

Course: Applications of Artificial Intelligence and Machine Learning

Code: PROG74000

Professor/Instructor(s): Akrem El-ghazal

Assignment/Lab 2: Winter 2025

Building Linear and Logistic Regression Models from Scratch

Objectives

- Implement Linear Regression and Logistic Regression from scratch without using machine learning libraries.
- Understand and apply **gradient descent** for optimizing model parameters.
- Evaluate model performance using appropriate performance measures.
- Use your implementation to perform regression and classification on the datasets provided in separate files.
- Compare your custom implementations with scikit-learn's built-in models.
- Reflect on challenges encountered and key takeaways from implementing regression models manually.

Instructions

Step 1: Implement the Linear Regression Algorithm

(1.5 pts.)

Your task is to implement the Linear Regression algorithm from scratch without using any machine learning libraries like scikit-learn for the core functionality. Follow these steps:

1. Create a LinearRegression class with the following methods:
 - a. `fit(X, y)`: Train the model using the given input features X and target values y.
 - b. `predict(X)`: Predict the target values for a given set of examples.
 - c. You may add other methods or modify the input arguments as needed
2. Use the **Mean Squared Error (MSE)** as the loss function
3. Implement **gradient descent** to optimize the model parameters. Allow the learning rate (lr) and the number of iterations to be adjustable
4. Ensure your implementation supports multiple features.

Step 2: Load the Dataset

(0.5 pt.)

You will receive files named *training_dataset_linear.csv* and *validation_dataset_linear.csv* containing the datasets. Perform the follow

1. Load the data from the provided CSV files.
2. Understand the dataset using visualizations and basic statistical summaries.
3. Preprocess the data if necessary (e.g., handle missing values, normalize features if needed).

Step 3: Train the Linear Regression Model

(0.5pt.)

1. Initialize your LinearRegression model with a learning rate of **lr** and **iter** iterations.
2. Train the model using the fit method with the provided training dataset.

Step 4: Test and Evaluate the Model

(1.5 pts.)

1. Use the predict method to make predictions on the validation dataset.
2. Compute the Mean Squared Error (MSE) and R-squared score to evaluate performance.
3. Plot the regression line generated by the model along with the training data on a single graph.
4. Compare your implementation with the result of LinearRegression from scikit-learn.

Step 5: Implement the Logistic Regression Algorithm

(1.5 pts.)

Your task is to implement the Logistic Regression algorithm from scratch without using any machine learning libraries like scikit-learn for the core functionality. Follow these steps

1. Create a LogisticRegression class with the following methods:
 - `fit(X, y)`: Train the model using the given input features X and target values y.
 - `predict(X)`: Predict the class labels for a given set of examples.
 - `predict_proba(X)`: Return the probability scores for each class.
 - You may add other methods or modify the input arguments as needed.
2. Use the **Binary Cross-Entropy (Log Loss)** as the loss function.
3. Implement **gradient descent** to optimize the model parameters. Allow the learning rate (lr) and the number of iterations to be adjustable.
4. Ensure your implementation supports multiple features.
5. Use the **sigmoid function** to map predictions to probabilities.

Step 6: Load the Dataset:

(0.5 pt.)

You will receive files named `training_dataset_logistic.csv` and `validation_dataset_logistic.csv` containing the datasets. Perform the following:

1. Load the data from the provided CSV files.
2. Understand the dataset using visualizations and basic statistical summaries.
3. Preprocess the data if necessary (e.g., handle missing values, normalize features if needed).

Step 7: Train the Logistic Regression Model:

(0.5 pt.)

1. Initialize your LogisticRegression model with a learning rate of **lr** and **iter** iterations.
2. Train the model using the fit method with the provided training dataset.

Step 8: Test and Evaluate the Model:

(1.5 pts.)

1. Use the predict method to classify examples from the validation dataset.
2. Compute the **accuracy, precision, recall, and F1-score** to evaluate the model.
3. Plot the decision boundary along with the training data on a single graph.
4. What is the equation of the decision boundary?
5. Compare your implementation with the result of LogisticRegression from scikit-learn.

The overall organization and Clarity of your solution

(2 pts.)

Ensure your solution is well-organized, clearly commented, and easy to follow

What to hand in?

Read the Assignment/Lab instructions section on the course shell.