CS412 Machine Learning, Fall 2024: Homework 3

April 19, 2024

Instructions

- Please submit your code as a notebook and a PDF. Name your submission as CS412-HW3-YourName.pdf and CS412-HW3-YourName-Code.ipynb, where you substitute your first and last names into the file-names in place of 'YourName'.
- You are required to code in Python. In preparing your notebook, please ensure that your code is well-organized and not confined to a single cell. Use separate cells for different sections of the code. This organization will help in making your notebook more readable.
- The responsible TA for this homework is Ekin Başar Gökce. But first, ask your questions on the Homework Forum.
- If you are submitting the homework late (see the late submission policy in the syllabus), we will grade your homework based on the time stamp of submission and your remaining late days.

Implementing Logistic Regression [100 pts]

In this homework, you are required to implement logistic regression using gradient descent from scratch. The goal is to solidify your understanding of the logistic regression model and the gradient descent optimization algorithm within the context of a binary classification task.

You will use a preprocessed version of the Titanic dataset, which contains data about Titanic passengers. You will predict if a passenger survived or not based on his/her age, sex and passenger class. In the sex column, 1 shows the passenger is female and 2 shows that the passenger is male.

1. Load the dataset and preprocess the data:
   - Set your random seed to 42.
   - Split the data into training, validation and test sets (60% , 20% , 20%).
   - As the data ranges vary significantly across the feature dimensions, you should scale your features. Scale them linearly within the 0-1 range. Be careful not to include the test data when scaling. You can use StandartScaler for that.
2. Implement the logistic regression model:
   - Initialize the model parameters w.
   - Implement the sigmoid function, $\sigma(z) = \frac{1}{1+e^{-z}}$, where z is the linear combination of the input features and the model parameters w.
   - Implement the cost function, $J(w) = -\frac{1}{m}\sum_{i=1}^{m}[y^{(i)}\log(\sigma(w^T x^{(i)})) + (1-y^{(i)})\log(1-\sigma(w^T x^{(i)}))]$.
   - Implement the gradient descent algorithm to minimize the cost function, updating the parameters as $w := w - \alpha\nabla_w J(w)$, where α is the learning rate.
3. Set the step size to 0.1. Train your model using the training data. Calculate the loss on the validation data. Plot both the training and validation losses across 100 iterations.

4. Now vary your step size and number of iterations, and calculate the validation loss in each case. Pick the one that gives you the best loss. Plot the loss curve across different iterations for the chosen values of these hyperparameters.
5. Combine the validation and training data and retrain the final model with the chosen hyperparameters.
6. Evaluate the accuracy of your model on the testing data and report the results.

Submission Guidelines

- Submit a report describing the results obtained. Properly add figure captions. Make sure the axis labels are legible, etc..
- Ensure that your code is well-commented and neatly formatted.

Grading Criteria

- Correct implementation of the logistic regression model.
- Accuracy of the model on the testing dataset.
- Quality and clarity of the report.
- Adherence to the submission guidelines.