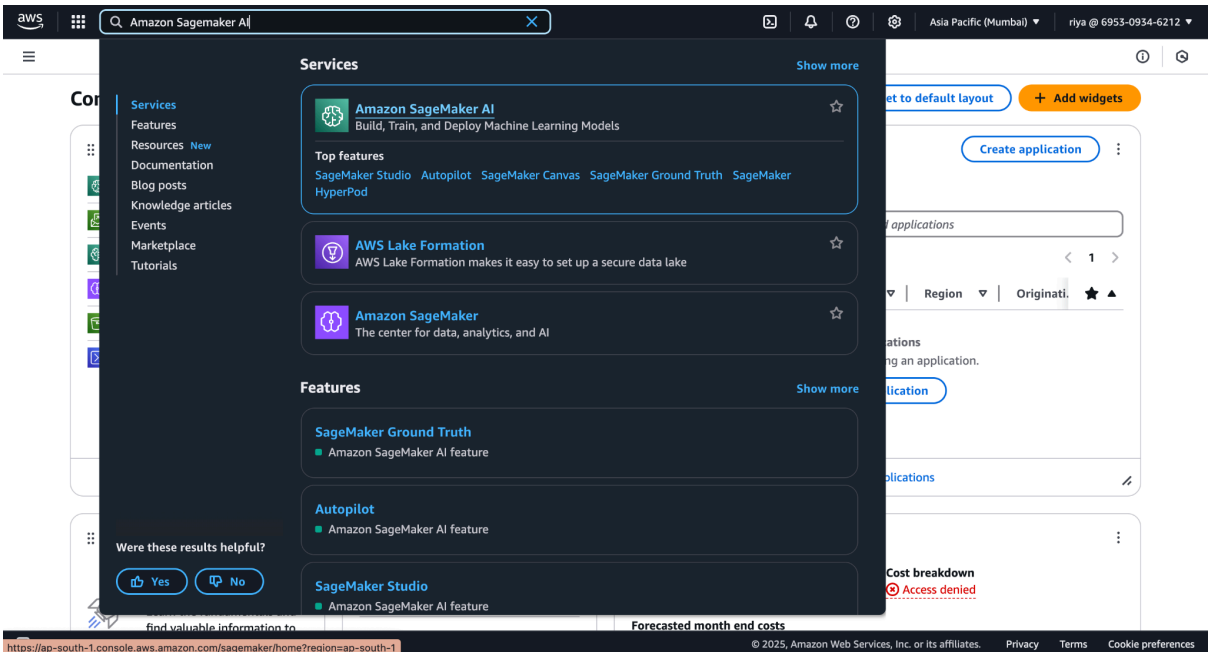
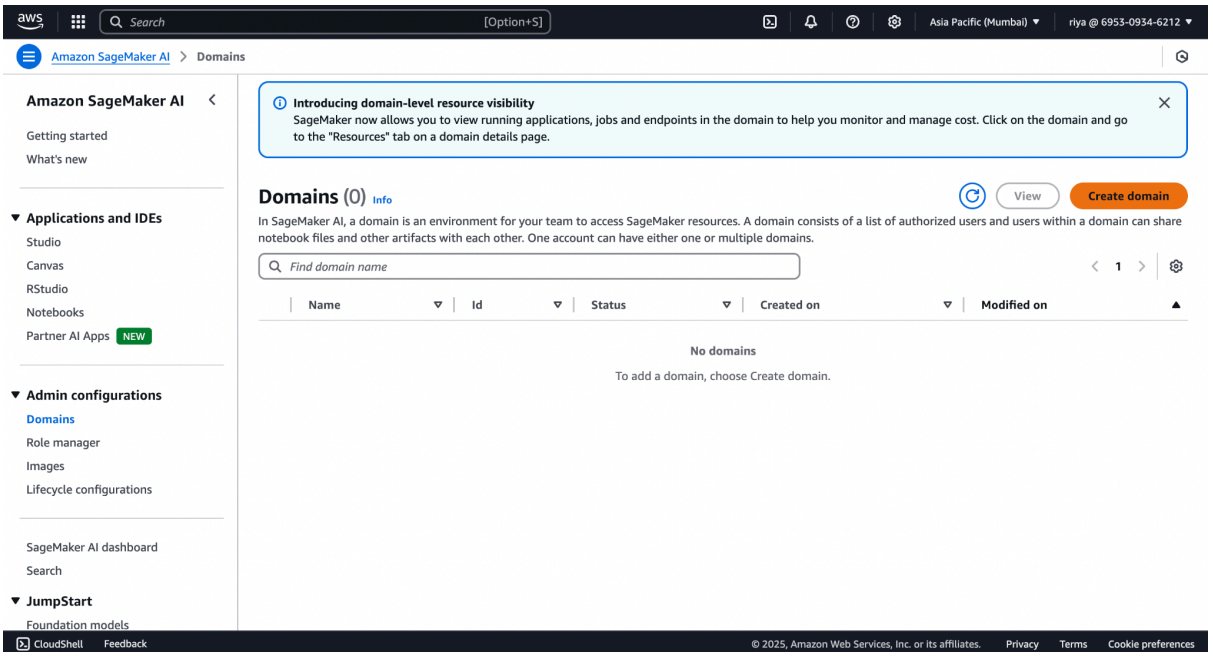
AMAZON SAGEMAKER AI- FROM DATA WRANGLING TO REST API

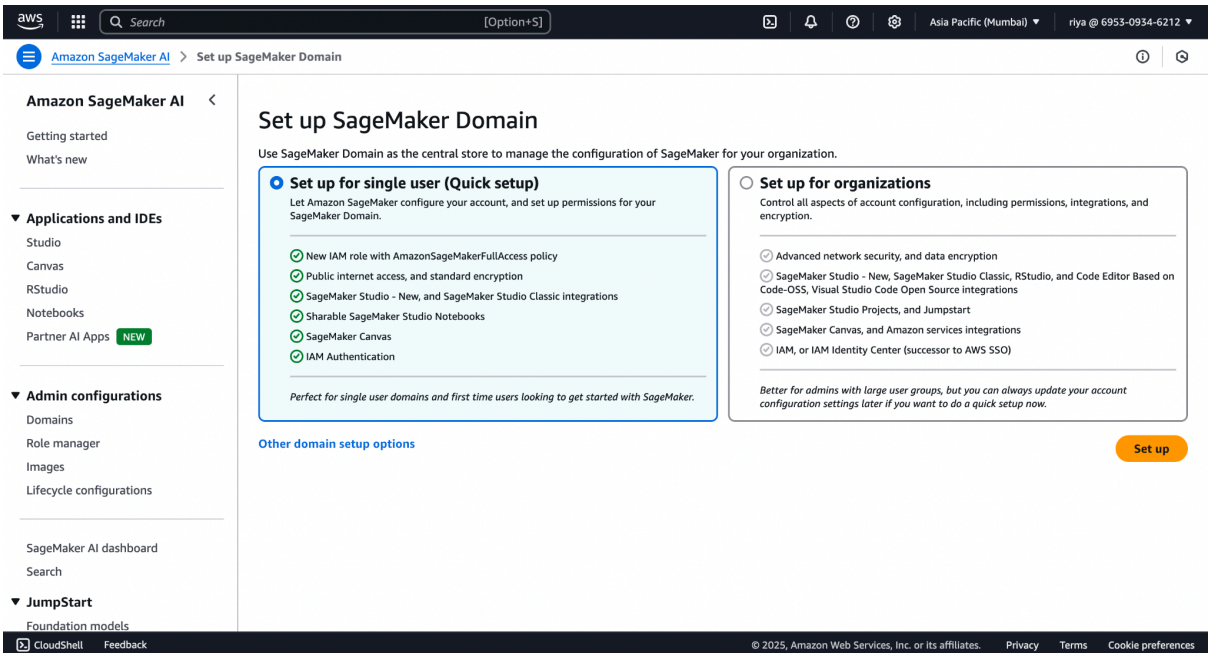
1. Open Amazon SageMaker AI

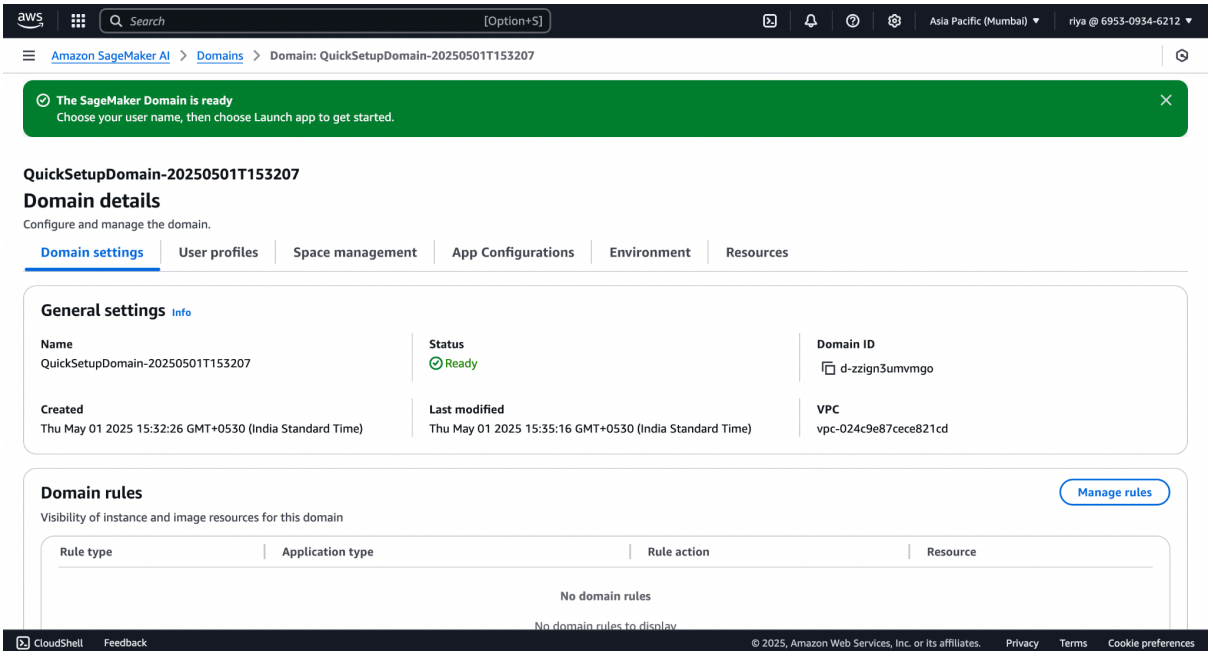


2. Click on Domains under Amazon Sagemaker AI and create a domain

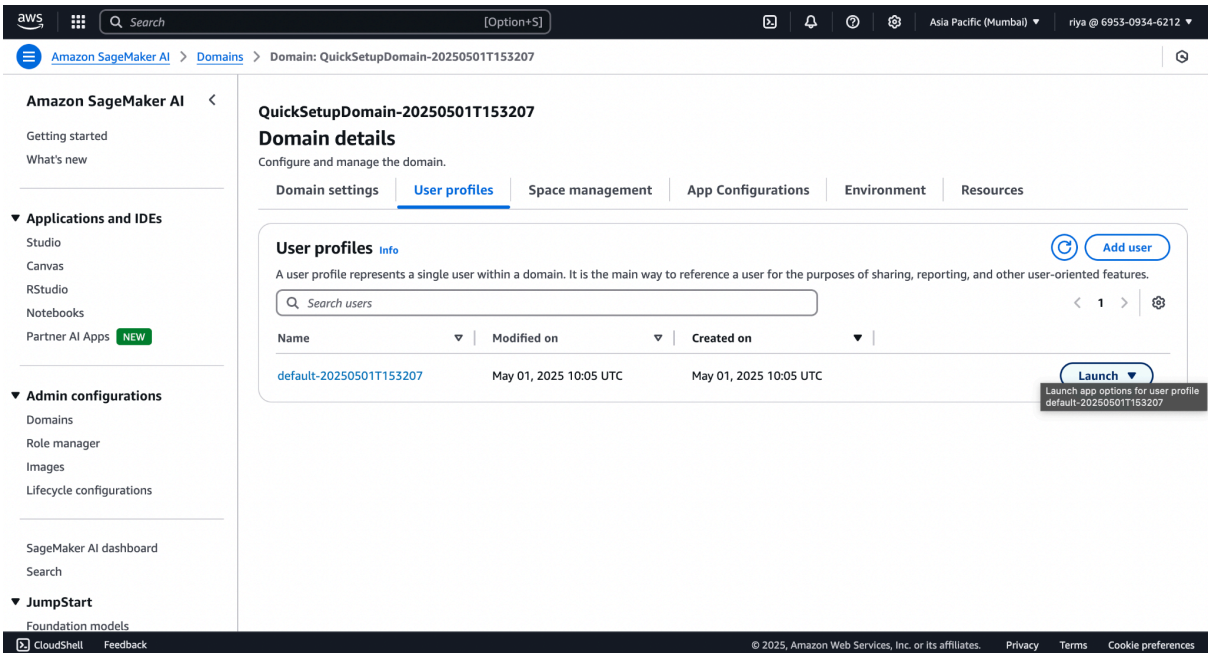


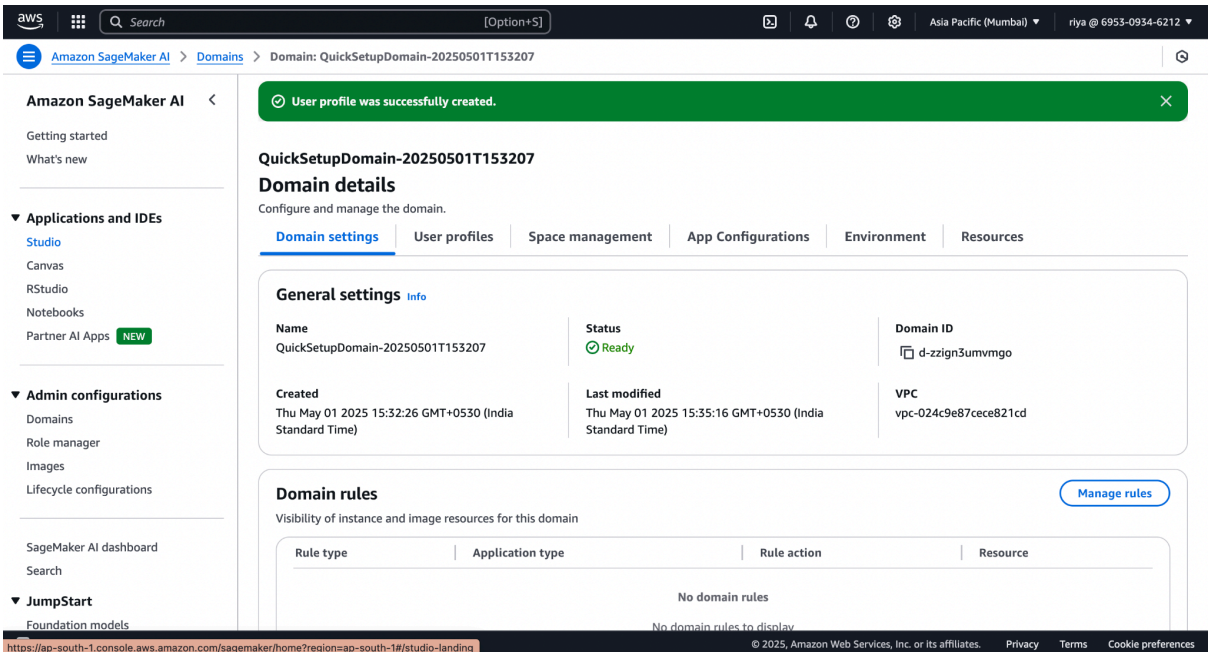
Click on set up for single user



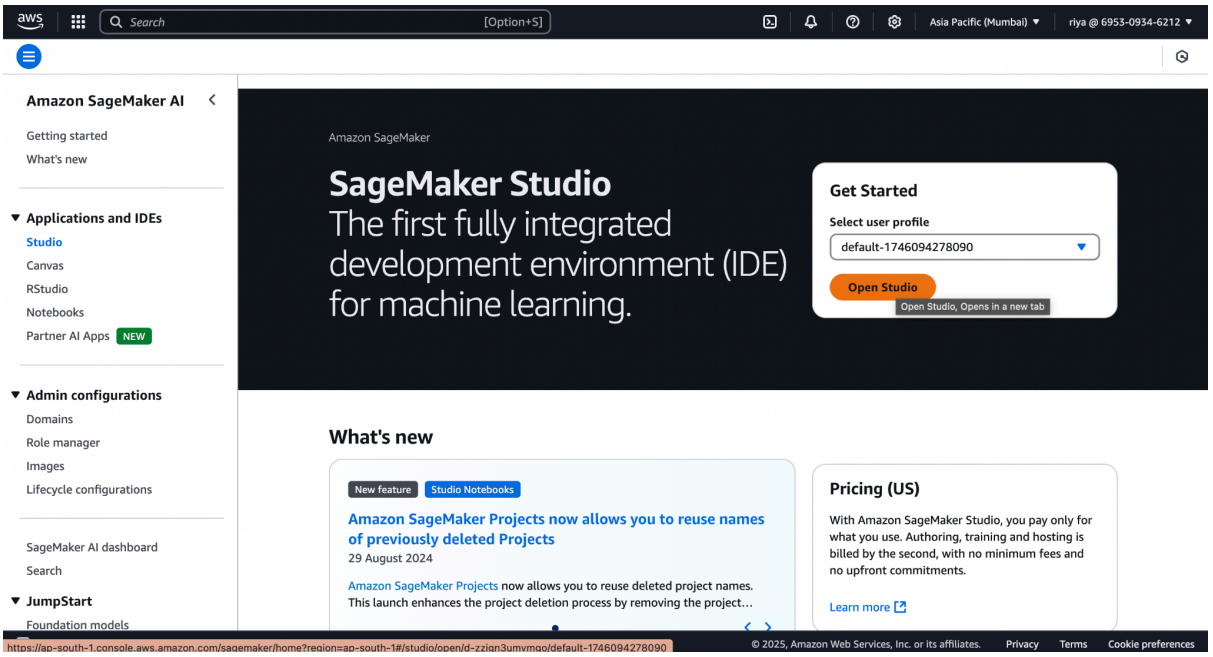


3.Create a user in the domain

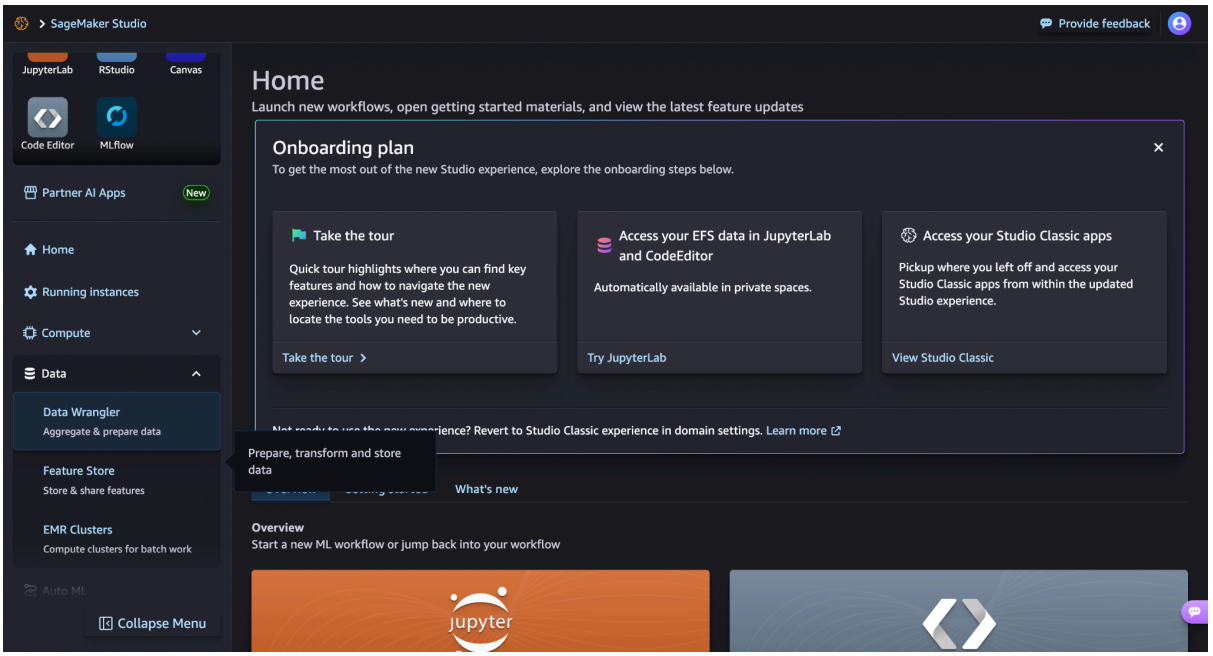




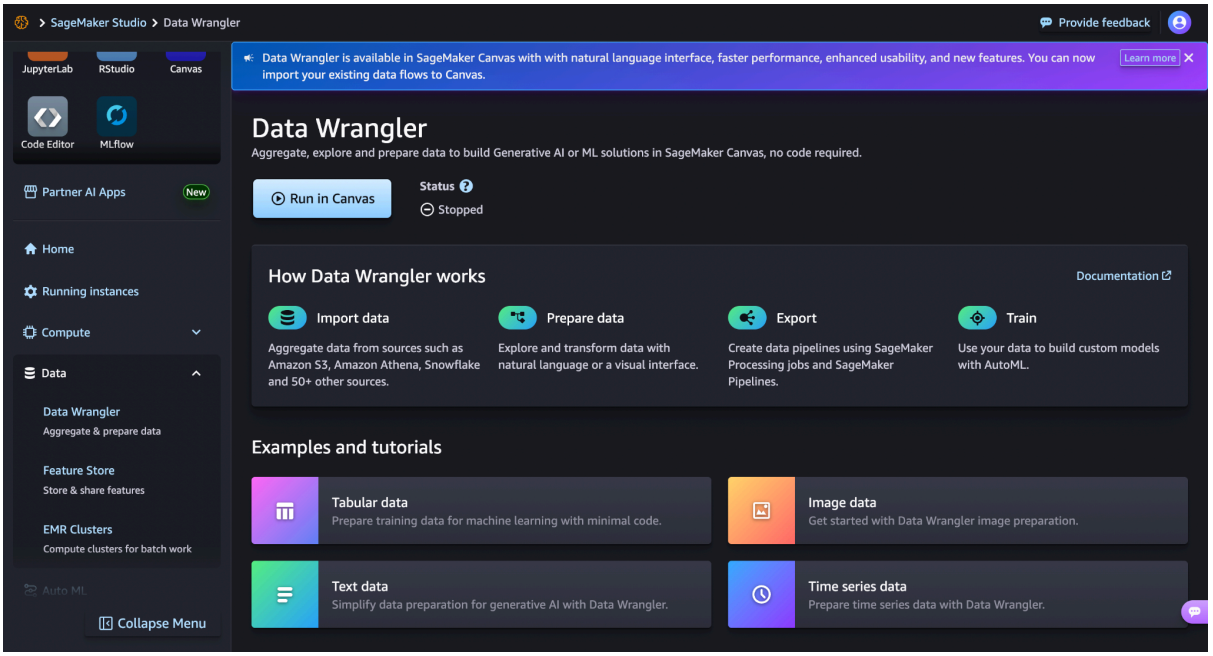
4. After creating the domain and user go to studio under Amazon Sagemaker AI and open studio with the user profile we created in the domain.



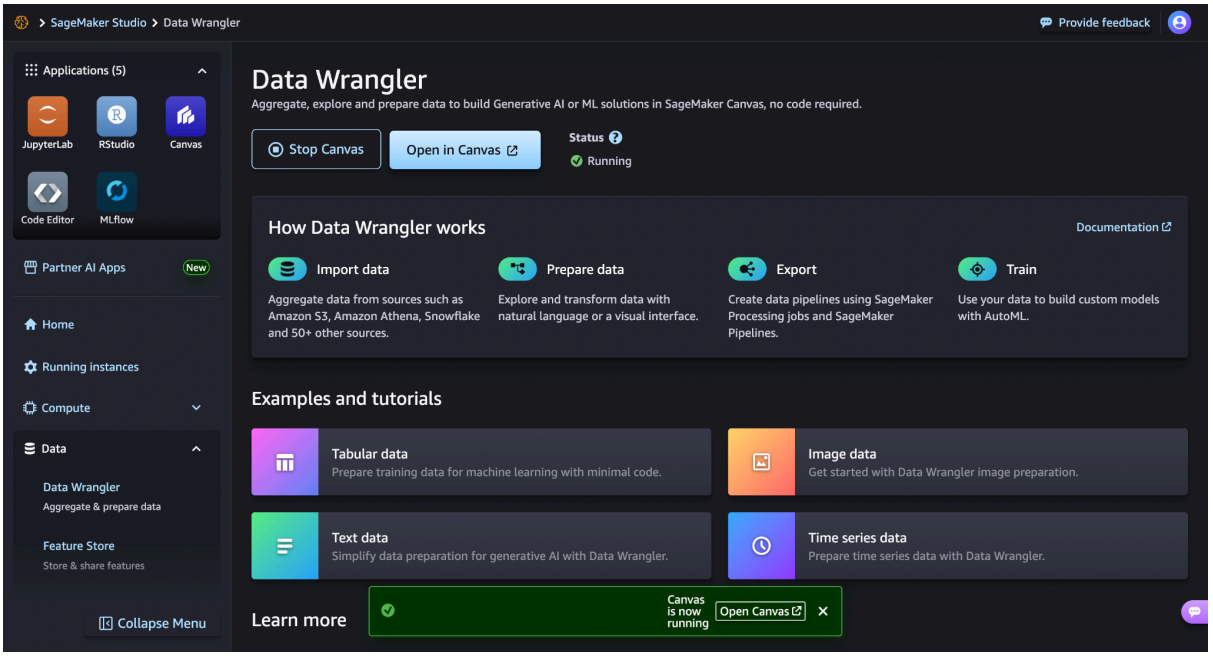
5.After opening Studio click on Data wrangler and open it.



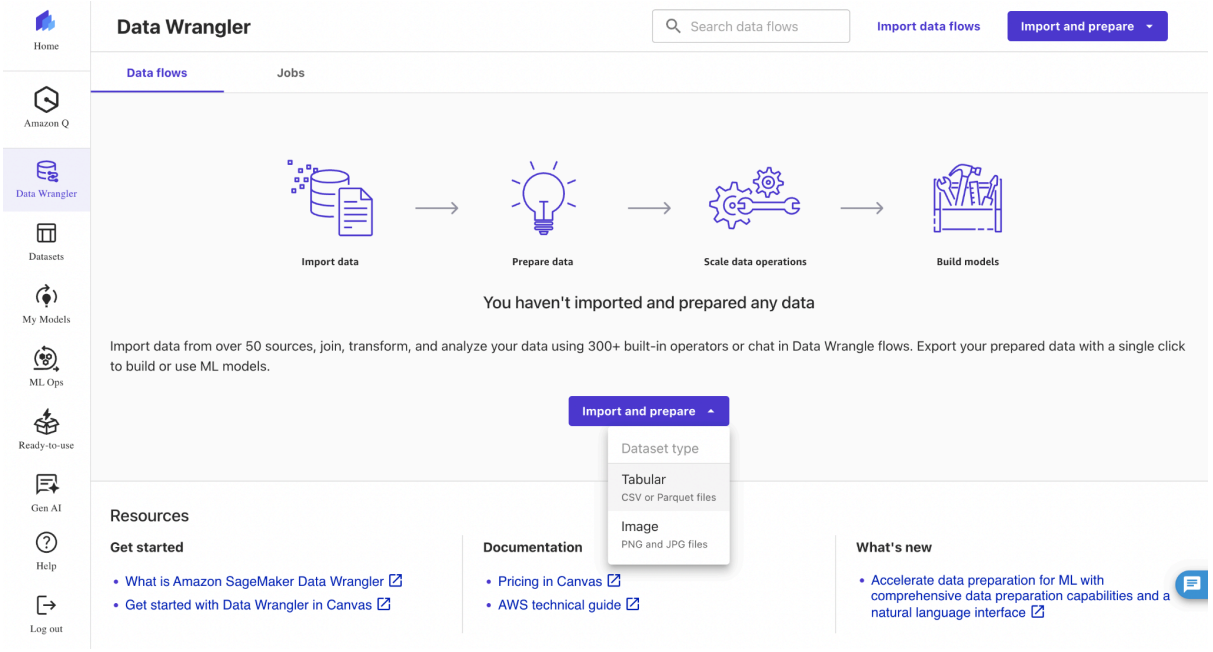
Click on Run in Canvas



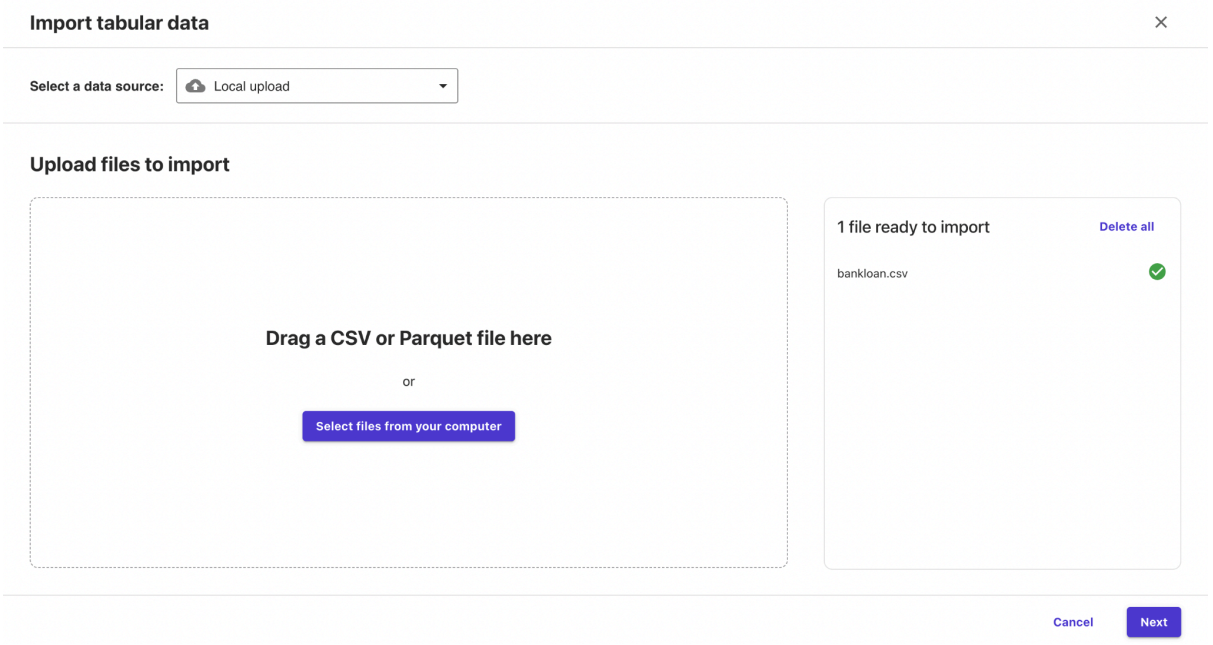
Click on Open in Canvas



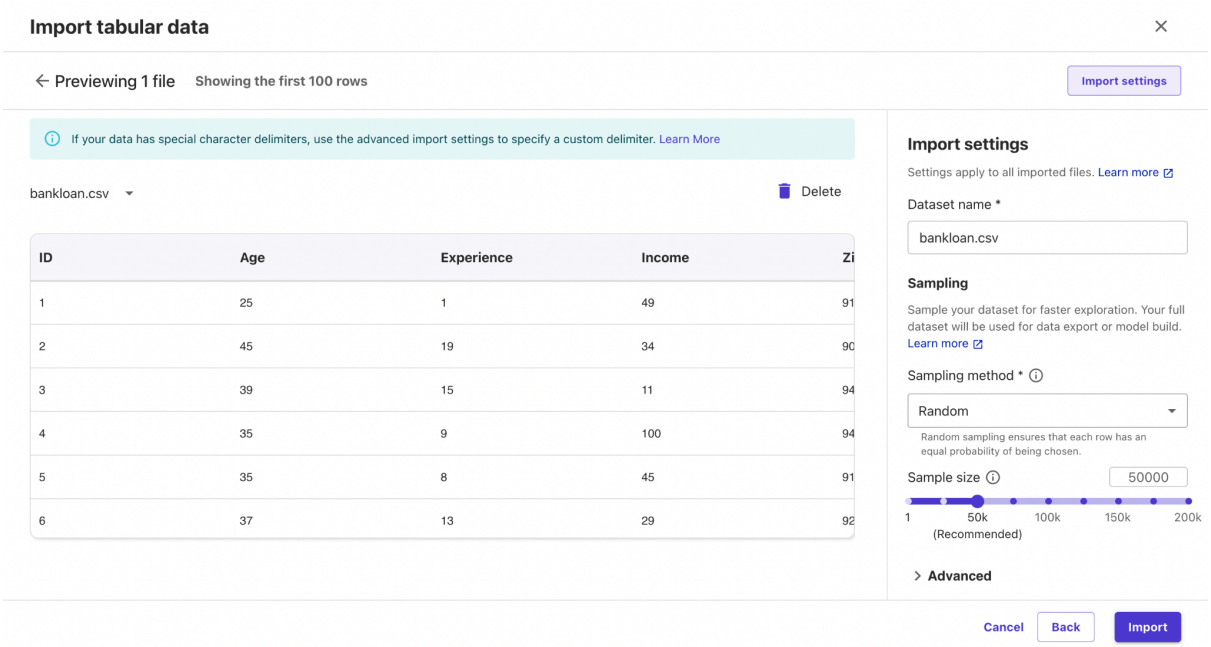
6.Click on Data wrangler and then import and prepare to import data needed.



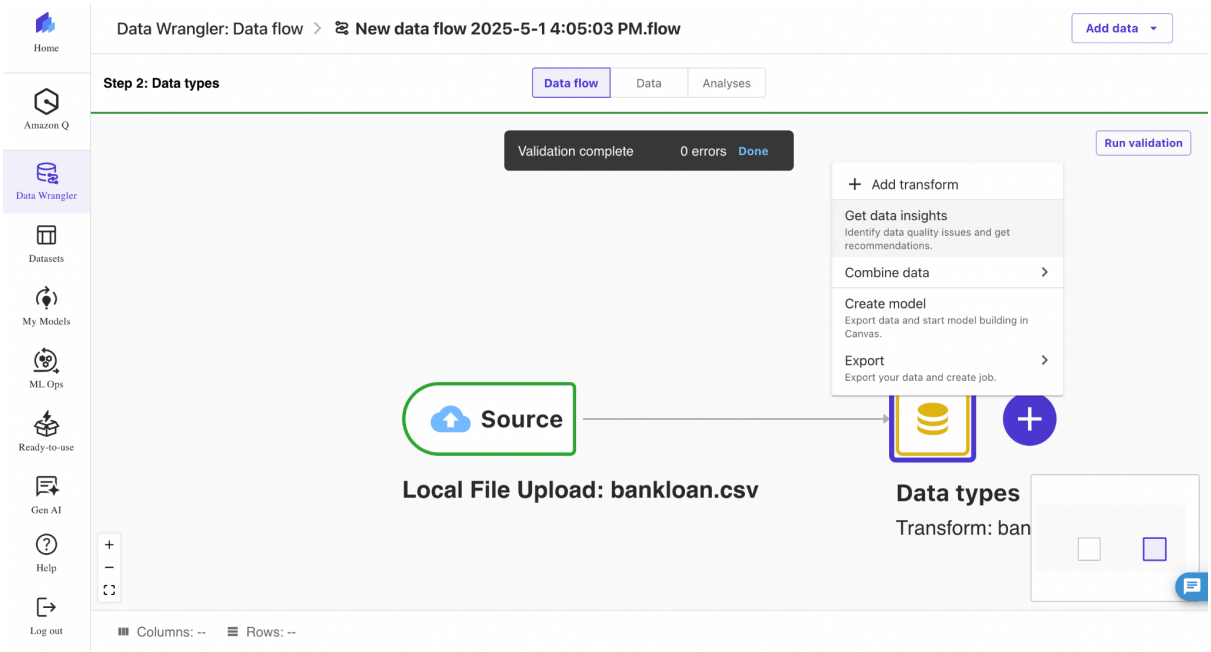
Select the data source type and upload it



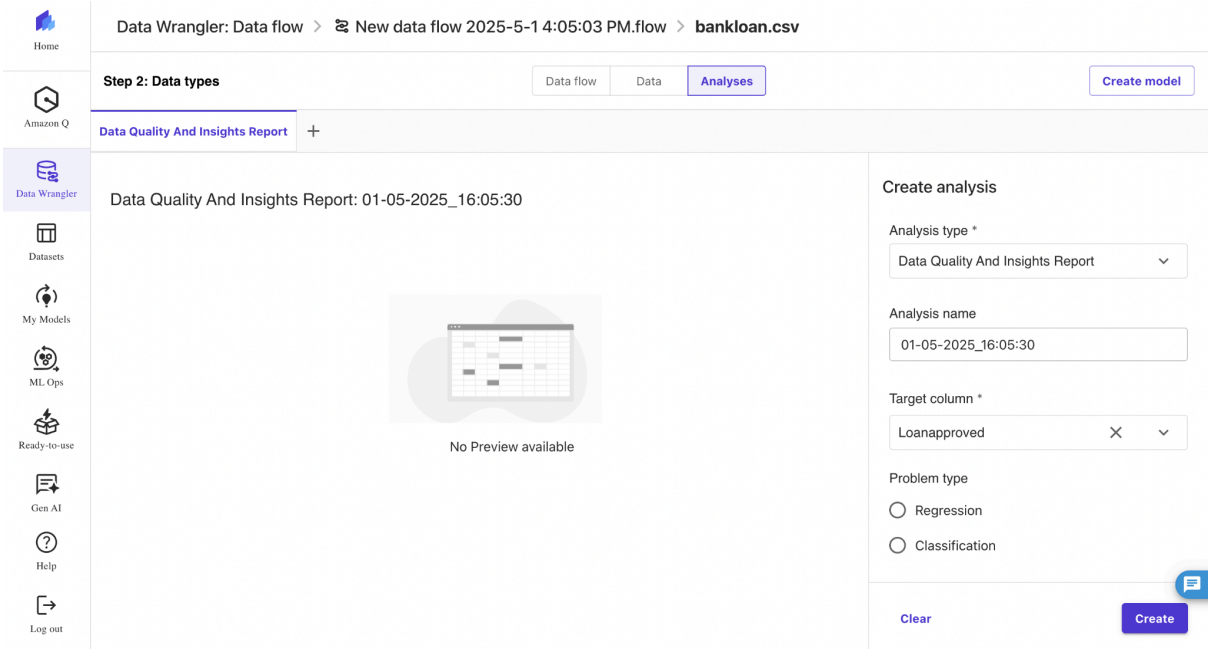
Click on import data



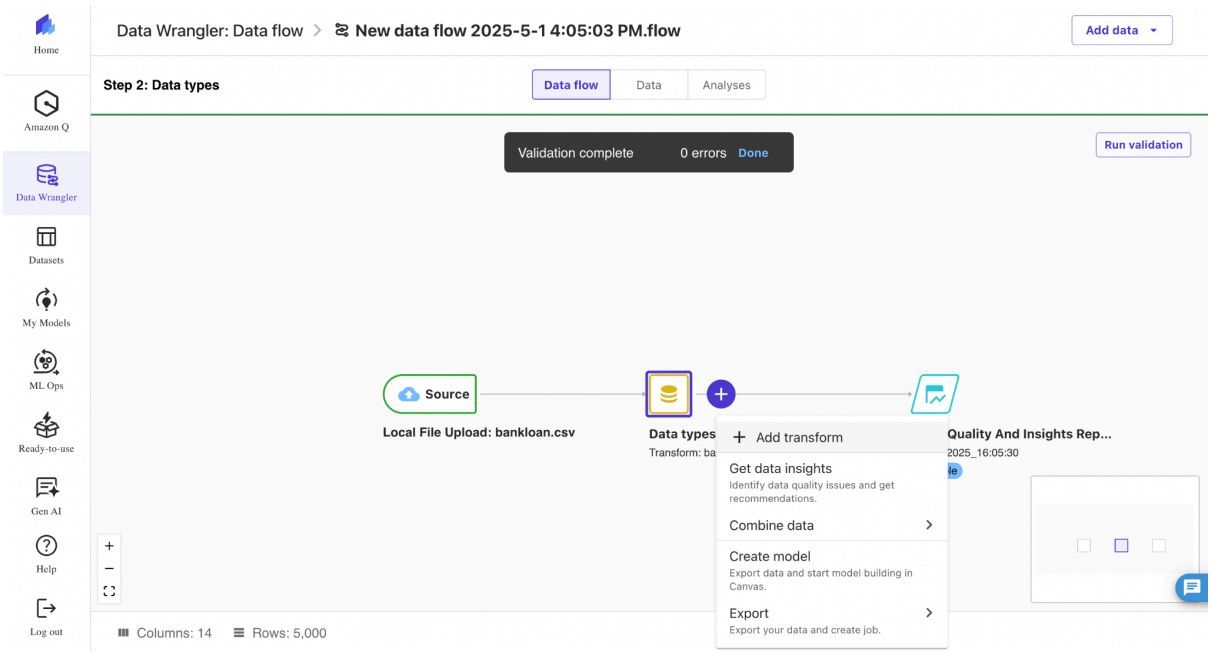
8.After Importing the Data it will appear as a flow- click on get data insights to create a Data Quality and Insights report.

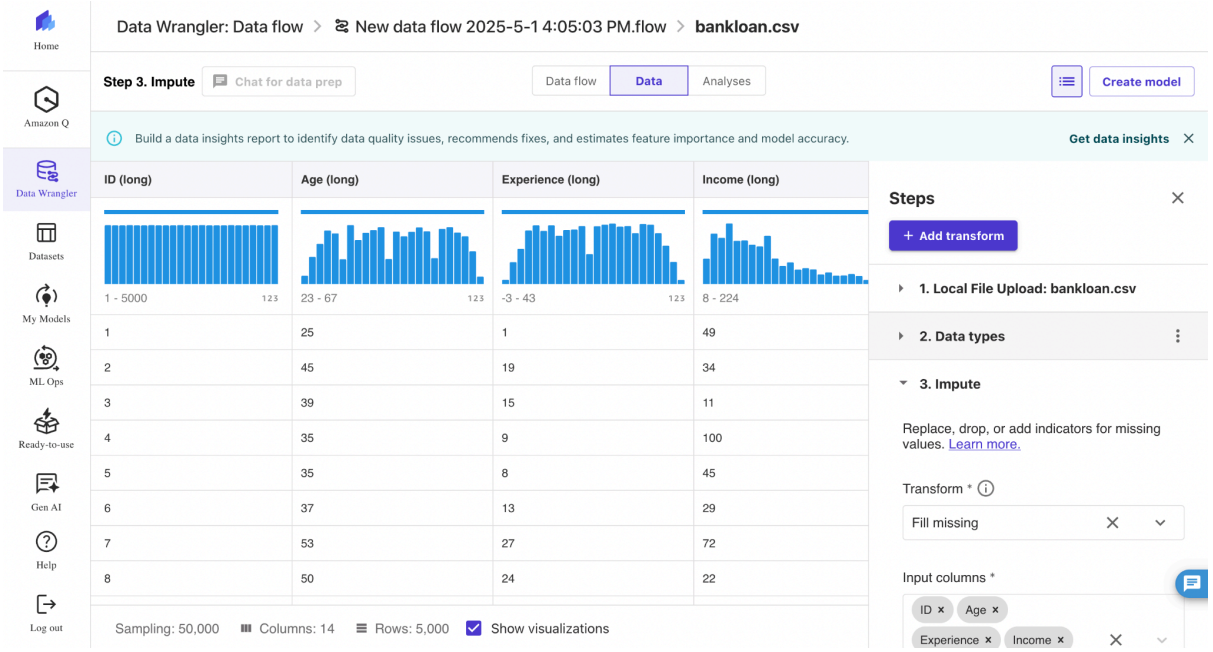


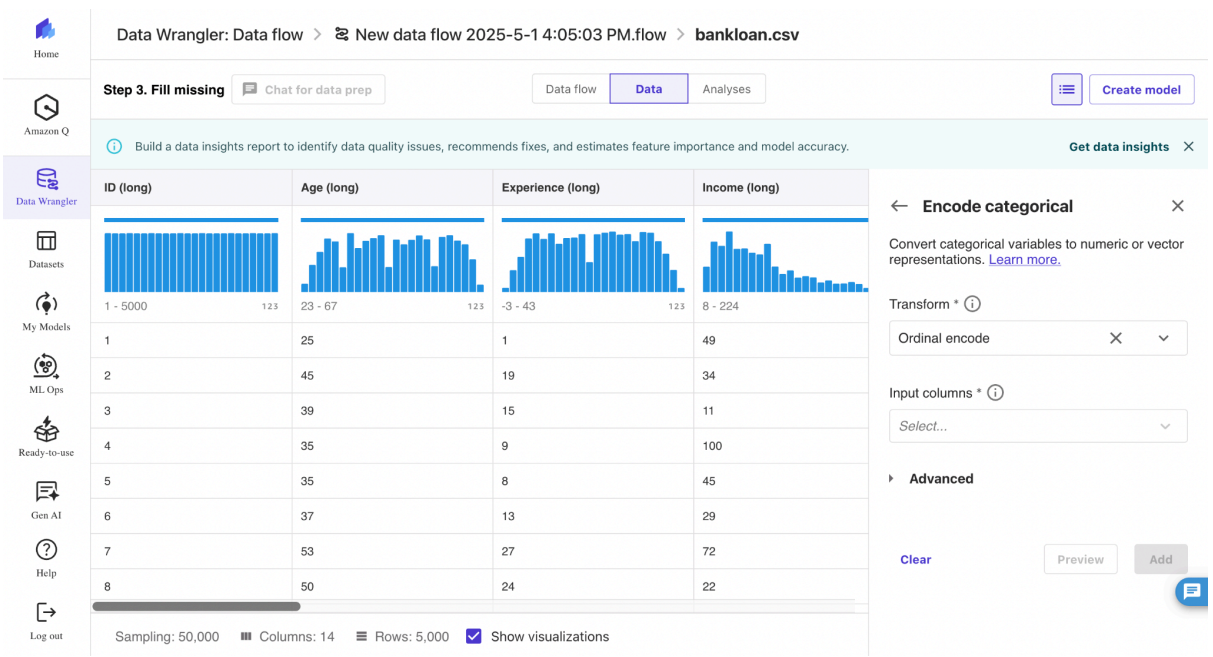
Click create

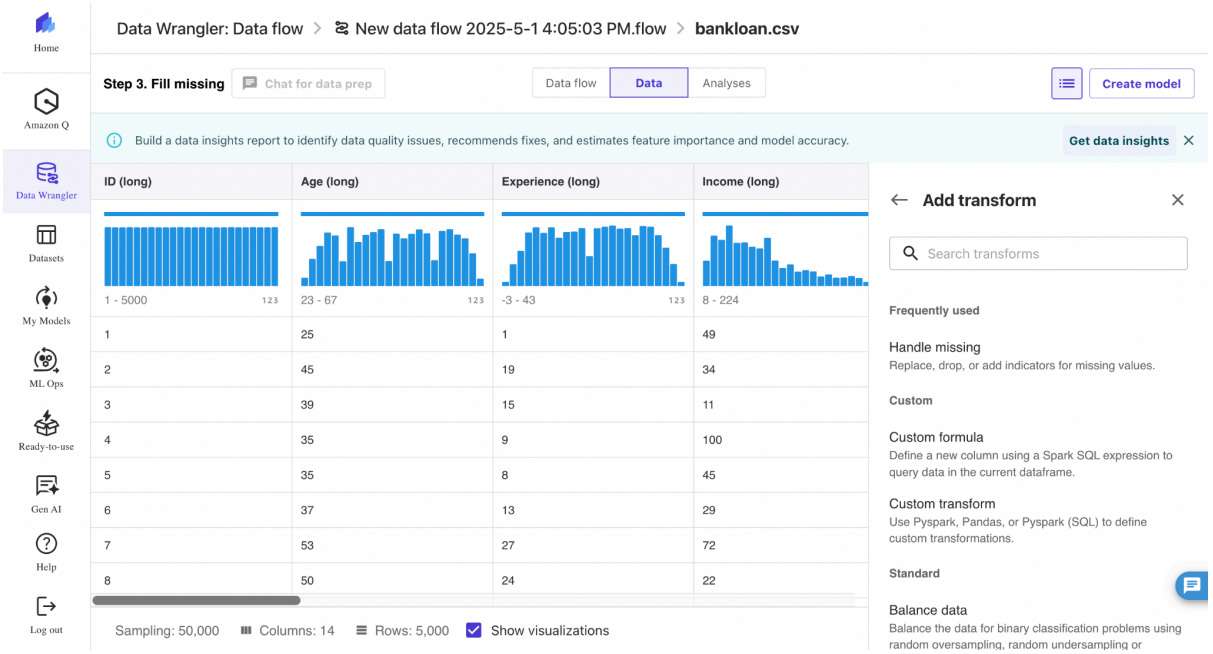


9.After Generating the report- go back to the data flow and click on add transform- this will allow you to apply any data preprocessing steps which are needed.

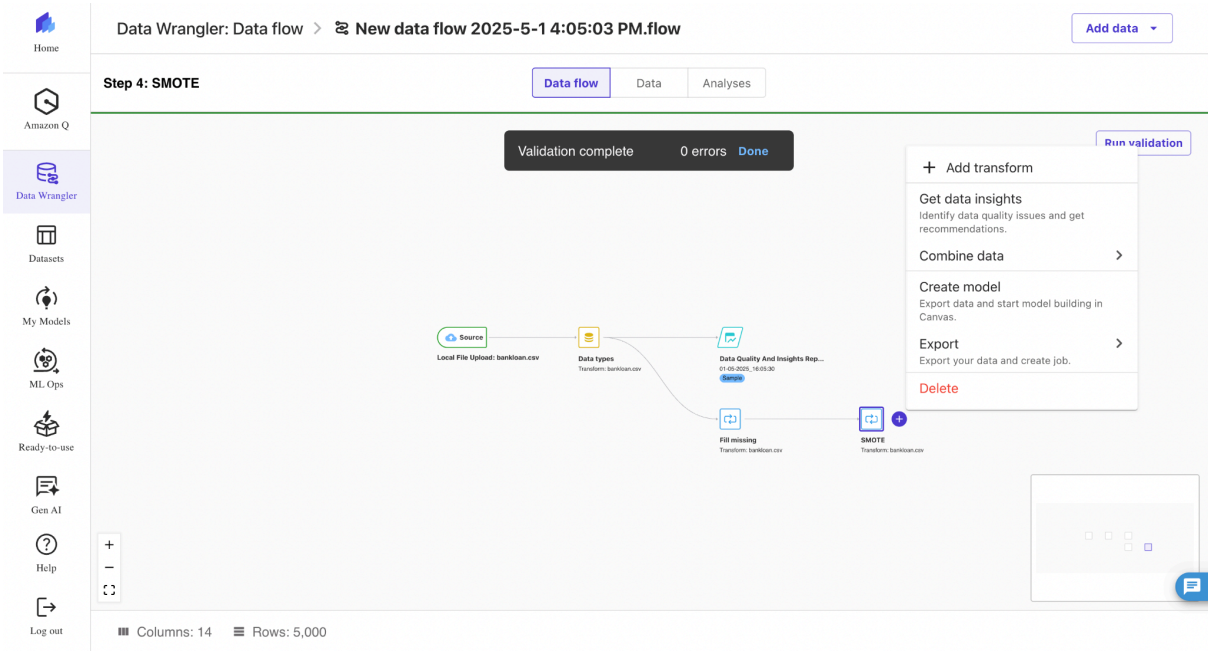



Home
Amazon Q
Data Wrangler
Datasets
My Models
ML Ops
Ready-to-use
Gen AI
Help
Log out
Data Wrangler: Data flow > New data flow 2025-5-1 4:05:03 PM.flow > bankloan.csv
Step 3. Impute
Chat for data prep
Data flow
Data
Analyses
Create model
Build a data insights report to identify data quality issues, recommends fixes, and estimates feature importance and model accuracy.
Get data insights
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
Age
Experience
Income

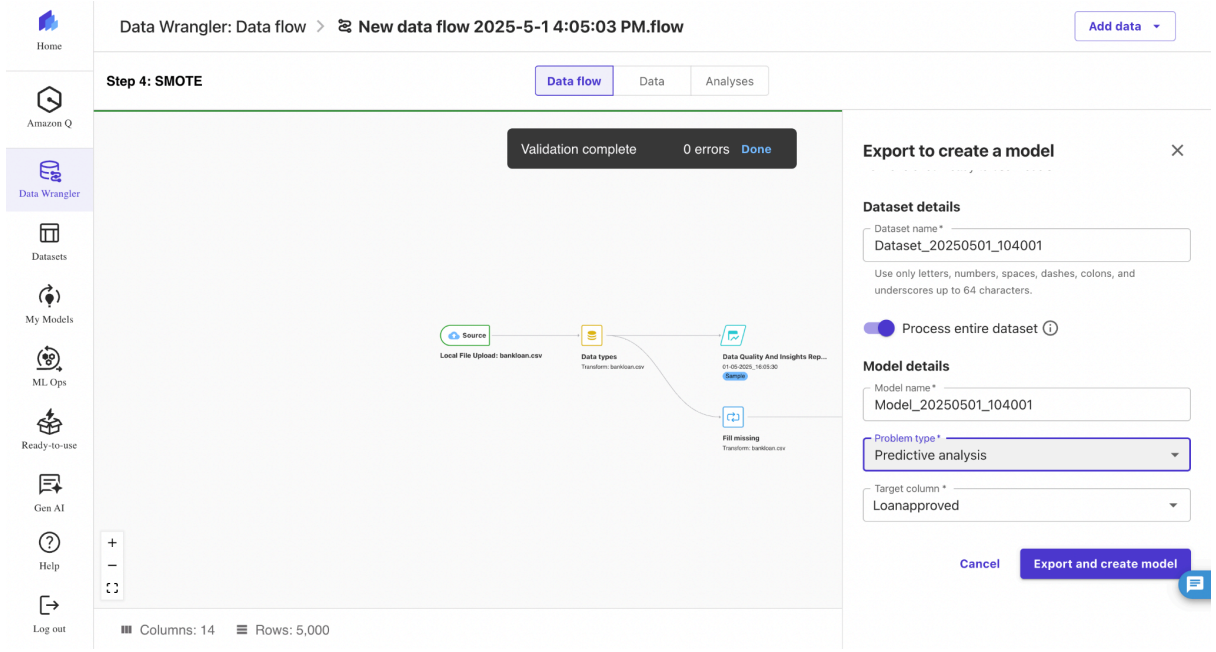

Home
Amazon Q
Data Wrangler
Datasets
My Models
ML Ops
Ready-to-use
Gen AI
Help
Log out
Data Wrangler: Data flow > New data flow 2025-5-1 4:05:03 PM.flow > bankloan.csv
Step 3. Fill missing
Chat for data prep
Data flow
Data
Analyses
Create model
Build a data insights report to identify data quality issues, recommends fixes, and estimates feature importance and model accuracy.
Get data insights
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

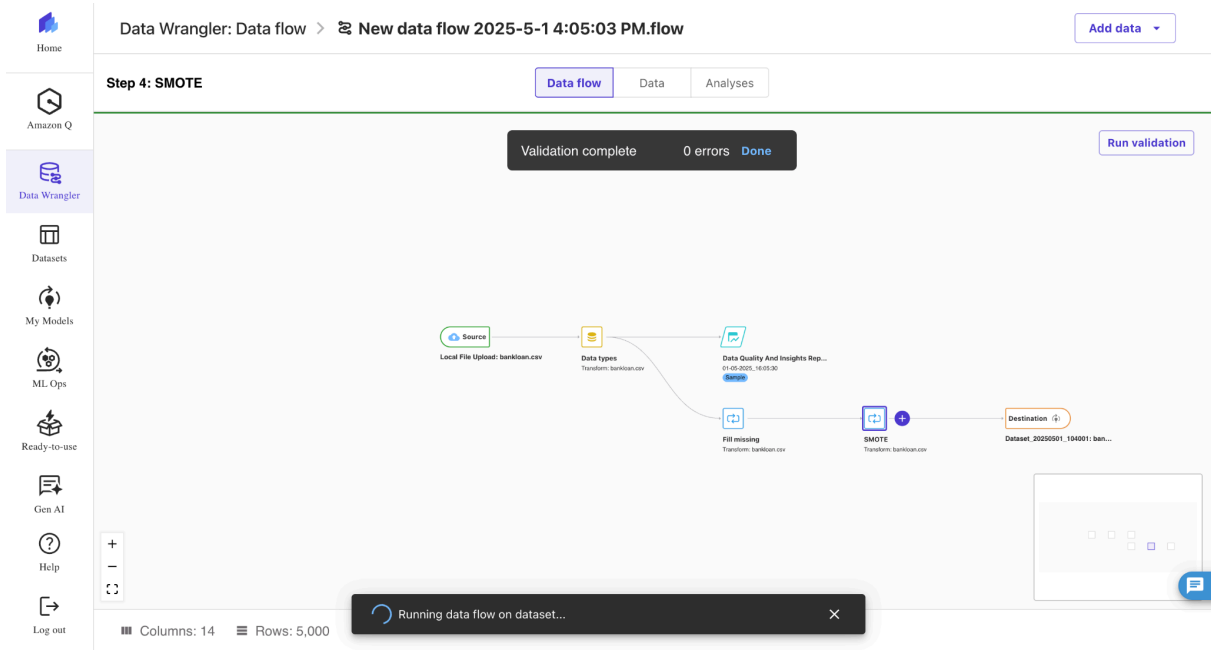


10.After adding any data transformation steps which are required click on create model

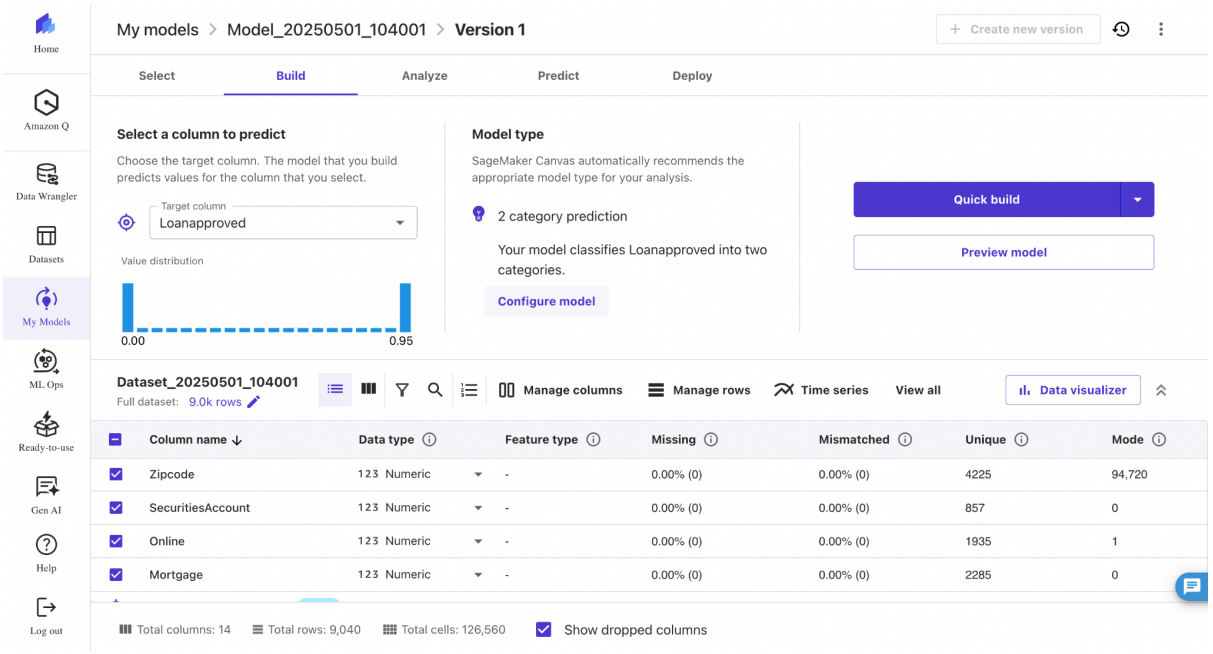


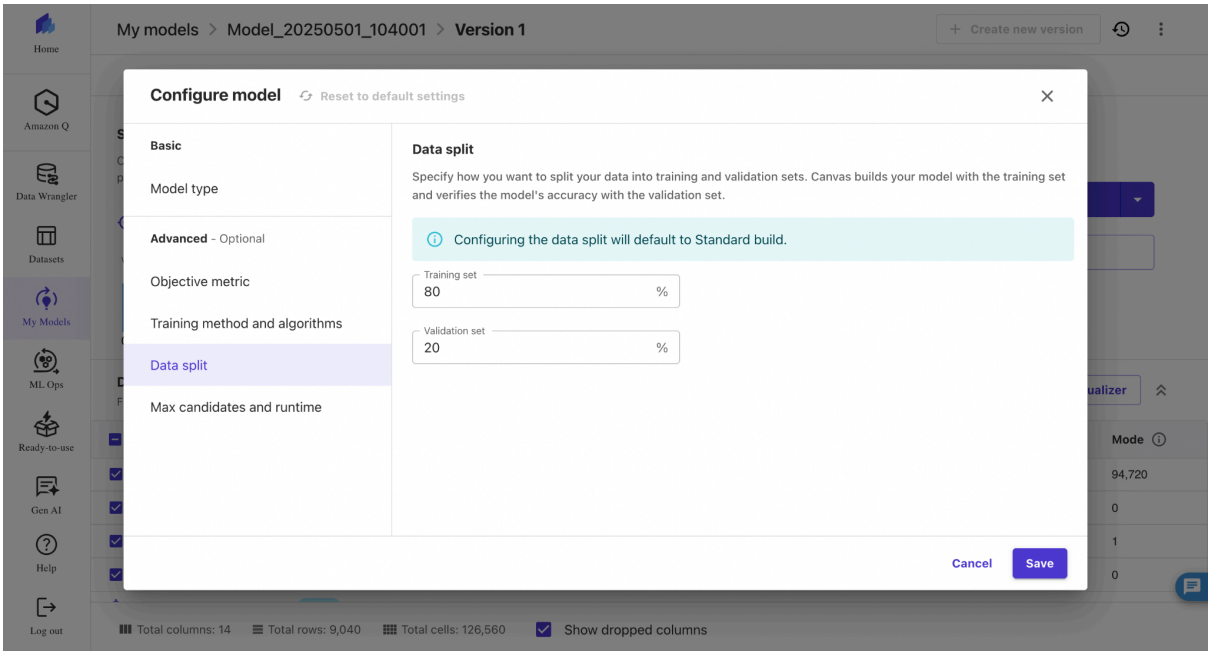
Click on export and create model



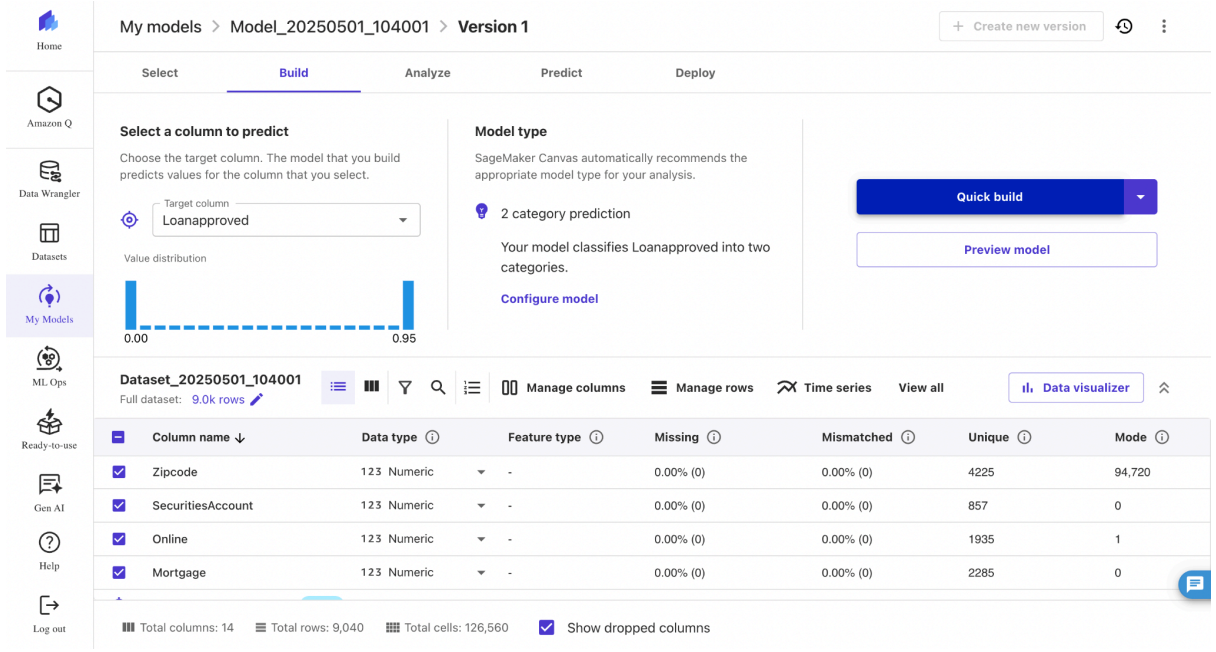


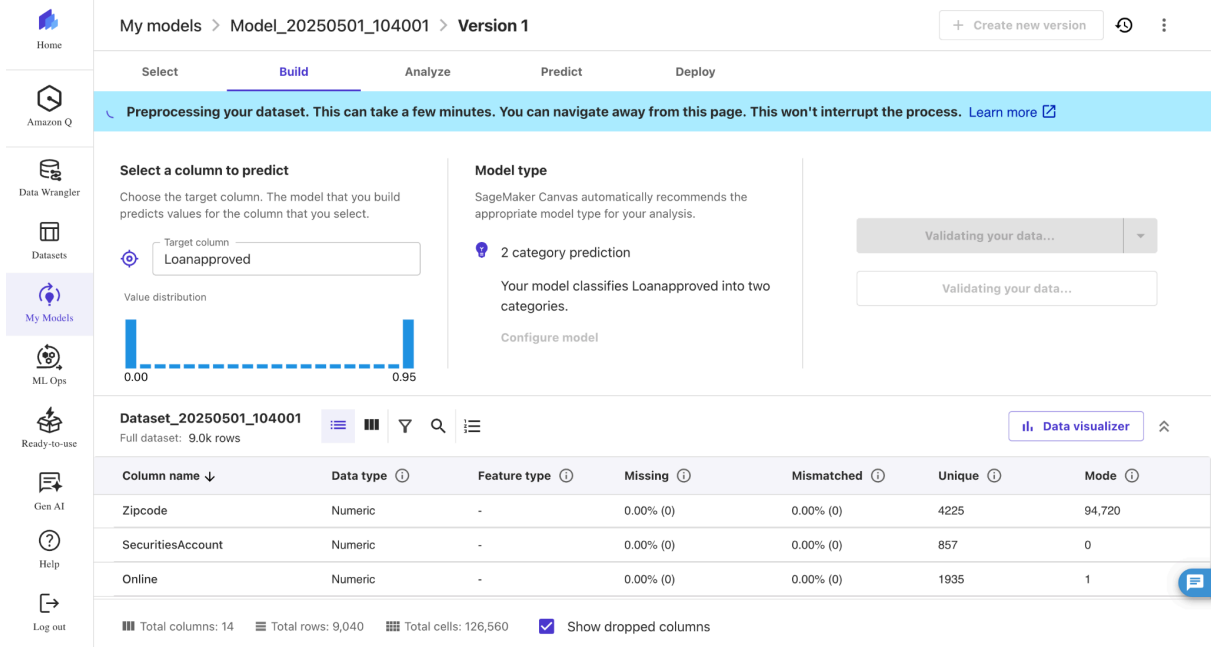
Click on configure model to choose the model type, objective metric, training method and algorithms, Data Split and Max candidates and runtime



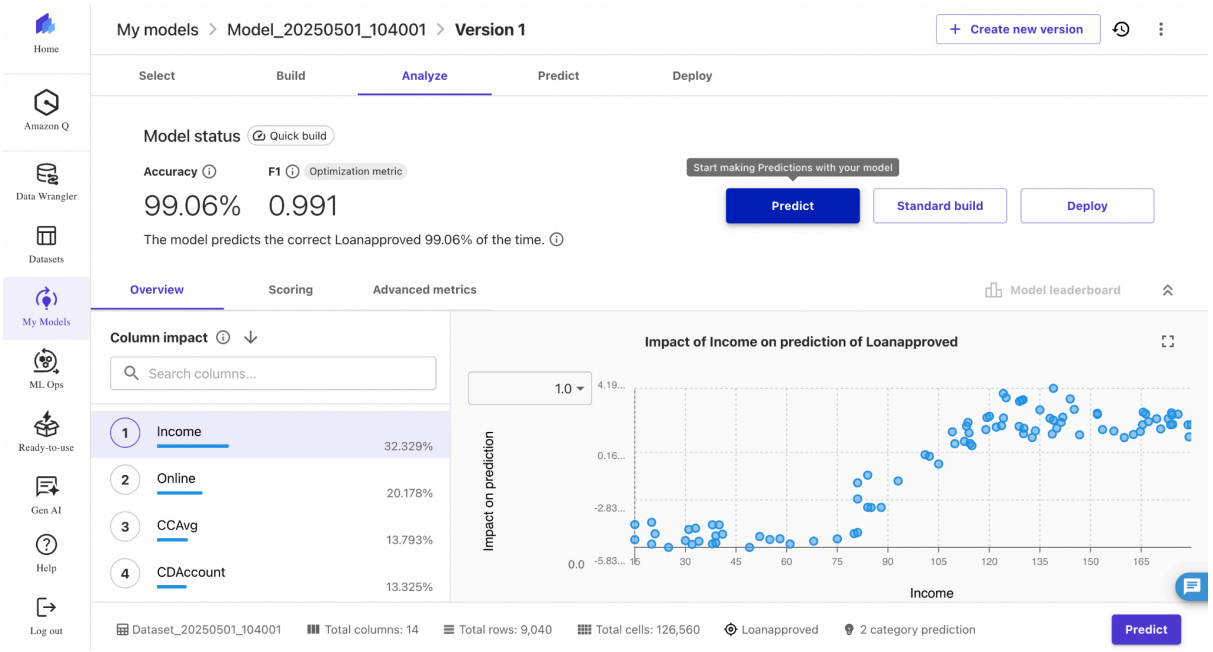


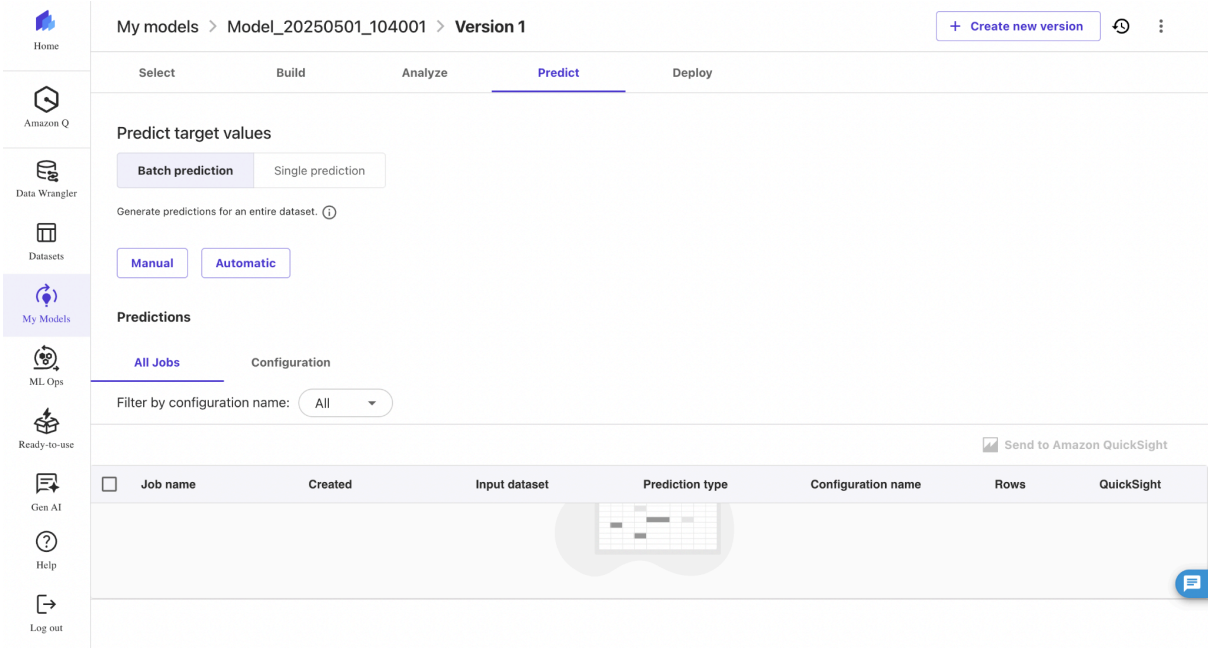
After configuring the model click on quick build





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





Select dataset for predictions

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
○	Dataset_20250501_104001	V1	14	9,040	126,560	05/01/2025 4:10 PM	Ready
○	canvas-sample-housing.csv	V1	10	1,000	10,000	05/01/2025 4:02 PM	Incompatible
○	canvas-sample-shipping-logs.csv	V1	12	1,000	12,000	05/01/2025 4:02 PM	Incompatible
○	canvas-sample-product-descriptions.csv	V1	5	120	600	05/01/2025 4:02 PM	Incompatible
○	canvas-sample-loans-part-2.csv	V1	5	1,000	5,000	05/01/2025 4:02 PM	Incompatible
○	canvas-sample-loans-part-1.csv	V1	19	1,000	19,000	05/01/2025 4:02 PM	Incompatible
○	canvas-sample-diabetic-readmission.csv	V1	16	1,000	16,000	05/01/2025 4:02 PM	Incompatible
○	canvas-sample-retail-electronics-forecasting.csv	V1	6	40,500	243,000	05/01/2025 4:02 PM	Incompatible

batchInfer-Model_20250501_104001-Dataset_20250501_104001-1746096461 predictions ready **View**

Close | Generate predictions

My models > Model_20250501_104001 > **Version 1**

+ Create new version

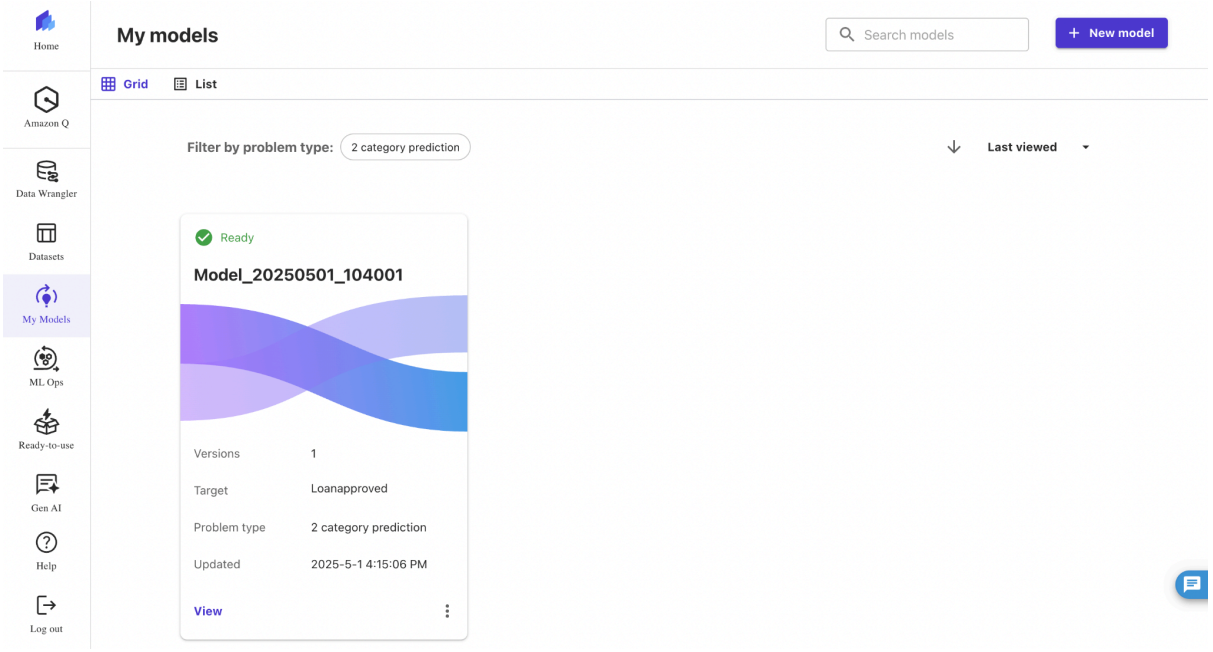
batchInfer-Model_20250501_104001-Dataset_20250501_104001-1746096461

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

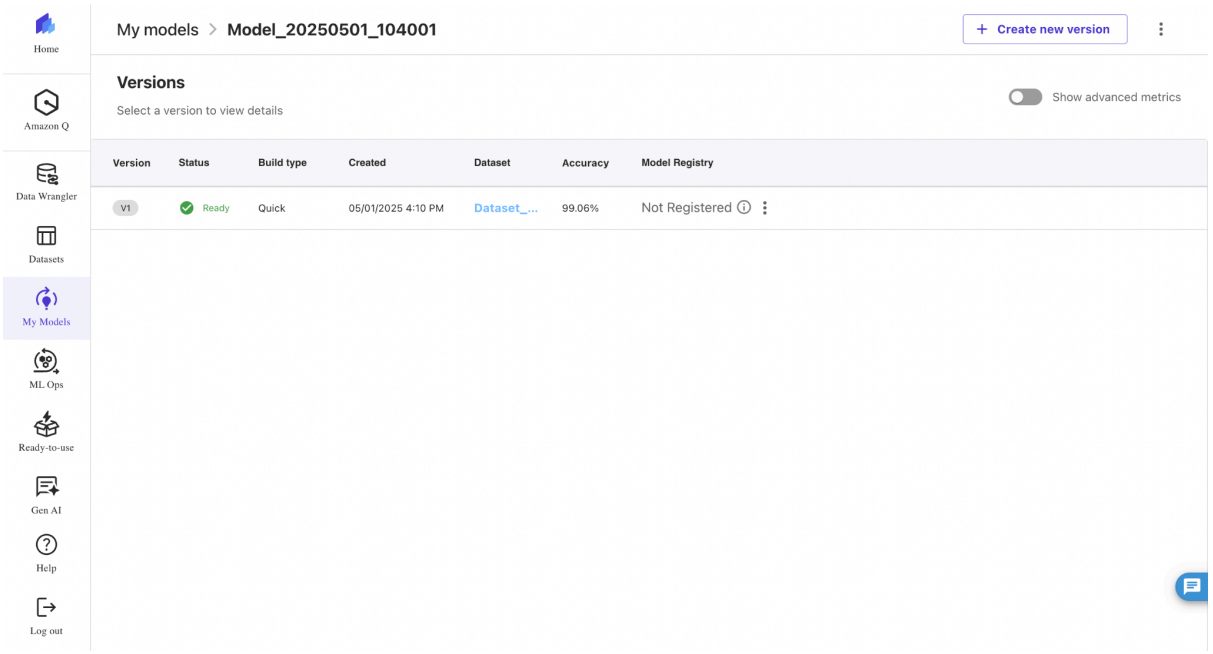
Send to Amazon QuickSight | Download

batchInfer-Model_20250501_104001-Dataset_20250501_104001-1746096461 predictions ready **View**

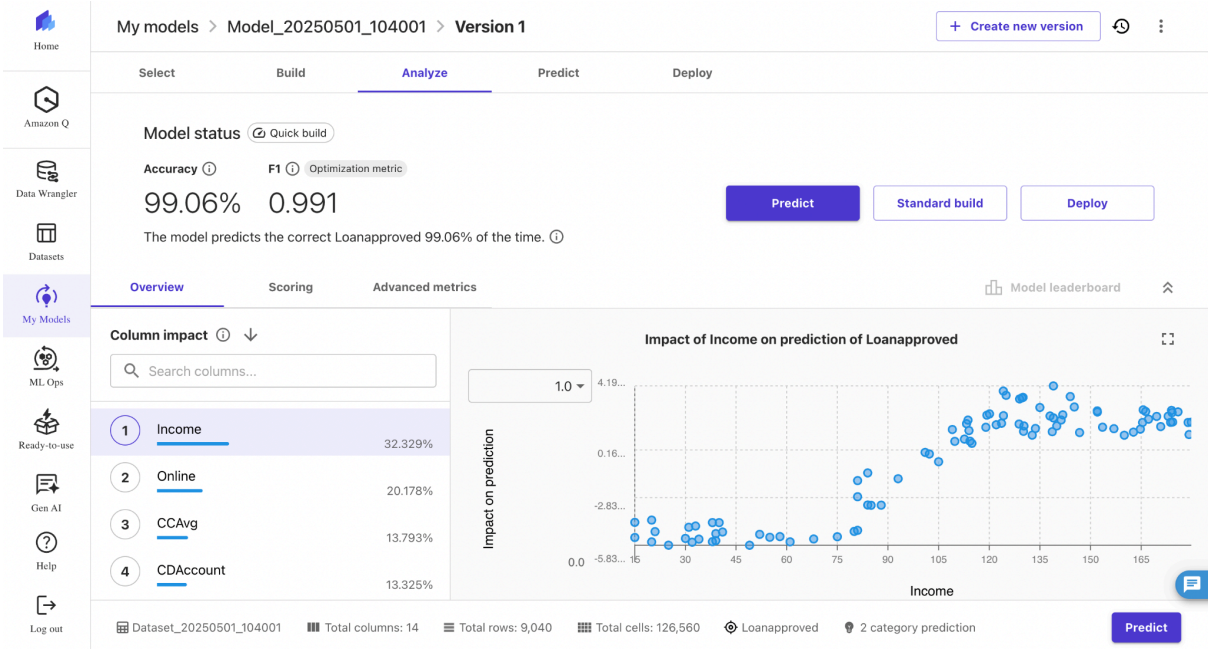
12. Deploying the model- Go to My Models and select view on model you want to deploy

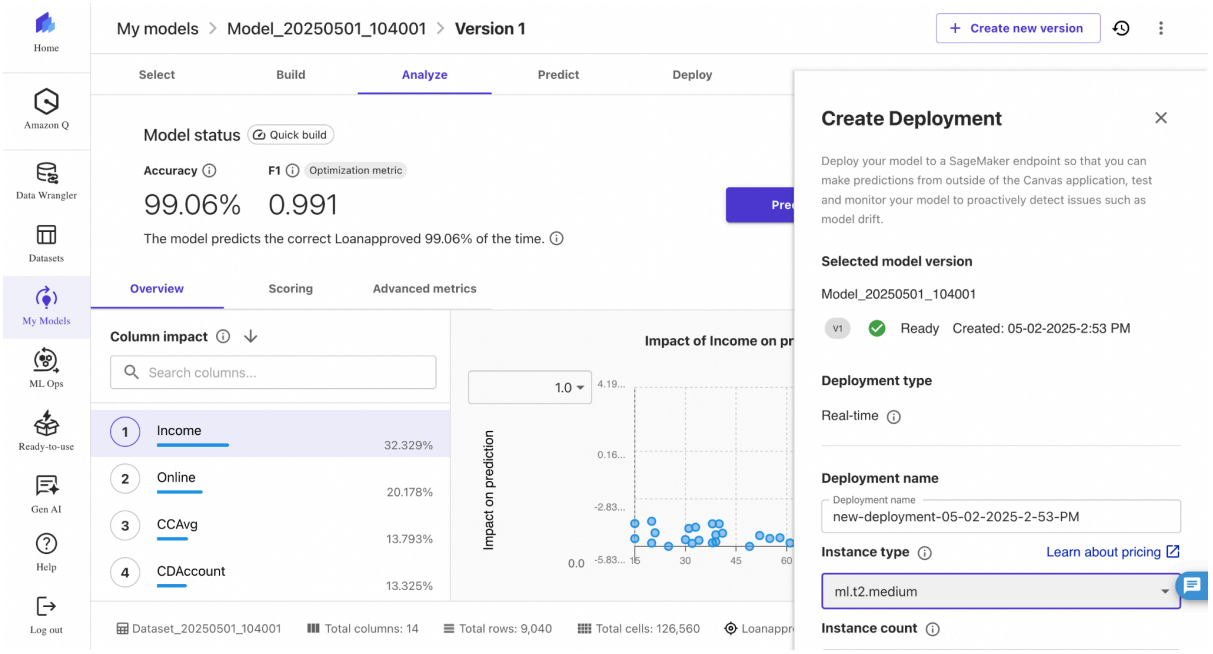


Click on the model again

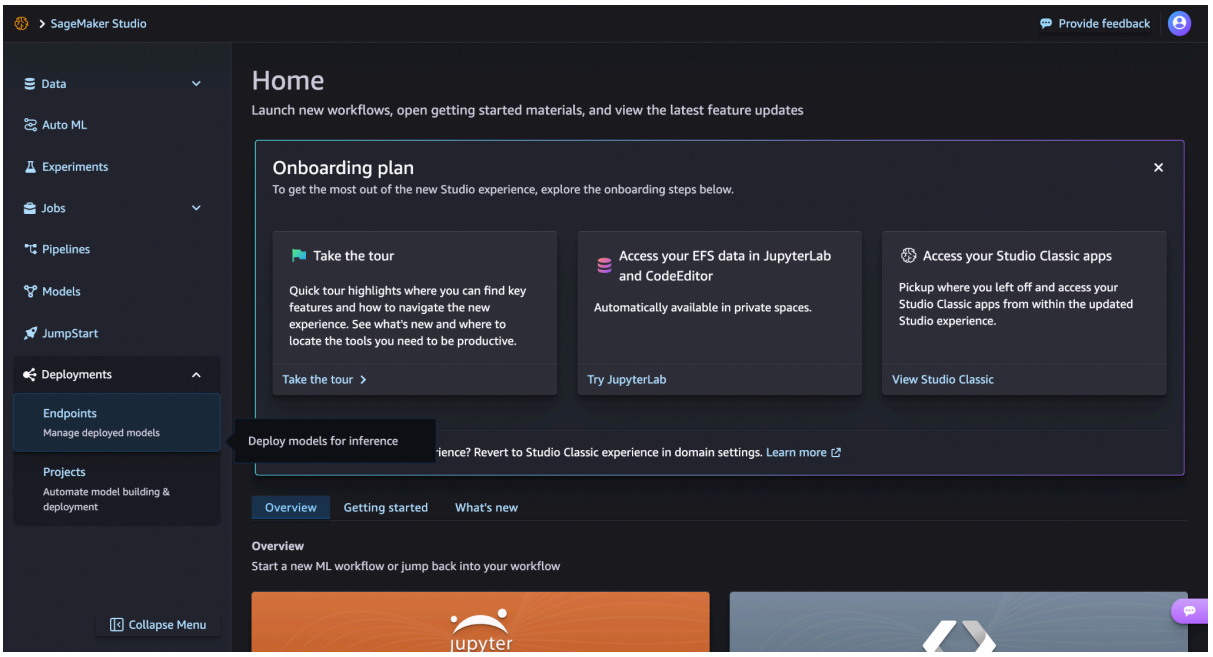


Click on deploy and create deployment

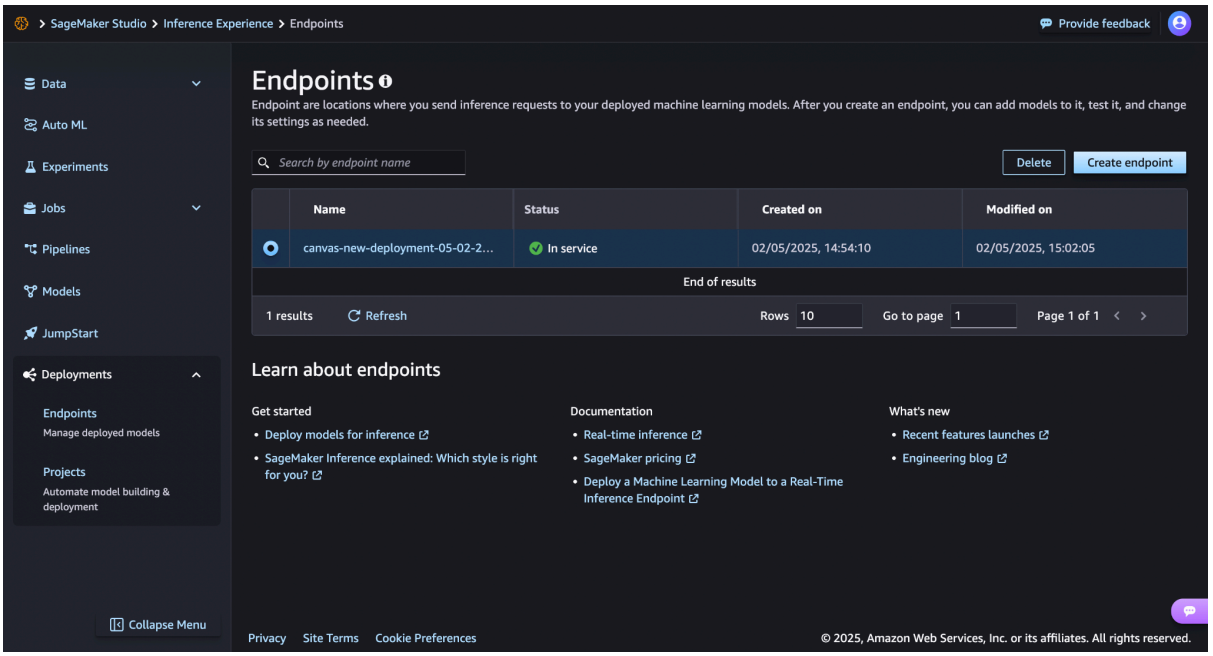




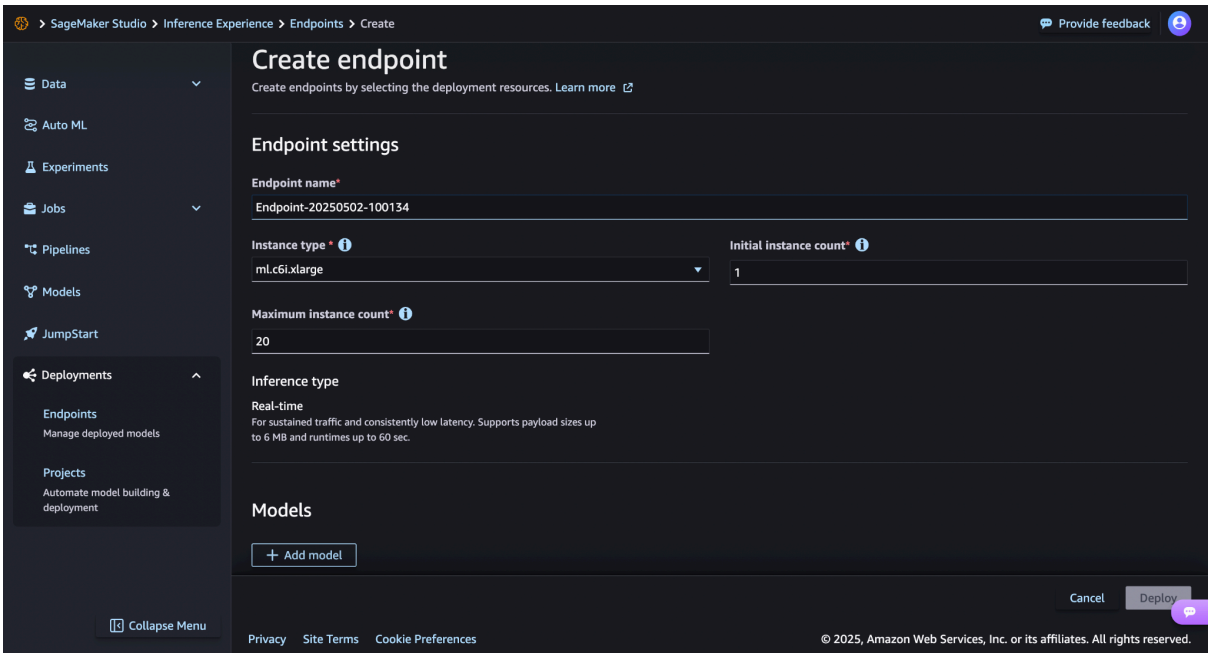
13- To create an endpoint- we go back to SageMaker AI studio and select endpoints from deployments



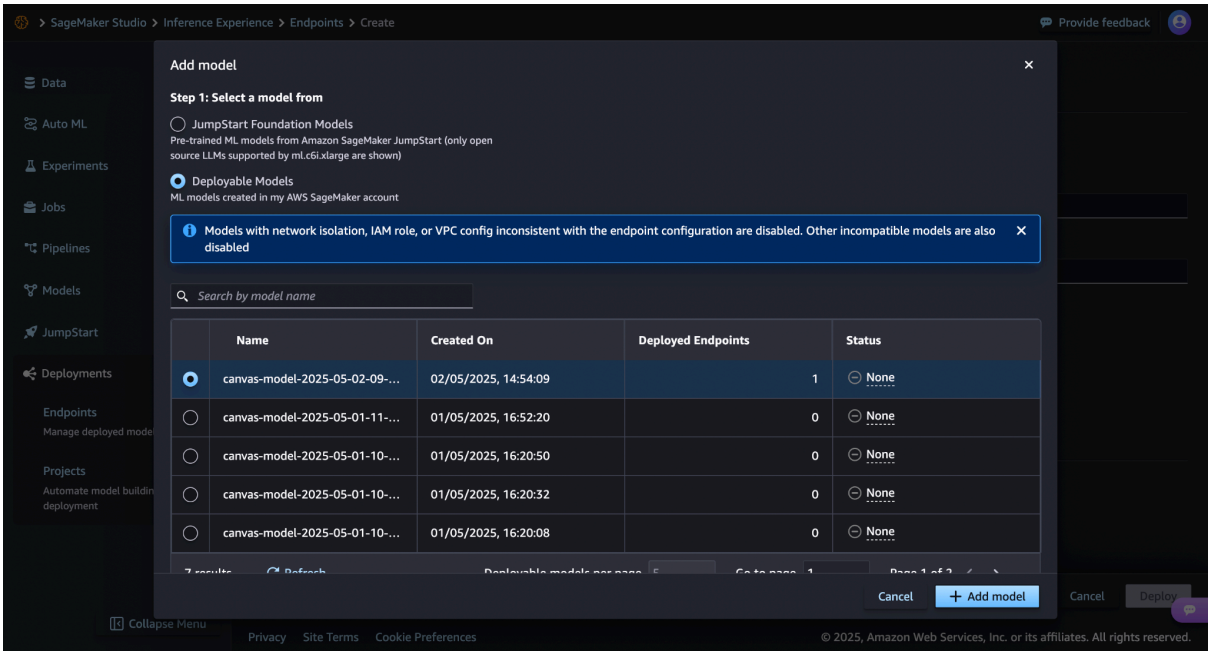
Here click on create endpoint



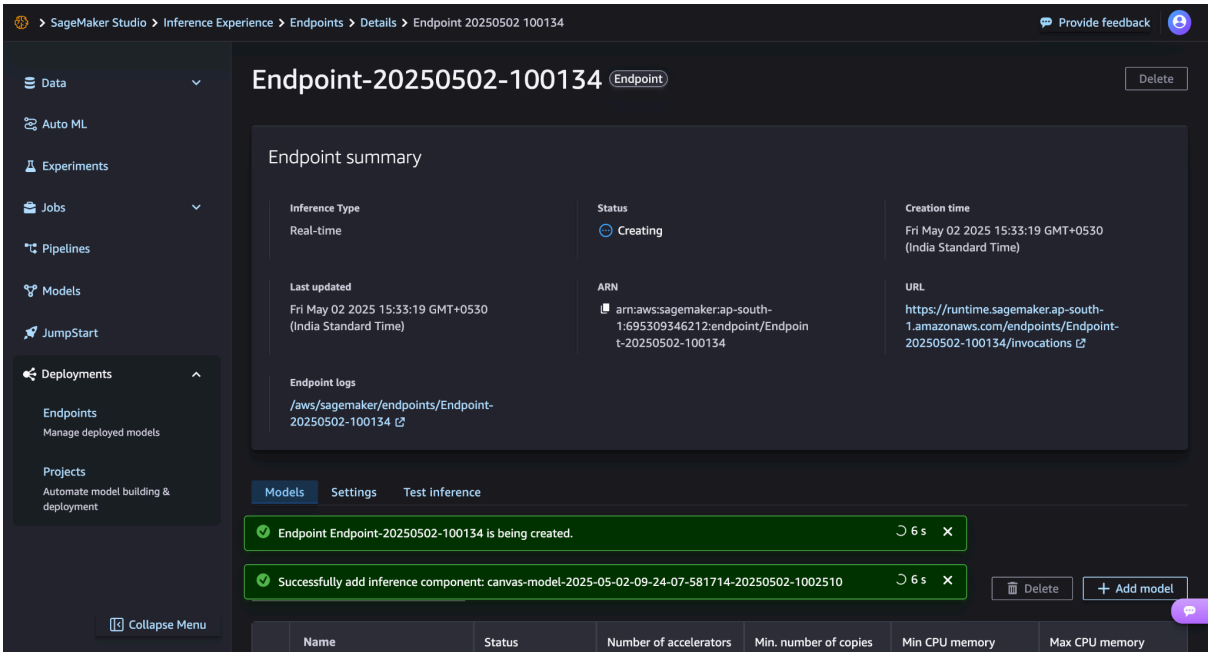
Click on add model and select the ML model we created



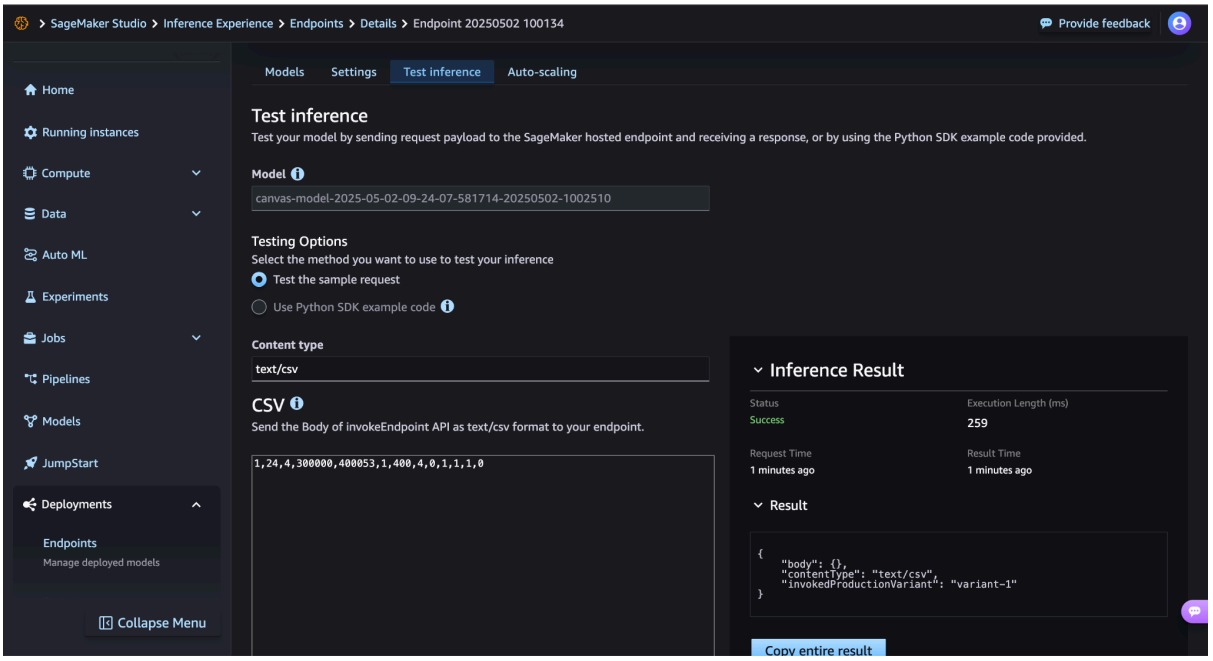
Choose deployable model as we have created the model on AWS Sagemaker



Create the endpoint

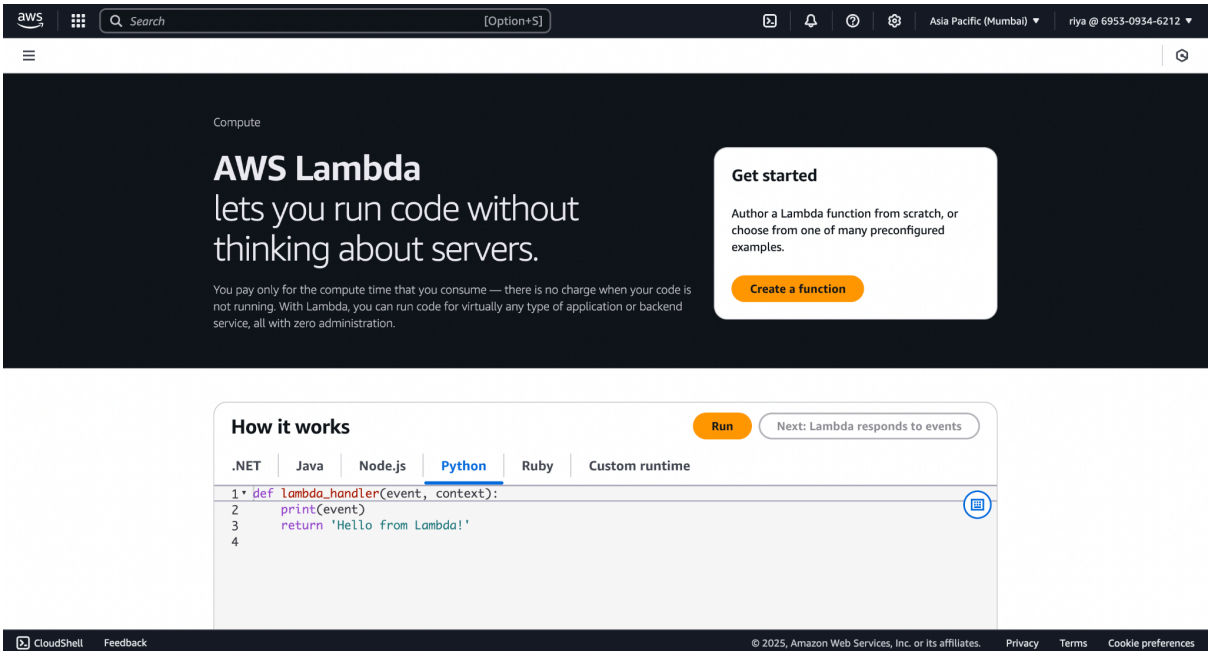


To test the endpoint click on
Test inference,
choose text/csv
paste- 1,24,4,300000,400053,1,400,4,0,1,1,1,0



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

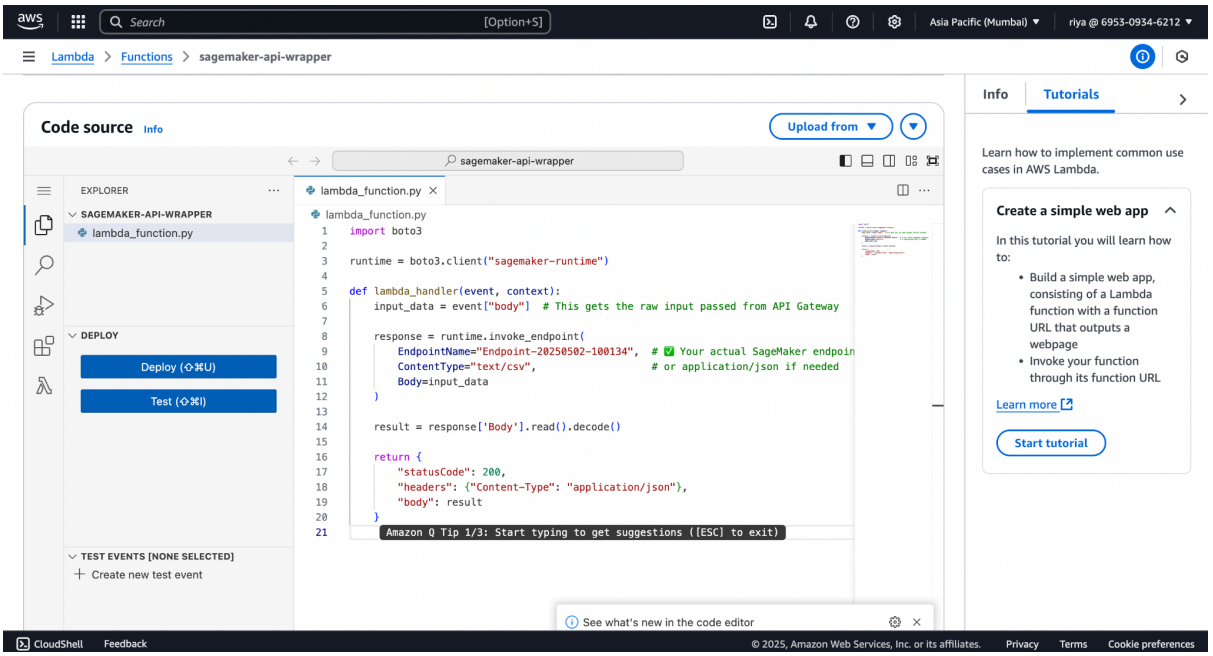
Open AWS Lambda and click on create a function



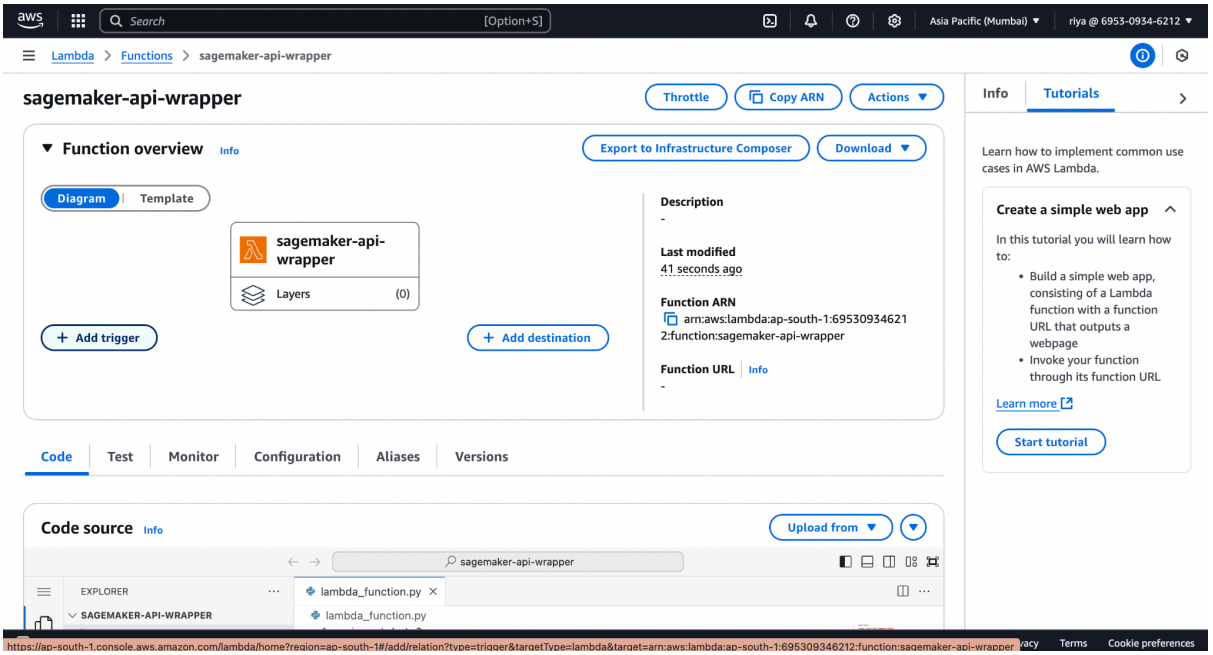
Scroll down to code source and put the following code in and click deploy-

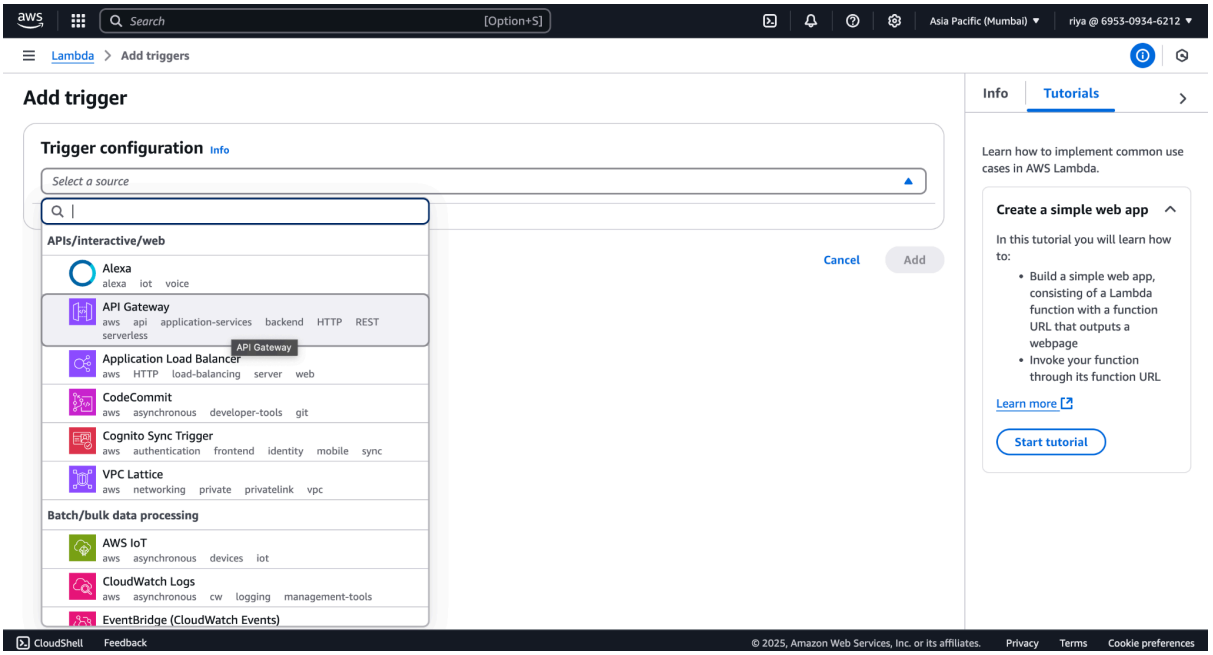
```
import boto3
import json

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)
    }
```

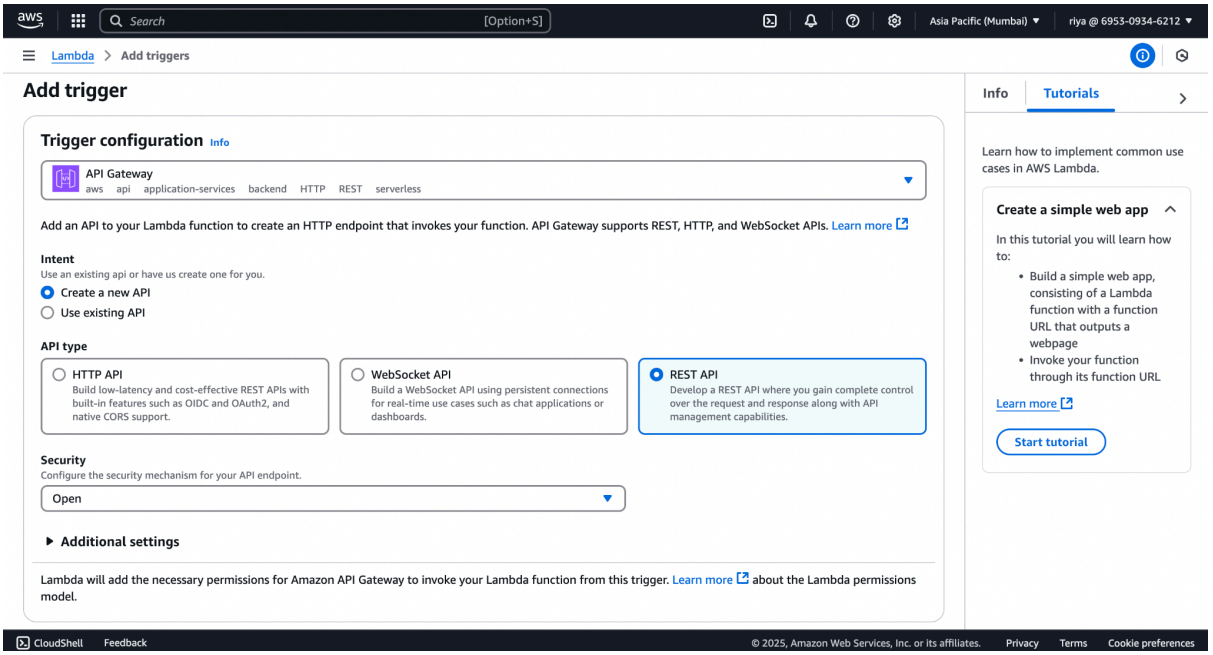


Then go to add trigger

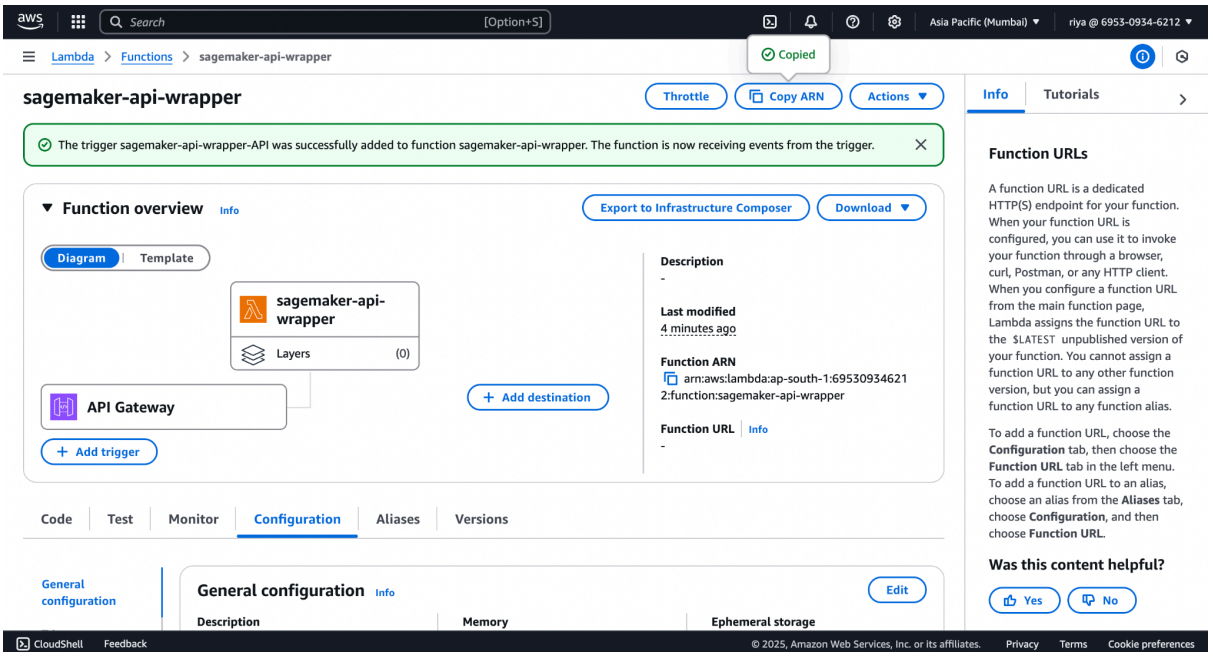




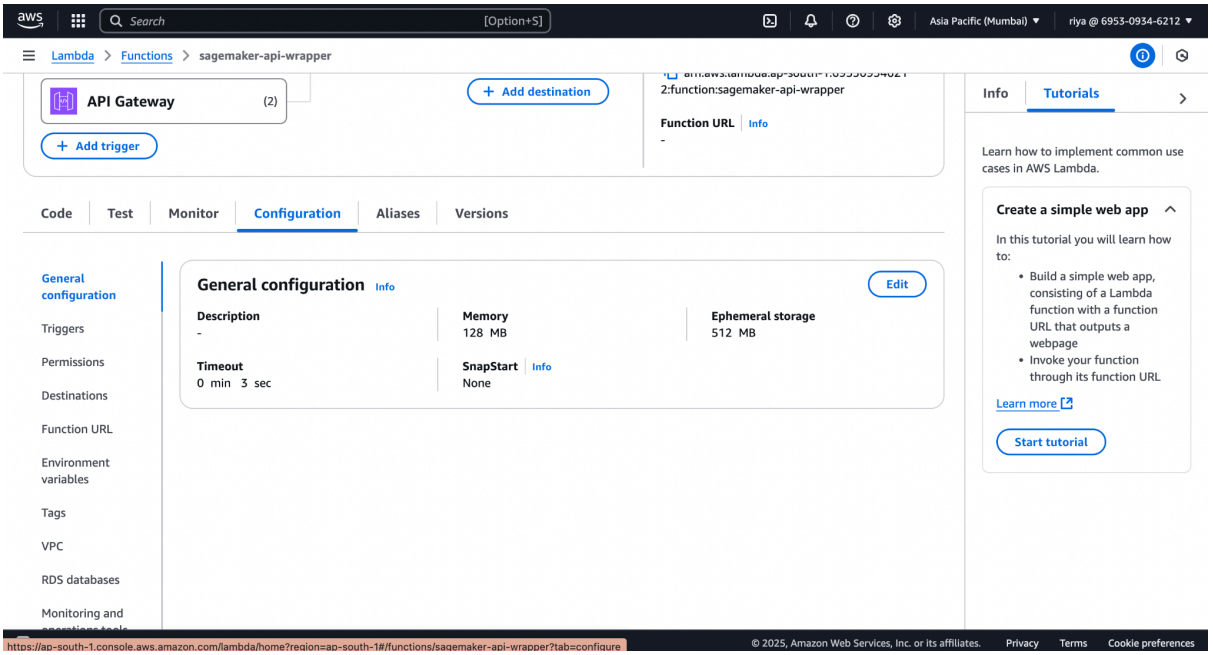
Select API gateway as the source then select creating new API as intent and API Type- REST API



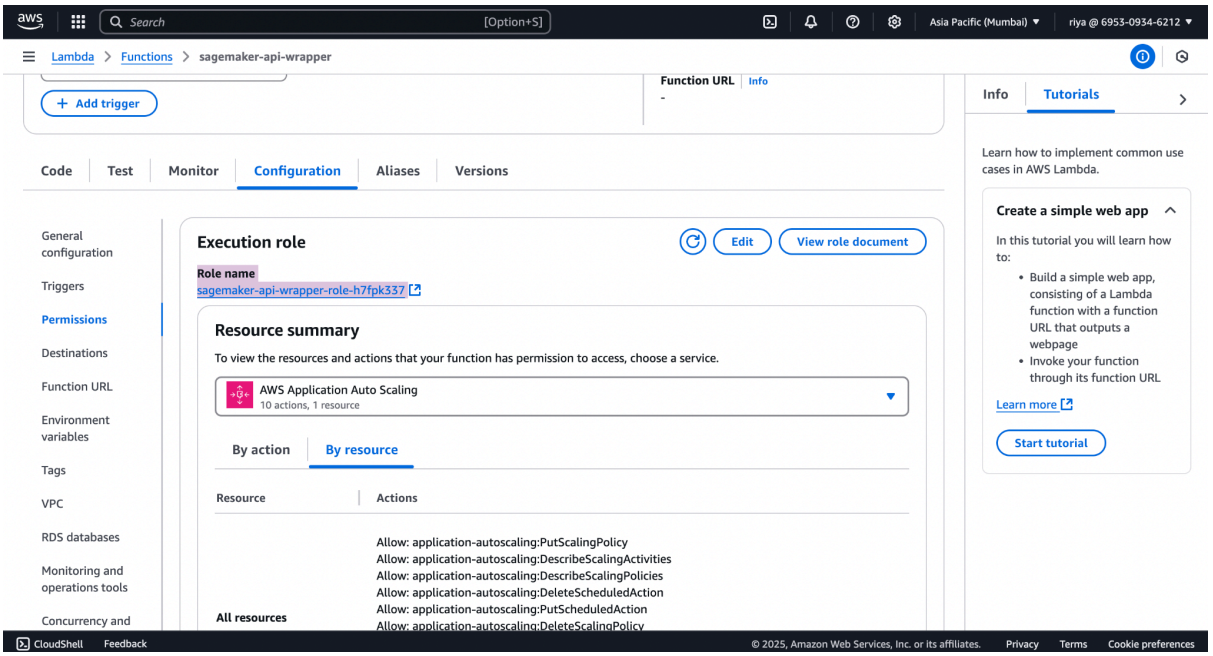
Copy the ARN



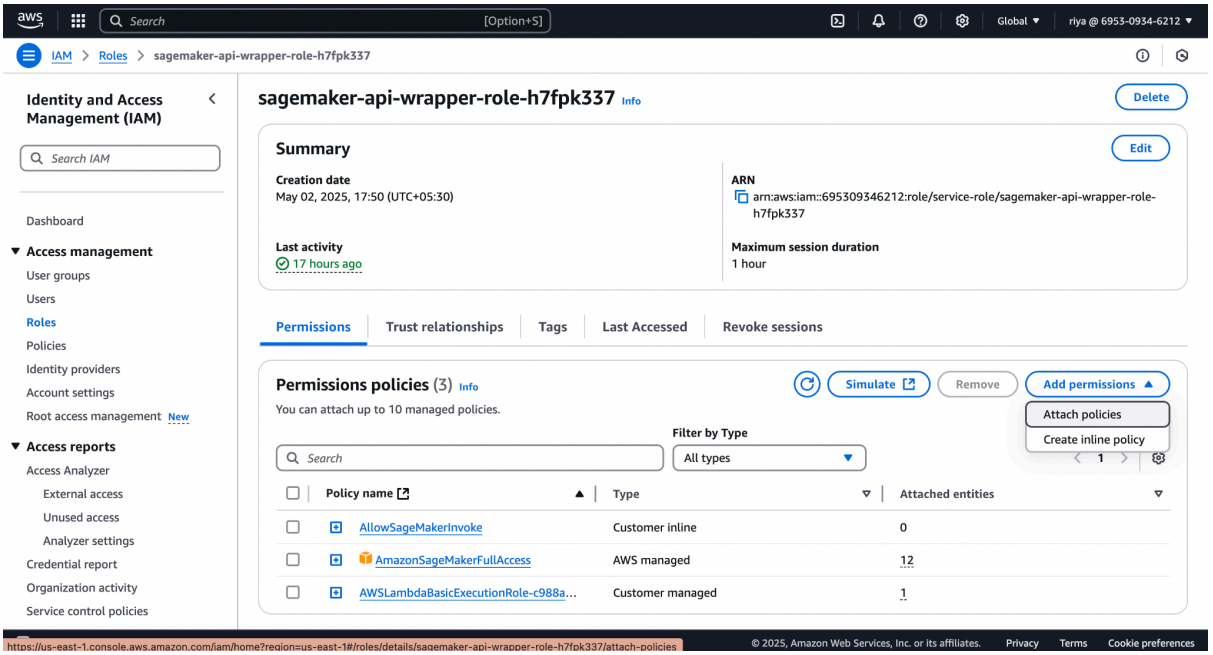
15- To add permissions- go to configuration



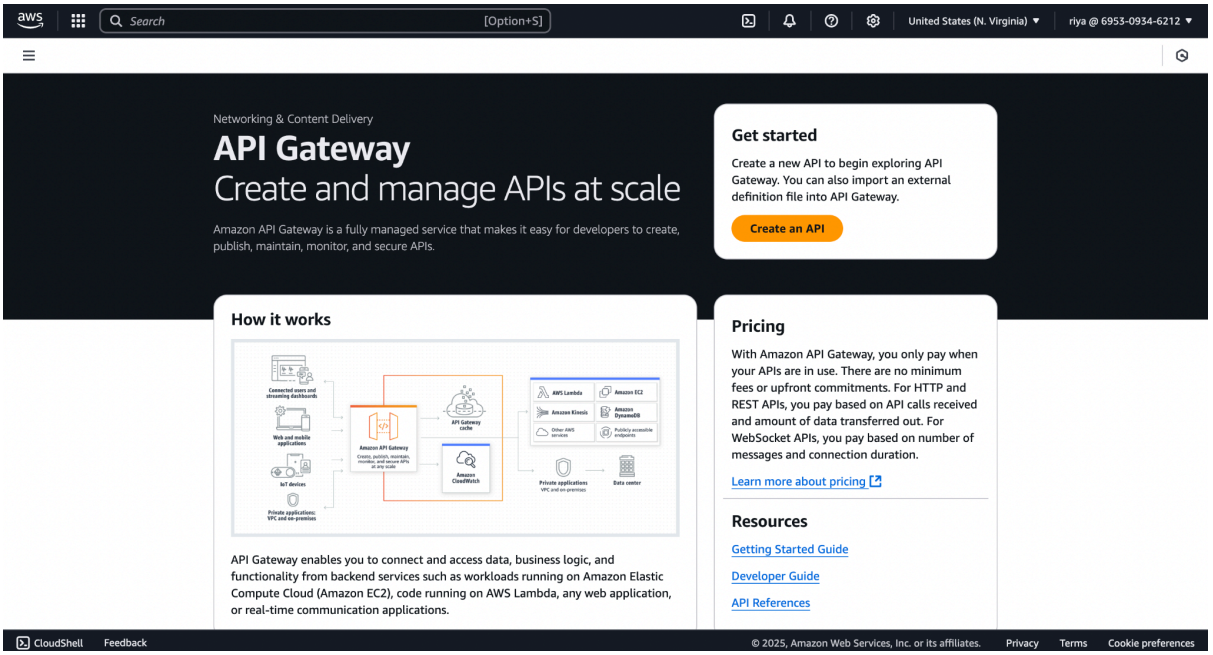
Then click on permissions and click on the role name link



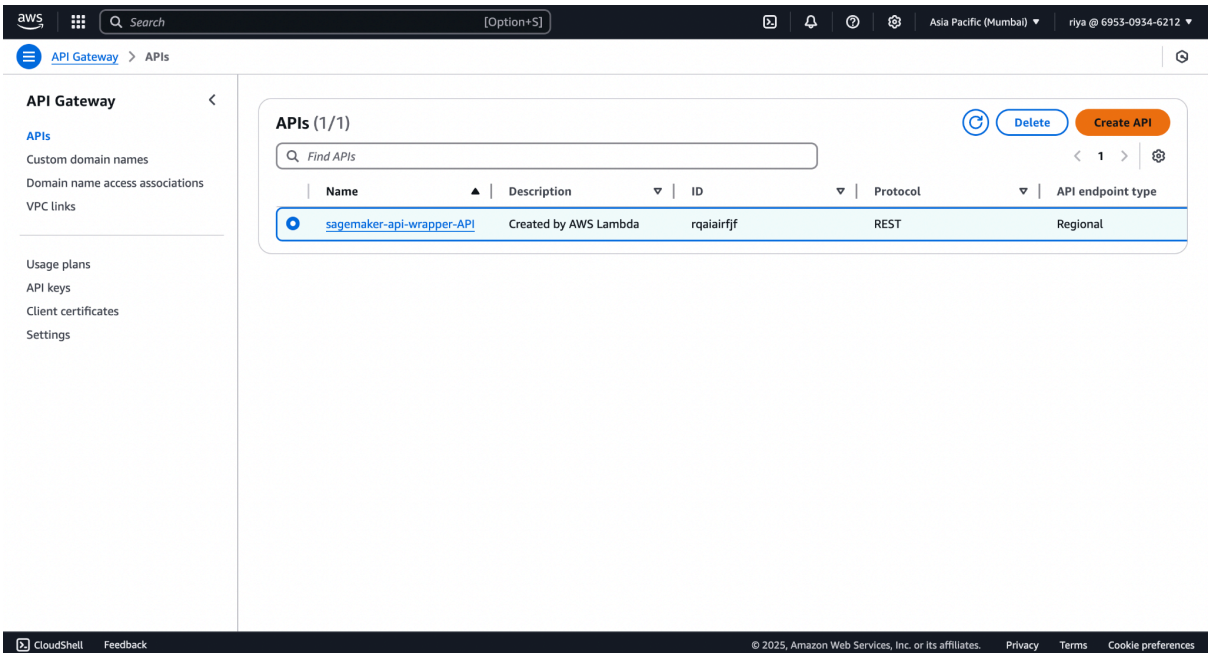
Then click on add permissions- attach policies and search for AmazonSageMakerFullAccess and add it



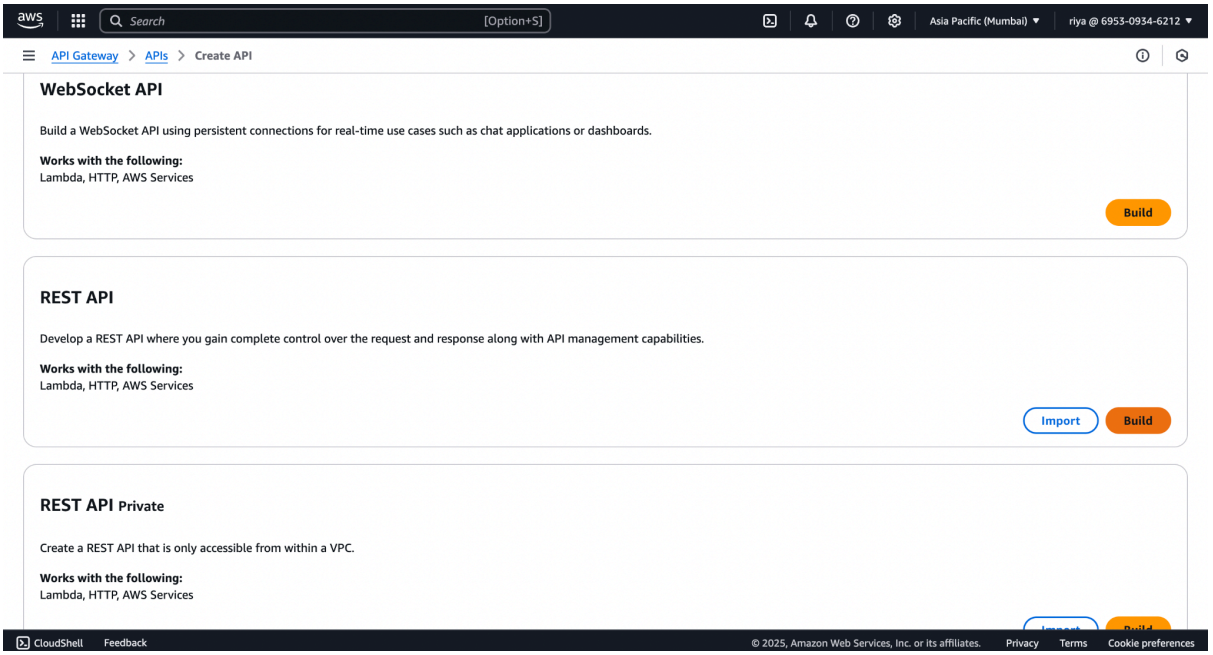
16- Go to API gateway.

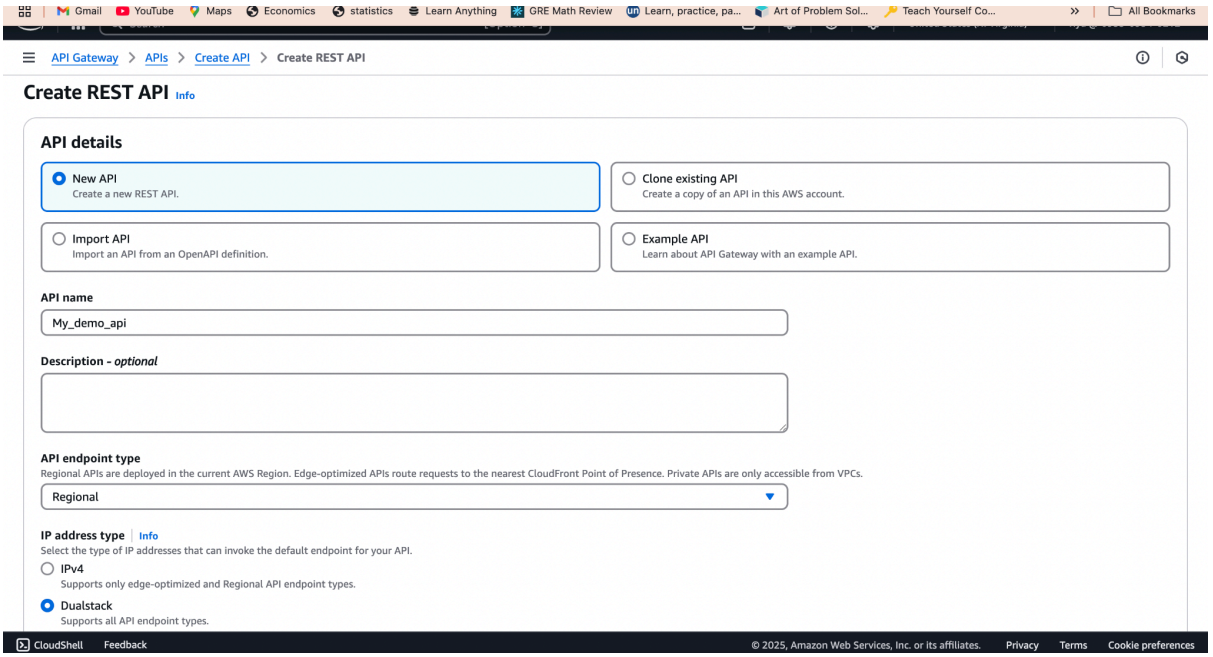


Click on Create API

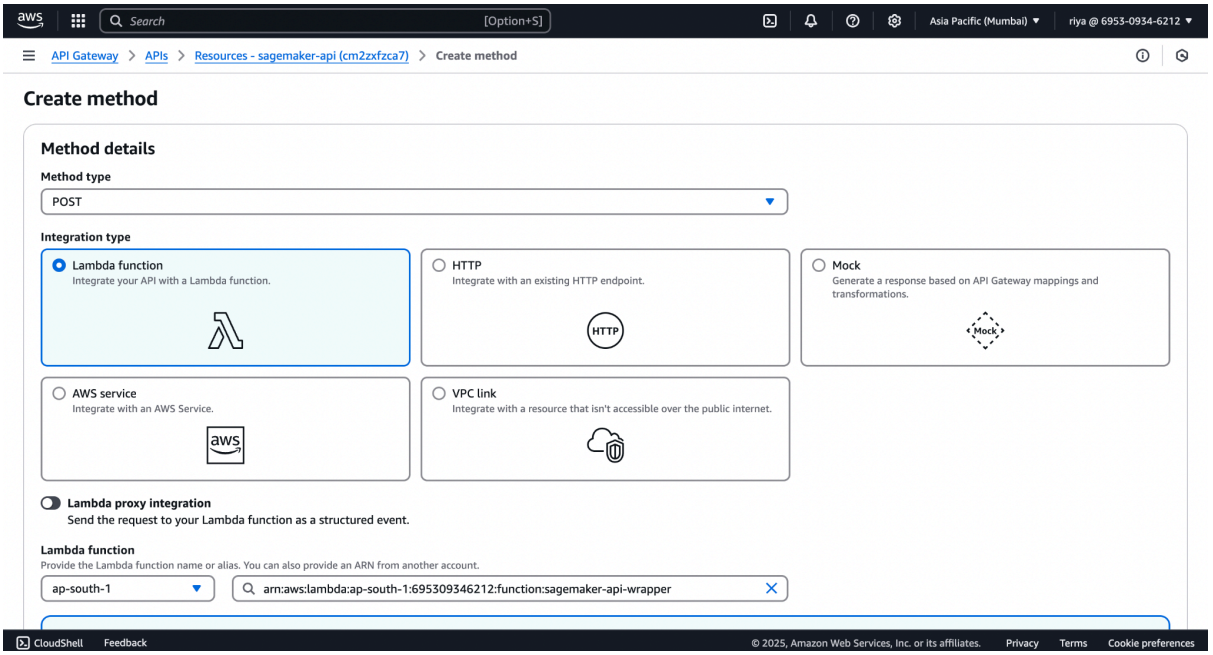


Choose REST API and then build

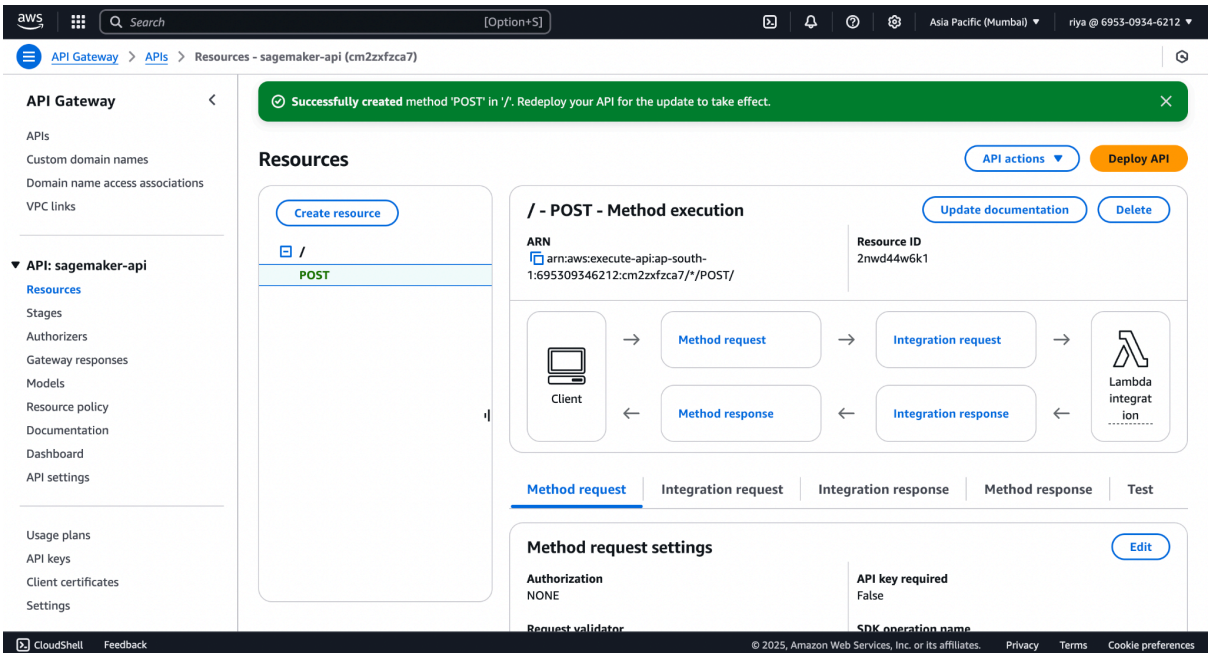




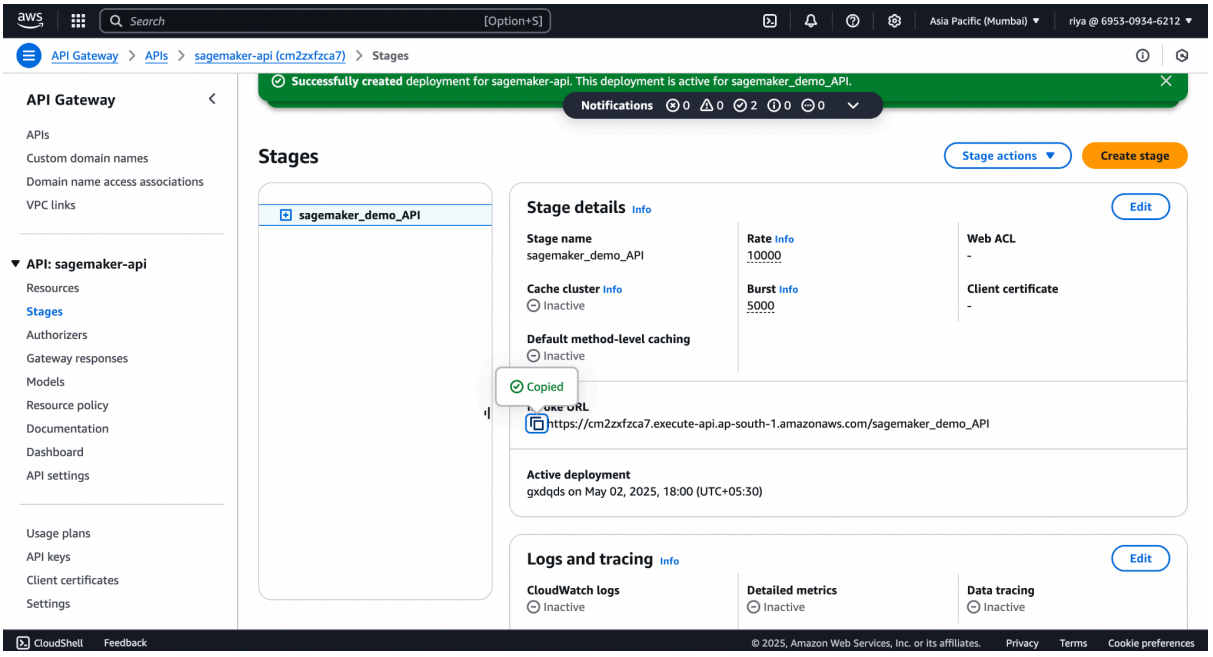
Choose POST method and then integration type as Lambda function and then connect the lambda function that we created in the link below



Click on deploy API



Copy the URL after deploying the API to use on postman



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
}
```