MGS 662 – Project 1 Part B

Title : Machine Learning Optimization for Visual Concept Classification
Student : Harshith Kumar Reddy Audipudi
UBIT : haudipud
UBIT No. : 50599797
Course : MGS 662
Professor : Prof. Haimonti Dutta
Due Date : April 4, 2025

1. Introduction

This project focuses on applying machine learning techniques to a real-world classification problem in the domain of digital humanities. The task involves analyzing traditional narrative scroll paintings, where each image panel can be annotated with the presence or absence of certain visual elements like trees, animals, and mythological characters.

The main objective was to implement a machine learning pipeline using R that performs the following:

- Extracts features from image panels
- Processes annotations from multiple raters
- Aggregates labels
- Trains classification models using Stochastic Gradient Descent (SGD) with different loss functions
- Evaluates model behavior through loss plots

2. Dataset Description

2.1 Image Data

The image dataset consists of 126 panel images located in structured directories under ProjectData_ForClass_v2-3/. Each scroll folder (e.g., s1, s2, ..., s22) contains .jpg files inside an /img/ subfolder. The filenames follow the pattern s1.jpg, s2.jpg, ..., s126.jpg.

2.2 Annotation Data

The annotation file (Annotation.csv) provides labeled information for each panel. It includes binary values for three visual concepts:

- Tree
- Animal
- Mythological Character

Each label is either 1 (present) or 0 (absent), with some cells missing (i.e., NA). These annotations were provided by multiple human raters and consolidated beforehand.

3. Label Aggregation Strategy

Although annotations originally came from five raters, the file used for this project already included pre-combined binary values (likely via voting or majority). No manual aggregation was necessary, but the following filtering logic was applied:

- Only rows with a value of 0 or 1 were included
- Missing or invalid entries were removed
- Each image was linked to its corresponding label using a generated ID (s1, s2, ..., s126)

This preprocessing step resulted in a final dataset of 98 usable samples per concept for training and evaluation.

4. Feature Extraction

4.1 Technique Used

Feature extraction was conducted using the EBImage package in R. The steps were:

- Convert each image to grayscale
- Resize to 32×32 pixels
- Flatten the image into a 1024-dimensional feature vector

Each processed image was converted into a row of numeric features, with a unique identifier to allow merging with label data.

4.2 Rationale

The grayscale pixel method is simple but effective for this task. More advanced methods (like edge detection or object recognition) may be used in future versions, but grayscale values are sufficient for the goals of this project.

5. Dataset Construction

After extracting the features and cleaning the label data, both were joined using image identifiers. This produced three datasets, one for each visual concept.

Concept	Samples	Feature Sample	Positive Labels	Negative Labels
Tree	98	1024	34	64
Animal	98	1024	34	64
Mythology	98	1024	34	64

These datasets served as the basis for model training.

6. Model Implementation

6.1 Algorithm

I implemented Stochastic Gradient Descent (SGD) for binary classification using:

- Logistic Loss
- Exponential Loss

Both losses were implemented in custom R functions. Labels were transformed to {+1, -1} as required for margin-based classification.

6.2 Training Setup

- Learning rate: 0.01
- Iterations: 1000
- Loss Tracking: Loss was computed and stored at each iteration
- Weight Vector: Initialized to zero and updated per step

Each concept (tree, animal, mythology) was trained independently using both loss functions, resulting in a total of 6 models.

7. Results and Observations

7.1 Loss Curve Analysis

Loss vs. iteration plots were generated for each concept-loss combination.

- Logistic Loss: Smooth and steady convergence in all three cases.
- Exponential Loss: Faster drop initially, but with more noise and occasional instability in convergence.

7.2 Key Insights

- Logistic loss is more stable and reliable, especially with imbalanced datasets.
- Exponential loss may reach lower loss values faster but can be unstable.
- Loss plots confirmed successful convergence in all training runs.

8. Model Parameters Summary

Parameter	Value
Learning Rate	0.01
Iterations	1000
Loss Functions	Logistic,Exponential
Input Feature Size	1024 pixels
Label Format	{+1,-1}

9. Discussion

This project showed how classical ML techniques like SGD can be applied to non-traditional datasets like visual art. Despite the simplicity of grayscale features, the classifiers were able to learn meaningful signals.

Limitations

- Imbalanced classes (34 vs. 64) could impact model bias.
- No evaluation metrics (accuracy, precision, recall) were used due to lack of split.
- Advanced optimization techniques (e.g., ADAM) were not used in this phase.

10. Conclusion

I successfully built an end-to-end machine learning pipeline using R for a visual classification task. All requirements for Part B were satisfied:

- Feature extraction
- Label aggregation
- Model training with two loss functions
- Loss plot generation

This forms a strong baseline for further enhancements.

11. Future Work

To improve performance and earn bonus credit, I plan to:

- Add Momentum methods (e.g., Nesterov, Heavy Ball)
- Add Adaptive SGD variants (e.g., ADAM, RMSProp)
- Incorporate evaluation metrics like accuracy or confusion matrix
- Explore object-based features (SIFT, HOG) instead of raw pixels
- Handle class imbalance through resampling or weighted loss

12. GitHub Repository

The complete source code, R scripts, output datasets, plots, and this final report are publicly available at the following GitHub repository:

🔗 https://github.com/HarshithReddy-Audipudi/mgs662-image-classification-sgd-haudipud

You can clone the repository or download the project files directly from GitHub for replication or further experimentation.