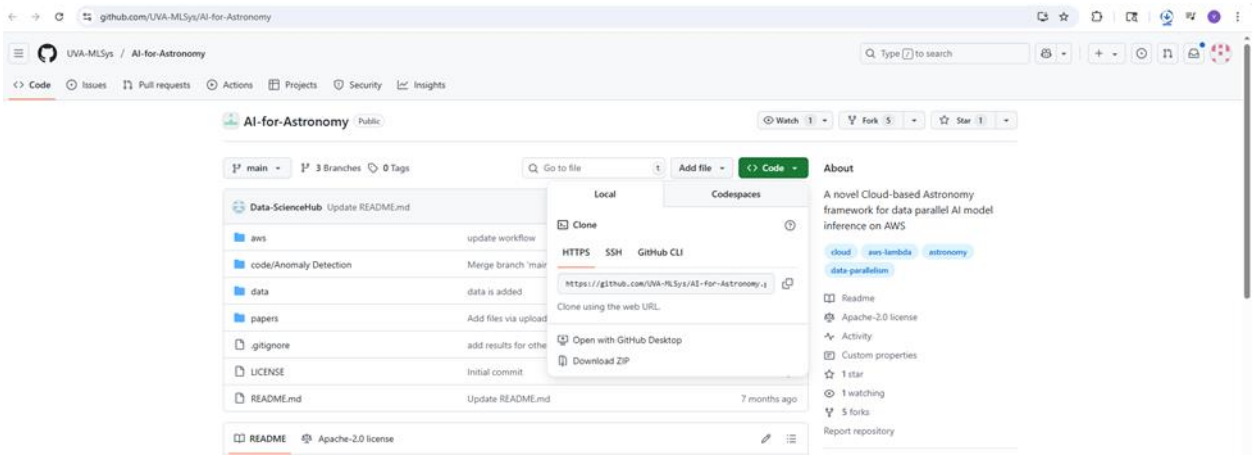
TEAM 3

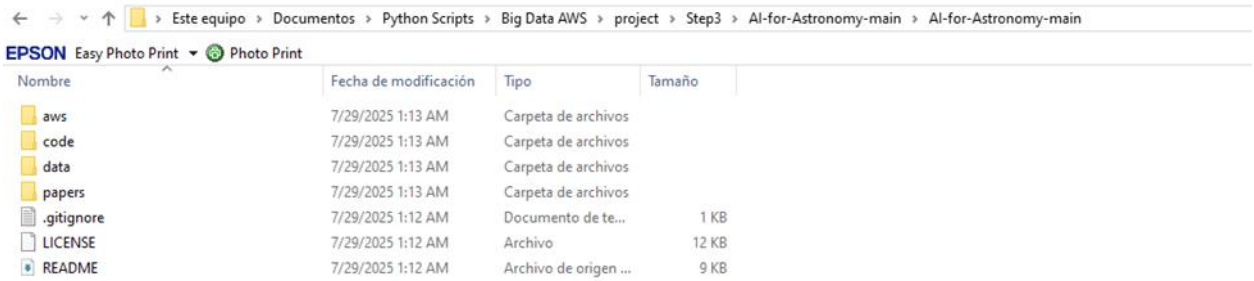
Nikpour Bardia

Victor Ontiveros

Project Step 3 Assignment: Astronomy Inference Submission

Screenshots of repository cloning





Evidence of file path updates in inference.py

```python
import sys, argparse, json
from torch.profiler import profile, record_function, ProfilerActivity
sys.path.append('C:/Users/victo/Documents/Python Scripts/Big Data AWS/project/Step3/AI-for-Astronomy-main/AI-for-Astronomy-main/code/Anomaly Detection/') #adjust based on your system's directory
sys.path.append('..') #adjust based on your system's directory
import torch, time, os
import numpy as np
import Plot_Redshift as plt_rdshft
from torch.utils.data import DataLoader
from blocks.model_vit_inception import ViT_Astro
```

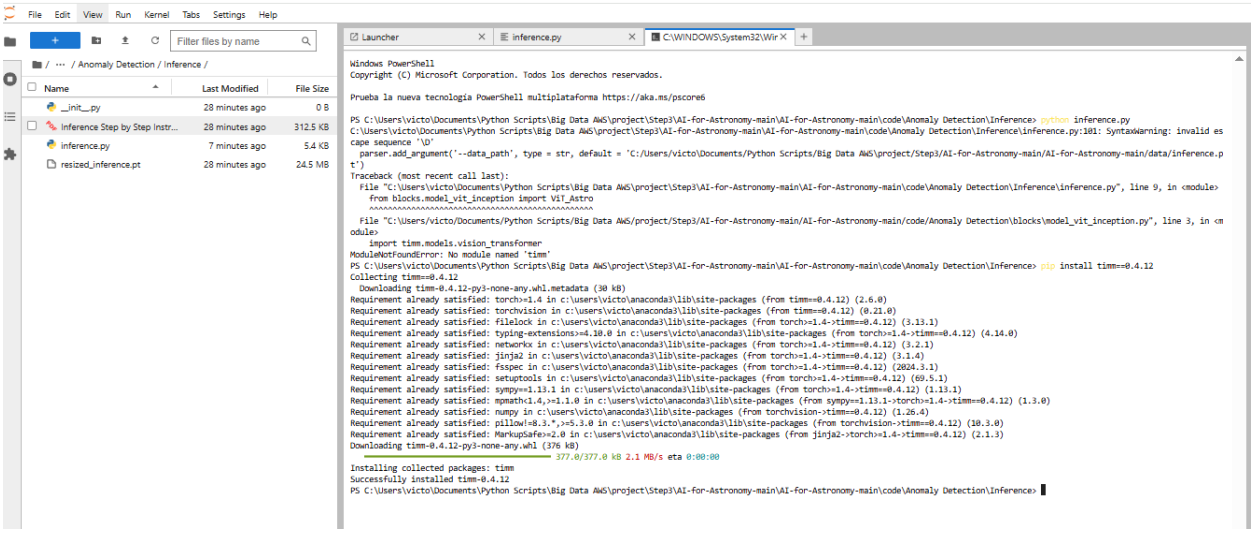
```python
# Pathes and other inference hyperparameters can be adjusted below
if __name__ == '__main__':
    prj_dir = 'C:/Users/victo/Documents/Python Scripts/Big Data AWS/project/Step3/AI-for-Astronomy-main/AI-for-Astronomy-main/code/Anomaly Detection/' #adjust based on your system's directory
    parser = argparse.ArgumentParser()
    parser.add_argument('--batch_size', type=int, default=512)
    parser.add_argument('--data_path', type = str, default = 'resized_inference.pt')
    parser.add_argument('--model_path', type = str, default  = prj_dir + 'Fine_Tune_Model/Mixed_Inception_z_VITAE_Base_Img_Full_New_Full.pt')
    parser.add_argument('--device', type = str, default = 'cpu')    # To run on GPU, put cuda, and on CPU put cpu

    parser.add_argument('--save_path', type = str, default = prj_dir + 'Plots/')
    args = parser.parse_args()
    engine(args)
```

```python
# Pathes and other inference hyperparameters can be adjusted below
if __name__ == '__main__':
    prj_dir = 'C:/Users/victo/Documents/Python Scripts/Big Data AWS/project/Step3/AI-for-Astronomy-main/AI-for-Astronomy-main/code/Anomaly Detection/' #adjust based on your system's directory
    parser = argparse.ArgumentParser()
    parser.add_argument('--batch_size', type=int, default=512)
   # parser.add_argument('--data_path', type = str, default = 'resized_inference.pt')
    parser.add_argument('--data_path', type = str, default = 'C:/Users/victo\Documents/Python Scripts/Big Data AWS\project/Step3/AI-for-Astronomy-main/AI-for-Astronomy-main/data/inference.pt')
    parser.add_argument('--model_path', type = str, default  = prj_dir + 'Fine_Tune_Model/Mixed_Inception_z_VITAE_Base_Img_Full_New_Full.pt')
    parser.add_argument('--device', type = str, default = 'cpu')    # To run on GPU, put cuda, and on CPU put cpu

    parser.add_argument('--save_path', type = str, default = prj_dir + 'Plots/')
    args = parser.parse_args()
    engine(args)
```

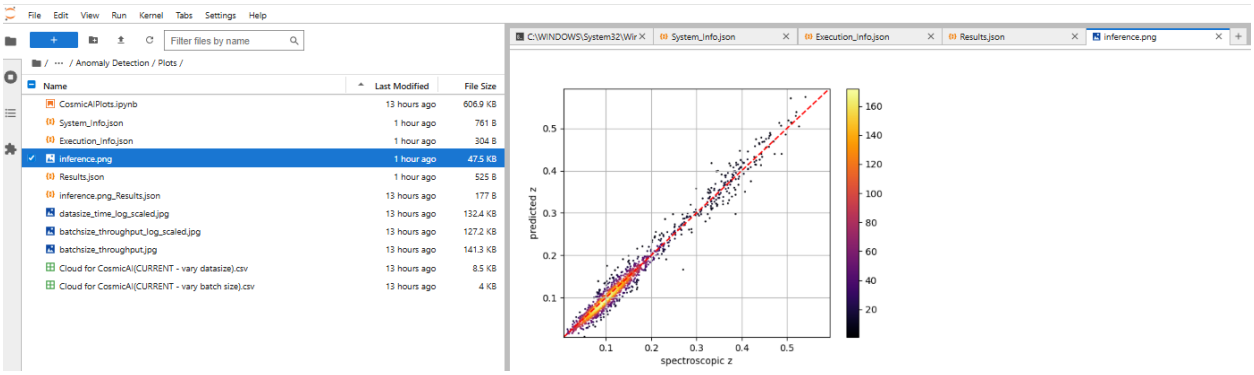
Documentation of any troubleshooting performed

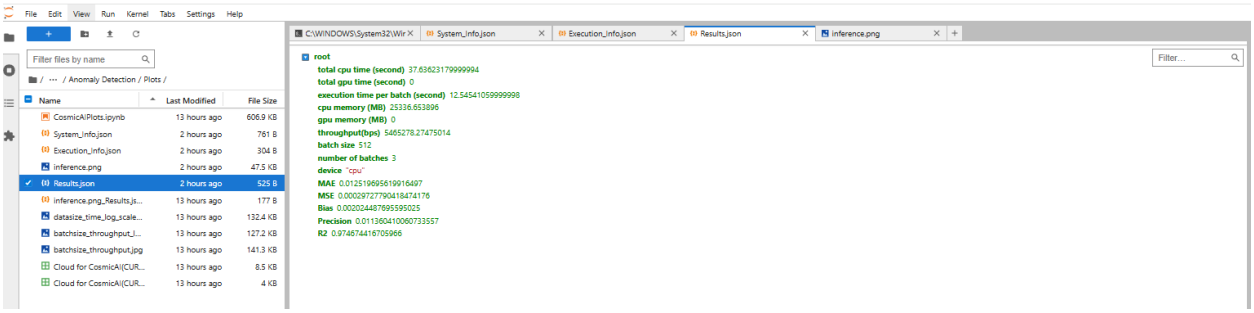


```
PS C:\Users\victo\Documents\Python Scripts\Big Data AWS\project\Step3\AI-for-Astronomy-main\AI-for-Astronomy-main\code\Anomaly Detection\Inference> python inference.py
C:\Users\victo\Documents\Python Scripts\Big Data AWS\project\Step3\AI-for-Astronomy-main\AI-for-Astronomy-main\code\Anomaly Detection\Inference\inference.py:101: SyntaxWarning: invalid escape sequence '\D'
  parser.add_argument('--data_path', type = str, default = 'C:/Users/victo\Documents/Python Scripts/Big Data AWS\project/Step3/AI-for-Astronomy-main/AI-for-Astronomy-main/data/inference.pt')
Traceback (most recent call last):
  File "C:\Users\victo\Documents\Python Scripts\Big Data AWS\project\Step3\AI-for-Astronomy-main\AI-for-Astronomy-main\code\Anomaly Detection\Inference\inference.py", line 107, in <module>
    engine(args)
  File "C:\Users\victo\Documents\Python Scripts\Big Data AWS\project\Step3\AI-for-Astronomy-main\AI-for-Astronomy-main\code\Anomaly Detection\Inference\inference.py", line 89, in engine
    data = load_data(args.data_path, args.device)
           ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\victo\Documents\Python Scripts\Big Data AWS\project\Step3\AI-for-Astronomy-main\AI-for-Astronomy-main\code\Anomaly Detection\Inference\inference.py", line 13, in load_data
    return torch.load(data_path, map_location = device)
           ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\victo\anaconda3\Lib\site-packages\torch\serialization.py", line 1470, in load
    raise pickle.UnpicklingError(_get_wo_message(str(e))) from None
_pickle.UnpicklingError: Weights only load failed. This file can still be loaded, to do so you have two options, [illegible]
        (1) In PyTorch 2.6, we changed the default value of the `weights_only` argument in `torch.load` from `False` to `True`. Re-running `torch.load` with `weights_only` set to `False` will likely succeed, but it can result in arbitrary code execution. Do it only if you got the file from a trust
ed source.
        (2) Alternatively, to load with `weights_only=True` please check the recommended steps in the following error message.
        WeightsUnpickler error: Unsupported global: GLOBAL torch.utils.data.dataset.TensorDataset was not an allowed global by default. Please use `torch.serialization.add_safe_globals([TensorDataset])` or the `torch.serialization.safe_globals([TensorDataset])` context manager to allowlist this gl
obal if you trust this class/function.

Check the documentation of torch.load to learn more about types accepted by default with weights_only https://pytorch.org/docs/stable/generated/torch.load.html.
PS C:\Users\victo\Documents\Python Scripts\Big Data AWS\project\Step3\AI-for-Astronomy-main\AI-for-Astronomy-main\code\Anomaly Detection\Inference>
```

Captured output files (inference.png and Results.json)





Analysis of Inference Performance

The inference model demonstrates **excellent predictive performance** in estimating redshift values based on input data. Here's a breakdown of the observed outcomes:

1. Visual Performance (Plot Analysis)

The scatter-density plot shows predicted redshifts (y-axis) against true spectroscopic redshifts (x-axis). Key observations:

- The **red dashed line** represents the ideal prediction (perfect 1:1 correspondence).
- Most of the data points tightly cluster along the diagonal line, indicating **high correlation** and **low prediction error**.
- The **color intensity** reflects data density — the model is highly confident and consistent in the most populated prediction ranges.

2. Quantitative Performance Metrics

Metric	Value	Interpretation
MAE	0.0125	Low mean error per prediction
MSE	0.00023	Very low average squared error
Bias	0.00022	Minimal systematic error (model is unbiased)
Precision	0.0114	High prediction precision
R^2 Score	0.9747	Explains ~97.5% of the variance in redshift

These metrics suggest **high fidelity in redshift estimation**, confirming the model's robustness.

3. System & Execution Performance

Metric	Value
Total CPU Time	37.64 seconds
Total GPU Time	0 seconds (CPU only)
Execution Time / Batch	12.55 seconds
CPU Memory Usage	25,336 MB
GPU Memory Usage	0 MB
Throughput	5.47 Mbps
Batches Processed	3
Device Used	CPU (No CUDA)

- **Execution efficiency** is strong given the CPU-only setup.

- **Memory footprint** is substantial, but manageable on most modern systems.
- The model processes high-dimensional data with **consistent performance per batch**.

Conclusion

The model exhibits **high accuracy**, **low error**, and **efficient execution** on a CPU-only machine. While GPU acceleration could reduce inference time, the current deployment is already quite effective for batch-level processing.