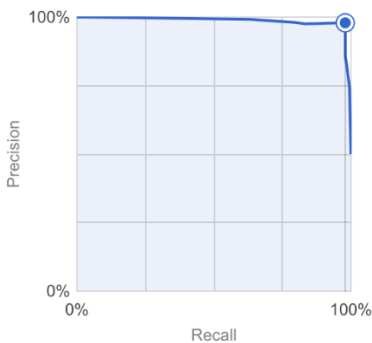
Confidence threshold ? 0.5

All labels

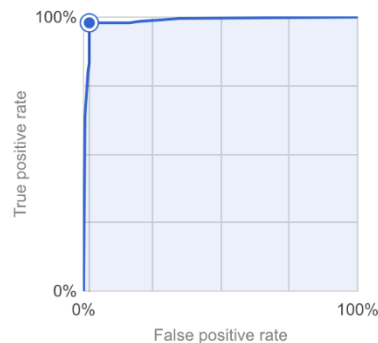
PR AUC ?	0.984
ROC AUC ?	0.986
Log loss ?	0.11
F1 score ?	0.9791667
Precision ?	97.9%
Recall ?	97.9%
Created	Dec 7, 2021, 6:27:33 PM

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.

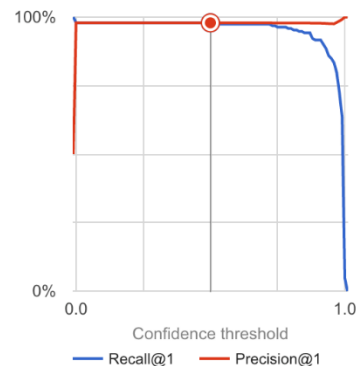
Precision-recall curve ?



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.

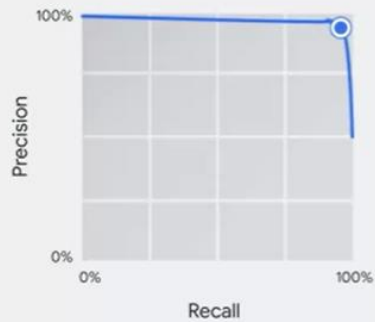
Confidence threshold  0.5

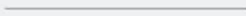
Determines how a machine learning model counts the positive cases

 Higher threshold Increases precision, decrease recall

 Lower threshold decreases precision, increases recall

Precision-recall curve




Confidence threshold  0

Determines how a machine learning model counts the positive cases.

 Higher threshold Increases precision, decrease recall

 Lower threshold decreases precision, increases recall

 Zero threshold produces the highest recall of 100%, and the lowest precision of 50%

Precision-recall curve



The model predicts that 100% of loan applicants will be able to repay a loan they take out. However, in actuality, only 50% of people were able to repay the loan.

Confidence threshold

1

Determines how a machine learning model counts the positive cases.



Higher threshold increases precision, decrease recall

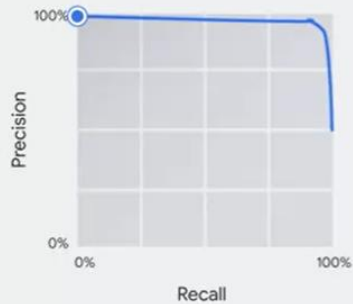


Lower threshold decreases precision, increases recall



A threshold of 1 produces the highest precision of 100%, with the lowest recall of 1%.

Precision-recall curve



Of all the people who were predicted to repay the loan, 100% of them actually did. However, you rejected 99% of loan applicants by only offering loans to 1% of them.

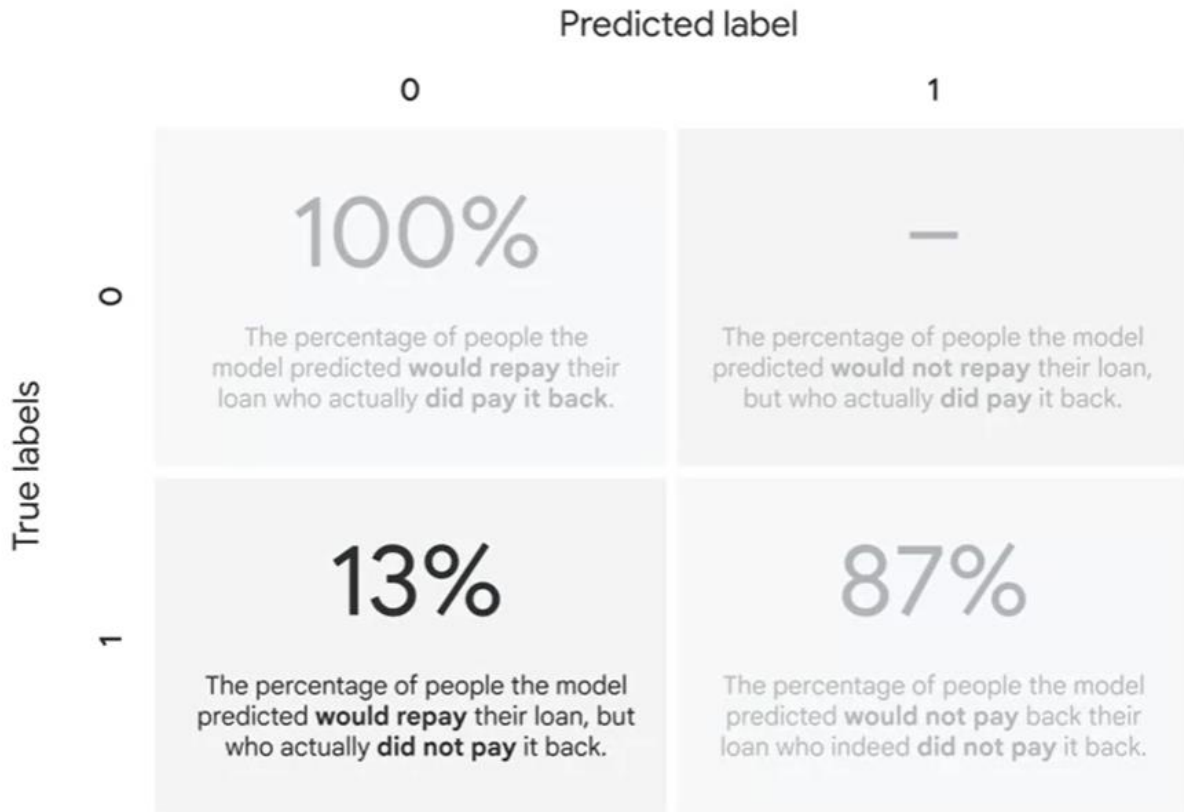
The confusion matrix

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%



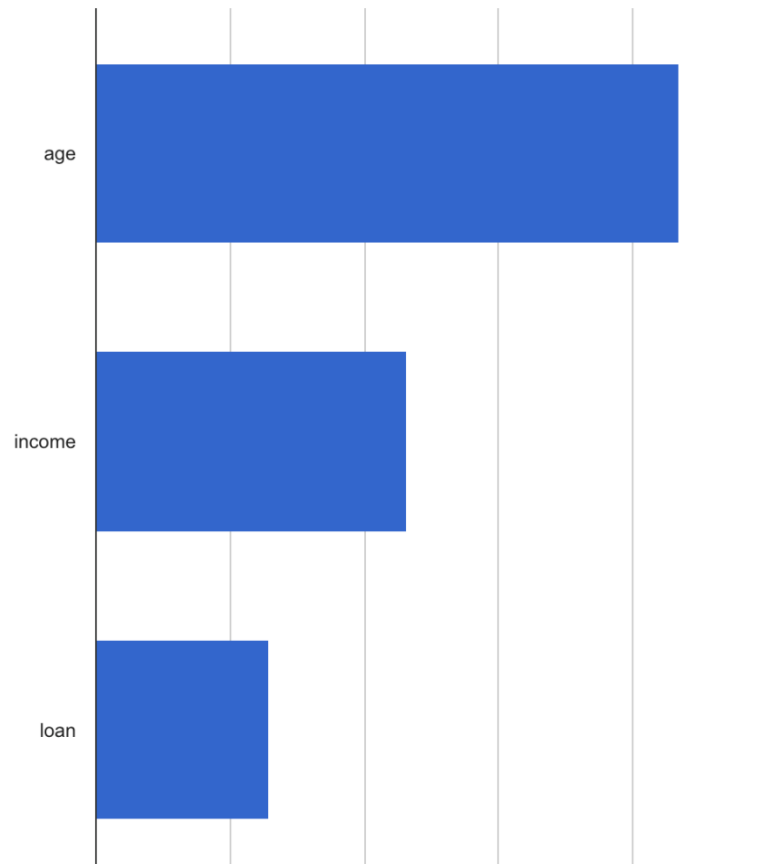
The confusion matrix shows that your initial model is able to predict 100% of the repay examples and 87% of the default examples in your test set correctly, which is not too bad.

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, click **Deploy & test**, and then click **Deploy to Endpoint**.
2. For **Endpoint name**, type **LoanRisk**
3. Click **Continue**.

Vertex AI

DATA

Feature Store

Datasets

Labeling tasks

MODEL DEVELOPMENT

Training

Experiments

Metadata

DEPLOY AND USE

Model Registry

Online prediction

Batch predictions

Vector Search

Marketplace

LoanRisk > Version 1

VIEW DATASET

EXPORT

EVALUATE

DEPLOY & TEST

BATCH PREDICT

VERSION DETAILS

Use your edge-optimized model

Container

Export your model as a TF Saved Model to run on a Docker container.

Deploy your model

Endpoints are machine learning models made available for online prediction requests. Endpoints are useful for timely predictions from many users (for example, in response to an application request). You can also request batch predictions if you don't need immediate results.

DEPLOY TO ENDPOINT

Name	ID	Status	Models	Deployment resource pool	Region	Monitoring	Most recent monitoring job	Most recent alerts
No active endpoints containing this model								

Deploy to endpoint

1 Define your endpoint

2 Model settings

3 Model monitoring

4 Monitoring objectives

DEPLOY

CANCEL

☒ Create new endpoint ☐ Add to existing endpoint

Endpoint name *

LoanRisk



Location



Some locations have been restricted due to a policy set by your organization. [Learn more about restricting locations.](#)

Region

us-west1 (Oregon)



Access

Determines how your endpoint can be accessed. By default, endpoints are available for prediction serving through a REST API. Endpoint access can't be changed after the endpoint is created.

☒ Standard

Makes the endpoint available for prediction serving through a REST API. AutoML and custom-trained models can be added to standard endpoints.

☐ Private

Create a private connection to this endpoint using a VPC network and [private services access](#). Only custom-trained and tabular models can be added to private endpoints. [Learn more](#)

ADVANCED OPTIONS

CONTINUE

Model settings and monitoring

1. Leave the traffic splitting settings as-is.

Deploy to endpoint

- ✓ Define your endpoint
- 2 Model settings
- 3 Model monitoring
- 4 Monitoring objectives

DEPLOY CANCEL

Model settings ?

New model

LoanRisk (Version 1)

Traffic split *

100

% ?

Compute resources

Choose how compute resources will serve prediction traffic to your model

- **Autoscaling:** If you set a minimum and maximum, compute nodes will scale to meet traffic demand within those boundaries
- **No scaling:** If you only set a minimum, then that number of compute nodes will always run regardless of traffic demand (the maximum will be set to minimum)

Once scaling settings are set, they can't be changed unless you redeploy the model. [Pricing guide](#)

Minimum number of compute nodes *

1

Default is 1. If set to 1 or more, then compute resources will continuously run even without traffic demand. This can increase cost but avoid dropped requests due to node initialization.

Maximum number of compute nodes (optional)

Enter a number equal to or greater than the minimum nodes. Can reduce costs but may cause reliability issues for high traffic.

✓ ADVANCED SCALING OPTIONS

2. For **Machine type**, select **e2-standard-8, 8 vCPUs, 30 GiB memory**.
3. For **Explainability Options**, click **Feature attribution**.
4. Click **Done**.
5. Click **Continue**.

Deploy to endpoint

- ✓ Define your endpoint
- 2 Model settings
- 3 Model monitoring
- 4 Monitoring objectives

DEPLOY CANCEL

✓ ADVANCED SCALING OPTIONS

Machine type *

e2-standard-8, 8 vCPUs, 32 GiB memory



Logging

Logging settings are permanent for this endpoint, and Logging charges will apply. To change your logging preference in the future, create a new endpoint. [Learn more](#)

- ☒ Enable access logging for this endpoint
- ☐ Disable container logging for this endpoint

Explainability options

- ☐ No explainability
- ☒ Feature attribution

Sampled Shapley 16 samples

EDIT

- ☐ Example-based explanation

It may take several minutes for endpoint settings to take effect.

DONE

ADD A MODEL

CONTINUE

6. In **Model monitoring**, click **Continue**.

Deploy to endpoint

- ✓ Define your endpoint
- ✓ Model settings
- 3 Model monitoring**
- 4 Monitoring objectives

DEPLOY CANCEL

Model monitoring

Models used in production require continuous monitoring to ensure that they perform as expected. Use model monitoring to track training-serving skew or prediction drift, then set up alerts to notify you when thresholds are crossed. [Learn more](#)

Model monitoring supports AutoML tabular and custom-trained models and incurs additional charges. [Learn more](#)

☒ Enable model monitoring for this endpoint

Monitoring job display name *
mm_LoanRisk_20231196550

Define the display name of the monitoring job

Monitoring interval *
24 hours

How frequently monitoring jobs will run

Monitoring data window hours

The length of the window to pull prediction traffic from. If left blank it will default to the monitoring interval. A short window can be good for endpoints with high prediction traffic, while a long window is useful for endpoints with low prediction traffic.

Notification emails *
student-02-894104d1d9ae@qwiklabs.net

All notifications, including status changes and alert events, are sent via email.

Notification channels

Sampling rate

Sampling rate *
10 %

7. In **Model objectives > Training data source**, select **Vertex AI dataset**.
8. Select your dataset from the drop down menu.
9. In **Target column**, type **Default**
10. Leave the remaining settings as-is and click **Deploy**.

Deploy to endpoint

- ✓ Define your endpoint
- ✓ Model settings
- ✓ Model monitoring
- 4 Monitoring objectives

DEPLOY

CANCEL

Monitoring objective

- ☒ Training-serving skew detection
Training-serving skew occurs when the feature data distribution in production is different from the feature data distribution in model training
- ☐ Prediction drift detection
Prediction drift occurs when feature data distribution in production changes significantly over time

Training-serving skew detection

Training data source

To detect training-serving skew, the monitoring job needs to compare the model training data to the dataset used to train the model

- ☐ Cloud Storage bucket
- ☐ BigQuery table
- ☒ Vertex AI dataset

Vertex AI dataset *

LoanRisk

Target column

The column name from the training data that the model is trained to predict. This column will be ignored when tracking feature skew.

Target column *

Default

Alert thresholds (Optional)

Determines which features to monitor and distance between the input feature distribution and its baseline. At the end of each monitoring run, if any thresholds

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.

←

LoanRisk > Version 1 ▾

VIEW DATASET

EXPORT

EVALUATE

DEPLOY & TEST

BATCH PREDICT

VERSION DETAILS

Export your model as a TF Saved Model to run on a Docker container.

Deploy your model

Endpoints are machine learning models made available for online prediction requests. Endpoints are useful for timely predictions from many users (for example, in response to an application request). You can also request batch predictions if you don't need immediate results.

DEPLOY TO ENDPOINT

Name	ID	Status	Models	Deployment resource pool	Region	Monitoring	Most recent monitoring job
LoanRisk	410276166834847744	✔ Active	0	—	us-west1	Disabled	—

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.

QWIKLABS

15 seconds

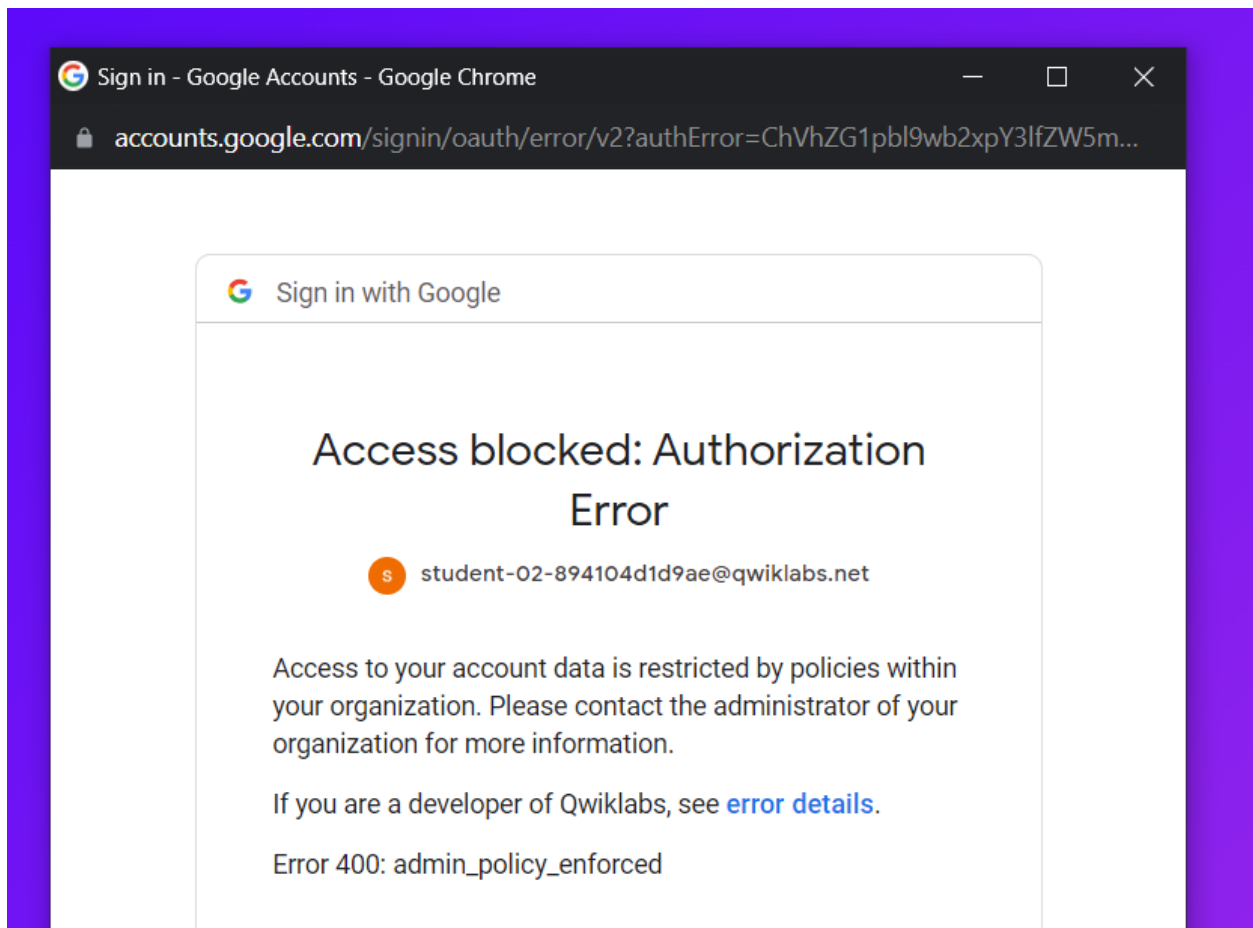


Signed in with Google

a21haWwuY29tliwiZW1haWwiOiJydWIAZnVsbHN0YWNrbW

Copy

Note: This token will only be available for about **60 seconds**, so copy and 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:

```
export AUTH_TOKEN="INSERT_SML_BEARER_TOKEN"
```

3. Download the lab assets:

```
gcloud storage cp gs://spl5/cbl455/cbl455.tar.gz .
```

4. Extract the lab assets:

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

5. Create an ENDPOINT environment variable:

```
export ENDPOINT=https://sml-api-vertex-kjyo252taq-uc.a.run.app/vertex/predict/tabular\_classification
```

6. Create a INPUT_DATA_FILE environment variable:

```
export INPUT_DATA_FILE="INPUT-JSON"
```

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
```

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
```

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 Vertex AI console. At the top, a table lists deployed models. The first model, 'LoanRisk', is highlighted with a red box around its 'Status' column, which shows a green checkmark and the word 'Active'. Below this, the 'Test your model' section is shown. It features a table with three input fields for 'income' (50000), 'age' (30), and 'loan' (20000), each with a red box around the input area. To the right of these inputs is a 'Predict label' section with a 'Prediction result' dropdown menu showing '0'. Below the prediction result, the baseline prediction value and confidence score are displayed. At the bottom left, there are 'PREDICT' and 'RESET' buttons, with 'PREDICT' highlighted by a red box.

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**.