Problem Set - 3

Please read all of the guidelines carefully before submitting the problem set. (Unless specified) each question is ***20 points*** and there are ***100 points*** in total.

Due date: Friday, February 11, 11:59 PM. Late submissions will be accepted with a penalty! (10% reduction per day – no submissions accepted two days after the deadline.)

Guidelines – Before You Start

1) **You should complete the problem set on your own.** Discussing ideas is fine; but, sharing answers and sharing code will be considered as plagiarism.
2) You will be using the **`Python`** programming language. You need to write your codes in an empty **.ipynb** file.
3) Make sure that you provide many comments to describe your code and the variables that you created.
4) Please use **LaTeX** or **MS Word** to submit your written responses (hand-written responses will not be graded).
5) For some of the coding exercises, you may need to do a little bit of "**Googling**" or review the documentation.

Deliverables:

1) The code of the problem set in **.ipynb** format (one file)
2) Short answers written with ***LaTeX*** or **MS Word** and exported in **.pdf** format (one file)

Questions

In this homework, you have three questions. The first question is worth ***20 points***. The remaining two questions are each worth ***40 points***.

1) Read the following highly-cited article by Fearon and Laitin (2003): https://cisac.fsi.stanford.edu/publications/ethnicity_insurgency_and_civil_war

   Answer the following:

   a. What is the paper about? (Please write a paragraph – max. 250 words)
   b. How many observations do authors have in their dataset? What does each observation represent? (=What is the unit of analysis?)
   c. What is the identification strategy of the authors? (=What are the different regression equations they are running?) Please write down in form of equations and explain. Identify the independent and dependent variables.
   d. What do the coefficient values listed in Table 1 represent? (*theoretically speaking*)
   e. Which independent variables have positive coefficients? Which independent variables have negative coefficients? Which ones are statistically significant?
   f. Thinking about the range of your independent variables, which variables do you think have a greater impact on the dependent variable(s)?

2) Build a two-class logistic regression model from scratch. You will need to work on the following:
   a. Implement the sigmoid function from scratch and call it `sigmoid_f`
   b. Implement the hypothesis function from scratch and call it `classifier_f`
   c. Implement the entropy function as your cost function and call it `binary_loss_f`
   d. Implement gradient descent for logistic regression and call it `gradient_f`
   e. Combining the functionalities of what you have coded above, create an optimizer function and call it `optimizer_f`. Note: You should find out the *input* and *output* to the functions above by reviewing the class notes and the textbook; in other words, this will be part of the challenge! If needed, use `265` as your random seed.

   Let's test your code on a dataset. Load the **Breast Cancer Wisconsin Dataset** provided by **`sklearn`**: https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_breast_cancer.html#sklearn.datasets.load_breast_cancer

   Now, do the following:

   a. Set the **`target`** column as your ***Y*** variable.
   b. Set all other *numeric* variables (excluding index) as your ***X*** matrix.
   c. Apply 0-1 normalization on both the ***X*** matrix and ***Y*** vector.
   d. Run logistic regression by using the code you have written (no need to do train/test split). Set the maximum number of iterations to **10,000**.
   e. Report the final equation you have obtained for logistic regression.
   f. Also indicate which coefficients are positively associated and which coefficients are negatively associated with the **`target`** variable. Rank them from positive to negative. Interpret the results.

3) Implement the three following cross-validation algorithms from scratch:
   a. Leave-one-out cross-validation
   b. K-fold cross-validation
   c. Train-test split cross-validation

   Test your results on the California Housing Dataset:
   https://scikit-learn.org/stable/modules/generated/sklearn.datasets.fetch_california_housing.html#sklearn.datasets.fetch_california_housing

   Now, do the following:

   a. Implement the ***cross-validation algorithms*** from scratch.

b. Choose the following features from the dataset as your ***X*** matrix: ***MedInc, HouseAge, AveRooms, AveBedrms, Population, AveOccup, Latitude, Longitude***

c. Choose the following feature from the dataset as your ***Y*** matrix: ***MedHouseVal***

d. Apply 0 – 1 normalization on ***X*** and ***Y***.

e. Apply the cross-validation algorithms that you implemented to train your model. (For splitting your data always use `265` as your random number or seed value). Note: You cannot use the pre-packaged algorithms for splitting the data. To split the data please do the following:

   a. Install the **`random`** package written for **`Python`**
   b. Set (the initial) `random.seed()` to `265`
   c. Create a list of integers that will function as your index numbers: `list(range(0,len(name_of_dataset))`
   d. Pick ***one*** integer for ***Train-Test Split CV*** from the list your created in ***c)*** to split 70% of your data as training set and the remaining 30% as the test set.
   e. For the ***K-fold CV***, set ***k = 5***. Please divide the dataset into 5 quasi-equal portions starting from index 0.
   f. For ***LOOCV***, start the training by randomly picking a feature vector associated with an index in your dataset (Reminder: random seed is `265`) – you will need to run the model on every point.

f. Using `scikit`'s `sklearn.linear_model.LinearRegression`, predict the house prices by using all of the data in your ***X*** matrix. Compare different techniques of CV. Which CV provides the lowest ***MSE***? Why? Interpret the results.