THE UNIVERSITY of TULSA
Tandy School of Computer Science

THE UNIVERSITY OF TULSA

Project 1: Linear and Logistic Regression

In the Machine Learning course, your projects require you to write and run Python code. You will implement or modify one or more learning algorithms, train models with multiple datasets, test your models' accuracy, and produce graphs and tables depicting your models' performances.

You will be given a code shell to start with. In the code, you will be instructed to import only certain Python libraries. Importing a library that implements an algorithm and training with a model from that library does not satisfy the requirement of implementing an algorithm. You will not need to import any libraries besides those provided. Use the code shell as a guide, but feel free to restructure the code as you prefer. You may discuss algorithms with other students or chatbots, but it is up to you to write the code.

1 Overview

For this project, you will implement the linear and logistic regression algorithms. Then you will train models using five datasets and perform a hyperparameter search to find the best learning rate and regularization parameters for each dataset.

2 Algorithms

Linear Regression: Implement two linear regression algorithms. The first algorithm you will implement is the least squares normal equation. You will also include ridge (L2) regularization. The second algorithm you will implement is gradient descent with least squares error. You will include LASSO (L1) regularization and ridge (L2) regularization. These algorithms will be used to train regression models for the regression tasks.

Logistic Regression: Implement the logistic regression algorithm. You will implement gradient descent with categorical cross-entropy loss. You will include LASSO (L1) and ridge (L2) regularization. These algorithms will be used to train classification models for classification tasks.

Cross Validation: Implement a K-fold cross-validation hyperparameter search algorithm that splits the provided training dataset into K equal subsets, then trains K models for each hyperparameter assignment. It should return mean error of the models for each hyperparameter, as well as mean accuracy for classification models.

3 Datasets for Experimentation

There are eight datasets you will use to test the models. In order to demonstrate the value of regularization, you will be using a smaller subset of the data. Sample 20 datapoints without replacement for training, and test on 100 unseen datapoints. For the full digits dataset, instead train on 100 datapoints and test on 100.

3.1 Regression Datasets

Synthetic regression (linear): This dataset includes points with only one feature and one target variable. The data follows a linear equation with added noise drawn from a normal distribution.

- `Eta` $\in [0.001 : 0.02]$
- `Lambda` $\in [0.01 : 0.2]$

Synthetic regression (quadratic): This dataset also includes points with only one feature and one target variable. The data follows a quadratic equation with a added noise drawn from a normal distribution.

- `Eta` $\in [0.001 : 0.02]$
- `Lambda` $\in [100.0 : 2000.0]$

California Housing: The goal of this dataset from Statlib is to predict the median price of houses on a block based on some numeric features of the block, such as the median income of residents and the density of rooms per population. It includes 20640 house datapoints consisting of eight numeric attributes and one target house price.

- `Eta` $\in [0.01 : 0.1]$
- `Lambda` $\in [0.1 : 2.0]$

3.2 Classification Datasets

Synthetic classification (linear): This dataset includes only one feature and two classes. The data follows a linear decision boundary with added noise drawn from a normal distribution.

- `Eta` $\in [0.01 : 0.2]$
- `Lambda` $\in [0.001 : 0.02]$

Breast Cancer Wisconsin: This UCI dataset contains cell measurements of patients, where the goal is to predict whether patient has breast cancer. There are 569 datapoints across 30 features.

- `Eta` $\in [0.1 : 2.0]$
- `Lambda` $\in [0.0001 : 0.002]$

Handwritten Digits: This version of the UCI ML handwritten digits dataset includes 1797 handwritten digit images. The images contain 8x8=64 grayscale pixels. There are 10 classes for classification, which correspond to the digits 0-9.

- `Eta` $\in [1.0 : 20.0]$
- `Lambda` $\in [0.0001 : 0.002]$

Handwritten 4's and 9's: This is a subset of the Handwritten Digits dataset for binary classification. It contains greyscale handwritten 4's and 9's. While this is a classification dataset, you will test using regression techniques to learn it alongside classification techniques.

- `Eta` $\in [0.001 : 0.02]$ for linear regression
- `Lambda` $\in [0.0001 : 0.002]$ for linear regression
- `Eta` $\in [0.1 : 2.0]$ for logistic regression
- `Lambda` $\in [0.0001 : 0.002]$ for logistic regression

4 Tests

Use 10-fold cross-validation on each data set to search for the best ridge regression hyperparameter. Then train a model on all the training data. Finally, test the accuracy of the model on the test data. Below is more detail on how you will run your linear and logistic regression models.

4.1 Linear Regression

Finding learning rate: Train the linear regression models with the gradient descent algorithm on each regression dataset without regularization (`lambda`=0). Perform a hyperparameter search to find the best value for the learning rate (`eta`).

Finding regularization parameter: Using the value found for `eta`, do a `lambda` hyperparameter search for the best ridge regularization parameter for gradient descent.

Compare models: Train a linear model on the test data, using the gradient descent algorithm with no regression (`lambda`=0), ridge regression, and LASSO regression. Also train a model using the normal equation algorithm with ridge regression. Compare mean error rates across the four models.

Using linear regression on digits: You will also test the effectiveness of linear regression on the 4's and 9's handwritten digits classification task. In addition to providing the error of the model, find and report the accuracy of the model by by rounding the predictions to 0 or 1 and treating the result as a classification.

4.2 Logistic Regression

Finding learning rate: Train the logistic regression model with the gradient descent algorithm on each categorical dataset without regularization (`lambda`=0). Perform a hyperparameter search to find the best value for the learning rate (`eta`).

Finding regularization parameter: Using the value found for `eta`, do a `lambda` hyperparameter search for the best ridge regression parameter for gradient descent.

Compare models: Train a logistic regression model on the test data, using the gradient decent algorithm with no regression (`lambda`=0), ridge regression, and LASSO regression. Compare mean error rates across the three models.

Compare accuracy with linear regression Logistic regression is designed for learning classification tasks. Compare the logistic regression models' accuracy on the handwritten 4's and 9's dataset with the accuracy of the linear regression model.

5 Report

In addition to submitting your code, submit a report that includes plots from your analysis. It should include the following:

5.1 Plots

Share plots for each hyperparameter search you perform. This should include:

- Gradient Descent Regression
  - Learning rate (eta) vs. mean error for gradient descent models on all regression datasets.
  - Regression coefficient (lambda) vs. mean error for gradient descent models on all regression datasets using ridge (L2) regression.
  - Decision boundaries of the synthetic dataset models for all four models labeled and plotted together.
- Gradient Descent Classification
  - Learning rate (eta) vs. accuracy for gradient descent models on all classification datasets.
  - Regression coefficient (lambda) vs. mean error for gradient descent models on all classification datasets using ridge (L2) regression.
  - Decision boundaries of the synthetic dataset models for all four models labeled and plotted together.

Above each hyperparameter search plot, report the best and worst value for the hyperparameter. Then, report the mean square error or accuracy on test data of a model trained on the full training set with the best and the worst of the hyperparameters.

There are a few additional plots to provide.

- Show a plot of the linear predictor for each of the models trained on the synthetic regression datasets. Comment on any differences that may occur.
- Show a plot of the decision boundary for each of the models trained on the synthetic classification dataset. Comment on any differences that may occur.