Due Date October 19, 2020
Late Submissions 30% per day
Teams You can do the assignment individually or in teams of 3.
Teams must submit only 1 copy of the assignment via the team leader's account.
Purpose The purpose of this assignment is to make you experiment with machine learning.

---

1 Experiments with Machine Learning

1.1 `scikit-learn`

For this assignment, you will use the `scikit-learn` machine learning framework to experiment with different machine learning algorithms and different data sets. The focus of this assignment lies more on the experimentations and analysis than on the implementation.

`scikit-learn` is an open-source machine learning library for Python (see http://scikit-learn.org/stable/), which provides an interface to program with a variety of different algorithms and built-in datasets. There are plenty of online documentation and examples of code online.

1.2 Data Sets

You must use the 2 datasets provided on Moodle (see the zip file `DataSet-Release1`). Both datasets are about the classification of black & white images of size 32×32 that represent a character. For example, the image  represents the character 'A'.

Dataset 1 contains images of the 26 uppercase letters [A – Z].

Dataset 2 contains images of 10 Greek letters.

Each character in the datasets is represented by an index, as indicated in the table below:

Dataset 1						Dataset 2	
index	char.	index	char.	index	char.	index	char.
0	A	10	K	20	U	0	π (pi)
1	B	11	L	21	V	1	α (alpha)
2	C	12	M	22	W	2	β (beta)
3	D	13	N	23	X	3	σ (sigma)
4	E	14	O	24	Y	4	γ (gamma)
5	F	15	P	25	Z	5	δ (delta)
6	G	16	Q			6	λ (lambda)
7	H	17	R			7	ω (omega)
8	I	18	S			8	μ (mu)
9	J	19	T			9	ξ (xi)

Each dataset is in `.csv` format, where each row is a data instance. Each instance is composed of 1024 binary features followed by its class (the index). Each dataset contains 3 splits:

- **training**: to be used for training your models.
- **validation**: to be used for validating/experimenting with your models.
- **test**: to be used to report your final output.

2 Your Task

For each dataset, write the necessary code to:

1. Plot the distribution of the number of the instances in each class.
2. Run 6 different ML models:
   (a) **GNB:** a Gaussian Naive Bayes Classifier, with default parameter values.
   (b) **Base-DT:** a baseline Decision Tree using entropy as decision criterion and using default values values for the rest of the parameters.
   (c) **Best-DT:** a better performing Decision Tree found by performing grid search to find the best combination of hyper-parameters. For this, you need to experiment with the following parameter values:
      - splitting criterion: gini and entropy
      - maximum depth of the tree: 10 and no maximum
      - minimum number of samples to split an internal node: experiment with values of your choice
      - minimum impurity decrease: experiment with values of your choice
      - class weight: None and balanced
   (d) **PER:** a Perceptron, with default parameter values..
   (e) **Base-MLP:** a baseline Multi-Layered Perceptron with 1 hidden layer of100 neurons, sigmoid/logistic as activation function, stochastic gradient descent, and default values for the rest of the parameters.
   (f) **Best-MLP:** a better performing Multi-Layered Perceptron found by performing grid search to find the best combination of hyper-parameters. For this, you need to experiment with the following parameter values:
      - activation function: sigmoid, tanh, relu and identity
      - 2 network architectures of your choice: for eg 2 hidden layers with 30+50 nodes, 3 hidden layers with 10+10
      - solver: Adam and stochastic gradient descent
3. For each model and each dataset, write the necessary code to generate a `csv` (comma separated values) output file that contains the output classification and the performance of each model for each dataset. This output file should be named `[model_name]-[dataset].csv`. Therefore you should generate 12 files:

   ```
   GNB-DS1  Base-DT-DS1  Best-DT-DS1  PER-DS1  Base-MLP-DS1  Best-MLP-DS1
   GNB-DS1  Base-DT-DS2  Best-DT-DS1  PER-DS1  Base-MLP-DS2  Best-MLP-DS2
   ```

   These files should contain:
   (a) the row number of the instance, followed by a comma, followed by the index of the predicted class of that instance, as in:

   ```
   1,24   // if your model's predicted class for instance 1 is 24 (Y)
   2,25   // if your model's predicted class for instance 2 is 25 (Z)
   3,4    // if your model's predicted class for instance 3 is 4 (E)
   ```

   (b) a plot the confusion matrix
   (c) the precision, recall, and f1-measure for each class
   (d) the accuracy, macro-average f1 and weighted-average f1 of the model

3 Deliverables

The submission of the assignment will consist of 3 deliverables:

(a) The code & output files
(b) The demo (8 min presentation & Q/A)

3.1 The Code & Output files

Submit all files necessary to run your code in addition to a `readme.md` which will contain specific and complete instructions on how to run your experiments. You do not need to submit the datasets. If the instructions in your readme file do not work, are incomplete or a file is missing, you will not be given the benefit of the doubt.

Generate one output file for which model and each dataset test sets as indicated in Section 2.

3.2 The Demos

You will have to demo your assignment for ≈ 12 minutes. Regardless of the demo time, you will demo the program that was uploaded as the official submission on or before the due date. The schedule of the demos will be posted on Moodle. The demos will consist in 2 parts: a presentation ≈ 8 minutes and a Q/A part (≈ 4 minutes). Note that the demos will be recorded.

3.2.1 The Presentation

Prepare an 8-minute presentation to analyse and compare the performance of your models. The intended audience of your presentation is your TAs. Hence there is no need to explain the theory behind the models. Your presentation should focus on **your** work and the comparison of the performance of the models when the hyper-parameters are modified.

Your presentation should contain at least the following:

- ☐ An analysis of the initial dataset given on Moodle. If there is anything particular about these datasets that might have an impact on the performance of some models, explain it.
- ☐ An analysis of the results of all the models with the data sets. In particular, compare and contrast the performance of each model with one another, and with the datasets. Please note that your presentation must be analytical. This means that in addition to stating the facts (e.g. the macro-F1 has this value), you should also analyse them (i.e. explain why some metric seems more appropriate than another, or why your model did not do as well as expected. Tables, graphs and contingency tables to back up your claims would be very welcome here.
- ☐ In the case of team work, a description of the responsibilities and contributions of each team member.

Any material used for the presentation (slides, ...) must be uploaded on EAS before the due date.

3.2.2 Q/A

After your presentation, your TA will proceed with a ≈ 4 minute question period. Each student will be asked questions on the code/assignment, and he/she will be required to answer the TA satisfactorily. In particular, each member should know what each parameters that you experimented with represent and their effect on the performance. Hence every member of team is expected to attend the demo.

In addition, your TA may give you a new dataset and ask you to train or run your models on this dataset. The output file generated by your program will have to be uploaded on EAS during your demo.

4 Evaluation Scheme

Students in teams can be assigned different grades based on their individual contribution to project. Individual grades will be based on:

(a) a peer-evaluation done after the submission.

(b) the contribution of each student as indicated on GitHub.

(c) the Q/A of each student during the demo.

The team grade will be based on:

Code	functionality, proper use of the datasets, design, programming style, ...	6
Output with initial datasets	correctness and format	1.5
Demo – Presentation	depth of the analysis, clarity and conciseness, presentation, time-management, ...	4
Demo – QA	correct and clear answers to questions, knowledge of the program, ...	2
Output with demo-dataset	correctness and format	1.5
Total		15

5 Submission

If you work in a team, identify one member as the team leader. The only additional responsibility of the team leader is to upload all required files (including the files at the demo) from her/his account and book the demo on the Moodle scheduler. If you work individually, by definition, you are the team leader of your one-person team.

5.1 Submission Schedule

Each deliverable is due on the date indicated below.

Deliverable	**Due Date**	**Upload as**
Submit your code, output files, presentation material	October 19, 2020, 11:59pm	Assignment 1
Submit the output files generated at demo time	during your demo	Assignment 1

5.2 Submission Checklist

In your GitHub project, include a `README.md` file that contains:

(a) on its first line: the URL of your GitHub repository,

(b) specific and complete instructions on how to run your program.

Code & Output files

- ☐ Create one zip file containing all your code, the output files for the initial test set on Moodle and the `README.md` file.
- ☐ Name your zip file: `472_Assignment1_ID1_ID2_ID3.zip` where `ID1` is the ID of the team leader.
- ☐ Have the team leader upload the zip file at: https://fis.encs.concordia.ca/eas/ as `Assignment1`.

Demo & Outut files During your actual demo with the TA:

- ☐ Prior to your demo, make your GitHub repository public.
- ☐ Generate the output files for the test set that the TA will give you.
- ☐ Create a zip file called: `472_Demo1_ID1_ID2_ID3` where `ID1` is the ID of the team leader.
- ☐ Have the team leader upload the zip file at: https://fis.encs.concordia.ca/eas/ as `Assignment1`.