

AMAZON SAGEMAKER AI- FROM DATA WRANGLING TO REST API

1. Open Amazon SageMaker AI

The screenshot shows the Amazon SageMaker AI service page. At the top, there's a search bar with 'Amazon Sagemaker AI' and a navigation bar with AWS logo, user info, and region selection (Asia Pacific (Mumbai)). The main content area is titled 'Services' and lists three items: 'Amazon SageMaker AI' (selected), 'AWS Lake Formation', and 'Amazon SageMaker'. Below this is a section titled 'Features' with 'SageMaker Ground Truth', 'Autopilot', and 'SageMaker Studio'. On the right, there's a sidebar with 'Create application' and 'Cost breakdown' sections, and a bottom footer with links like 'Privacy', 'Terms', and 'Cookie preferences'.

2. Click on Domains under Amazon Sagemaker AI and create a domain

The screenshot shows the 'Domains' page under the Amazon SageMaker AI service. The left sidebar has sections for 'Getting started', 'What's new', 'Applications and IDEs' (Studio, Canvas, RStudio, Notebooks, Partner AI Apps), 'Admin configurations' (Domains, Role manager, Images, Lifecycle configurations), and 'JumpStart' (Foundation models). The main content area has a message about introducing domain-level resource visibility. It shows a table header for 'Domains (0) Info' with columns: Name, Id, Status, Created on, and Modified on. Below the table, it says 'No domains' and 'To add a domain, choose Create domain.' The bottom footer includes links for 'CloudShell' and 'Feedback'.

Click on set up for single user

The screenshot shows the 'Set up SageMaker Domain' page. On the left, there's a sidebar with navigation links like 'Getting started', 'What's new', 'Applications and IDEs' (Studio, Canvas, RStudio, Notebooks, Partner AI Apps), 'Admin configurations' (Domains, Role manager, Images, Lifecycle configurations), 'JumpStart' (Foundation models), and 'CloudShell'.

The main content area has two sections: 'Set up for single user (Quick setup)' and 'Set up for organizations'. The 'Set up for single user' section is selected, showing a list of features: New IAM role with AmazonSageMakerFullAccess policy, Public internet access, and standard encryption; SageMaker Studio - New, and SageMaker Studio Classic integrations; Sharable SageMaker Studio Notebooks; SageMaker Canvas; and IAM Authentication. It also includes a note: 'Perfect for single user domains and first time users looking to get started with SageMaker.' Below this is a 'Set up' button.

The 'Set up for organizations' section is also present, listing more advanced features: Advanced network security, and data encryption; SageMaker Studio - New, SageMaker Studio Classic, RStudio, and Code Editor Based on Code-OSS, Visual Studio Code Open Source Integrations; SageMaker Studio Projects, and Jumpstart; SageMaker Canvas, and Amazon services integrations; and IAM, or IAM Identity Center (successor to AWS SSO). It includes a note: 'Better for admins with large user groups, but you can always update your account configuration settings later if you want to do a quick setup now.'

At the bottom right of the main content area, there are links for 'Privacy', 'Terms', and 'Cookie preferences'.

The screenshot shows the 'Domain: QuickSetupDomain-20250501T153207' page. At the top, there's a green banner with the message: 'The SageMaker Domain is ready. Choose your user name, then choose Launch app to get started.'

The main content area has tabs for 'Domain settings' (selected), 'User profiles', 'Space management', 'App Configurations', 'Environment', and 'Resources'. Under 'Domain settings', there's a 'General settings' section with fields for Name (QuickSetupDomain-20250501T153207), Status (Ready), Domain ID (d-zzign3umvng0), Created (Thu May 01 2025 15:32:26 GMT+0530 (India Standard Time)), and Last modified (Thu May 01 2025 15:35:16 GMT+0530 (India Standard Time)).

Below this is a 'Domain rules' section with a table:

Rule type	Application type	Rule action	Resource
No domain rules			

At the bottom right of the main content area, there's a 'Manage rules' button.

At the very bottom, there are links for 'CloudShell', 'Feedback', 'Privacy', 'Terms', and 'Cookie preferences'.

3.Create a user in the domain

The screenshot shows the Amazon SageMaker AI console interface. On the left, a sidebar navigation includes sections for Getting started, What's new, Applications and IDEs (Studio, Canvas, RStudio, Notebooks, Partner AI Apps), Admin configurations (Domains, Role manager, Images, Lifecycle configurations), and JumpStart (Foundation models, CloudShell, Feedback). The main content area displays the 'Domain details' for 'QuickSetupDomain-20250501T153207'. The 'User profiles' tab is selected. A success message at the top states 'User profile was successfully created.' Below it, the 'General settings' section shows the domain name as 'QuickSetupDomain-20250501T153207', status as 'Ready', and domain ID as 'd-zzign3umvrmgo'. The 'Domain rules' section indicates 'No domain rules'.

Amazon SageMaker AI <

Getting started
What's new

▼ Applications and IDEs

Studio
Canvas
RStudio
Notebooks
Partner AI Apps NEW

▼ Admin configurations

Domains
Role manager
Images
Lifecycle configurations

SageMaker AI dashboard
Search

▼ JumpStart

Foundation models
CloudShell Feedback

QuickSetupDomain-20250501T153207

Domain details

Configure and manage the domain.

Domain settings User profiles Space management App Configurations Environment Resources

User profiles Info

A user profile represents a single user within a domain. It is the main way to reference a user for the purposes of sharing, reporting, and other user-oriented features.

Search users

Name Modified on Created on

default-20250501T153207 May 01, 2025 10:05 UTC May 01, 2025 10:05 UTC

Launch

Launch app options for user profile default-20250501T153207

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Amazon SageMaker AI <

Getting started
What's new

▼ Applications and IDEs

Studio
Canvas
RStudio
Notebooks
Partner AI Apps NEW

▼ Admin configurations

Domains
Role manager
Images
Lifecycle configurations

SageMaker AI dashboard
Search

▼ JumpStart

Foundation models

User profile was successfully created.

QuickSetupDomain-20250501T153207

Domain details

Configure and manage the domain.

Domain settings User profiles Space management App Configurations Environment Resources

General settings Info

Name: QuickSetupDomain-20250501T153207 Status: Ready Domain ID: d-zzign3umvrmgo

Created: Thu May 01 2025 15:32:26 GMT+0530 (India Standard Time) Last modified: Thu May 01 2025 15:35:16 GMT+0530 (India Standard Time) VPC: vpc-024c9e87cece821cd

Domain rules

No domain rules

No domain rules to display.

Manage rules

https://ap-south-1.console.aws.amazon.com/sagemaker/home?region=ap-south-1#/studio-landing

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

4. After creating the domain and user go to studio under Amazon Sagemaker AI and open studio with the user profile we created in the domain.

The screenshot shows the Amazon SageMaker AI interface. On the left, there's a sidebar with navigation links: 'Getting started', 'What's new', 'Applications and IDEs' (with 'Studio' selected), 'Admin configurations', 'JumpStart', and 'JumpStart' again. The main content area features the 'SageMaker Studio' logo and a callout for 'Get Started' with a dropdown menu for 'Select user profile' (set to 'default-1746094278090') and an 'Open Studio' button. Below this, there's a 'What's new' section with a card for 'Amazon SageMaker Projects now allows you to reuse names of previously deleted Projects' (published 29 August 2024) and a 'Pricing (US)' section. At the bottom, the URL is https://ap-south-1.console.aws.amazon.com/sagemaker/home?region=ap-south-1#/studio/open/d-zz1on3umvmao/default-1746094278090, and the footer includes links for 'Privacy', 'Terms', and 'Cookie preferences'.

5. After opening Studio click on Data wrangler and open it.

The screenshot shows the SageMaker Studio Home page. The sidebar on the left has sections for 'JupyterLab', 'RStudio', 'Canvas', 'Code Editor', 'MLflow', 'Partner AI Apps' (with a 'New' badge), 'Home', 'Running instances', 'Compute' (with a dropdown arrow), 'Data' (with a dropdown arrow), 'Feature Store', 'Store & share features', 'EMR Clusters', 'Compute clusters for batch work', and 'Auto ML'. The 'Data' section is currently active, with 'Data Wrangler' highlighted. The main content area features an 'Onboarding plan' with three cards: 'Take the tour', 'Access your EFS data in JupyterLab and CodeEditor', and 'Access your Studio Classic apps'. Below this, there's a note about switching back to the classic experience if needed, followed by 'Prepare, transform and store data', 'Overview', and a large orange 'jupyter' logo.

Click on Run in Canvas

The screenshot shows the SageMaker Studio interface for the Data Wrangler application. The left sidebar has a 'Compute' section expanded, showing 'Data Wrangler' under 'Data'. The main area title is 'Data Wrangler' with the sub-instruction 'Aggregate, explore and prepare data to build Generative AI or ML solutions in SageMaker Canvas, no code required.' Below the title are four main sections: 'Import data', 'Prepare data', 'Export', and 'Train'. Under 'Import data', it says 'Aggregate data from sources such as Amazon S3, Amazon Athena, Snowflake and 50+ other sources.' Under 'Prepare data', it says 'Explore and transform data with natural language or a visual interface.' Under 'Export', it says 'Create data pipelines using SageMaker Processing jobs and SageMaker Pipelines.' Under 'Train', it says 'Use your data to build custom models with AutoML.' At the top right, there's a 'Provide feedback' button and a 'Learn more' link. In the center, there's a 'Status' indicator showing 'Stopped' and a 'Run in Canvas' button.

Click on Open in Canvas

This screenshot is identical to the previous one, showing the SageMaker Studio Data Wrangler interface. The key difference is the status indicator at the top right, which now shows 'Running' with a green checkmark, indicating that the canvas has been successfully opened.

6.Click on Data wrangler and then import and prepare to import data needed.

The screenshot shows the Data Wrangler interface. On the left is a sidebar with icons for Home, Amazon Q, Data Wrangler (selected), Datasets, My Models, ML Ops, Ready-to-use, Gen AI, Help, and Log out. The main area has a header with a search bar, 'Import data flows', and 'Import and prepare'. Below is a flow diagram: Import data (database icon) → Prepare data (lightbulb icon) → Scale data operations (cogs icon) → Build models (workshop icon). A message says 'You haven't imported and prepared any data'. Below the flow is a text block: 'Import data from over 50 sources, join, transform, and analyze your data using 300+ built-in operators or chat in Data Wrangle flows. Export your prepared data with a single click to build or use ML models.' A dropdown menu under 'Import and prepare' shows 'Dataset type' selected, with 'Tabular (CSV or Parquet files)' and 'Image (PNG and JPG files)' options. The 'Resources' section includes 'Get started' (links to 'What is Amazon SageMaker Data Wrangler?' and 'Get started with Data Wrangler in Canvas'), 'Documentation' (links to 'Pricing in Canvas' and 'AWS technical guide'), and 'What's new' (link to 'Accelerate data preparation for ML with comprehensive data preparation capabilities and a natural language interface').

Select the data source type and upload it

This is a screenshot of the 'Import tabular data' step in the Data Wrangler wizard. It shows a 'Select a data source:' dropdown set to 'Local upload'. Below is a large dashed box for file upload with the text 'Drag a CSV or Parquet file here' and a 'Select files from your computer' button. To the right, a panel shows '1 file ready to import' with 'bankloan.csv' listed, a 'Delete all' button, and a 'Next' button at the bottom.

Click on import data

Import tabular data

← Previewing 1 file Showing the first 100 rows Import settings

If your data has special character delimiters, use the advanced import settings to specify a custom delimiter. [Learn More](#)

bankloan.csv Delete

ID	Age	Experience	Income	Zi
1	25	1	49	91
2	45	19	34	90
3	39	15	11	94
4	35	9	100	94
5	35	8	45	91
6	37	13	29	92

Import settings

Settings apply to all imported files. [Learn more](#)

Dataset name *

bankloan.csv

Sampling

Sample your dataset for faster exploration. Your full dataset will be used for data export or model build. [Learn more](#)

Sampling method * ⓘ

Random

Random sampling ensures that each row has an equal probability of being chosen.

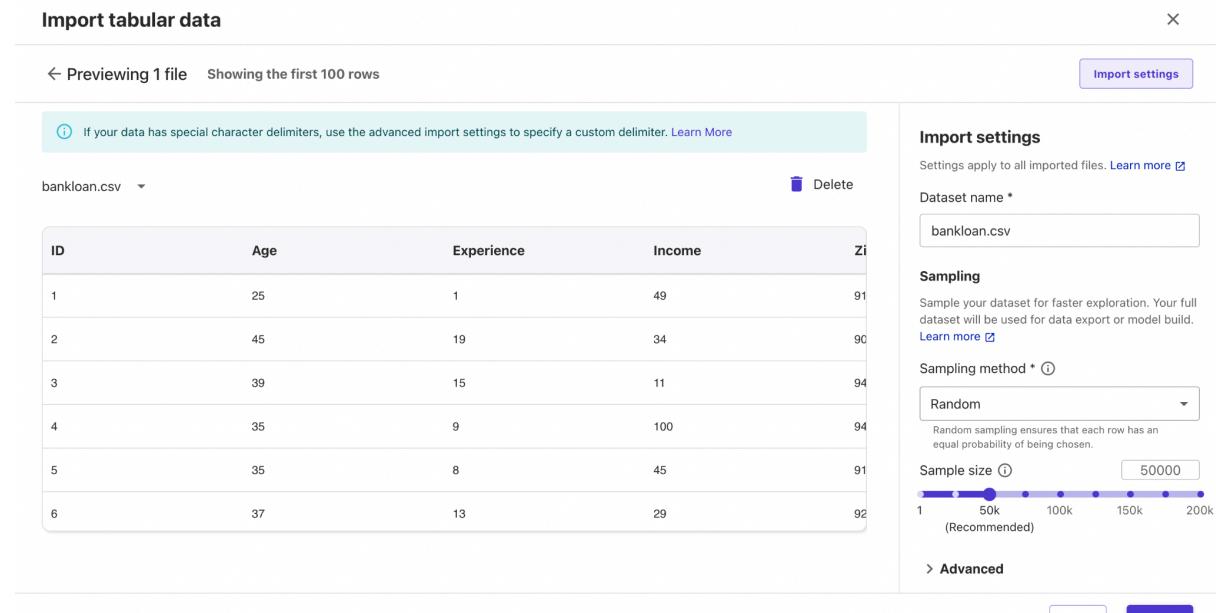
Sample size ⓘ

50000

1 50k 100k 150k 200k (Recommended)

> Advanced

Cancel Back Import



8.After Importing the Data it will appear as a flow- click on get data insights to create a Data Quality and Insights report.

Data Wrangler: Data flow > New data flow 2025-5-1 4:05:03 PM.flow Add data

Step 2: Data types Data flow Data Analyses

Validation complete 0 errors Done Run validation

+ Add transform

Get data insights Identify data quality issues and get recommendations.

Combine data

Create model Export data and start model building in Canvas.

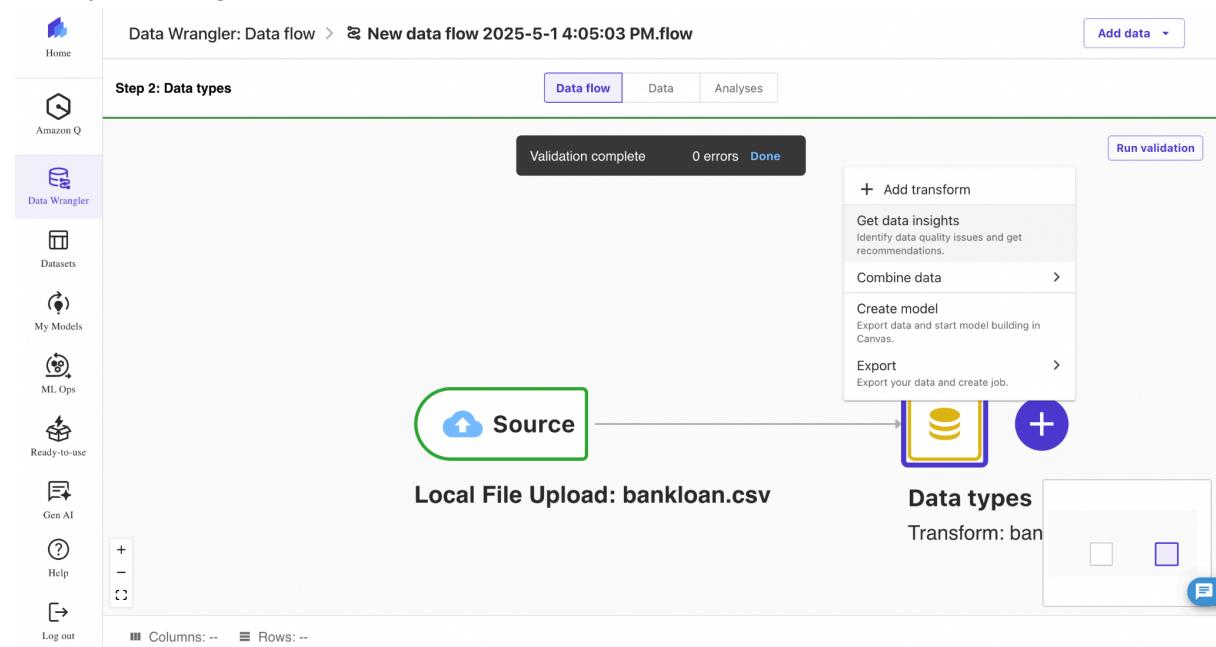
Export Export your data and create job.

Source Local File Upload: bankloan.csv

+

Data types Transform: ban

Columns: -- Rows: --



Click create

9. After Generating the report- go back to the data flow and click on add transform- this will allow you to apply any data preprocessing steps which are needed.

Data Wrangler: Data flow > New data flow 2025-5-1 14:05:03 PM.flow > bankloan.csv

Step 3. Impute Chat for data prep Data flow Data Analyses Create model

Build a data insights report to identify data quality issues, recommends fixes, and estimates feature importance and model accuracy. Get data insights X

ID (long)	Age (long)	Experience (long)	Income (long)
1 - 5000	23 - 67	-3 - 43	8 - 224
1	25	1	49
2	45	19	34
3	39	15	11
4	35	9	100
5	35	8	45
6	37	13	29
7	53	27	72
8	50	24	22

Sampling: 50,000 Columns: 14 Rows: 5,000 Show visualizations

Steps

- + Add transform
- 1. Local File Upload: bankloan.csv
- 2. Data types
- 3. Impute

Replace, drop, or add indicators for missing values. [Learn more.](#)

Transform * ⓘ Fill missing

Input columns * ID x Age x Experience x Income x

Data Wrangler: Data flow > New data flow 2025-5-1 14:05:03 PM.flow > bankloan.csv

Step 3. Fill missing Chat for data prep Data flow Data Analyses Create model

Build a data insights report to identify data quality issues, recommends fixes, and estimates feature importance and model accuracy. Get data insights X

ID (long)	Age (long)	Experience (long)	Income (long)
1 - 5000	23 - 67	-3 - 43	8 - 224
1	25	1	49
2	45	19	34
3	39	15	11
4	35	9	100
5	35	8	45
6	37	13	29
7	53	27	72
8	50	24	22

Sampling: 50,000 Columns: 14 Rows: 5,000 Show visualizations

← Encode categorical

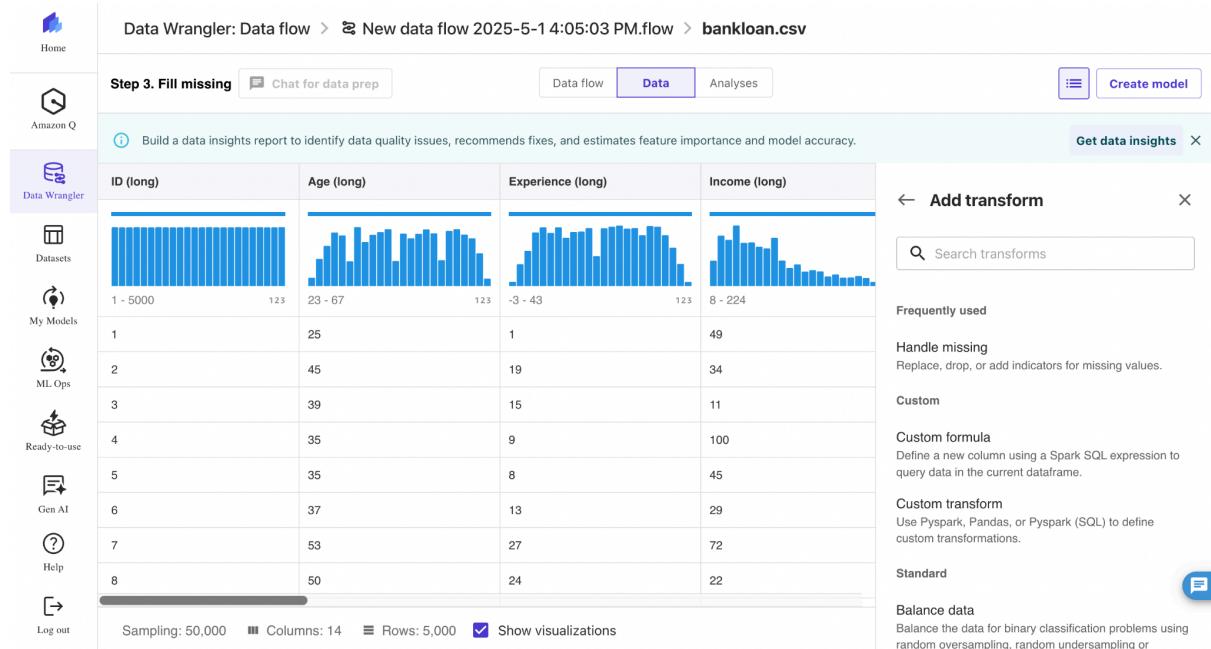
Convert categorical variables to numeric or vector representations. [Learn more.](#)

Transform * ⓘ Ordinal encode

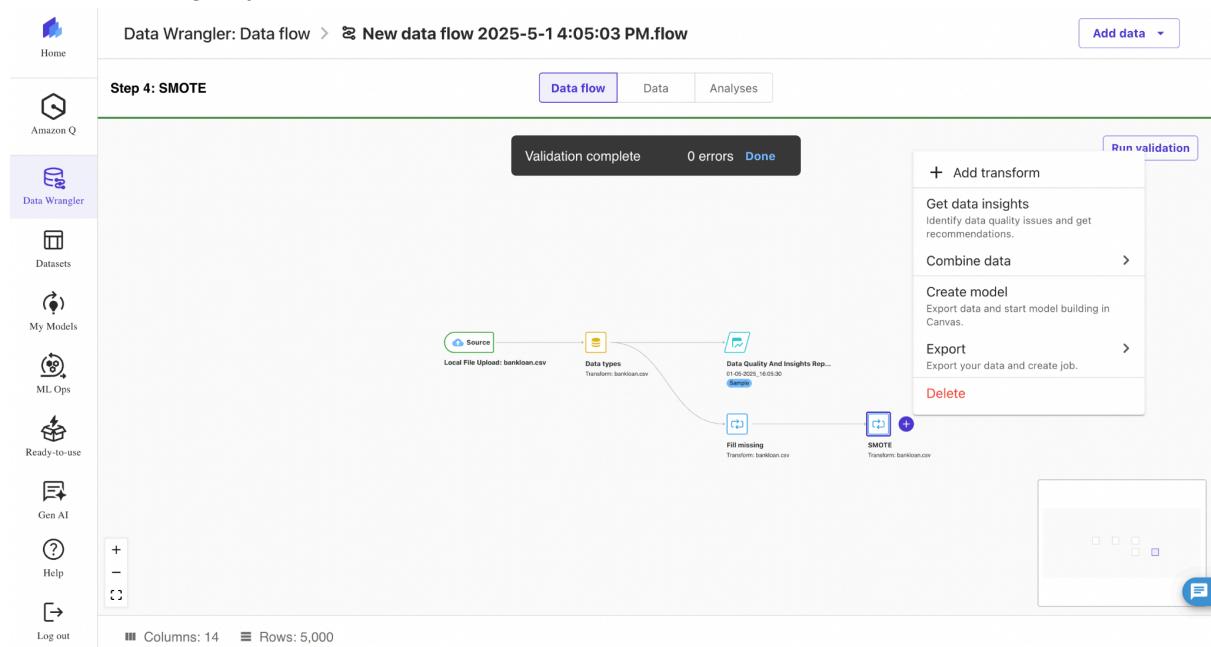
Input columns * ⓘ Select...

Advanced

Clear Preview Add



10. After adding any data transformation steps which are required click on create model



Click on export and create model

Data Wrangler: Data flow > **New data flow 2025-5-1 4:05:03 PM.flow**

Step 4: SMOTE

Validation complete 0 errors Done

Export to create a model

Dataset details
Dataset name * **Dataset_20250501_104001**
Use only letters, numbers, spaces, dashes, colons, and underscores up to 64 characters.

Process entire dataset ⓘ

Model details
Model name * **Model_20250501_104001**

Problem type * Predictive analysis

Target column * Loanapproved

Cancel **Export and create model**

Data Wrangler: Data flow > **New data flow 2025-5-1 4:05:03 PM.flow**

Step 4: SMOTE

Validation complete 0 errors Done Run validation

Running data flow on dataset...

Local File Upload: bankloan.csv Data types Data Quality And Insights Rep...
 Transform: bankloan.csv Transform: bankloan.csv Transform: bankloan.csv
 Fill missing SMOTE Destination

Dataset_20250501_104001: ban...

Columns: 14 Rows: 5,000

Click on configure model to choose the model type, objective metric, training method and algorithms, Data Split and Max candidates and runtime

My models > Model_20250501_104001 > Version 1

Select Build Analyze Predict Deploy

Select a column to predict

Choose the target column. The model that you build predicts values for the column that you select.

Target column: Loanapproved

Value distribution:

Model type

SageMaker Canvas automatically recommends the appropriate model type for your analysis.

2 category prediction

Your model classifies Loanapproved into two categories.

Configure model

Quick build

Preview model

Dataset_20250501_104001

Full dataset: 9.0k rows

Column name	Data type	Feature type	Missing	Mismatched	Unique	Mode
Zipcode	123 Numeric	-	0.00% (0)	0.00% (0)	4225	94,720
SecuritiesAccount	123 Numeric	-	0.00% (0)	0.00% (0)	857	0
Online	123 Numeric	-	0.00% (0)	0.00% (0)	1935	1
Mortgage	123 Numeric	-	0.00% (0)	0.00% (0)	2285	0

Total columns: 14 Total rows: 9,040 Total cells: 126,560 Show dropped columns

My models > Model_20250501_104001 > Version 1

Configure model Reset to default settings

Basic

Model type

Advanced - Optional

Objective metric

Training method and algorithms

Data split

Specify how you want to split your data into training and validation sets. Canvas builds your model with the training set and verifies the model's accuracy with the validation set.

Configuring the data split will default to Standard build.

Training set: 80 %

Validation set: 20 %

Max candidates and runtime

Cancel **Save**

Total columns: 14 Total rows: 9,040 Total cells: 126,560 Show dropped columns

After configuring the model click on quick build

My models > Model_20250501_104001 > Version 1

Select Build Analyze Predict Deploy

Select a column to predict

Choose the target column. The model that you build predicts values for the column that you select.

Target column: Loanapproved

Value distribution:

Model type

SageMaker Canvas automatically recommends the appropriate model type for your analysis.

2 category prediction

Your model classifies Loanapproved into two categories.

Configure model

Quick build

Preview model

Dataset_20250501_104001

Full dataset: 9.0k rows

Column name	Data type	Feature type	Missing	Mismatched	Unique	Mode
Zipcode	123 Numeric	-	0.00% (0)	0.00% (0)	4225	94,720
SecuritiesAccount	123 Numeric	-	0.00% (0)	0.00% (0)	857	0
Online	123 Numeric	-	0.00% (0)	0.00% (0)	1935	1
Mortgage	123 Numeric	-	0.00% (0)	0.00% (0)	2285	0

Total columns: 14 Total rows: 9,040 Total cells: 126,560 Show dropped columns

My models > Model_20250501_104001 > Version 1

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: Loanapproved

Value distribution:

Model type

SageMaker Canvas automatically recommends the appropriate model type for your analysis.

2 category prediction

Your model classifies Loanapproved into two categories.

Configure model

Validating your data...

Validating your data...

Dataset_20250501_104001

Full dataset: 9.0k rows

Column name	Data type	Feature type	Missing	Mismatched	Unique	Mode
Zipcode	Numeric	-	0.00% (0)	0.00% (0)	4225	94,720
SecuritiesAccount	Numeric	-	0.00% (0)	0.00% (0)	857	0
Online	Numeric	-	0.00% (0)	0.00% (0)	1935	1

Total columns: 14 Total rows: 9,040 Total cells: 126,560 Show dropped columns

11. After the model has been created you can click on predict to test the models prediction on a dataset

My models > Model_20250501_104001 > Version 1

Select Build Analyze Predict Deploy

Model status (Quick build)

Accuracy (Optimization metric: F1) 99.06% 0.991

The model predicts the correct Loanapproved 99.06% of the time.

Predict Standard build Deploy

Overview Scoring Advanced metrics Model leaderboard

Column impact (Search columns...)

Rank	Column	Impact (%)
1	Income	32.329%
2	Online	20.178%
3	CCAvg	13.793%
4	CDAccount	13.325%

Impact of Income on prediction of Loanapproved

Dataset_20250501_104001 Total columns: 14 Total rows: 9,040 Total cells: 126,560 Loanapproved 2 category prediction Predict

My models > Model_20250501_104001 > Version 1

Select Build Analyze Predict Deploy

Predict target values

Batch prediction Single prediction

Generate predictions for an entire dataset.

Manual Automatic

Predictions

All Jobs Configuration

Filter by configuration name: All

Send to Amazon QuickSight

Job name	Created	Input dataset	Prediction type	Configuration name	Rows	QuickSight
[Placeholder]						

Select dataset for predictions

X

To make predictions on a dataset, select it or import it. The dataset that you select must have the same number of feature columns as the training dataset. [?](#)

[+ Create dataset](#)

Search datasets in Canvas

Name	Columns	Rows	Cells	Created	Status
<input checked="" type="radio"/> Dataset_20250501_104001	V1	14	9,040	126,560	05/01/2025 4:10 PM
<input type="radio"/> canvas-sample-housing.csv	V1	10	1,000	10,000	05/01/2025 4:02 PM
<input type="radio"/> canvas-sample-shipping-logs.csv	V1	12	1,000	12,000	05/01/2025 4:02 PM
<input type="radio"/> canvas-sample-product-descriptions.csv	V1	5	120	600	05/01/2025 4:02 PM
<input type="radio"/> canvas-sample-loans-part-2.csv	V1	5	1,000	5,000	05/01/2025 4:02 PM
<input type="radio"/> canvas-sample-loans-part-1.csv	V1	19	1,000	19,000	05/01/2025 4:02 PM
<input type="radio"/> canvas-sample-diabetic-readmission.csv	V1	16	1,000	16,000	05/01/2025 4:02 PM
<input type="radio"/> canvas-sample-retail-electronics-forecasting.csv	V1	6	40,500	243,000	05/01/2025 4:02 PM

batchInfer-Model_20250501_104001-Dataset_20250501_104001-1746096461 predictions ready [View](#) X

[Close](#)

[Generate predictions](#)

My models > Model_20250501_104001 > Version 1

+ Create new version

batchInfer-Model_20250501_104001-Dataset_20250501_104001-1746096461

Prediction (Loanapproved)	Probability	ID	Age	Experience	Income	Zipcode	Family
0	99.9%	1	25	1	49	91107	4
0	99.9%	2	45	19	34	90089	3
0	99.9%	3	39	15	11	94720	1
0	96.5%	4	35	9	100	94112	1
0	99.9%	5	35	8	45	91330	4
0	99.9%	6	37	13	29	92121	4
0	99.9%	7	53	27	72	91711	2
0	99.9%	8	50	24	22	93943	1
0	99.9%	9	35	10	81	90089	3

[Send to Amazon QuickSight](#) [Download](#)

batchInfer-Model_20250501_104001-Dataset_20250501_104001-1746096461 predictions ready [View](#) X

12. Deploying the model- Go to My Models and select view on model you want to deploy

The screenshot shows the 'My models' section of a machine learning platform. On the left is a sidebar with icons for Home, Amazon Q, Data Wrangler, Datasets, My Models (which is selected and highlighted in blue), ML Ops, Ready-to-use, Gen AI, Help, and Log out. The main area has tabs for 'Grid' and 'List'. A search bar at the top right says 'Search models' and a button says '+ New model'. Below the tabs is a filter for 'problem type: 2 category prediction'. A large card displays a summary for 'Model_20250501_104001': it is 'Ready' (indicated by a green checkmark). It has '1' version, 'Loanapproved' as the target, '2 category prediction' as the problem type, and was updated on '2025-5-1 4:15:06 PM'. There are 'View' and more options buttons at the bottom of the card.

Click on the model again

The screenshot shows the 'Versions' page for the model 'Model_20250501_104001'. The sidebar on the left is identical to the previous screenshot. The main title is 'My models > Model_20250501_104001'. A 'Create new version' button is at the top right. Below it is a 'Show advanced metrics' toggle switch. The 'Versions' table has columns: Version, Status, Build type, Created, Dataset, Accuracy, and Model Registry. One row is visible: Version V1, Status Ready (green checkmark), Build type Quick, Created 05/01/2025 4:10 PM, Dataset 'Dataset_...', Accuracy 99.06%, and Model Registry Not Registered. There is a more options button at the end of the row.

Click on deploy and create deployment

My models > Model_20250501_104001 > Version 1

Select Build **Analyze** Predict Deploy

Model status (Quick build)

Accuracy (Optimization metric) F1 (Optimization metric)
99.06% 0.991

The model predicts the correct Loanapproved 99.06% of the time.

Predict **Standard build** **Deploy**

Overview Scoring Advanced metrics Model leaderboard

Column impact (Search columns...)

Rank	Column	Impact (%)
1	Income	32.329%
2	Online	20.178%
3	CCAvg	13.793%
4	CDAccount	13.325%

Impact of Income on prediction of Loanapproved

Impact on prediction

Income

Dataset_20250501_104001 Total columns: 14 Total rows: 9,040 Total cells: 126,560 Loanapproved 2 category prediction **Predict**

+ Create new version ⏪ ⋮

Create Deployment X

Deploy your model to a SageMaker endpoint so that you can make predictions from outside of the Canvas application, test and monitor your model to proactively detect issues such as model drift.

Selected model version
Model_20250501_104001
v1 Ready Created: 05-02-2025-2:53 PM

Deployment type
Real-time ⏪

Deployment name
Deployment name
new-deployment-05-02-2025-2-53-PM

Instance type ⏪ Learn about pricing ⏪
ml.t2.medium

Instance count ⏪

13- To create an endpoint- we go back to SageMaker AI studio and select endpoints from deployments

The screenshot shows the SageMaker Studio Home page. On the left, a sidebar menu includes Data, Auto ML, Experiments, Jobs, Pipelines, Models, JumpStart, Deployments (selected), and Projects. Under Deployments, there are sub-options for Endpoints (Manage deployed models) and Projects (Automate model building & deployment). A large central panel displays the "Onboarding plan" with three sections: "Take the tour" (Quick tour highlights where you can find key features and how to navigate the new experience. See what's new and where to locate the tools you need to be productive.), "Access your EFS data in JupyterLab and CodeEditor" (Automatically available in private spaces.), and "Access your Studio Classic apps" (Pickup where you left off and access your Studio Classic apps from within the updated Studio experience.). Below the onboarding plan, there is a "Deploy models for inference" section with a link to "Revert to Studio Classic experience in domain settings. Learn more". At the bottom, there are tabs for Overview, Getting started, and What's new, followed by a "jupyter" logo.

Here click on create endpoint

The screenshot shows the SageMaker Studio Endpoints page. The sidebar menu is identical to the Home page, with Deployments selected. The main content area is titled "Endpoints" and contains a table of existing endpoints. The table has columns for Name, Status, Created on, and Modified on. One entry is visible: "canvas-new-deployment-05-02-2..." with status "In service", created on "02/05/2025, 14:54:10", and modified on "02/05/2025, 15:02:05". Below the table, there is a "Learn about endpoints" section with links for Get started, Documentation, and What's new. At the bottom, there are links for Privacy, Site Terms, and Cookie Preferences, along with a copyright notice: "© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved."

Click on add model and select the ML model we created

The screenshot shows the 'Create endpoint' settings page in AWS SageMaker Studio. On the left, a sidebar navigation includes 'Data', 'Auto ML', 'Experiments', 'Jobs', 'Pipelines', 'Models', 'JumpStart', 'Deployments' (selected), 'Endpoints' (under Deployments), and 'Projects'. The main area is titled 'Create endpoint' with the sub-section 'Endpoint settings'. It contains fields for 'Endpoint name' (set to 'Endpoint-20250502-100134'), 'Instance type' (set to 'ml.c6i.xlarge'), 'Initial instance count' (set to '1'), and 'Maximum instance count' (set to '20'). Under 'Inference type', 'Real-time' is selected, with a note: 'For sustained traffic and consistently low latency. Supports payload sizes up to 6 MB and runtimes up to 60 sec.' A 'Models' section has a '+ Add model' button. At the bottom right are 'Cancel', 'Deploy' (disabled), and a message bubble icon.

Choose deployable model as we have created the model on AWS Sagemaker

The screenshot shows the 'Add model' dialog in AWS SageMaker Studio. The sidebar on the left is identical to the previous screenshot. The dialog title is 'Add model' and the sub-section is 'Step 1: Select a model from'. It offers two options: 'JumpStart Foundation Models' (radio button not selected) and 'Deployable Models' (radio button selected). A note below states: 'Models with network isolation, IAM role, or VPC config inconsistent with the endpoint configuration are disabled. Other incompatible models are also disabled'. A search bar 'Search by model name' is present. A table lists five models, all of which are currently 'None' in the 'Status' column. The table has columns: Name, Created On, Deployed Endpoints, and Status. The first model listed is 'canvas-model-2025-05-02-09-...'. At the bottom are 'Cancel', '+ Add model' (disabled), and 'Deploy' (disabled).

Name	Created On	Deployed Endpoints	Status
canvas-model-2025-05-02-09-...	02/05/2025, 14:54:09	1	None
canvas-model-2025-05-01-11-...	01/05/2025, 16:52:20	0	None
canvas-model-2025-05-01-10-...	01/05/2025, 16:20:50	0	None
canvas-model-2025-05-01-10-...	01/05/2025, 16:20:32	0	None
canvas-model-2025-05-01-10-...	01/05/2025, 16:20:08	0	None

Create the endpoint

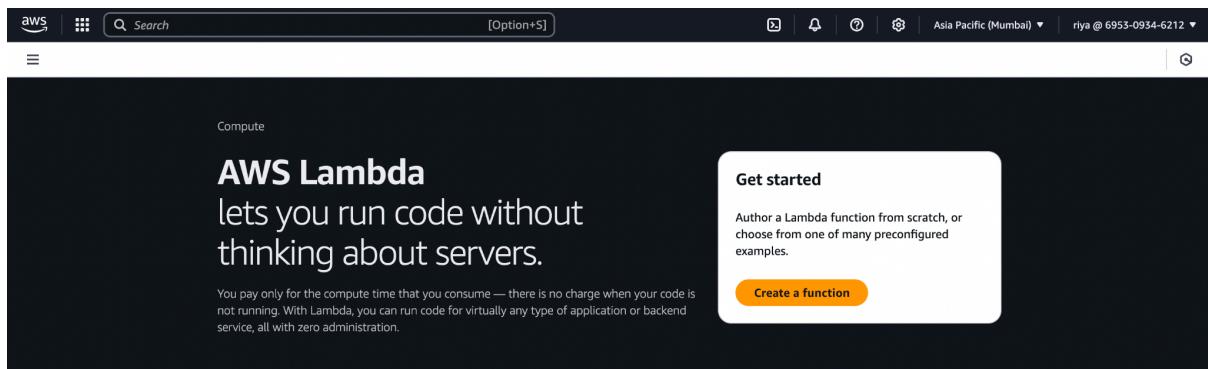
The screenshot shows the SageMaker Studio interface for creating a new endpoint. The left sidebar is collapsed. The main area displays the 'Endpoint summary' for 'Endpoint-20250502-100134'. The endpoint is currently in the 'Creating' status. The 'Inference Type' is set to 'Real-time'. The 'Last updated' timestamp is 'Fri May 02 2025 15:33:19 GMT+0530 (India Standard Time)'. The ARN is listed as 'arn:aws:sagemaker:ap-south-1:695309346212:endpoint/Endpoint-20250502-100134'. The 'Endpoint logs' path is '/aws/sagemaker/endpoints/Endpoint-20250502-100134'. Below the summary, there are tabs for 'Models', 'Settings', and 'Test inference'. A green success message states 'Endpoint Endpoint-20250502-100134 is being created.' and 'Successfully add inference component: canvas-model-2025-05-02-09-24-07-581714-20250502-1002510'. At the bottom, there are buttons for 'Delete' and '+ Add model'.

To test the endpoint click on
Test inference,
choose text/csv
paste- 1,24,4,300000,400053,1,400,4,0,1,1,1,0

The screenshot shows the 'Test inference' tab in SageMaker Studio. The 'Model' dropdown is set to 'canvas-model-2025-05-02-09-24-07-581714-20250502-1002510'. Under 'Testing Options', the radio button for 'Test the sample request' is selected. The 'Content type' is set to 'text/csv'. The 'CSV' section contains the input data: '1,24,4,300000,400053,1,400,4,0,1,1,1,0'. On the right, the 'Inference Result' panel shows the response: 'Status: Success', 'Execution Length (ms): 259', 'Request Time: 1 minutes ago', and 'Result Time: 1 minutes ago'. The result body is shown as a JSON object: '{ "body": {}, "contentType": "text/csv", "invokedProductionVariant": "variant-1" }'. A blue 'Copy entire result' button is at the bottom.

14-After the endpoint has been created we will make a Rest API-

Open AWS Lambda and click on create a function

This screenshot shows the "How it works" section of the AWS Lambda console. It displays a code editor with Python runtime selected. The code shown is a simple lambda handler:1 def lambda_handler(event, context):
2 print(event)
3 return 'Hello from Lambda!'
4A "Run" button is visible above the code editor, along with a link to "Next: Lambda responds to events". The footer includes links for CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

Scroll down to code source and put the following code in and click deploy-

```
import boto3  
import json  
  
runtime = boto3.client("sagemaker-runtime")  
  
def lambda_handler(event, context):  
    try:  
        input_json = json.loads(event["body"])  
  
        feature_order = [  
            "ID", "Age", "Experience", "Income", "Zipcode", "Family",  
            "CCAvg", "Education", "Mortgage", "SecuritiesAccount",  
            "CDAccount", "Online", "CreditCard"  
        ]  
  
        csv_values = [str(input_json[feature]) for feature in feature_order]  
        csv_string = ",".join(csv_values)  
  
        response = runtime.invoke_endpoint(  
            EndpointName="Endpoint-20250502-100134", #Enter your endpoint name here  
            ContentType="text/csv",  
            Body=csv_string,  
            Accept="application/json",  
  
            InferenceComponentName="canvas-model-2025-05-02-09-24-07-581714-20250502-10025  
            10"  
        )  
  
        result = response["Body"].read().decode("utf-8")
```

```

return {
    "statusCode": 200,
    "headers": {"Content-Type": "application/json"},
    "body": result
}

except Exception as e:
    return {
        "statusCode": 500,
        "body": str(e)
}

```

The screenshot shows the AWS Lambda code editor for a function named 'sagemaker-api-wrapper'. The code in 'lambda_function.py' is as follows:

```

import boto3
runtime = boto3.client("sagemaker-runtime")
def lambda_handler(event, context):
    input_data = event["body"] # This gets the raw input passed from API Gateway
    response = runtime.invoke_endpoint(
        EndpointName="Endpoint-20250502-100134", # Your actual SageMaker endpoint
        ContentType="text/csv", # or application/json if needed
        Body=input_data
    )
    result = response['Body'].read().decode()
    return {
        "statusCode": 200,
        "headers": {"Content-Type": "application/json"},
        "body": result
    }

```

The interface includes a sidebar with 'EXPLORER', 'TEST EVENTS [NONE SELECTED]', and deployment buttons for 'Deploy' and 'Test'. A right-hand panel titled 'Create a simple web app' provides a tutorial on building a Lambda function.

Then go to add trigger

Screenshot of the AWS Lambda Function Overview page for 'sagemaker-api-wrapper'.

Function Overview:

- Description:** Last modified 41 seconds ago.
- Function ARN:** arn:aws:lambda:ap-south-1:695309346212:function:sagemaker-api-wrapper
- Function URL:** -

Code Source: Info

Add trigger:

Trigger configuration: Info

Select a source: API Gateway

APIs/Interactive/web:

- Alexa
- API Gateway
- Application Load Balancer
- CodeCommit
- Cognito Sync Trigger
- VPC Lattice

Batch/bulk data processing:

- AWS IoT
- CloudWatch Logs
- EventBridge (CloudWatch Events)

Tutorials: Create a simple web app

In this tutorial you will learn how to:

- Build a simple web app, consisting of a Lambda function with a function URL that outputs a webpage
- Invoke your function through its function URL

[Learn more](#) [Start tutorial](#)

Select API gateway as the source then select creating new API as intent and API Type- REST API

15- To add permissions- go to configuration

aws Search [Option+S] Lambda > Functions > sagemaker-api-wrapper

2:function:sagemaker-api-wrapper

Function URL | Info

Code | Test | Monitor | Configuration | Aliases | Versions

General configuration

Description: - | Memory: 128 MB | Ephemeral storage: 512 MB

Timeout: 0 min 3 sec | SnapStart: None

Triggers

Permissions

Destinations

Function URL

Environment variables

Tags

VPC

RDS databases

Monitoring and operations tools

General configuration

Description: - | Memory: 128 MB | Ephemeral storage: 512 MB

Timeout: 0 min 3 sec | SnapStart: None

Edit

Create a simple web app ^

In this tutorial you will learn how to:

- Build a simple web app, consisting of a Lambda function with a function URL that outputs a webpage
- Invoke your function through its function URL

Learn more ↗ Start tutorial

https://ap-south-1.console.aws.amazon.com/lambda/home?region=ap-south-1#/functions/sagemaker-api-wrapper?tab=configure

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Then click on permissions and click on the role name link

aws Search [Option+S] Lambda > Functions > sagemaker-api-wrapper

Function URL | Info

Code | Test | Monitor | Configuration | Aliases | Versions

General configuration

Role name: sagemaker-api-wrapper-role-h7fpk337

Resource summary

To view the resources and actions that your function has permission to access, choose a service.

AWS Application Auto Scaling

By action | By resource

Resource | Actions

All resources

Allow: application-autoscaling:PutScalingPolicy
Allow: application-autoscaling:DescribeScalingActivities
Allow: application-autoscaling:DescribeScalingPolicies
Allow: application-autoscaling>DeleteScheduledAction
Allow: application-autoscaling:PutScheduledAction
Allow: application-autoscaling>DeleteScalingPolicy

CloudShell Feedback

Create a simple web app ^

In this tutorial you will learn how to:

- Build a simple web app, consisting of a Lambda function with a function URL that outputs a webpage
- Invoke your function through its function URL

Learn more ↗ Start tutorial

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Then click on add permissions- attach policies and search for AmazonSageMakerFullAccess and add it

Screenshot of the AWS IAM Roles page showing the details of the 'sagemaker-api-wrapper-role-h7fpk337' role.

Identity and Access Management (IAM)

sagemaker-api-wrapper-role-h7fpk337

Summary

Creation date: May 02, 2025, 17:50 (UTC+05:30)

Last activity: 17 hours ago

ARN: arn:aws:iam::695309346212:role/service-role/sagemaker-api-wrapper-role-h7fpk337

Maximum session duration: 1 hour

Permissions

Permissions policies (3)

You can attach up to 10 managed policies.

Policy name	Type	Attached entities
AllowSageMakerInvoke	Customer inline	0
AmazonSageMakerFullAccess	AWS managed	12
AWSLambdaBasicExecutionRole-c988a...	Customer managed	1

16- Go to API gateway.

Screenshot of the AWS API Gateway landing page.

Networking & Content Delivery

API Gateway

Create and manage APIs at scale

Amazon API Gateway is a fully managed service that makes it easy for developers to create, publish, maintain, monitor, and secure APIs.

Get started

Create a new API to begin exploring API Gateway. You can also import an external definition file into API Gateway.

Create an API

How it works

API Gateway enables you to connect and access data, business logic, and functionality from backend services such as workloads running on Amazon Elastic Compute Cloud (Amazon EC2), code running on AWS Lambda, any web application, or real-time communication applications.

Pricing

With Amazon API Gateway, you only pay when your APIs are in use. There are no minimum fees or upfront commitments. For HTTP and REST APIs, you pay based on API calls received and amount of data transferred out. For WebSocket APIs, you pay based on number of messages and connection duration.

Resources

- Getting Started Guide
- Developer Guide
- API References

Click on Create API

The screenshot shows the AWS API Gateway interface. On the left, a sidebar titled 'API Gateway' has a 'APIs' section with links to 'Custom domain names', 'Domain name access associations', 'VPC links', 'Usage plans', 'API keys', 'Client certificates', and 'Settings'. The main content area is titled 'APIs (1/1)' and shows a table with one row. The table columns are 'Name', 'Description', 'ID', 'Protocol', and 'API endpoint type'. The row contains the values: 'sagemaker-api-wrapper-API', 'Created by AWS Lambda', 'rqaiairffj', 'REST', and 'Regional'. At the top right of the table are 'Delete' and 'Create API' buttons. Below the table is a search bar with placeholder text 'Find APIs'.

Choose REST API and then build

The screenshot shows the 'Create API' wizard. The first step, 'WebSocket API', is selected. It contains a brief description: 'Build a WebSocket API using persistent connections for real-time use cases such as chat applications or dashboards.' Below this is a note: 'Works with the following: Lambda, HTTP, AWS Services'. A large orange 'Build' button is at the bottom right. The next steps are 'REST API' and 'REST API Private', each with their own descriptions and 'Import' and 'Build' buttons. The bottom navigation bar includes 'CloudShell', 'Feedback', '© 2025, Amazon Web Services, Inc. or its affiliates.', 'Privacy', 'Terms', and 'Cookie preferences'.

API Gateway > APIs > Create API > Create REST API

Create REST API Info

API details

- New API Create a new REST API.
- Clone existing API Create a copy of an API in this AWS account.
- Import API Import an API from an OpenAPI definition.
- Example API Learn about API Gateway with an example API.

API name
My_demo_api

Description - optional

API endpoint type
Regional APIs are deployed in the current AWS Region. Edge-optimized APIs route requests to the nearest CloudFront Point of Presence. Private APIs are only accessible from VPCs.

Regional

IP address type Info
Select the type of IP addresses that can invoke the default endpoint for your API.

- IPv4 Supports only edge-optimized and Regional API endpoint types.
- Dualstack Supports all API endpoint types.

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Choose POST method and then integration type as Lambda function and then connect the lambda function that we created in the link below

API Gateway > APIs > Resources - sagemaker-api (cm2zxfca7) > Create method

Create method

Method details

Method type
POST

Integration type

- Lambda function Integrate your API with a Lambda function.

- HTTP Integrate with an existing HTTP endpoint.

- Mock Generate a response based on API Gateway mappings and transformations.

- AWS service Integrate with an AWS Service.

- VPC link Integrate with a resource that isn't accessible over the public internet.


Lambda proxy integration
Send the request to your Lambda function as a structured event.

Lambda function
Provide the Lambda function name or alias. You can also provide an ARN from another account.
ap-south-1

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Click on deploy API

The screenshot shows the AWS API Gateway Resources page. On the left, there's a sidebar with navigation links for APIs, VPC links, and the selected 'API: sagemaker-api' section, which includes Resources, Stages, Authorizers, Gateway responses, Models, Resource policy, Documentation, Dashboard, and API settings. Under 'Resources', a 'Create resource' button is visible, followed by a path entry field containing '/POST'. A green success message at the top right says 'Successfully created method 'POST' in '/'. Redeploy your API for the update to take effect.' Below this, the 'Method execution' diagram shows the flow from Client to Method request to Integration request to Lambda integration, and back from Integration response to Method response. The 'Method request' tab is selected. In the 'Method request settings' section, it shows 'Authorization: NONE', 'API key required: False', and 'SDK operation name'. The ARN is listed as arn:aws:execute-api:ap-south-1:695309346212:cm2zxfca7/*/POST/ and the Resource ID is 2nwd44w6k1.

Copy the URL after deploying the API to use on postman

The screenshot shows the AWS API Gateway Stages page. The sidebar is identical to the previous screen. The main area shows a deployment for the 'sagemaker_demo_API' stage. A green success message at the top right says 'Successfully created deployment for sagemaker-api. This deployment is active for sagemaker_demo_API.' Below this, the 'Stage details' section shows the Stage name as 'sagemaker_demo_API', Rate Info set to 10000, Cache cluster set to Inactive, and Default method-level caching set to Inactive. It also shows a 'Copied' message and a 'Take URL' link with the URL https://cm2zxfca7.execute-api.ap-south-1.amazonaws.com/sagemaker_demo_API. The 'Logs and tracing' section shows CloudWatch logs set to Inactive, Detailed metrics set to Inactive, and Data tracing set to Inactive.

17- Test in Postman-

Choose POST method and then enter your URL

Then enter the following in the body and click send to get your prediction-

```
{  
  "ID": 1,  
  "Age": 24,  
  "Experience": 4,  
  "Income": 300000,  
  "Zipcode": 400053,  
  "Family": 1,
```

```

    "CCAvg": 400,
    "Education": 4,
    "Mortgage": 0,
    "SecuritiesAccount": 1,
    "CDAccount": 1,
    "Online": 1,
    "CreditCard": 0
}

```

The screenshot shows the Postman application interface. On the left, there's a sidebar with 'My Workspace' containing 'Collections', 'Environments', 'Flows', and 'History'. A 'Create a collection for your requests' section is also present. The main area shows a POST request to https://cm2zxfca7.execute-api.ap-south-1.amazonaws.com/Postman_API. The 'Body' tab is selected, showing the following JSON payload:

```

1  {
2      "ID": 1,
3      "Age": 24,
4      "Experience": 4,
5      "Income": 300000,
6      "Zipcode": 400053,
7      "Family": 1,
8      "CCAvg": 400,
9      "Education": 4,
10     "Mortgage": 0,
11     "SecuritiesAccount": 1,
12     "CDAccount": 1,

```

The response tab shows a 200 OK status with a response body:

```

1  {
2      "predictions": [
3          {
4              "predicted_label": "1.0",
5              "probability": 0.9950631856918335,
6              "probabilities": "[0.9950631856918335, 0.004936841782182455]",
7              "labels": "[1.0, 0.0]"
8          }
9      ]
10 }

```