

## Perceptron learning rule

Over the course of two years, data was collected from one viewer of Netflix regarding the movies that they watched and whether they liked it or not. A movie may have any of 10 features. For example, movie 1 is described by vector

$$\mathbf{x}_1 = [1 \ 0 \ 0 \ 1 \ 0 \ 0 \ 1 \ 1 \ 0 \ 0]$$

Our viewer liked this movie, so we label it as:  $y_1 = 1$

In the file input.dat you have the features of 200 movies that our viewer watched. In the file output.dat you have the information on whether the viewer liked this movie (+1) or did not like it (-1).

Using the perceptron learning rule, find the weight vector  $\mathbf{w}$  that allows you to predict whether the viewer will like a movie or not.

- 1) Starting from a 10x1 vector  $\mathbf{w} = [0 \ 0 \ 0 \ 0 \ \dots \ 0]$ , use the following to predict viewer's choice on each movie:

$$\hat{y}_i = \text{sign}[\mathbf{w}^T \mathbf{x}_i]$$

Note: the prediction of the viewer's choice can also be equal to 0, meaning that it was not determinable. In case of prediction of 0, count that as an error.

- 2) Count the number of wrong predictions for all the movies. For each movie that your prediction is wrong, use the following learning rule to update your weight vector:

$$\mathbf{w} \leftarrow \mathbf{w} + y_i \mathbf{x}_i$$

- 3) Repeat steps 2 & 3 for 1000 times, and plot the number of wrong predictions per each step.