* Part I. Implementation
  + Part1

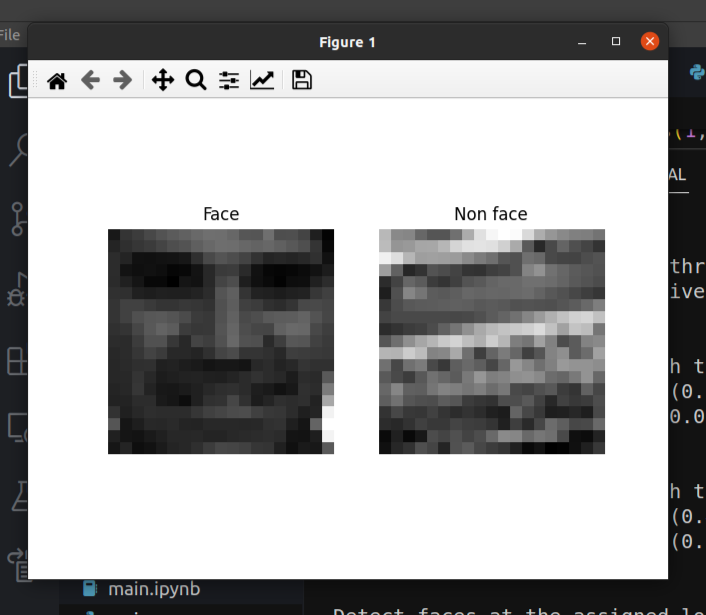
1. Get the path of the directory.
2. Go through every file in the directory.
3. Read the image in grayscale.
4. Collect the images and store into the dataset list.
5. Return the dataset.
   * Part2
6. Set zero as threshold of the feature value, if it is negative, we consider it is the feature of face and therefore equals to one.
7. Calculate the amount of data that can be classified correctly by comparing the feature value with the corresponding label.
8. Calculate the score of error while taking the weight into consideration.
9. Select the best classifier which has the smallest error.
   * Part4
10. Read the text file in the folder.
11. Extract the name of the image file, coordinate, width, and length.
12. Read the image, resize to 19x19, and then convert into gray scale.
13. Use clf.classify() to detect whether the area is face.
14. Plot the bounding box on the face in color according to the result of step 4.
15. Show the image on screen.



* + Part6

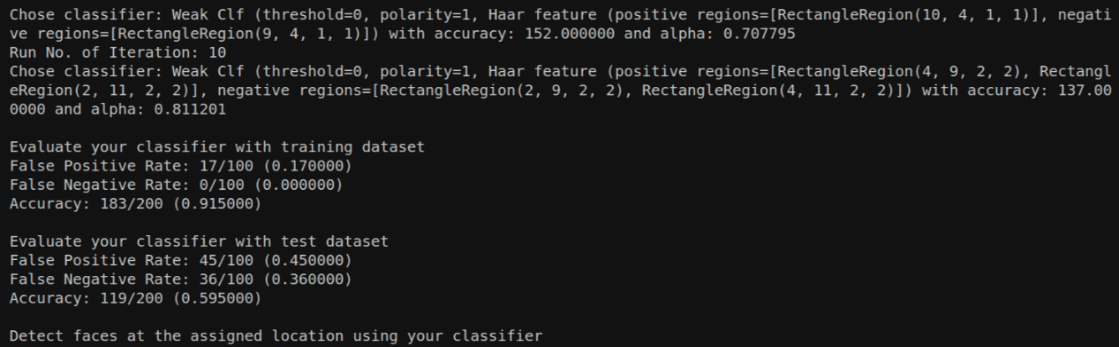
1. Create two weak classifiers for each feature.
2. Calculate the amount of data that can be classified correctly by comparing the return of the classifier with the corresponding label.
3. Calculate the score of error while taking the weight into consideration.
4. Select the best classifier which has the smallest error.

* Part II. Results & Analysis
  + Part1

1. If Part1 is done correctly, we obtain the following Picture-1.

Picture-1

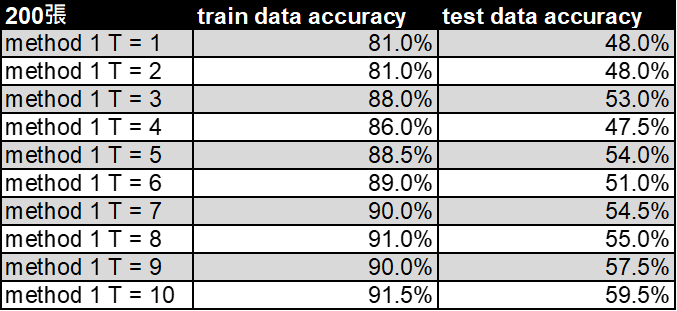
* + Part2

1. If Part2 is done correctly, we get the following Picture-2

Picture-2

* + Part3

1. Testing T from 1 to 10, we obtain the following data.
2. The accuracies of both data are growing slightly in each generation and the accuracy of training data is higher about thirty percent than that of test data. In the first few generations, the accuracy of test data is about fifty percent, which means it can barely classify the data, and it reaches sixty percent in the end.
3. For the false positive rate, the rates of two data are decreasing slightly in each generation. And for the false negative rate, the rate of training data decrease to about zero after generation three. However, for the false negative rate of test data, it fluctuated and has a higher false rate compare to that of training data.
4. From the data, we can anticipate that the model trained is not good at distinguish the none face data.



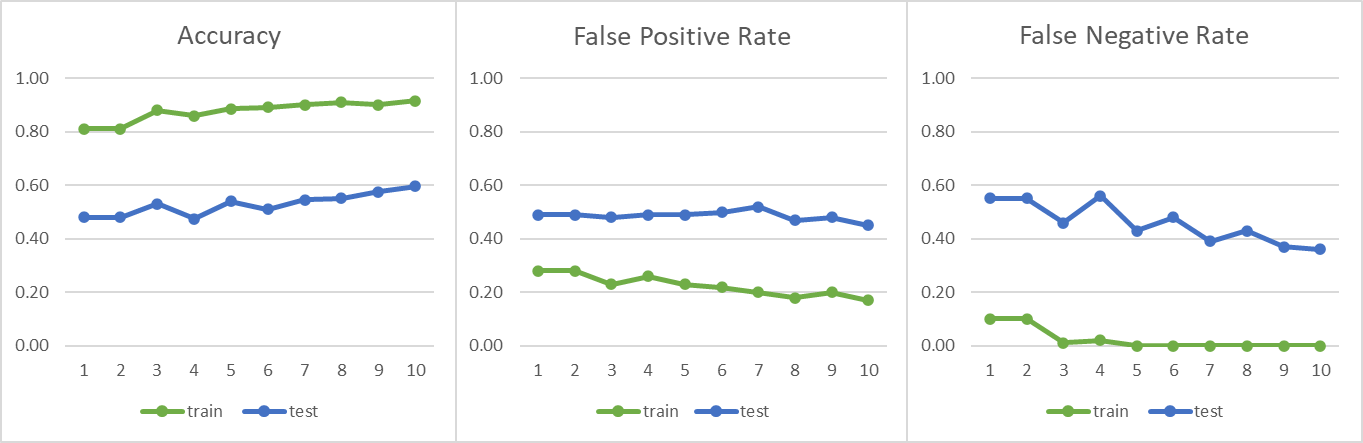
Chart-1

Chart-2

* + Part4

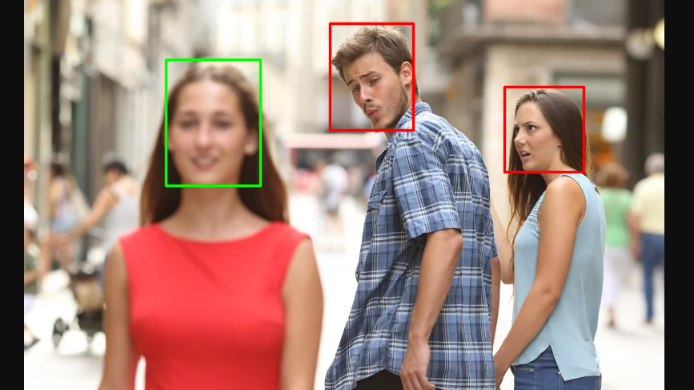
1. 一張含有 人員, 戶外, 擺姿勢 的圖片

   自動產生的描述If Part4 is done correctly, we will obtain the following Picture-3,4

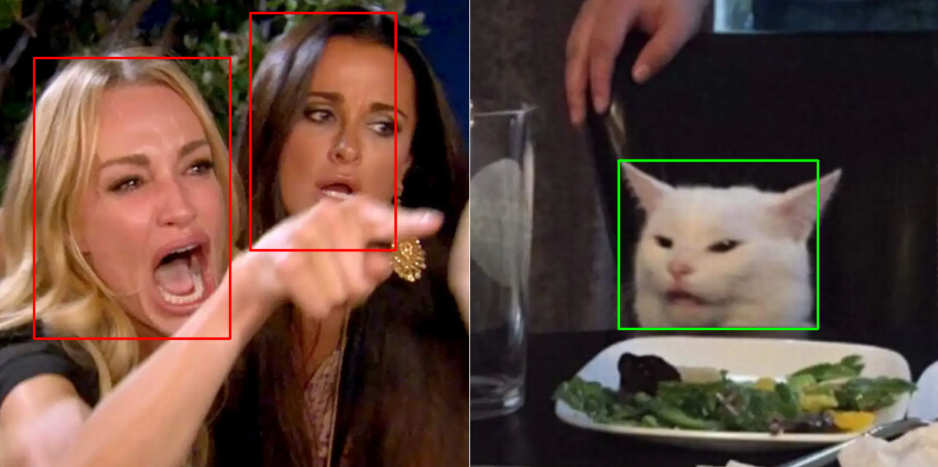
Picture-3 Picture-4

* + Part5

1. If Part5 is done correctly, we will obtain the following Picture-5,6,7
2. From Picture-6, we found that it cannot distinguish side face, any little angle could cause the error. But if the face is in front view, it can be detected, even if it is a little obscure.



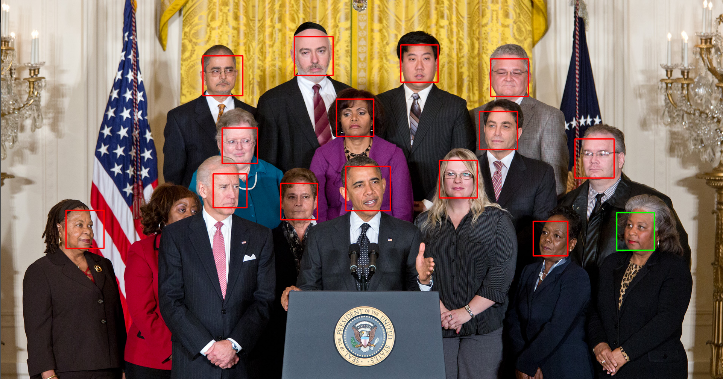
Picture-5 Picture-6



Picture-7

* + Part6

1. Test T from 1 to 10, using the custom select method, we obtain the following Picture-8,9 and Chart-3,4.
2. The accuracy of this method is higher than method about ten percent, however the performance on test picture is worse than method 1 and has a extreme low accuracy in Picture-8.
3. The false negative rate of test dataset of both method is fluctuated and is much higher than that of test data.
4. I think this problem is raised from how we select the best classifier in each generation. To optimize the method might be able to reduce the false rate.



Picture-8 Picture-9



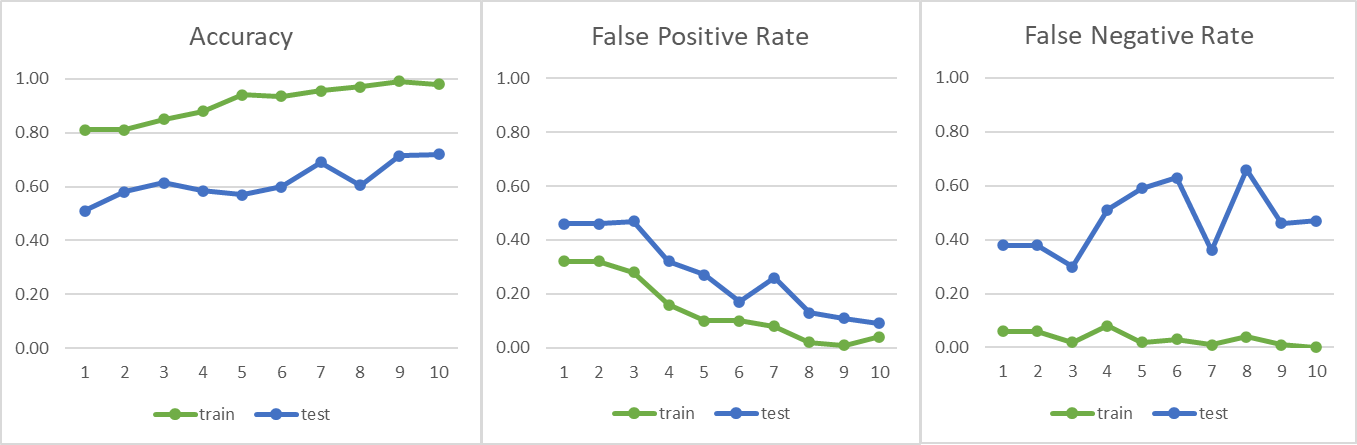
Chart-3

Chart-4

* Part III. Answer the questions (12%):

1. Please describe a problem you encountered and how you solved it.
2. While calculating the score in part2, I did not realize that the weight is necessary to be taken into consideration. I struggle with the problem that the selectBest() return the same classifier in each generation which make the training process not taking effort for a long time.
3. Until I decide to study the entire process of adaboost again, I finally realize that the adaboost update the weight depend on the best classifier, and the weight will be used in calculating the error.

2. What are the limitations of the Viola-Jones’ algorithm?

1. The Haar feature is based on changes in light, so detection accuracy can be affected by changes in light. The same object may have different detection results in different lighting environments.
2. When a target is partially blocked or has a small area, it may not perform normally.
3. The system is designed primarily for frontal faces, so the accuracy for data from other angles may be lower.

3. Based on Viola-Jones’ algorithm, how to improve the accuracy except

changing the training dataset and parameter T?

1. Design a better method to choose classifier in each round, namely optimize part2.

4. Other than Viola-Jones’ algorithm, please propose another possible face

detection method (no matter how good or bad, please come up with an idea).

Please discuss the pros and cons of the idea you proposed, compared to the

Adaboost algorithm.

1. Yolo is an algorithm that uses convolutional neural networks for object detection, which has the advantages of high speed and high accuracy, but requires a large amount of training data.