Introduction to Machine Learning - Exercise 2

Due Date: November 23th 22:00, 2021

Yosi Shrem, Yael Segal, Roni Chernyak and Yossi Keshet

November 9, 2021

Guidelines

1. You are not allowed to use external packages other than numpy (KNN should be fully implemented , using an external implementation is forbidden).
2. In order to submit your solution please upload your files to `Submit` and check your inbox for the feedback mail.
3. Technical questions about this exercise should be asked at the course' piazza.
4. Private/Personal issues regarding the deadline should be directed to **Yael Segal**.

1 KNN, Perceptron, SVM, and Passive Aggressive (PA)

In this exercise you will implement and compare KNN, Perceptron, SVM, and PA. You are provided with a training set of 240 examples where you need to classify between 3 classes.

To make things easier for you, here are the update rules for all models:

Perceptron

$$w_{t+1}^{y} = w_t^{y} + \eta * x$$

$$w_{t+1}^{\hat{y}} = w_t^{\hat{y}} - \eta * x$$

$$w_{t+1}^{i \neq \hat{y}, y} = w_{t+1}^{i \neq \hat{y}, y}$$

SVM

$$w_{t+1}^{y} = (1 - \eta\lambda) w_t^{y} + \eta * x$$

$$w_{t+1}^{\hat{y}} = (1 - \eta\lambda) w_t^{\hat{y}} - \eta * x$$

$$w_{t+1}^{i \neq \hat{y}, y} = (1 - \eta\lambda) w_{t+1}^{i \neq \hat{y}, y}$$

PA

$$w_{t+1}^{y} = w_t^{y} + \tau * x$$

$$w_{t+1}^{\hat{y}} = w_t^{\hat{y}} - \tau * x$$

$$w_{t+1}^{i \neq \hat{y}, y} = w_{t+1}^{i \neq \hat{y}, y}$$

where τ is set to:

$$\tau = \frac{\ell(w, x, y)}{2 \cdot ||x||^2}$$

2 Dataset

Your dataset is the Iris flower classification. In this dataset, you are provided with five features per instance (all the features are numerical) and three labels, which are correspond to the Iris flower species. You should try different normalization techniques as well as feature selection.

3 Code

Your main file should be called: `ex2.py`. Your code should get as input four arguments.

The run command to your program should be:

```
$ python ex2.py <train_x_path> <train_y_path>  <test_x_path> <output_log_name>

For example:
$ python ex2.py train_x.txt train_y.txt test_x.txt out.txt
```

The first parameter will be the training examples (`train_x.txt`), the second param is the training labels (`train_y.txt`), the third param will be the testing examples (`test_x.txt`) , and the fourth one will be the output file name. Notice, `train_x.txt` and `test_x.txt` will have the same format. You should train 4 algorithms learned in class: KNN, Perceptron, SVM and PA (in that order). Next, you should output your predictions (only) for `test_x.txt` to a file in the following format:

```
knn: 0, perceptron: 0, svm: 0, pa: 1
knn: 1, perceptron: 1, svm: 1, pa: 1
knn: 2, perceptron: 2, svm: 2, pa: 2
...
use the following line to match your output to the requested format:
f.write(f"knn: {knn_yhat}, perceptron: {perceptron_yhat}, svm: {svm_yhat}, pa: {pa_yhat}\n")
```

where each line in your output corresponds to a line in `test_x.txt` and the numbers represent your class predictions for classes 0,1,2. Notice, all your hyper-parameters should be hardcode, no external arguments will be provided during runtime.

Notice: The dataset is almost linearly separable (can be seen in 1). As a result, you should limit your training to a max number of epochs. In addition, for each algorithm, you should use the best model (the best w).

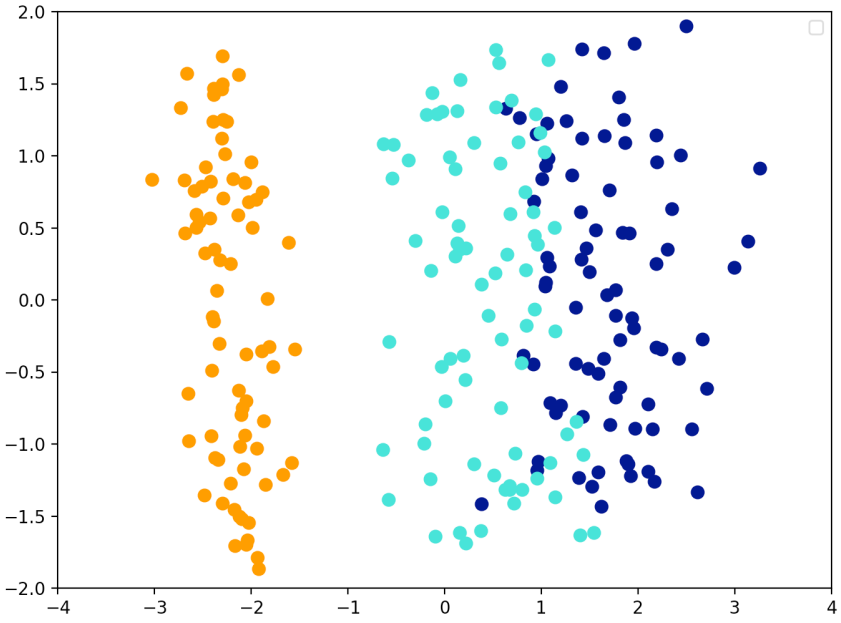

Figure 1: PCA reduction of the features

4 What to submit?

You should submit the following files:

- A `txt` file, named `details.txt` with your name and ID.

  ```
  name: FIRST_NAME LAST_NAME
  ID: 123456789
  ```

- Python 3.6+ code. Your main function should reside in a file called `ex2.py`. The main function should train all four models and output the predictions as described in 3.
- A `PDF` report named `report.pdf` with all the implementation details and hyper-parameters, (How did you choose the learning rate? What values did you try and how was the performance? How did you choose the lambda value for SVM? How did you choose the k for KNN? etc.).
- Part of your grade will consist of automatic checks using the `Submit` system. Make sure your output matches the expected output described above.
- **Note**: your code should run up to 5 minutes long. Submissions that will run longer than that, will be graded as 0.

Overall : `ex2.py, details.txt` and `report.pdf`