Midterm Project [ROBT407]

Brief Information

In this project, students will practice the theoretical knowledge learned about a class of learning machines, such as Linear and Logistic Regression. Besides, you will be asked to work with real-world problems and participate in the Kaggle competition. The main objective of this project is to give you essential hands-on experience in implementing a specific algorithm and using available tools in ScikitLearn.

Method of Delivery

Assignment deliverables should be submitted via Moodle to the course instructor before the due date.

Level of Collaboration Allowed

team should consist of at most three students and at least two students on this assignment. If you cannot form a group or have difficulties finding team partners, please inform the instructor.

Assignment Deliveries

- A well-documented Jupyter Notebook report describing in sufficient detail the work includes the source codes, the approach for solving the problem, implementation, results, difficulties, limitations, etc.
- Your report should contain a statement clearly stating the contribution of each member in a team.
- Archive all the files with .zip or .tar extension and submit as one file to the Moodle.

Grading Criteria

- 35% - Implementation (a well documented Jupyter Notebook)
- 10% - The accuracy of the best model that has been selected after cross-validation
- 20% - Overall work and report quality
- 20% - Discussion (for example of success/failure;limitations, etc.)
- 15% - A video presentation of the entire workflow using free screen capture software. The presentation should explain each code and summarize the work, including your Kaggle submission, in detail. The duration of presentation should be around 20 - 30 minutes.

Midterm Project [ROBT407]

1 Linear Regression -20%

Implement the linear regression algorithm in Section 3.2 of LFD to compute the optimal $(d+1)$-dimensional w that solves

$$\underset{w}{\text{minimize}} \sum_{n=1}^{N} (y_n - (w^\intercal x_n))^2 \tag{1}$$

- Generate a training data set of size 100 as directed by Exercise 3.2 of LFD. Generate a test set of size 1000 of the same nature.
- Run the pocket algorithm (Homework 1) on the training set for $T = 1000$ to get w_{pocket}.
- Run the linear regression algorithm to get w_{lin}. Estimate the performance of the two weight vectors with the test set to get $E_{test}(w_{pocket})$ and $E_{test}(w_{lin})$, in terms of the 0/1 loss (classification).
- Repeat the experiment (with fresh data sets) 100 times and plot $E_{test}(w_{pocket})$ versus $E_{test}(w_{lin})$ as a scatter plot.
- Based on your findings in the previous problem, which algorithm would you recommend to your boss for this data set? Why?

2 Logistic Regression and Gradient Descent -20%

Consider the formulation,

$$\underset{w}{\text{Minimize}} \quad E(w)$$

$$\text{where,} \quad E(w) = \frac{1}{N}\sum_{n=1}^{N} E^{(n)}(w)$$

$$E^{(n)}(w) = \ln\Big(1 + \exp(-y_n(w^{\intercal}x_n))\Big)$$

Implement the fixed learning rate stochastic gradient descent algorithm for Eq.2.

a) initialize a $(d+1)$ -dimensional vector $w^{(0)}$, say, $w^{(0)} \leftarrow (0, 0, ..., 0)$

b) for $t = 1, 2, ..., T$

- randomly pick one n from $\{1, 2, ..., N\}$.
- update

$$w^{(t)} \rightarrow w^{(t-1)} - \eta \nabla E^{(n)}(w^{(t-1)}) \tag{2}$$

- Assume that

$$g_1^t(x\ ) = \text{sigmoid}\Big((w^{(t)})^T x\Big) \tag{3}$$

- where $w(t)$ are generated from gradient descent algorithm above.

  Run the algorithm with $\eta = 0.001$ and $T = 2000$ on the IRIS data set after splitting it into D_{train} (80%) and on the D_{test} (20%) You can use get the IRIS data as follows or from the Scikitlearn

```
import seaborn as sns
iris = sns.load_dataset('iris')
iris.head()
```

  .

- Plot $E_{in}(g_1^{(t)})$ and $E_{test}(g_1^{(t)})$ as a function of t, and briefly state your findings.

You can use One-Versus-All (OVA) decomposition to solve the three class problem as below.

- for $k \in \mathcal{Y}$ obtain $\mathbf{w}_{[k]}$ by running logistic regression on

$$\mathcal{D}_{[k]} = \{(\mathbf{x}_n, y_n' = 2\,[y_n = k] - 1)\}_{n=1}^{N}$$

- return $g(\mathbf{x}) = \text{argmax}_{k\in\mathcal{Y}}\left(\mathbf{w}_{[k]}^T\mathbf{x}\right)$

Or you can select classes from the data and work on a binary classification task. However, the OVA approach is preferred.

3 Practical design of a learning algorithm - 30%

- Given training data consisting of input-output pairs, **model selection** in machine learning is a process that builds a model to predict the output from the input, usually by learning optimal adjustable parameters. Many models exist in literature to perform such tasks, including linear, logistic models, SVMs, etc. The main objective of this task is to find and compare methods to optimally select a model which will perform best on new test data.

The following steps summarize the important steps in model selection:

1. Consider a dataset D (from any domain e.g. credit/ medical / digit recognition)
2. Apply an algorithm of your choice (e.g. Lin Reg, Logistic Reg. etc) on D
3. Estimate its generalization error (E_{test})
4. **If**: generalization error smaller than what exists in the literature for the same dataset:
   - End of the process: Outcome
5. **Else**:
   - Go back to step 2 with another algorithm or change the learning strategy

In the model validation lecture (see Textbook section 4.3), the V-fold cross-validation is a commonly used method to estimate generalization error (or perform model selection), especially when there is little training data.

Specific tasks:

Task 1: Review the lecture 12 on validation

Task 2: Watch the lectures 13, 14 on scikit learn machine learning library

Task 3: Load Optical Recognition of Handwritten Digits Data Set

```
from sklearn import datasets
digits = datasets.load_digits()
# read the description
print(digits.DESCR)
```

Task 4: Using the Linear & Logistic Regression coded in Section `1` and `2` implement a routine that uses tenfold cross validation for model selection on digits dataset.

For Task 4 implement the following model selection steps:

(a) Write a function that divides the on digits dataset (training) set (of size m) into n disjoint sets $S_1, ..., S_n$ of equal size n/m

(b) For each S_i:

- Train a classifier (e.g. Lin Reg. Log Reg) on $S \backslash S_i$
- Test it on $S_i \leftarrow error(i)$

(c) Output the average error

This estimates the classifier's generalization error when trained on $n - n/m$ data. Report the comparative analysis of Linear and Logistic regression performance and when you change the number of folds invalidation (5 fold vs. ten fold vs. 20 fold vs. loocv). Refer to chapter 4 of the textbook for more details.

Task 5: Perform GridSearchCV based tenfold cross-validation using the Scikit-learn module and compare the performance of Linear and Logistic regression on digits dataset.

* For each model, plot the validation curves using Scikit-Learn to justify your selection of a final model. Explain how you addressed the bias-variance tradeoff and that your model has not overfitted the digits dataset.

4 KAGGLE Competition - 30%

- Check out the following website: (https://www.kaggle.com/competitions)

In this task, you need to choose a problem of interest from the Kaggle website and participate in a competition. Using the scikit-learn: machine learning toolbox and jupyter notes as your primary tool would be best. You are expected to submit your solutions to the website and share the link to your submission and well-documented Jupyter notebook solutions.

For instance, you can choose the following problems to work on (or many others based on your choice on the Kaggle platform):

1) Netflix Prize Dataset

Netflix Prize data set gives 100 million records of the form "user X rated movie Y a 4.0 on 2/12/05". The data is available here: **Netflix Prize**

Project Task:

- Can you predict the rating a user will give on a movie from the movies the user has rated in the past and the ratings similar users have given similar movies?
- Can you discover clusters of similar movies or users?
- Can you predict which users rated which movies in 2006? In other words, your task is to predict the probability that each pair was rated in 2006. Note that the current rating is irrelevant, and we want whether that user rated the movie sometime in 2006. The date in 2006 when the rating was given is also irrelevant. The test data can be found on this website.
- Reference

  https://www.kaggle.com/netflix-inc/netflix-prize-data

2) Fashon-MNIST

- **Context**

  Fashion-MNIST is a dataset of Zalando's article images—consisting of a training set of 60,000 examples and a test set of 10,000 examples. Each example is a 28x28 grayscale image associated with a label from 10 classes. Zalando intends Fashion-MNIST to direct drop-in replacement for the original MNIST dataset for benchmarking machine learning algorithms. It shares the same image size and structure of training and testing splits. The original MNIST dataset contains a lot of handwritten digits. Members of the AI/ML/Data Science community love this dataset and use it as a benchmark to validate their algorithms. MNIST is often the first dataset researchers try. "If it doesn't work on MNIST, it won't work at all", they said. "Well, if it does work on MNIST, it may still fail on others."

- **Content**

  Each image is 28 pixels in height and 28 pixels in width for 784 pixels in total. Each pixel has a single pixel value associated with it, indicating the lightness or darkness of that pixel, with higher numbers meaning darker. This pixel-value is an integer between 0 and 255. The training and test data sets have 785 columns. The first column consists of the class labels (see above) and represents clothing. The rest of the columns contain the pixel-values of the associated image. To locate a pixel on the image, suppose that we have decomposed x as x = i * 28 + j, where i and j are integers between 0 and 27. The pixel is located on row i and column j of a 28 x 28 matrix. For example, pixel31 indicates the pixel in the fourth column from the left and the second row from the top, as in the ASCII diagram below.

3) Chinese-MNIST

Consider working on the Chinese MNIST dataset as well:
https://www.kaggle.com/gpreda/chinese-mnist

4) Other

Other projects that you may be interested in working with...