Homework Assignment 3

Due: 11:59pm, February 19, 2021

This assignment includes three parts for studying the logistic regression method and model selection strategies.

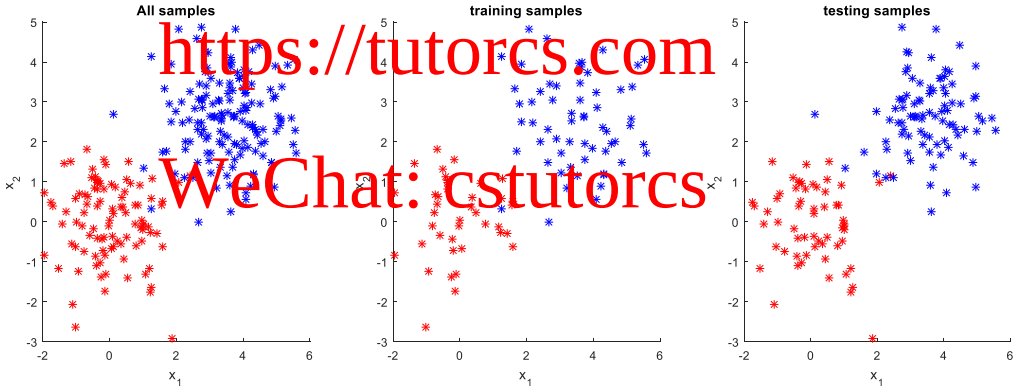
Part I. Logistic Regression

Overview. The goal of this mini-project is to predict the binary class label for any sample described with two features. You will be instructed to use Logistic Regression (LR) method for solving a simulated problem.

Sample Codes. The file "main_part1.py" provides sample codes to start with. There are four major steps: generating the dataset, training the logistic regression model, making predictions over testing samples and evaluating the prediction results.

The **first** step is to generate data (provided) and split it into training and testing subsets. You will need to write your codes to randomly split the data points into two even halves: one for training and the other one for testing (placeholder 1). Then, the code will display the splitting results as the following figures.

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs



Note that your data would be randomly generated and might look very different from the data used for generating the above figures.

The **second** step is to train a logistic regression model using the training data. To do so, you will need to use the codes we provided in the CCLE course site (week 4). Remember that there are two different implementations. Please try both implementations in this placefolder and report their performance differences (Placeholder 2).

The **third** step is to apply the learned model to get the binary classes of testing samples. This step should be modified according to the implementation of the second step (Placeholder 3). To get binary classification results, please classify all samples with 0.5 or above predictions to be positive.

The **fourth** step is to compare the binary predictions with the ground-truth labels and calculate average errors and standard deviation. Please add codes to calculate the ROC curves and AUC of the trained model over testing samples (Placeholder 4). You might use the Scikit-Learn module for this step. No need to develop your own codes for calculating ROC curves/AUC. Please include in your .pdf report the ROC Curves/AUC for both implementations of the logistic regression model.

In brief, you will need to replace the PLACEHOLDERs with your codes for splitting datasets (step 1) , training logistic regression models (step 2) , testing (step 3) and evaluation (step 4). Most of these codes are provided in the sample codes in CCLE course site (Week4, code_logistic.zip). All you need is to apply the codes to solve this given problem.

Part II. Confusion matrix

There are two questions in this part:

Question 1: Suppose that there is a trained classifier for predicting the animal classes (e.g., cat, dog) of a photo. The following table lists the prediction class and ground-truth class for each test image.

Assignment Project Exam Help

Image ID	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
True class	C	C	C	C	C	D	D	D	D	D	D	D	D	M	M	M	M	M	M	M
Predicted class	D	C	D	D	M	D	D	C	C	M	M	D	D	C	C	M	M	D	D	M

https://tutorcs.com

Notes: C, cat; D, dog; M, monkey

Please manually compute and report the confusion matrix and accuracy. For each of the three categories, calculate its precision and recall rates.

WeChat: cstutorcs

Question 2: Please write a Python function to calculate the confusion matrix for the prediction results of a binary classifier. This function should take the form:

```
def func_calConfusionMatrix(predY, trueY)
```

where predY is the vector of the binary predicted classes (1 being positive, 0 being negative) and trueY is the vector of binary true classes (1 being positive, 0 being negative). This function should return accuracy, per-class precision, and per-class recall rate.

Please use above function in the script "main_part1.py", and report the confusion matrix of both logistic regression implementations.

Part III. Cross-Validation

In lecture 3, we study how k-fold cross-validation can be used for model selection. In this part, you will need to implement a 5-fold cross-validation procedure to select the optimal *learning rate* for the gradient descendent method used in main_part1.py. Please use the self-developed implementation of logistic regression (not the scikit-learn

function). To select the optimal learning rate, you need to evenly divide the training set into 5 folds. The candidate values for the learning rate are: [0.001, 0.002, 0.005, 0.01, 0.02,0.05, 0.07, 0.1,0.5, 1.0]. For each of the above 10 learning rate, you need to

- For each of the 5 folds, train the logistic model using the other folds of training samples, and calculate the classification accuracy over the current fold.
- Average the accuracy over the five results to get the mean validation accuracy.

Please report the learning rate which achieves the highest mean validation accuracy. This is the optimal learning rate. To get binary classification results, we will classify all samples with 0.5 or above prediction values to be positive.

For each of the 10 candidate learning rates, please repeat the following experiment: train the logistic model with all the training samples, and apply the trained model over the testing samples; calculate the model's accuracy over testing samples. Please report if the optimal learning rate actually achieve the best testing accuracy. Note that it is possible the model with the optimal learning rate does not outperform all other candidates.

Assignment Project Exam Help

What to submit

- **Part I**: Submit the main_part1.py script with completed placeholders; Include the required writeup (results, figures, observations, etc.) in the PDF.
- **Part II**: Submit the python function for calculating the confusion matrix. You might include this function in main_part1.py script. Include the needed writeup in the PDF report.
- **Part III**: Modify the main_part1.py to implement the 5-fold cross-validation. Submit the modified .py file. Include the needed writeup in the PDF report.

https://tutorcs.com

WeChat: cstutorcs

How to submit

- Submit your source codes and the report through the `course site`. The codes should be self-contained, and run without any error.
- ***No hard copy required.***