ID6001W: Applied Deep Learning

Programming Homework 1

Homework due Jan 20th 2023 4:00 PM

Submit as single .ipynb file

Linear Regression (10 Marks)

1. The given Housing price dataset is divided into Training, Validation and Test dataset using a 70: 20: 10 ratio. (For the description of the dataset please refer this link)
2. The first 13 columns indicate features and the last column 'Price' indicates Target.
3. Use the sci-kit learn library for fitting the linear regression model. Use validation data for hyperparameter tuning. Use R-squared value (coefficient of determination) and mean Square error as evaluation metrics.
4. Using the trained model, predict the housing price for the given data in the submission.csv file.
5. Obtain the following values.
    a. Mean Squared error and R-squared value for training data and validation data respectively.
    b. Plot the scatter plot for y_{true} against $y_{predicted}$ for training data and validation respectively
6. For this part, you will have to write Python code from scratch without using any in-built python functions or external libraries, for gradient descent and use the same to train the linear model and provide the output for Questions 4 and 5 using the model you build from scratch.

    Submit a .ipynb notebook which displays the outputs as asked for in the questions above

Logistic Regression Assignment (10 marks)

Your task is to build a logistic regression model to predict whether there will be a surgical complication or not for a given patient.

Dataset consists of Train, Test and sample submission .csv files.

Both train and test dataset contain patient information as described in columns.

The last column in train file is output column which tells about the surgical complication. 1 - complication, 0 - no complication

Dataset :
https://drive.google.com/drive/folders/1FuY7PI4MGVaE2otpaGIPXJZmE9fAQtgz?usp=share_link

1. Split the given train dataset into train and validation in a ratio of 80:20
2. Plot the confusion matrix.
3. Compute the Accuracy and F1 score of the validation dataset
4. Predict the complication column of the test dataset and display

You can use the inbuilt ML library for splitting the dataset, confusion matrix, accuracy, F1 score.
Submit an ipynb notebook which shows the accuracy, F1 score, classification report, and the plot of confusion matrix. For question 4 just have a print statement to display the output.
You should use the gradient descent code you developed for training the linear model to update the parameters of the logistic regression model (Use Binary cross entropy loss). Do not use any in-built Python ML library or external libraries.

Multi-class classification Assignment(10 marks)

You are given a dataset that consists of 13 attributes which consist of the patient information and the target variable from 0 (no disease) to 4 which tells us the severity of heart disease. The goal of this assignment is to predict this target given these 13 attributes.

Downloading the dataset

You can download the dataset i.e., dataset.csv from this link. You can use this dataset for the tasks given below.

Loading the dataset

You can use the pandas to load the dataset as a dataset frame and explore the dataset provided to you. For example, you can find which features are categorical and numerical, where there are any rows/columns with null values, etc.

Data pre-processing steps

1. You will find that there are many categorical attributes in the given dataset. For the **binary attributes** i.e., sex, fasting_bloodsugar, and exercise_induced angina you can convert it as 0 or 1 .

2. For the **multi-label attributes**, use can use one hot encoding which can be done using pd.get_dummies i.e., a method in the pandas package. You can read more about one hot encoding from this link.
3. You might also find that the dataset consists of so many null values i.e., so many values are missing. If we drop all the data with null values we will lose more that 70 % of our data. In order to avoid that you can use any imputation techniques in sklearn package i.e., mean, median, or k-means imputation. For the mean imputation, you can refer to this link. You are encouraged to experiment with all the imputations mentioned above.
4. The study location attribute is not necessary to predict the severity of the disease so you can drop that column using df.drop i.e., a method in pandas dataframe

Data splitting and model training

1. Once these pre-processing steps are complete you can keep the y (which we need to predict) as the target column and all others as the X (i.e., features)
2. You can use sklearn.model_selection's method i.e., train_test_split to split the data into training and test set. You should keep the ***train_test_split(X, y, stratify=y, test_size=0.2,shuffle=True, random_state=12)*** while splitting the dataset for grading it correctly.
3. You can use MinMaxScaler()from sklearn.preprocessing to normalize our training and test data.
4. Use sklearn.linear_model's LogisticRegression for training the data and find the accuracy and plot the confusion matrix for the test data

Submit the final ipynb notebook with accuracy, classification report, and the plot of confusion matrix.

Bonus Question (5 marks)

You might see that the data is imbalanced i.e., there is more 0 compared to 1, 2, 3, 4. You can use SMOTE from the imblearn package to upsample (i.e., create data) the data so that the number of minority classes will be the same as the majority samples. Then train this dataset the same as the steps mentioned before.