Introduction to Machine Learning Homework 2 Announcement

Presenter: TA Jui-Che (Ben)

Lastest update: 2023/10/30 10:00

Homework 2

- Deadline: 23:59, Nov. 14th (Tue), 2023
- Coding (50%): Implement linear classification methods by only using **numpy.**
  - Part 1: Logistic Regression
  - Part 2: Fisher's Linear Discriminant
  - Submit your python file (.py).
  - Answer the questions (by screenshots) in the report (.pdf).
- Handwritten Questions (50%): Answer questions about linear classification methods.
  - Answer the questions (handwritten, typed, digital, etc.) in the report.

Links

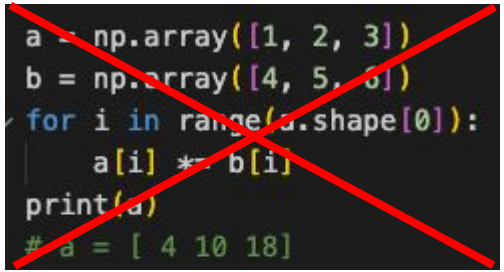
- Questions: Link
- Sample code: Link
- Dataset: Link
- Report template: Link (same as HW1)

Environment

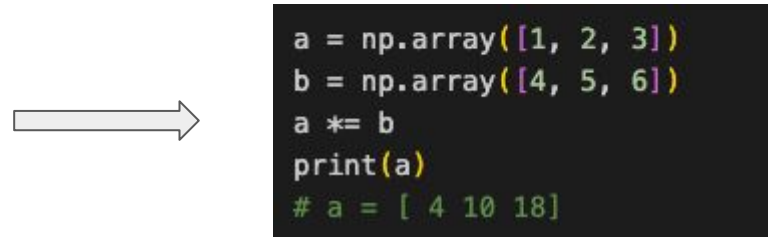
- Python version: 3.8 or newer
- Tips
  - We recommend that you use **virtual environments** when implementing your homework assignments.
  - Here are some popular virtual environment management tools:
    - Conda
    - Miniconda
    - virtualenv

Numpy

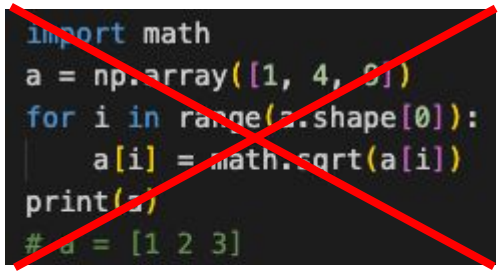
- Build-in array operations.
- Numpy Tutorial: [Link]



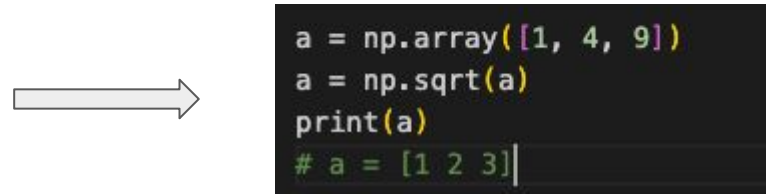
```
a = np.array([1, 2, 3])
b = np.array([4, 5, 6])
for i in range(a.shape[0]):
    a[i] *= b[i]
print(a)
# a = [ 4 10 18]
```



```
a = np.array([1, 2, 3])
b = np.array([4, 5, 6])
a *= b
print(a)
# a = [ 4 10 18]
```



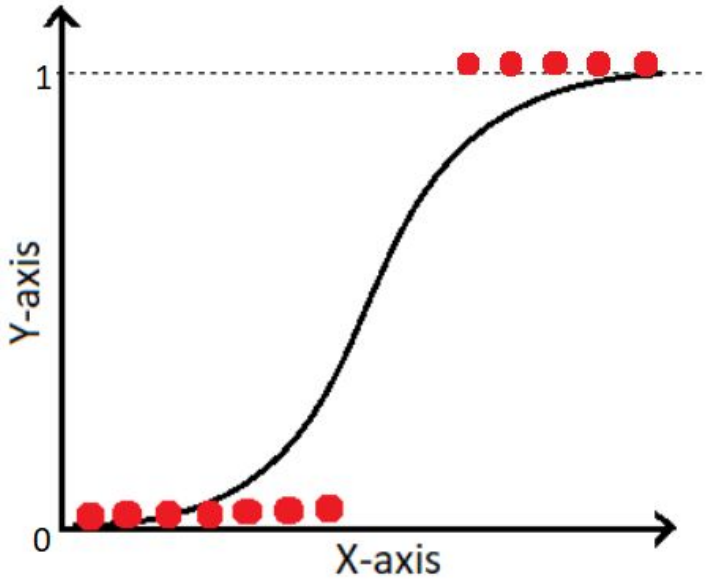
```
import math
a = np.array([1, 4, 9])
for i in range(a.shape[0]):
    a[i] = math.sqrt(a[i])
print(a)
# a = [1 2 3]
```



```
a = np.array([1, 4, 9])
a = np.sqrt(a)
print(a)
# a = [1 2 3]
```

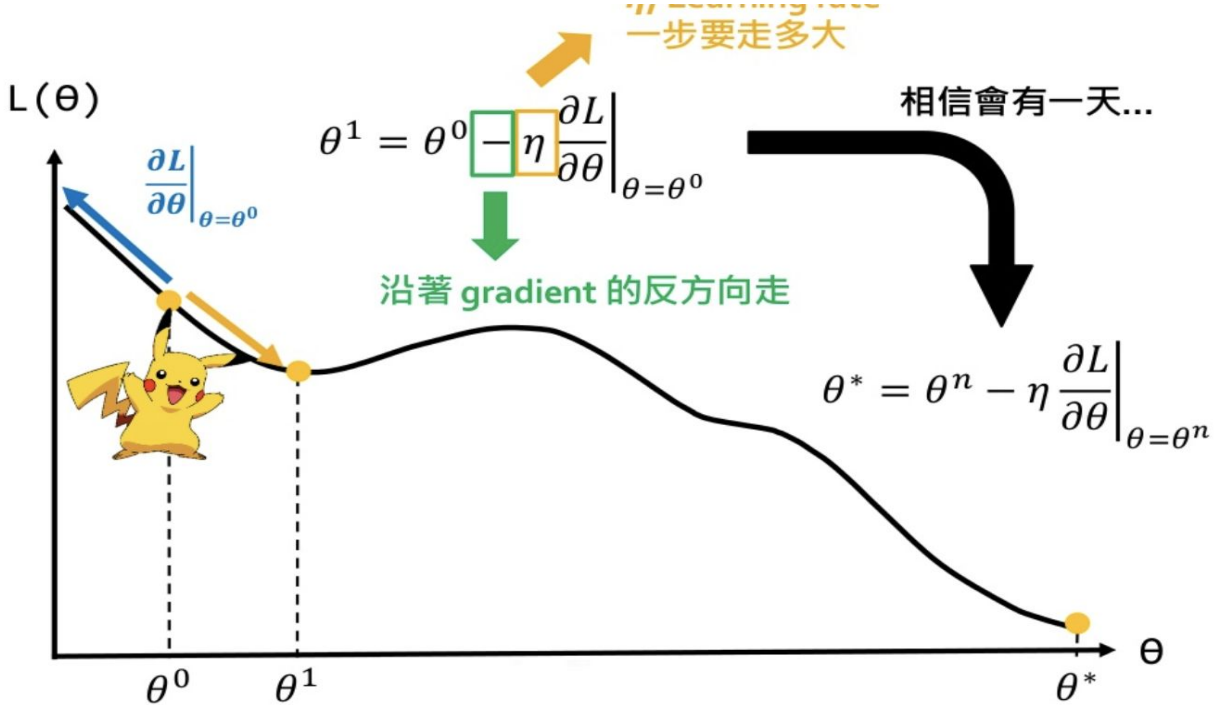
Logistic Regression

- Find the best value of the weights and the intercept of a logistic model



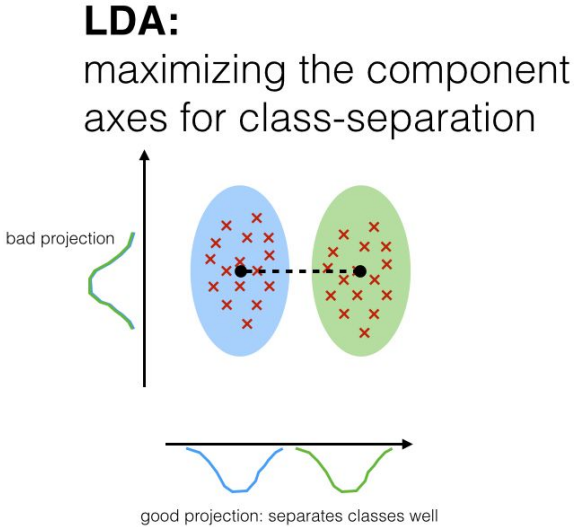
Logistic Regression

- Gradient Descent



Fisher's Linear Discriminant

- FLD seeks the projection w that gives a large distance between the projected data means while giving a small variance within each class.



$$J(\boldsymbol{W}) = \frac{(m_2 - m_1)^2}{s_1^2 + s_2^2}$$

Between-class variance

Within-class variance

https://sebastianraschka.com/Articles/2014_python_lda.html, https://sthalles.github.io/fisher-linear-discriminant/

Dataset

- Heart Attack Dataset
- Features
  - age
  - sex
  - cp: chest pain type (4 values)
  - fbs: fasting blood sugar > 120 mg/dl
  - thalach: maximum heart rate achieved
  - thal: 0 = normal; 1 = fixed defect; 2 = reversable defect
- Target
  - target (0 = no heart attack, 1 = heart attack)

Logistic Regression

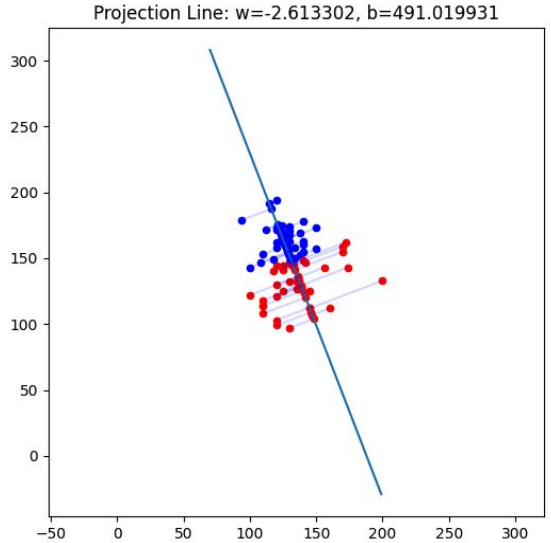
- Requirements:
  - Use Gradient Descent to update your model
  - Use CE (Cross-Entropy) as your loss function.
- Criteria:
  - (0%) Show the hyperparameters (learning rate and iteration) that you used.
  - (5%) Show your weights and intercept of your model.
  - (10%) Show the accuracy score of your model on the testing set.

Fisher's Linear Discriminant

- Requirements:
  - Implement FLD to reduce the dimension of the data from 2-dimensional to 1-dimensional.
- Criteria:
  - (0%) Show the mean vectors m_i (i=0, 1) of each class of the training set.
  - (5%) Show the within-class scatter matrix S_W of the training set.
  - (5%) Show the between-class scatter matrix S_B of the training set.
  - (5%) Show the Fisher's linear discriminant w of the training set.
  - (10%) Obtain predictions for the testing set by measuring the distance between the projected value of the testing data and the projected means of the training data for the two classes. Show the accuracy score on the testing set.

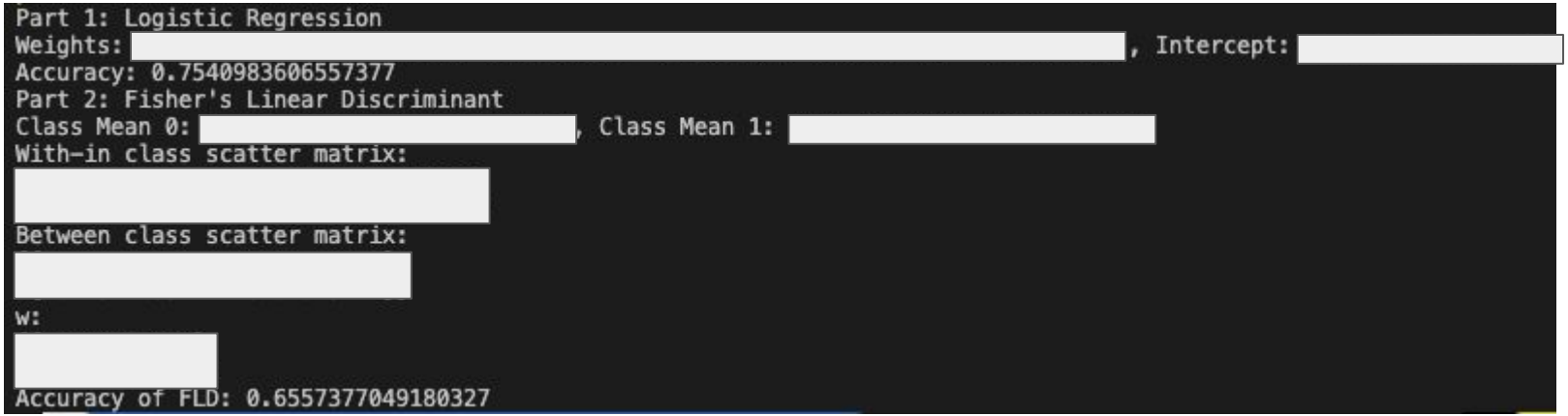
Fisher’s Linear Discriminant

- Criteria:
  - (10%) Plot the projection line.
    - Plot the projection line trained on the training set and show the slope and intercept on the title.
    - Obtain the prediction of the testing set, plot and colorize them based on the prediction.
    - Project all testing data points on your projection line.



Code Output

- Do not modify the main function architecture.
- Your code output will look like this:



Report

- Please follow the same report template format just like HW1.
- Link

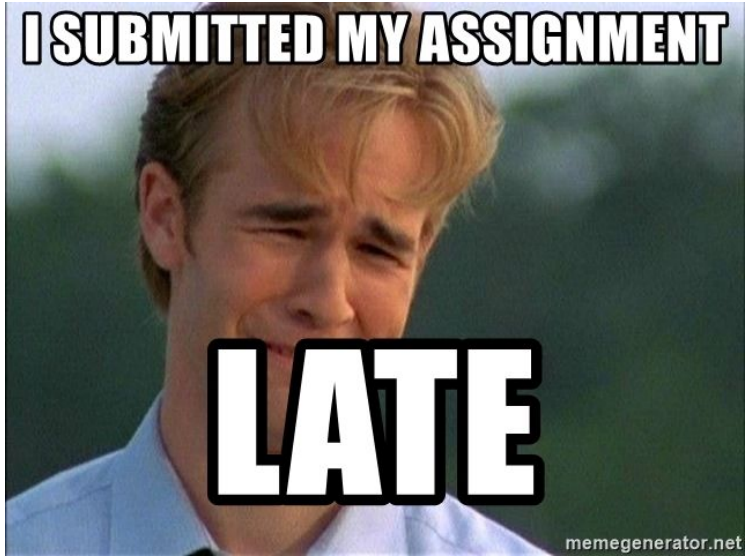
Submission

- Compress your code and report into a **.zip file** and submit it on E3.
- <STUDENT ID>_HW2.zip
  - <STUDENT ID>_HW2.py
  - <STUDENT ID>_HW2.pdf (do not submit .doc, .docx or others format)

```
 zip 0716040.zip 0716040_HW2.py 0716040_HW2.pdf
adding: 0716040_HW2.py (stored 0%)
adding: 0716040_HW2.pdf (deflated 3%)
```

Late policy

- We will deduct a late penalty of 20 points per additional late day.
- For example, If you get 90 points but delay for two days, your will get only 50 points!



Have Fun!