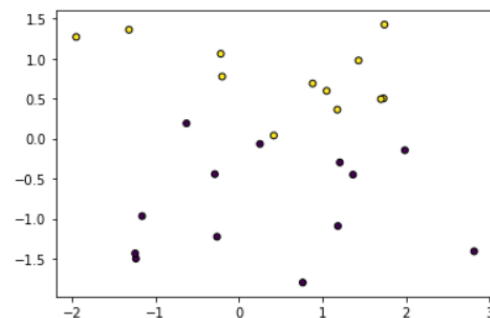


Description

- You are required to implement:
 - Batch Perceptron Algorithm
 - Online Training AlgorithmFor **binary** classification.
- Initialize your model with **random weights** between $(-1, 1)$.
- Try both algorithms on problems 1, 4 in sheet 3. **Include bias** in your model.
- Compare the results between both algorithms. Your comparison must include:
 - Number of epochs needed to achieve almost zero error.
 - Number of times each algorithm updated model weights.
 - Plot of deltas (changes in weights).
- Repeat all the above using the data generated by the following code:
This time train the model **without** adding **bias**.

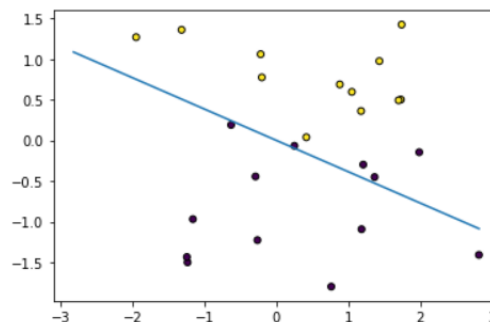
```
x, y = make_classification(25, n_features=2, n_redundant = 0,  
                           n_informative = 1, n_clusters_per_class = 1)  
mask_for_y = y == 0  
y[mask_for_y] = -1  
plt.scatter(x[:, 0], x[:, 1], marker='o', c=y, s=25, edgecolor='k')
```

It should plot the data generated like the following figure.



Use 75% as training data and 25% as testing data. And calculate model accuracy.

- Visualize the model (a line) on the plot. Note that: you did not add bias in the model to be able to visualize in 2D. Example of visualization is:



- Write a report that describes both algorithms (how they work) and each method you have implemented (documentation to your code). Include the comparisons required in this report.
- Use your own words in the report. Never copy and paste code. Plagiarism will cause zero grade.

Submission

- Perform this task in groups of two.
- **Deadline:** Wednesday 12nd April.
- Upload code and report to your **Github** profile and send link to peter315@eng.cu.edu.eg.
- It is preferred (not a must) to use one **Jupyter Notebook** to implement your code therefore, you will be able to describe (in your own words) each method you have implemented in separate cell after implementation. The report and comparisons can be written separately as **markdown readme file**.

Grading Criteria

Each method is described clearly	5
Both algorithms are described (how do they work)	10
Comparison (3 parts) are fulfilled	15 (5 marks per part)
Correct implementation of algorithms	20
All the above repeated with the randomly generated data and models visualization.	50