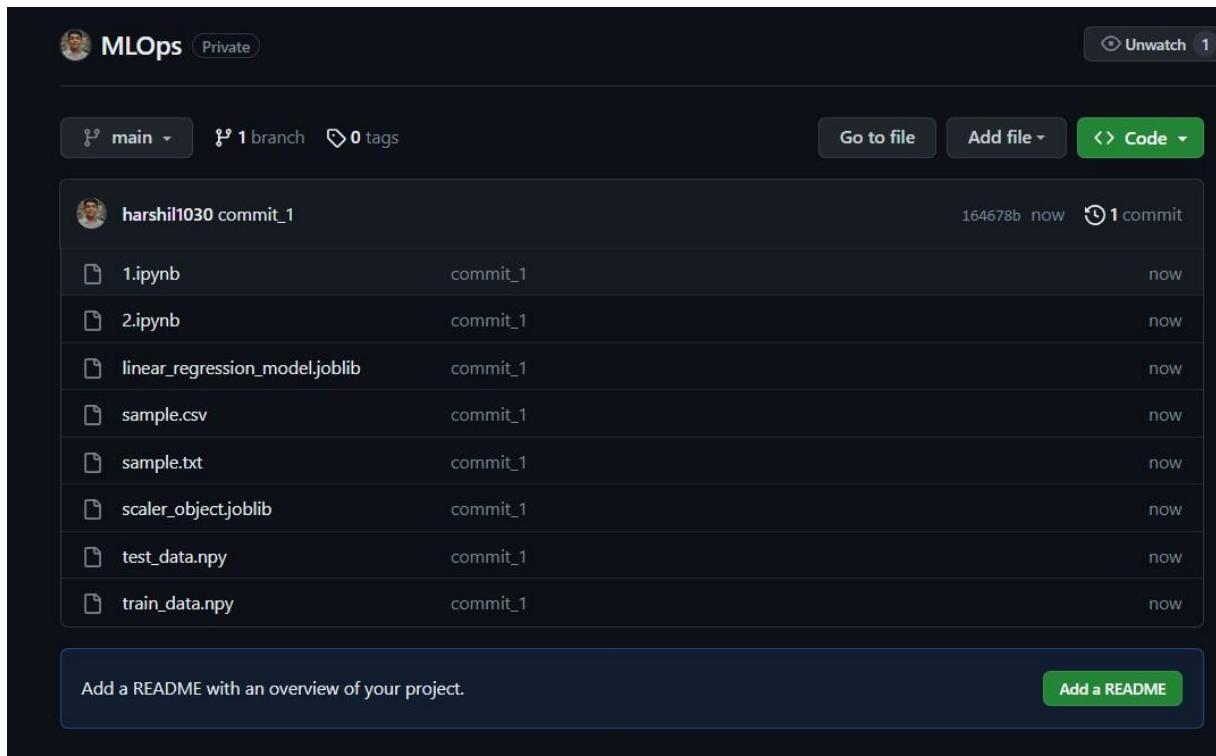


Practical-3

Generation of Reproducible and Interactive ML Project.

Task 1: Create the Github repository for the house rate prediction project created in practical 2.



Task 2: Integrate your repository with the binder to make your project interactive. (Hint: refer to the following link for the steps: (<https://mybinder.org/>)

Build and launch a repository

GitHub repository name or URL
GitHub ▾ https://github.com/harshil1030/MLOps

Git ref (branch, tag, or commit)
HEAD

Path to a notebook file (optional)
https://github.com/harshil1030/MLOps/blob/main

File ▾ launch

Copy the URL below and share your Binder with others:
https://mybinder.org/v2/gh/harshil1030/MLOps/HEAD?labpath=https%3A%2F%2Fgithub.com%2Fharshil1030%2FMLOps%2Fblob%2Fmain

Expand to see the text below, paste it into your README to show a binder badge: [launch](#) [binder](#)

Waiting Building Pushing

Launcher

File Edit View Run Kernel Tabs Settings Help

+ Filter files by name

Name	Last Modified
1.ipynb	2 minutes ago
2.ipynb	2 minutes ago
linear_regr...	2 minutes ago
sample.csv	2 minutes ago
sample.txt	2 minutes ago
scaler_obje...	2 minutes ago
test_data.npy	2 minutes ago
train_data.n...	2 minutes ago

Notebook

Python 3 (ipykernel)

Console

Python 3 (ipykernel)

The screenshot shows a Jupyter Notebook interface with the title "2.ipynb". The left sidebar displays a file tree with files like 1.ipynb, 2.ipynb, linear_regr..., sample.csv, sample.txt, scaler_obje..., test_data.npy, and train_data.n... All files were modified 2 minutes ago. The main area contains two code cells:

```
[6]: import numpy as np
from sklearn.preprocessing import StandardScaler
import joblib

# Load the data from sample.csv using np.genfromtxt
data = np.genfromtxt('sample.csv', delimiter=',')
# # Load the data from Sample.txt
# data = np.loadtxt('sample.csv')

# Create a StandardScaler object and fit_transform the data
scaler = StandardScaler()
scaled_data = scaler.fit_transform(data)

# Store the scaler object for reproducibility
joblib.dump(scaler, 'scaler_object.joblib')

# Now scaled_data contains the normalized dataset
```

```
[6]: ['scaler_object.joblib']

[7]: from sklearn.model_selection import train_test_split

# Split the data into training and testing sets
train_data, test_data = train_test_split(scaled_data, test_size=0.2, random_state=42)

# train_data and test_data now contain the training and testing datasets
```