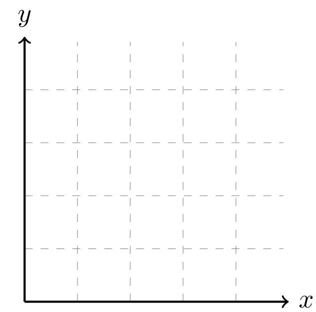
Discussion 6D

1 Perceptron

You want to predict if movies will be profitable based on their screenplays. You hire two critics A and B to read a script you have and rate it on a scale of 1 to 4. The critics are not perfect; here are five data points including the critics' scores and the performance of the movie:

#	Movie Name	A	B	Profit?
1	Pellet Power	1	1	-
2	Ghosts!	3	2	+
3	Pac is Bac	2	4	+
4	Not a Pizza	3	4	+
5	Endless Maze	2	3	-



(a) Plot the data above and determine if the points are linearly separable.

(b) Now you decide to use a perceptron to classify your data. Suppose you directly use the scores given above as features, together with a bias feature. That is $f_0 = 1$, $f_1 =$ score given by A and $f_2 =$ score given by B.

Run one pass through the data with the perceptron algorithm, filling out the table below. Go through the data points in order, e.g. using data point #1 at step 1.

step	Weights	Score	Correct?
1	[-1, 0, 0]	$-1 \cdot 1 + 0 \cdot 1 + 0 \cdot 1 = -1$	yes
2			
3			
4			
5			

Final weights:

(c) Have weights been learned that separate the data?

(d) More generally, irrespective of the training data, you want to know if your features are powerful enough to allow you to handle a range of scenarios. Circle the scenarios for which a perceptron using the features above can indeed perfectly classify movies which are profitable according to the given rules:

(a) Your reviewers are awesome: if the total of their scores is more than 5, then the movie will definitely be profitable, and otherwise it won't be.

(b) Your reviewers are art critics. Your movie will be profitable if and only if each reviewer gives either a score of 2 or a score of 3.

(c) Your reviewers have weird but different tastes. Your movie will be profitable if and only if both reviewers agree.

2 Optimization

We would like to classify some data. We have N samples, where each sample consists of a feature vector $\mathbf{x} = [x_1, \cdots, x_k]^T$ and a label $y \in \{0, 1\}$.

Logistic regression produces predictions as follows:

$$P(Y = 1 \mid X) = h(\mathbf{x}) = s\left(\sum_i w_i x_i\right) = \frac{1}{1 + \exp(-(\sum_i w_i x_i))}$$

$$s(\gamma) = \frac{1}{1 + \exp(-\gamma)}$$

where $s(\gamma)$ is the logistic function, $\exp x = e^x$, and $\mathbf{w} = [w_1, \cdots, w_k]^T$ are the learned weights.

Let's find the weights w_j for logistic regression using stochastic gradient descent. We would like to minimize the following loss function (called the cross-entropy loss) for each sample:

$$L = -[y \ln h(\mathbf{x}) + (1 - y) \ln(1 - h(\mathbf{x}))]$$

(a) Show that $s'(\gamma) = s(\gamma)(1 - s(\gamma))$

(b) Find $\frac{dL}{dw_j}$. Use the fact from the previous part.

(c) Now, find a simple expression for $\nabla_{\mathbf{w}} L = [\frac{dL}{dw_1}, \frac{dL}{dw_2}, ..., \frac{dL}{dw_k}]^T$

(d) Write the stochastic gradient descent update for $\mathbf{w}$. Our step size is η.