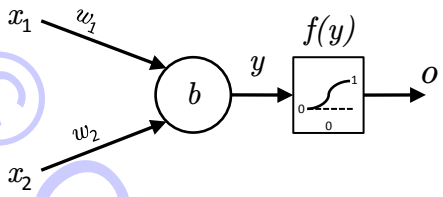
1. An interesting modification of the basic, single-layer neural classifier considered in the lectures is a *Logistic Regression* machine. This accepts inputs which are members of two possible classes, C_1 and C_2, and returns an output which is the *probability* of a particular input vector being a member of (say) class C_1.

   The architecture of a *Logistic Regression* engine is basically identical to a single-layer network (like a Perceptron or Adaline), but it uses a sigmoid nonlinearity instead of the hard nonlinearity normally associated with these classifiers. Here, for example, is a 2-d logistic regressor.



   One interesting feature of the logistic regressor is that we can *prove* that the output $o = f(y)$ is the probability that the input $\vec{x}$ is an element of class C_1, if the classifier is *parametric*, i.e., if we know, *a priori*, that the inputs are drawn from one of a small number of common probability distributions.

   The most common distributions assumed for a parametric classifier are *multivariate normal*, i.e., the probability of a particular vector $\vec{x}$ appearing at the classifier inputs, given that the vector is of class C_i is

$$p(\vec{x}|C_i) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{1}{2\sigma^2}||\vec{x} - \vec{\mu}_i||^2\right)$$

   where $\vec{x} = [x_1, x_2]^T$ is the input vector, $\vec{\mu_i} = [\mu_{i_{x_1}}, \mu_{i_{x_2}}]^T$ the mean vector of the class i distribution, and σ^2 the common variance of the two distributions. Note that the variance has to be the same for the two distributions, but can have a more complex form, i.e., a 2×2 covariance matrix. We will work with the simpler form in this question.

   Given that the probabilities of the inputs are multivariate normal as described above, show that the probability that an input vector is of class C_1 for the 2-d classifier shown above can be described by

$$p(C_1|\vec{x}) = \frac{1}{1 + \exp(-y)}$$

   where $y = w_1x_1 + w_2x_2 - b$ is the activation level of the neuron. Note, you may assume that the *class priors* are equal, i.e., $p(C_1) = p(C_2)$.

2. In the notes it was claimed that the partial derivative of the Log-Loss with respect to w_{ij}, a weight incident on the *hidden* layer of a network, is given by:

$$\frac{\partial L_{s_p}}{\partial w_{ij}} = -\delta_{pj} o_{pi}$$

where

$$\delta_{pj} = f'(y_{pj}) \sum_{k=1}^{K} w_{jk} \delta_{pk}.$$

This is just a claim that the rule for back-propagation of δ's holds for multinomial logistic regression networks as well as for the usual backpropagation (squared-difference error) classifier networks. Prove this to be so.