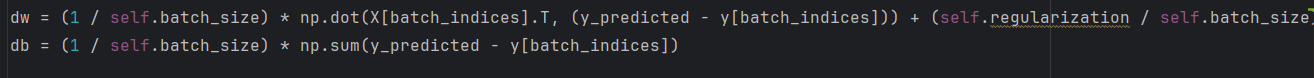
**CSE 6363: Assignment 1**

*REVANTH NAGARAJ MALLOL [ ID:1002244969]*

This assignment covers Linear Models for regression and classification. Linear Regression is a method of predicting real values given some input. Logistic Regression and Linear Discriminant Analysis will be used as classifiers.

**LINEAR REGRESSION**

1. **The fit method:**  In this method, only one column(feature) of the iris dataset will be given as input. All parameters except input, target and regularization will remain same for all linear regression models. Here, the input and target are features of the iris dataset. There will be a slight difference in the output when L2 regularization is applied to the model.

****

1. **The predict method:** This method contains code which predicts the target values based on the feature input.

**A group of white letters on a black background

Description automatically generated**

**A screen shot of a computer code

Description automatically generated**

1. **The score method:** Here, the mean squared error of difference between predicted and actual values is calculated using the formula given below.

**A number and mathematical symbols

Description automatically generated with medium confidence**

****

1. **Training:** For training each model separate scripts are created such as train\_regression1 etc. Each script uses different input data and target combination to train the model.

A screenshot of a computer code

Description automatically generated

First the iris dataset is loaded to the script. Then the X (input) is taken as the sepal length column and y (target) is taken as sepal width. By using Sklearn’s train\_test\_split, the data was divided into training and testing data with test size 10% of the data.

Then the fit method is called for the data as shown below. In this case, the output for data with and without regularization will be computed:



For regularization, and extra term is added to the weights which were calculated using gradient descent. This term is shown below.



When the train script is run, it provides the loss after each epoch (0-99).

A screenshot of a computer

Description automatically generated A screen shot of a computer

Description automatically generated

The loss decreases with each epoch of training. Hence, we plot a loss versus step number graph. In case of early stop, this message is printed.

A screenshot of a computer

Description automatically generated

The image on the left is the step loss plotted without early stop. The image on right is the loss plotted after early stop is implemented.

A graph with a line

Description automatically generated A graph with a line

Description automatically generated

With regularization set to 0:

The best weights and bias are as follows.



Also, the loss for each epoch is shown:



Now, when regularization parameter is set to 0.5, we get the following values:



The loss values are as follows:



We can see that there is a small difference in the weights and loss when regularization is applied.

After this, the model parameters are saved using pickle. The following code will create a file in the same directory as the python script which saves the model parameters such as weights. This can the be used for any other testing data.

A computer screen shot of a computer screen

Description automatically generated

1. **Testing:** In the testing script, the model parameters are loaded into the evaluation script and predicts method is called. This script prints out the predictions and mean square error. Also, the plot for the regression line is also implemented.

A computer screen shot of a program code

Description automatically generated

The same has been implemented for other 3 scripts with different combination of input and targets. Here are few examples.

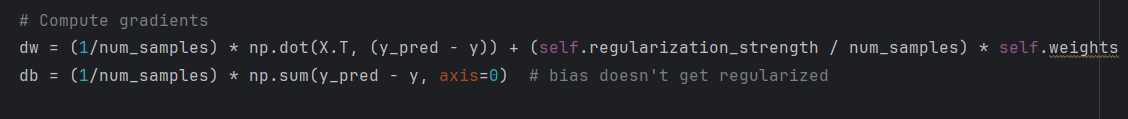
A graph with a red line and blue dots

Description automatically generated A graph with blue dots and a red line

Description automatically generated

**REGRESSION WITH MULTIPLE OUTPUTS**

1. **The fit method:**  In this method, only one column(feature) of the iris dataset will be given as input. All parameters except input, target and regularization will remain same for all linear regression models. Here, the input and target are features of the iris dataset. There will be a slight difference in the output when L2 regularization is applied to the model. These are implemented in the *eval\_MLR* script.

****

This function also plots the loss graph using the following code. The loss curve is also shown below:

A computer screen shot of a program

Description automatically generated

A graph with a blue line

Description automatically generated

1. **The predict method:** This method contains code which predicts the target values based on the feature input. The predict method is implemented in the *eval\_MLR.py* script.

****

**A black screen with white text

Description automatically generated**

The MSE is calculated as follows:

**A black background with text

Description automatically generated**

The predictions are made based on the test data and printed in the console along with mean squared error.

A black background with white text

Description automatically generated

A black background with white numbers

Description automatically generated

The graph of sepal values versus predicted petal values is also plotted in this method.

A chart with red and green dots

Description automatically generated

**LOGISTIC REGRESSION**

1. **The fit method:**  In this method, only one column(feature) of the iris dataset will be given as input. All parameters except input, target and regularization will remain same for all linear regression models. Here, the input and target are classes of the iris dataset.

**A computer screen with white text

Description automatically generated**

1. **The predict method:** This method contains code which predicts the target values based on the feature input.

The dot products of the input and weights must be passed through the sigmoid to scale the values so that they are between 0 and 1.

A screen shot of a computer

Description automatically generated

**A computer screen shot of a code

Description automatically generated**

1. **Evaluation:** For evaluation three scripts are made in which each script the iris data will be loaded, split into train-test data, and then fit method is called on the test data for model training. For training the input is 1 feature (petal width, sepal length, and all four features for *eval\_classifier* 1,2,3 respectively) and target is the species.
   1. **For petal width as input (*eval\_classifier* 1):**  We get the following predictions with an accuracy of 93.33%. The decision boundary plot for this is also shown below.

A screenshot of a computer screen

Description automatically generatedA graph of a logistic regression region

Description automatically generated

* 1. **For sepal length as input (*eval\_classifier* 2):**  We get the following predictions with an accuracy of 73.33%. The decision boundary plot for this is also shown below.

A screenshot of a computer code

Description automatically generated

A graph of a logistic regression region

Description automatically generated

* 1. **For all features as input (*eval\_classifier* 3):**  We get the following predictions with an accuracy of 66.6%.

A screenshot of a computer code

Description automatically generated