

LAB REPORT

Name -Devyani Gorkar

Roll no.-B20ME027

Question 1

Part A

- 1) Importing necessary libraries like matplotlib library and pandas and numpy to plot the scatter plot of the data.

Part B

- 1) For implementing the linear regression using the inbuilt function we need to import the `sklearn.linear_model` class to get the `LinearRegression` function. `X` is the 2D array of weight samples and `Y` is the 2D array of height samples. `Fit` function of linear regression model fits the training data to the model. `Predict` function predicts the value of the test Data. `pred` are the predicted values of the training dataset
- 2) `.coef_[0]` gives the coefficient/weight after training the model. And `intercept_` gives the intercept/bias after training the model.
- 3) `X` is the input array containing the weight values. `Y` is the array of Height samples. `Theta` is the array of weights and bias.

The code consists of 4 functions: `hypothesis`, `cost`, `gradient`, `gradientdesc`.

1. Hypothesis: This function takes 2 inputs -`X` and `theta` and returns `y_` which is the calculated height from assumed `theta`.
2. Cost: It takes 4 inputs-`X`, `Y`, `theta` and `y_`. `m` is the no. samples. It calculates the mean squared error of the hypothesis(`y_`).
3. Gradient: It takes three inputs, `X`, `Y` `theta` and calculates the the gradient of the cost function. It calculates the partial derivative of cost function with respect to weights and bias. Since it is linear regression we get two values of gradient descent .One for bias and one for weight.
4. Gradientdesc: It takes 4 inputs `X`, `Y`, `alpha`, `epochs`. Where `epochs` is the no. of iterations required to get final `theta` values. This function updates the value of `theta` in each iteration.

`y_new` is the predicted values after we obtain final `theta` values from `gradient desc` function.

Question 2

- 1) From `sklearn.preprocessing` we need to import `train_test_split` to split the dataset into training and testing parts.
- 2) The given dataset has a column of user ids which is not really helpful so we drop it from the input (`X`). The Gender column is encoded using `OneHotEncoder` and two new columns are created. So the input dataset `X` contains total 4 columns (Male, Female, Age, Estimated salary). Output is `Y` (Purchased). To train dataset we need to import `LogisticRegression` class from `sklearn.linear_model`. `logistic_reg.fit()` fits the training

dataset to get the correct model.`linear_model.predict()` predicts the output of testing data.

- 3) Using the library we get the confusion matrix and accuracy