Example 1 Notes:

- The decision boundary is linear so we will use logistic regression.
- Denote the observation for number i by

$$\underline{x}^{(i)} = \begin{bmatrix} x_1^{(i)} \\ x_2^{(i)} \end{bmatrix} \quad \text{where } x_1^{(i)}, x_2^{(i)} \in [-1, 1] \quad \text{and}$$

$$y^{(i)} \in \{0, 1\}$$

- The logistic regression model is:

$$y^{(i)} \sim \text{Bernoulli}\left(f(x^{(i)})\right)$$

this means that $y^{(i)}$ is either 0 or 1 and $P(y^{(i)}=1) = f(x^{(i)})$

$$f(x^{(i)}) = \frac{e^{b + w_1 x_1^{(i)} + w_2 x_2^{(i)}}}{1 + e^{b + w_1 x_1^{(i)} + w_2 x_2^{(i)}}}$$

$$= \frac{e^{b + \underline{w}' \underline{x}^{(i)}}}{1 + e^{b + \underline{w}' x^{(i)}}} \quad \text{where } \underline{w} = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}$$

$$= \sigma(b + w' \underline{x}^{(i)}) \quad \text{where } \sigma(z) = \frac{\exp(z)}{1 + \exp(z)}$$

The decision boundary is the set of points $\begin{bmatrix} x_1^{(i)} \\ x_2^{(i)} \end{bmatrix}$ where there is equal probability of the two classes: $P(Y^{(i)}=1) = 0.5$

$$0.5 = f(x^{(i)}) = \frac{e^{b+w'x^{(i)}}}{1+e^{b+w'x^{(i)}}}$$

$$\Rightarrow 0.5 + 0.5e^{b+w'x^{(i)}} = e^{b+w'x^{(i)}}$$

$$\Rightarrow 0.5 = 0.5e^{b+w'x^{(i)}}$$

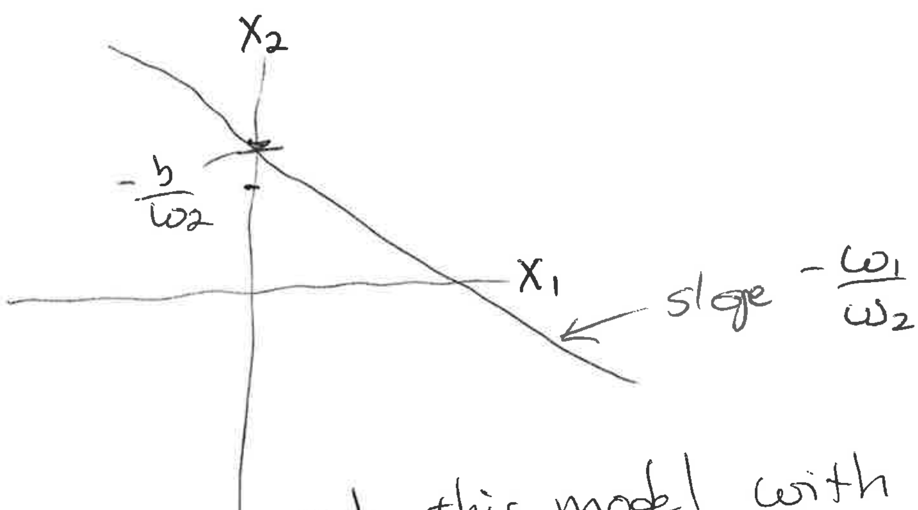
$$\Rightarrow 1 = e^{b+w'x^{(i)}}$$

$$\Rightarrow 0 = b + w'x^{(i)}$$

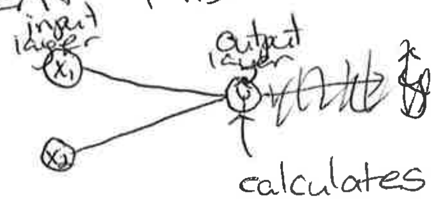
$$\Rightarrow 0 = b + w_1 x_1^{(i)} + w_2 x_2^{(i)}$$

$$\Rightarrow x_2^{(i)} = \frac{-b}{w_2} - \frac{w_1}{w_2}$$

↑ a line in the (X_1, X_2) plane



We can represent this model with a graph as follows



Example 2 Notes:

We want an elliptical decision boundary.

The equation of an ellipse is

$$b + w_1 x_1^{(i)} + w_2 x_2^{(i)} + w_3 (x_1^{(i)})^2 + w_4 (x_2^{(i)})^2 = 0$$

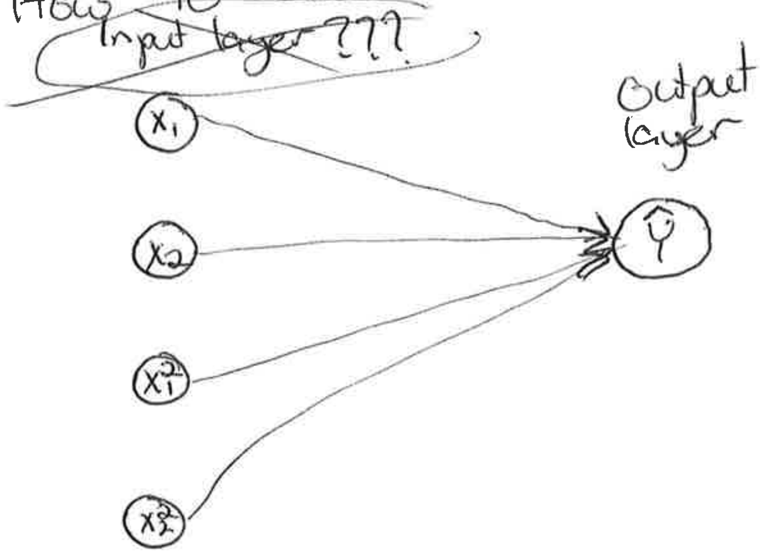
We can use the model

$$y^{(i)} \sim \text{Bernoulli}\left(f(x^{(i)})\right)$$

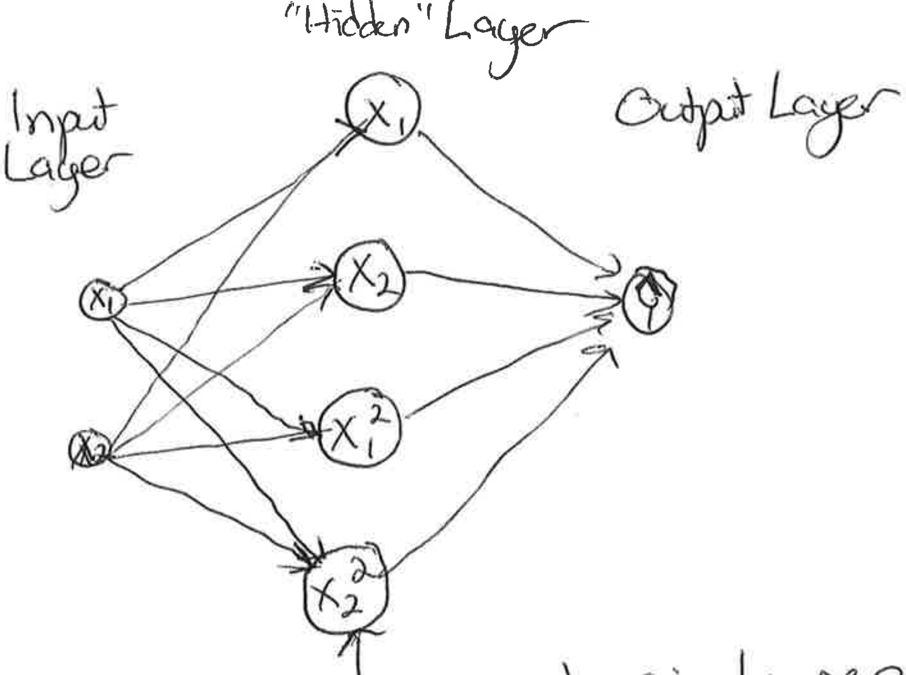
$$f(x^{(i)}) = \frac{e^{b + w_1 x_1^{(i)} + w_2 x_2^{(i)} + w_3 (x_1^{(i)})^2 + w_4 (x_2^{(i)})^2}}{1 + e^{b + w_1 x_1^{(i)} + w_2 x_2^{(i)} + w_3 (x_1^{(i)})^2 + w_4 (x_2^{(i)})^2}}$$

The same algebra as before shows $f(\underset{\sim}{x}^{(i)}) = 0.5$ when the equation for the ellipse above is satisfied.

How to draw / think about this model?



This is not the full process. We really started with x_1, x_2.



each circle represents an intermediate quantity calculated by a "unit" or "neuron", and used as an input to the next layer.

~~Notation:~~

Take observed features as inputs $\longrightarrow$ Calculate some <u>nonlinear</u> functions of them as intermediate quantities $\longrightarrow$ Calculate an estimate of the output based on intermediate quantities

Notation: $a_j^{(i)[l]}$ is the <u>activation</u> (output) from unit j in layer l for observation i.

The input layer is "layer 0".

$$\left.\begin{array}{l} \cdot\ a_1^{(i)[0]} = x_1^{(i)} \\ \cdot\ a_2^{(i)[0]} = x_2^{(i)} \end{array}\right\}\ \text{input layer, layer 0}$$

$$\left.\begin{array}{l} \cdot\ a_1^{(i)[1]} = x_1^{(i)} \quad a_2^{(i)[1]} = x_2^{(i)}, \\ \quad a_3^{(i)[1]} = (x_1^{(i)})^2, \quad a_4^{(i)[1]} = (x_2^{(i)})^2 \end{array}\right\}\ \text{hidden layer, layer 1}$$

$\cdot\ \cancel{a_1^{(i)[2]} = \ldots}$

$$\bullet\ a_1^{(i)[2]} = \frac{e^{b^{[2]} + w_{11}^{[2]} x_1^{(i)} + w_{21}^{[2]} x_2^{(i)} + w_{31}^{[2]} (x_1^{(i)})^2 + w_{41}^{[2]} (x_2^{(i)})^2}}{1 + e^{b^{[2]} + w_{11}^{[2]} x_1^{(i)} + \cdots + w_{14}^{[2]} (x_2^{(i)})^2}}$$

$$= \frac{e^{b^{[2]} + w_{11}^{[2]} a_1^{(i)[1]} + w_{21}^{[2]} a_2^{(i)[1]} + w_{31}^{[2]} a_3^{(i)[1]} + w_{41}^{[2]} a_4^{(i)[1]}}}{1 + e^{b^{[2]} + \cdots + w_{14}^{[2]} a_4^{(i)[1]}}}$$

$$= \frac{e^{b_1^{[2]} + \underset{\sim}{w}_1^{[2]T} \underset{\sim}{a}^{(i)[1]}}}{1 + e^{b_1^{[2]} + \underset{\sim}{w}_1^{[2]T} \underset{\sim}{a}^{(i)[1]}}}$$

where $\underset{\sim}{w}_1^{[2]} = \begin{bmatrix} w_{11}^{[2]} \\ w_{21}^{[2]} \\ w_{31}^{[2]} \\ w_{41}^{[2]} \end{bmatrix}$ and $\underset{\sim}{a}^{(i)[1]} = \begin{bmatrix} a_1^{(i)[1]} \\ a_2^{(i)[1]} \\ a_3^{(i)[1]} \\ a_4^{(i)[1]} \end{bmatrix}$

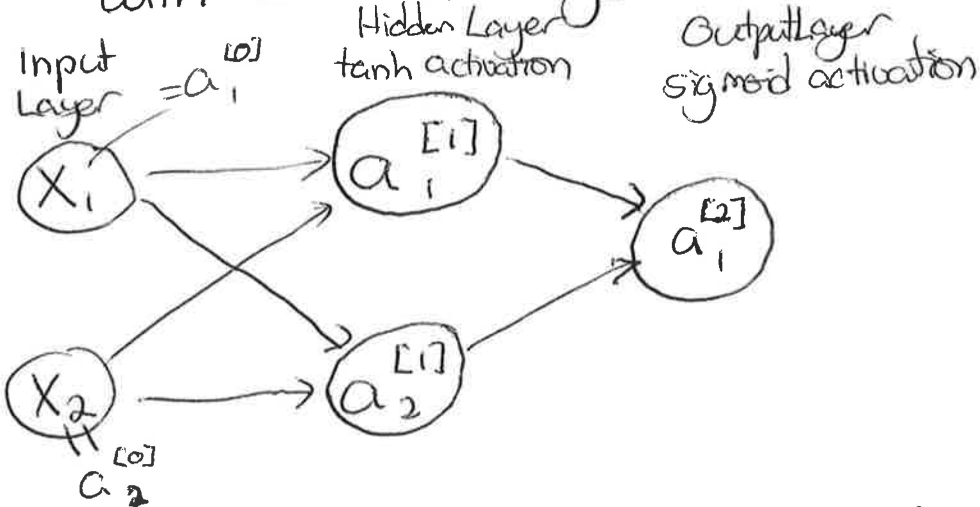
weights used to calculate activation number 1 in layer 2.

This looks terrifying, but it's exactly the same as what we first wrote, with a lot more notation.

We could also use different non-linear functions in building the hidden layers.

One common example is $\tanh(z) = \dfrac{e^{2z}-1}{e^{2z}+1}$

Ex.: Maybe we could use a network with 1 hidden layer that has 2 units in it:



$$a_1^{[1]} = \tanh\left(b_1^{[1]} + \underset{\sim}{w}_1^{[1]T} \underset{\sim}{a}^{[0]}\right)$$

$$a_2^{[1]} = \tanh\left(b_2^{[1]} + \underset{\sim}{w}_2^{[1]T} \underset{\sim}{a}^{[0]}\right)$$

$$a_1^{[2]} = \sigma\left(b_1^{[2]} + \underset{\sim}{w}_1^{[2]T} \underset{\sim}{a}^{[1]}\right)$$

We could also fit a model with more hidden layers and more units in each layer

Ex: 2 hidden layers, 20 units in each: