

Configuration Manual

MSc Research Project
Cloud Computing

Sameera Bano
Student ID: 23244950

School of Computing
National College of Ireland

Supervisor: Shaguna Gupta

**National College of Ireland
Project Submission Sheet
School of Computing**



Student Name:	Sameera Bano
Student ID:	23244950
Programme:	Cloud Computing
Year:	2024
Module:	MSc Research Project
Supervisor:	Shaguna Gupta
Submission Due Date:	29/01/2025
Project Title:	Configuration Manual
Word Count:	836
Page Count:	14

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Sameera Bano
Date:	29th January 2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	<input type="checkbox"/>
Attach a Moodle submission receipt of the online project submission , to each project (including multiple copies).	<input type="checkbox"/>
You must ensure that you retain a HARD COPY of the project , both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	<input type="checkbox"/>

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Sameera Bano
23244950

Building a Flask Web Application with Machine Learning Models Using Docker

This guide details the steps to build and deploy a Flask web application that incorporates three machine learning models: Random Forest, Support Vector Machine (SVM), and Logistic Regression. The project is powered by Python and Docker for efficient development and deployment.

1 System Requirements

1.1 Hardware Requirements

- Operating System: Windows 10
- Processor: AMD Ryzen 3
- System Type: 64-bit Operating System
- RAM: 4 GB
- Storage:
 - SSD: 256 GB
 - HDD: 1 TB
- Display: HD with a refresh rate of 60 Hz

1.2 Software Requirements

- Python: Version 3.12
- VSCode: Version 1.95.3
- VSCode Extensions:
 - Pylance: Enabled
 - Docker Extension: Enabled
- Docker Desktop: Version 4.3.1

2. Installation Steps

1. Download the Flask Web App Code:

Download the ZIP file containing the Flask application code and Docker configuration.

2. Extract the Files:

Extract the ZIP file to a folder on your computer.

3. Open the Project in VSCode:

Use VSCode to navigate to the extracted directory.

4. Run the Flask Application Locally:

Start the application by running:

```
```bash
python app.py
```
```

5. Install Dependencies:

Install the required libraries, including Flask and machine learning packages, by running:

```
```bash
pip install -r requirements.txt
```
```

3. Configuration

The application is designed to run in a containerized environment. The Dockerfile uses python:3.12-slim to minimize the container size while supporting all necessary libraries. Additionally, Docker Desktop was configured with WSL2 to ensure compatibility with Linux-based images.

4. Usage Instructions

4.1 Running the Flask Application

1. Start the Application:

Run the app locally:

```
```bash
python app.py
```
```

The application will be available at <http://localhost:5000>.

2. Machine Learning Models Integration:

The app includes the following models:

- Random Forest: Predicts outcomes using an ensemble of decision trees.

- SVM: Performs classification tasks using hyperplanes.
- Logistic Regression: for checking if deployments were successful or not

3. Interact with the Application:

Use the web interface to input data and receive predictions from the models.

4.2 Deploying the Application with Docker

1. Create a Dockerfile:

```
``Dockerfile
# Use the official Python image
FROM python:3.12

# Set the working directory
WORKDIR /app

# Copy the dependencies file and install libraries
COPY requirements.txt /app/
RUN pip install --no-cache-dir -r requirements.txt

# Copy the Flask app and models
COPY . /app/

# Expose the port Flask will run on
EXPOSE 5000

# Run the Flask app
CMD ["python", "app.py"]
``
```

2. Build the Docker Image:

```
``bash
docker build -t flask-ml-app .
``
```

3. Run the Docker Container:

```
``bash
docker run -p 5000:5000 flask-ml-app
``
```

Key Features of the Application

- Random Forest: Predictive analytics using an ensemble learning method.
- SVM: Efficient classification of datasets with clear margins.

- Logistic Regression: for checked if deployments were successful or not.

By combining these models, the application serves as a versatile tool for making predictions based on user inputs.

Screenshots:

1) To create container registry:

The screenshot shows the 'Create container registry' page in the Microsoft Azure portal. The page has a blue header with the Microsoft Azure logo and a search bar. Below the header, there's a breadcrumb trail: 'Home > Create a resource >'. The main heading is 'Create container registry' with a three-dot menu icon. Below this, there are tabs for 'Basics', 'Networking', 'Encryption', 'Tags', and 'Review + create'. The 'Basics' tab is selected. The page content describes Azure Container Registry and provides instructions on how to use it. Below the description, there are two sections: 'Project details' and 'Instance details'. The 'Project details' section has two dropdown menus: 'Subscription' (set to 'Azure subscription 1') and 'Resource group' (with a 'Create new' link below it). The 'Instance details' section has three fields: 'Registry name' (with a placeholder 'Enter the name' and '.azurecr.io' suffix), 'Location' (set to 'East US'), and 'Use availability zones' (unchecked). Below the 'Use availability zones' field, there is an information icon and a note: 'Availability zones are activated on premium registries and in regions that support availability zones. [Learn more](#)'. At the bottom, there is a 'Pricing plan' dropdown set to 'Standard'. At the very bottom, there are three buttons: 'Review + create' (blue), '< Previous' (grey), and 'Next: Networking >' (grey).

Microsoft Azure

Search resources, services, and docs (G+)

Home > Create a resource >

Create container registry

Basics Networking Encryption Tags Review + create

Azure Container Registry allows you to build, store, and manage container images and artifacts in a private registry for all types of container deployments. Use Azure container registries with your existing container development and deployment pipelines. Use Azure Container Registry Tasks to build container images in Azure on-demand, or automate builds triggered by source code updates, updates to a container's base image, or timers. [Learn more](#)

Project details

Subscription * Azure subscription 1

Resource group * [Create new](#)

Instance details

Registry name * Enter the name .azurecr.io

Location * East US

Use availability zones ☐

i Availability zones are activated on premium registries and in regions that support availability zones. [Learn more](#)

Pricing plan * ☐ Standard

[Review + create](#) [< Previous](#) [Next: Networking >](#)

2) To run the pipeline I have used Agent pool with the name 'self'

The screenshot shows the 'Project Settings' page in Azure DevOps. On the left sidebar, under 'Pipelines', the 'Agent pools' option is highlighted with a red dashed box. In the main content area, the 'self' agent pool is highlighted with a red solid box. Below this, a table lists jobs running on this pool.

| Name | Project | Agent | Queued | Wait time | Duration |
|--|------------------|--------|-----------------|-----------|----------|
| Job 80
20241121.17 Machine_Learning | Machine_Learning | agent1 | 21 Nov at 09:24 | <1s | 22m 2s |
| Job 79 | Machine_Learning | agent1 | 21 Nov at 09:23 | <1s | 20s |

3) Jobs run in the pipeline :

The screenshot shows the Azure DevOps interface for a pipeline run. The breadcrumb path is: Azure DevOps / sameerabanonci0642 / Machine_Learning / Pipelines / Machine_Learning / 20241121.17. The left sidebar shows the 'Pipelines' section with 'Pipelines' selected. The main area displays 'Jobs in run #20241121.17' for the 'Machine_Learning' pipeline. A table lists the jobs in the run, and a detailed view of the 'Job' is shown on the right.

| Jobs | Duration |
|---------------------------------|----------|
| Job | 21m 46s |
| Initialize job | <1s |
| Checkout Machine_Lear... | 6s |
| List all files in the direct... | 2s |
| Run Logistic Regression... | 8s |
| Log in to Azure Contain... | 1s |
| Build and push D... | 21m 25s |
| Post-job: Checkout Mac... | 1s |
| Finalize Job | <1s |
| Report build status | <1s |

Job Details:

- 1 Pool: self
- 2 Agent: agent1
- 3 Started: 21 Nov at 09:24
- 4 Duration: 21m 46s
- 5
- 6 ▶ Job preparation parameters

Azure DevOps sameerabanonci0642 / Machine_Learning / Pipelines / Machine_Learning / 20241121.17

Machine_Learning

Overview

Boards

Repos

Pipelines

Pipelines

Environments

Library

Test Plans

Artifacts

Project settings

Jobs in run #20241121.17

Machine_Learning

Jobs

Job 21m 46s

Initialize job <1s

Checkout Machine_Lear... 6s

List all files in the direct... 2s

Run Logistic Regression... 8s

Log in to Azure Contain... 1s

Build and push D... 21m 25s

Post-job: Checkout Mac... 1s

Finalize Job <1s

Report build status <1s

Initialize job

3 Agent machine name: 'DESKTOP-SEUNG14'

4 Current agent version: '3.246.0'

5 Agent running as: 'hp'

6 Prepare build directory.

7 Set build variables.

8 Download all required tasks.

9 Checking job knob settings.

10 Knob: DockerActionRetries = true Source: \$(VSTSAGENT_DOCKER_ACTION_RETRIES)

11 Knob: UseGitLongPaths = true Source: \$(USE_GIT_LONG_PATHS)

12 Knob: EnableIssueSourceValidation = true Source: \$(ENABLE_ISSUE_SOURCE_VALIDATION)

13 Knob: AgentEnablePipelineArtifactLargeChunkSize = true Source: \$(AGENT_ENABLE_PIPELINEARTIFACT_LARGE_CHUNK_SIZE)

14 Knob: ContinueAfterCancelProcessTreeKillAttempt = true Source: \$(VSTSAGENT_CONTINUE_AFTER_CANCEL_PROCESSTREEKILL_ATTEMPT)

15 Knob: ProcessHandlerSecureArguments = false Source: \$(AZP_75787_ENABLE_NEW_LOGIC)

16 Knob: ProcessHandlerSecureArguments = false Source: \$(AZP_75787_ENABLE_NEW_LOGIC_LOG)

17 Knob: ProcessHandlerTelemetry = true Source: \$(AZP_75787_ENABLE_COLLECT)

18 Knob: UseNewNodeHandlerTelemetry = true Source: \$(DistributedTask.Agent.UseNewNodeHandlerTelemetry)

19 Knob: ProcessHandlerEnableNewLogic = true Source: \$(AZP_75787_ENABLE_NEW_PH_LOGIC)

20 Knob: EnableResourceMonitorDebugOutput = true Source: \$(AZP_ENABLE_RESOURCE_MONITOR_DEBUG_OUTPUT)

21 Knob: EnableResourceUtilizationWarnings = true Source: \$(AZP_ENABLE_RESOURCE_UTILIZATION_WARNINGS)

22 Knob: IgnoreVSTaskLib = true Source: \$(AZP_AGENT_IGNORE_VSTASKLIB)

23 Knob: FailJobWhenAgentDies = true Source: \$(FAIL_JOB_WHEN_AGENT_DIES)

24 Knob: CheckForTaskDeprecation = true Source: \$(AZP_AGENT_CHECK_FOR_TASK_DEPRECATION)

25 Knob: CheckIfTaskRunnerIsDeprecated246 = True Source: \$(DistributedTask.Agent.CheckIfTaskRunnerIsDeprecated246)

26 Knob: UseNode20ToStartContainer = True Source: \$(DistributedTask.Agent.UseNode20ToStartContainer)

27 Knob: LogTaskNameInUserAgent = true Source: \$(AZP_AGENT_LOG_TASKNAME_IN_USERAGENT)

28 Knob: UseFetchFilterInCheckoutTask = true Source: \$(AGENT_USE_FETCH_FILTER_IN_CHECKOUT_TASK)

29 Knob: Rosetta2Warning = true Source: \$(ROSETTA2_WARNING)

Azure DevOps sameerabanonci0642 / Machine_Learning / Pipelines / Machine_Learning / 20241121.17

Machine_Learning

Overview

Boards

Repos

Pipelines

Pipelines

Environments

Library

Test Plans

Artifacts

Project settings

Jobs in run #20241121.17

Machine_Learning

Jobs

Job 21m 46s

Initialize job <1s

Checkout Machine_Lear... 6s

List all files in the direct... 2s

Run Logistic Regression... 8s

Log in to Azure Contain... 1s

Build and push D... 21m 25s

Post-job: Checkout Mac... 1s

Finalize Job <1s

Report build status <1s

Checkout Machine_Learning@master to s

47 git remote --mirror origin

18 HEAD is now at 2a285f7 Updated azure-pipelines.yml

19 git config gc.auto 0

20 git config core.longpaths true

21 git config --get-all http.https://sameerabanonci0642@dev.azure.com/sameerabanonci0642/Machine_Learning/_git/Machine_Learning.extraheader

22 git config --get-all http.extraheader

23 git config --get-regexp .*extraheader

24 git config --get-all http.proxy

25 git config http.version HTTP/1.1

26 git --config-env=http.extraheader=env_var.http.extraheader fetch --force --tags --prune --prune-tags --progress --no-recurse-submodules

27 remote: Azure Repos

28 remote:

29 remote: Found 12 objects to send. (0 ms)

30 From https://dev.azure.com/sameerabanonci0642/Machine_Learning/_git/Machine_Learning

31 * [new ref] f77693a2b2c738d8dc03486cbea1172ea0ea391 -> origin/f77693a2b2c738d8dc03486cbea1172ea0ea391

32 \$[warning]free memory is lower than 5% currently used: 95.74%

33 git --config-env=http.extraheader=env_var.http.extraheader fetch --force --tags --prune --prune-tags --progress --no-recurse-submodules

34 remote: Azure Repos

35 remote:

36 remote: Found 0 objects to send. (0 ms)

37 From https://dev.azure.com/sameerabanonci0642/Machine_Learning/_git/Machine_Learning

38 * branch f77693a2b2c738d8dc03486cbea1172ea0ea391 -> FETCH_HEAD

39 git checkout --progress --force refs/remotes/origin/f77693a2b2c738d8dc03486cbea1172ea0ea391

40 Previous HEAD position was 2a285f7 Updated azure-pipelines.yml

41 HEAD is now at f77693a Updated azure-pipelines.yml

42 Finishing: Checkout Machine_Learning@master to s

Azure DevOps sameerabanonci0642 / Machine_Learning / Pipelines / Machine_Learning / 20241121.17

Machine_Learning

Overview
Boards
Repos
Pipelines
Pipelines
Environments
Library
Test Plans
Artifacts

Project settings

Jobs in run #20241121.17
Machine_Learning

Jobs

| Job | Duration |
|---------------------------------|----------|
| Initialize job | <1s |
| Checkout Machine_Lear... | 6s |
| List all files in the direct... | 2s |
| Run Logistic Regression... | 8s |
| Log in to Azure Contain... | 1s |
| Build and push D... | 21m 25s |
| Post-job: Checkout Mac... | 1s |
| Finalize Job | <1s |
| Report build status | <1s |

List all files in the directory (Windows)

```

1 Starting: List all files in the directory (Windows)
2 -----
3 Task : Command line
4 Description : Run a command line script using Bash on Linux and macOS and cmd.exe on Windows
5 Version : 2.246.1
6 Author : Microsoft Corporation
7 Help : https://docs.microsoft.com/azure/devops/pipelines/tasks/utility/command-line
8 -----
9 [Warning]Free memory is lower than 50, currently used: 49,748
10 Generating script.
11 ----- Starting Command Output -----
12 "C:\Windows\system32\cmd.exe" /D /E:ON /V:OFF /S /C "CALL "C:\agents_work_temp\78c29841-4c26-4ecb-9f9c-1d752613324c.cmd""
13 "Listing all files in the working directory:"
14 Volume in drive C has no label.
15 Volume Serial Number is BE56-B163
16
17 Directory of C:\agents_work\1\
18
19 21-11-2024 14:54 <DIR> .
20 21-11-2024 14:54 <DIR> ..
21 21-11-2024 14:54 1,400 azure-pipelines.yml
22 08-11-2024 16:31 <DIR> data
23 21-11-2024 13:14 456 Dockerfile
24 21-11-2024 13:38 6,567 model.py
25 08-11-2024 13:14 985 README.md
26 21-11-2024 13:14 99 requirements.txt
27 21-11-2024 13:14 <DIR> templates

```

Azure DevOps sameerabanonci0642 / Machine_Learning / Pipelines / Machine_Learning / 20241121.17

Machine_Learning

Overview
Boards
Repos
Pipelines
Pipelines
Environments
Library
Test Plans
Artifacts

Project settings

Jobs in run #20241121.17
Machine_Learning

Jobs

| Job | Duration |
|---------------------------------|----------|
| Initialize job | <1s |
| Checkout Machine_Lear... | 6s |
| List all files in the direct... | 2s |
| Run Logistic Regression... | 8s |
| Log in to Azure Contain... | 1s |
| Build and push D... | 21m 25s |
| Post-job: Checkout Mac... | 1s |
| Finalize Job | <1s |
| Report build status | <1s |

Log in to Azure Container Registry

```

1 Starting: Log in to Azure Container Registry
2 -----
3 Task : Docker
4 Description : Build or push Docker images, login or logout, start or stop containers, or run a Docker command
5 Version : 2.248.1
6 Author : Microsoft Corporation
7 Help : https://aka.ms/azpipes-docker-tsg
8 -----
9 Finishing: Log in to Azure Container Registry

```

➤ To create the Kubernetes cluster:

Microsoft Azure

Search resources, services, and docs (G+)

Copilot

Home >

Kubernetes services

Default Directory

+ Create Manage view Refresh Export to CSV Open query Assign tags

Automatic Kubernetes cluster (preview)
Automated operations for streamlined application deployment.

Kubernetes cluster
Customizable setup for added control and flexibility.



Add a Kubernetes cluster with Azure Arc
Create a Kubernetes cluster with Azure Arc

Type equals all Resource group equals all Location equals all Add filter

No grouping

| Type | Resource group | Kuberne... | Location | Subscription |
|--------------------|----------------|------------|-------------|----------------------|
| Kubernetes service | ML_Project | 1.29.10 | West Europe | Azure subscription 1 |

- To create a starter application to be deployed :

 Microsoft Azure  Search resources, services, and docs (G+/)

Home > MLProject | Workloads > Create a starter application >

Create a single-image application ...

1 Get image

2 Application details

3 Review YAML

4 Deploy

This experience will walk through the process of selecting an image from a registry or adding an image to a registry, configuring key deployment parameters, and reviewing the YAML file that will be sent to the cluster.

If you already have a YAML file or want to create a more complex deployment, [add with YAML](#) instead.

Container registry details

Before containers can be deployed to your cluster, a container image must be uploaded to the container registry.

Container registry type ⓘ

☒ Azure Container Registry

☐ Other registry

Container registry * ⓘ

MLproject

Create new

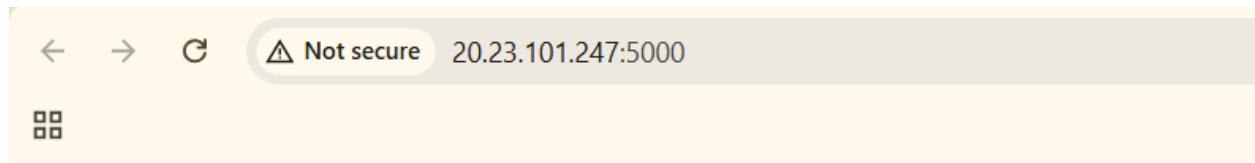
Image details

After choosing a registry, you must choose or create a container image. A container image is the foundation for a Kubernetes deployment and is the blueprint used to create all containers.

Repository ⓘ *

Select an image

- Public Address to check Machine Learning models evaluation in Azure-
<http://20.23.101.247:5000/>



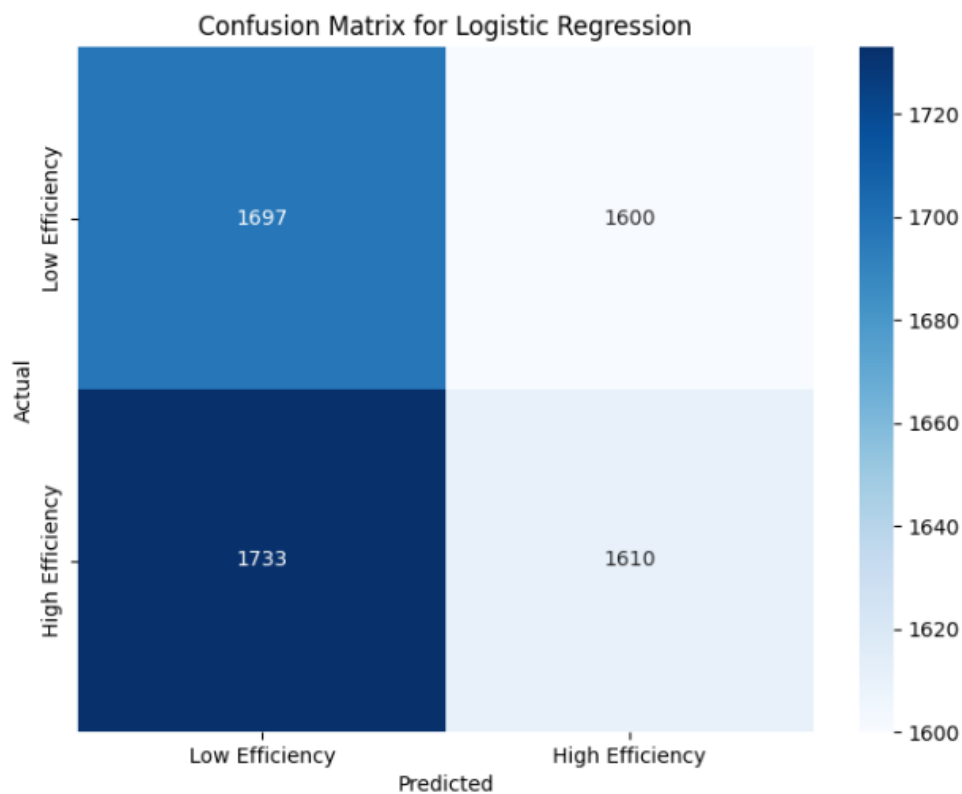
Model Evaluation Results

Logistic Regression

Accuracy: 0.4980421686746988

| | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| 0 | 0.49 | 0.51 | 0.50 | 3297 |
| 1 | 0.50 | 0.48 | 0.49 | 3343 |
| accuracy | | | 0.50 | 6640 |
| macro avg | 0.50 | 0.50 | 0.50 | 6640 |
| weighted avg | 0.50 | 0.50 | 0.50 | 6640 |

Confusion Matrix:



AWS:

➤ To create the security group:

The screenshot shows the 'Create security group' page in the AWS Management Console. The page is divided into several sections:

- Inbound rules:** A section stating 'This security group has no inbound rules.' with an 'Add rule' button.
- Outbound rules:** A section with a table for adding rules. The table has columns: Type, Protocol, Port range, Destination, and Description - optional. A rule is being added with Type 'All traffic', Protocol 'All', Port range 'All', and Destination '0.0.0.0/0'. There is a warning message below the table: 'Rules with destination of 0.0.0.0/0 or ::/0 allow your instances to send traffic to any IPv4 or IPv6 address. We recommend setting security group rules to be more restrictive and to only allow traffic to specific known IP addresses.'
- Tags - optional:** A section stating 'A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.' with an 'Add new tag' button.

➤ CodeBuild:

The screenshot shows the AWS CodeBuild console. The left sidebar contains a navigation menu with the following items:

- Developer Tools
- CodeBuild
- Source • CodeCommit
- Artifacts • CodeArtifact
- Build • CodeBuild
 - Getting started
 - Build projects
 - Build project
 - Settings
 - Build history
 - Report groups

The main content area shows the 'Build history' tab. It includes a table with the following columns: Build run, Status, Build number, Source version, Submitter, Duration, and Completed. A single build is listed with a status of 'Succeeded'.

| Build run | Status | Build number | Source version | Submitter | Duration | Completed |
|--|-----------|--------------|--|------------------|---------------------|-------------|
| SimpleDockerProject-06a3c99926c9c0e5cf5a-76cb-4117-977d-9f64524af9a2 | Succeeded | 12 | arn:aws:codepipeline:eu-central-1:77580454909/m12/SourceArt1/28MwdD5 | codepipeline/ml2 | 1 minute 13 seconds | 16 days ago |

➤ Pipeline:

The screenshot shows the AWS CodePipeline console. On the left, the 'Developer Tools' sidebar is open, with 'CodePipeline' selected. The main panel displays the 'Pipeline type: V2' and 'Execution mode: QUEUED'. The pipeline execution ID is '1483cda1-28bc-47f8-af2f-766e0bd88df'. The pipeline consists of two stages: 'Source' and 'Build'. The 'Source' stage is 'Succeeded' and the 'Build' stage is 'Succeeded'. The 'Build' stage is highlighted with a 'Start rollback' button. The 'Build' stage is using 'AWS CodeBuild' and 'Source: Update Dockerfile'.

➤ Elastic Container Service containing the public IP:

The screenshot shows the Amazon Elastic Container Service console. The 'Configuration' tab is selected. The 'Task overview' section shows the 'ARN' as 'arn:aws:ecs:eu-central-1:084828579234:task/machine-modals/Oe74bd72029a4f9986253ca9b0b571'. The 'Last status' is 'Running'. The 'Desired status' is 'Running'. The 'Started/created at' is '10 December 2024 at 06:30 (UTC)'. The 'Fargate ephemeral storage' section shows 'Encryption' as 'Default AWS Fargate encryption' and 'Size (GiB)' as '20'. The 'Configuration' section shows 'Operating system/Architecture' as 'Linux/X86_64', 'Capacity provider' as 'FARGATE', 'Launch type' as 'FARGATE', 'Container instance ID' as '-', and 'Task definition: revision' as 'v1'. The 'ENI ID' is 'eni-0847db2f48f7fa719'. The 'Network mode' is 'awsvpc'. The 'Subnet ID' is 'subnet-028d75415a422afeb'. The 'Public IP' is '3.72.15.26 | open address'. The 'Private IP' is '172.31.32.18'. The 'MAC address' is '06:1c:b2:2b:25:17'. The 'Container details for modals' section is expanded, showing 'Details', 'Log configuration', 'Restart policy', 'Network bindings', 'Docker labels and hosts', 'Environment variables and files', and 'Volume configuration'. The 'Details' tab is selected, showing 'Image URI' as 'essential' and 'Command' as 'http://3.72.15.26:5000/'.

AWS public IP to analyse Machine learning models evaluation- <http://3.72.15.26:5000/>

Evaluation result:

Model Evaluation Results

Logistic Regression

Accuracy: 0.4935

| | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| 0 | 0.51 | 0.38 | 0.43 | 5117 |
| 1 | 0.49 | 0.61 | 0.54 | 4883 |
| accuracy | | | 0.49 | 10000 |
| macro avg | 0.50 | 0.50 | 0.49 | 10000 |
| weighted avg | 0.50 | 0.49 | 0.49 | 10000 |

Confusion Matrix:

