

Due: Sunday April 13<sup>th</sup>, 2025 at 11:59 PM

Upload the lab to D2L Labs, Lab 9 dropbox

## The Perceptron:

### Objective:

Implement a single-layer perceptron from scratch in Python and use it to classify linearly separable data.

### Dataset:

Create a synthetic dataset using `sklearn.datasets.make_classification` with 2 features, 1 target, and 2 classes. Ensure the data is linearly separable.

### Tasks:

Implement the Perceptron. Write a Python class `Perceptron` that initializes with learning rate (`lr`) and number of epochs (`n_epochs`). Include methods for fitting (`fit`) to the data and predicting (`predict`) new data points. The `fit` method should update weights based on the perceptron learning rule.

Train the Perceptron. Use the synthetic dataset to train your perceptron. Initialize your perceptron with a learning rate of 0.01 and epochs of 100.

Evaluation. Plot the decision boundary of your perceptron on the same plot as your training data. Report the accuracy of your model on the training data.

### Questions

1. Describe how the perceptron learning rule updates the weights. Why might this rule fail for non-linearly separable data?
2. Experiment with different learning rates and epochs. How do these parameters affect the model's performance and convergence?

You can use the Perceptron code that comes with the book:

<https://homepages.ecs.vuw.ac.nz/~marslast/Code/Ch3/pcn.py>