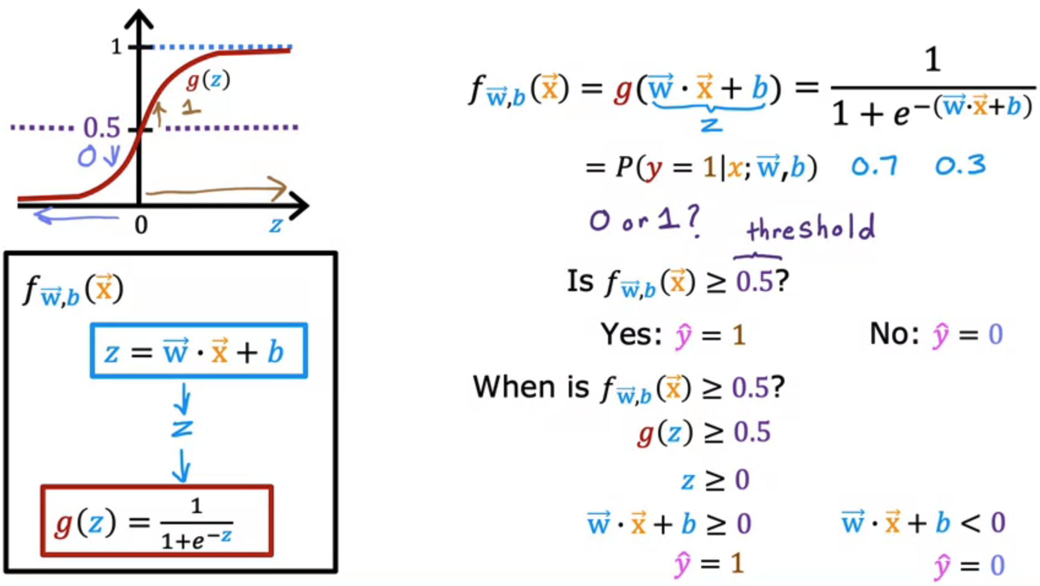
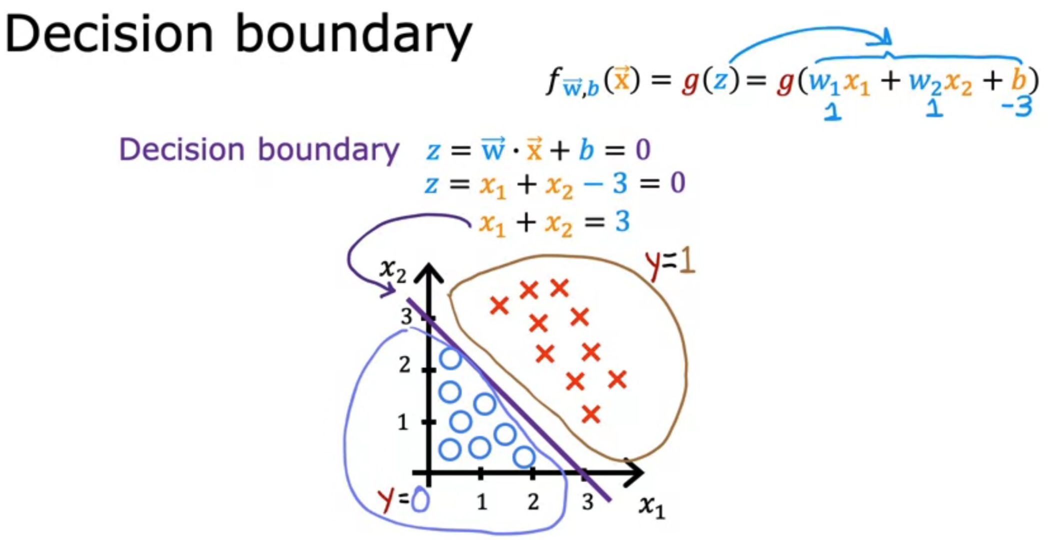
Optional Lab: Logistic Regression, Decision Boundary

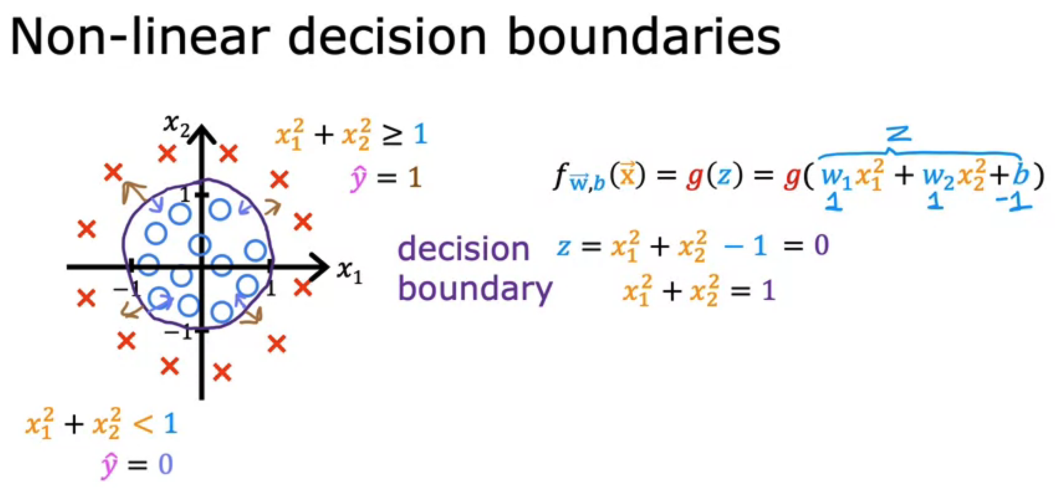
Goals

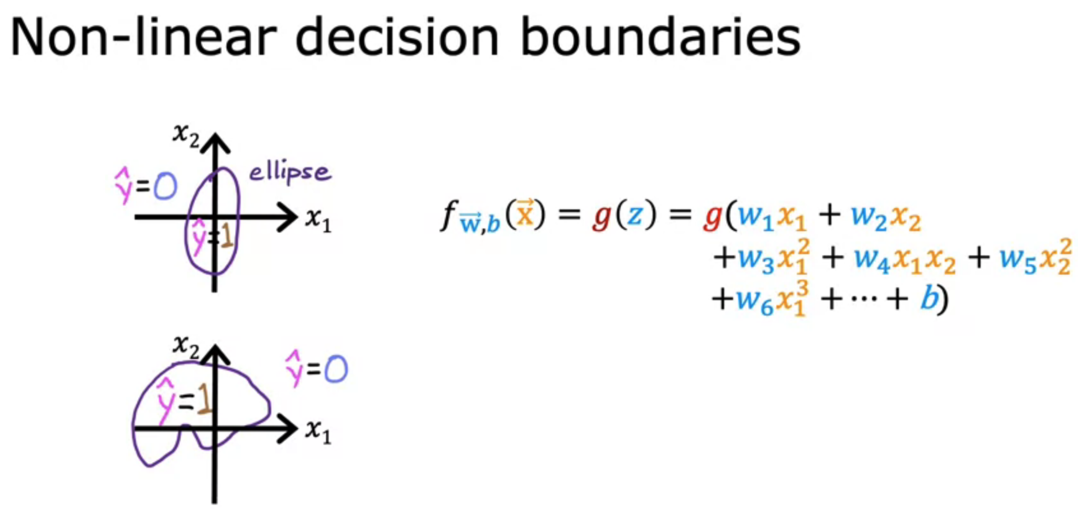
In this lab, you will:

- Plot the decision boundary for a logistic regression model. This will give you a better sense of what the model is predicting.









In [1]:

```
import numpy as np
%matplotlib widget
import matplotlib.pyplot as plt
from lab_utils_common import plot_data, sigmoid, draw_vthresh
plt.style.use('./deeplearning.mplstyle')
```

Dataset

Let's suppose you have following training dataset

- The input variable `X` is a numpy array which has 6 training examples, each with two features
- The output variable `y` is also a numpy array with 6 examples, and `y` is either `0` or `1`

In [2]:

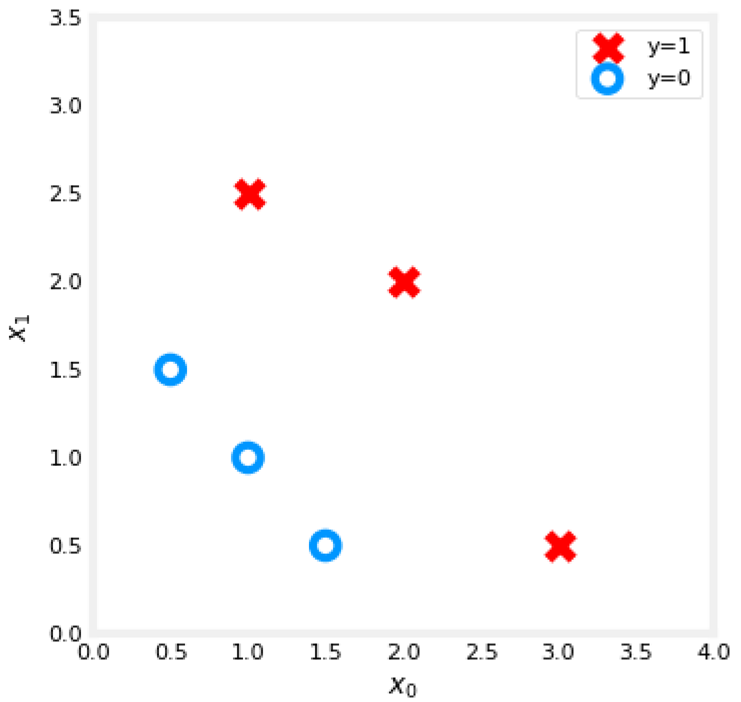
```
X = np.array([[0.5, 1.5], [1,1], [1.5, 0.5], [3, 0.5], [2, 2], [1, 2
y = np.array([0, 0, 0, 1, 1, 1]).reshape(-1,1)
```

Plot data

Let's use a helper function to plot this data. The data points with label $y = 1$ are shown as red crosses, while the data points with label $y = 0$ are shown as blue circles.

In [3]:
```
fig,ax = plt.subplots(1,1,figsize=(4,4))
plot_data(X, y, ax)

ax.axis([0, 4, 0, 3.5])
ax.set_ylabel('$x_1$')
ax.set_xlabel('$x_0$')
plt.show()
```



Logistic regression model

- Suppose you'd like to train a logistic regression model on this data which has the form

  $f(x) = g(w_0 x_0 + w_1 x_1 + b)$

  where $g(z) = \frac{1}{1+e^{-z}}$, which is the sigmoid function

- Let's say that you trained the model and get the parameters as $b = -3, w_0 = 1, w_1 = 1$. That is,

  $f(x) = g(x_0 + x_1 - 3)$

  (You'll learn how to fit these parameters to the data further in the course)

Let's try to understand what this trained model is predicting by plotting its decision boundary

Refresher on logistic regression and decision boundary

- Recall that for logistic regression, the model is represented as

$$f_{\mathbf{w},b}(\mathbf{x}^{(i)}) = g(\mathbf{w} \cdot \mathbf{x}^{(i)} + b) \tag{1}$$

  where $g(z)$ is known as the sigmoid function and it maps all input values to values between 0 and 1:

$$g(z) = \frac{1}{1+e^{-z}} \tag{2}$$

  and $\mathbf{w} \cdot \mathbf{x}$ is the vector dot product:

$$\mathbf{w} \cdot \mathbf{x} = w_0 x_0 + w_1 x_1$$

- We interpret the output of the model ($f_{\mathbf{w},b}(x)$) as the probability that $y = 1$ given $\mathbf{x}$ and parameterized by $\mathbf{w}$ and b.
  - Therefore, to get a final prediction ($y = 0$ or $y = 1$) from the logistic regression model, we can use the following heuristic -

  if $f_{\mathbf{w},b}(x) >= 0.5$, predict $y = 1$

  if $f_{\mathbf{w},b}(x) < 0.5$, predict $y = 0$

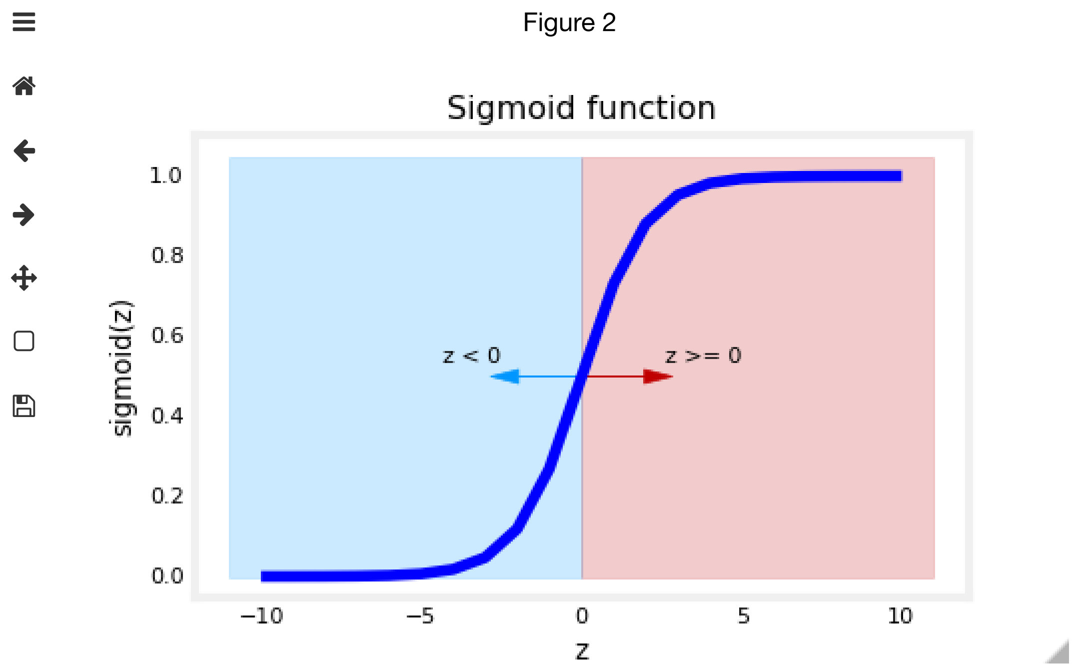
- Let's plot the sigmoid function to see where $g(z) >= 0.5$

In [4]:
```
# Plot sigmoid(z) over a range of values from -10 to 10
z = np.arange(-10,11)

fig,ax = plt.subplots(1,1,figsize=(5,3))
# Plot z vs sigmoid(z)
ax.plot(z, sigmoid(z), c="b")

ax.set_title("Sigmoid function")
ax.set_ylabel('sigmoid(z)')
ax.set_xlabel('z')
draw_vthresh(ax,0)
```

Figure 2



- As you can see, $g(z) >= 0.5$ for $z >= 0$
- For a logistic regression model, $z = \mathbf{w} \cdot \mathbf{x} + b$. Therefore,

  if $\mathbf{w} \cdot \mathbf{x} + b >= 0$, the model predicts $y = 1$

  if $\mathbf{w} \cdot \mathbf{x} + b < 0$, the model predicts $y = 0$

Plotting decision boundary

Now, let's go back to our example to understand how the logistic regression model is making predictions.

- Our logistic regression model has the form

  $f(\mathbf{x}) = g(-3 + x_0 + x_1)$

- From what you've learnt above, you can see that this model predicts $y = 1$ if $-3 + x_0 + x_1 >= 0$

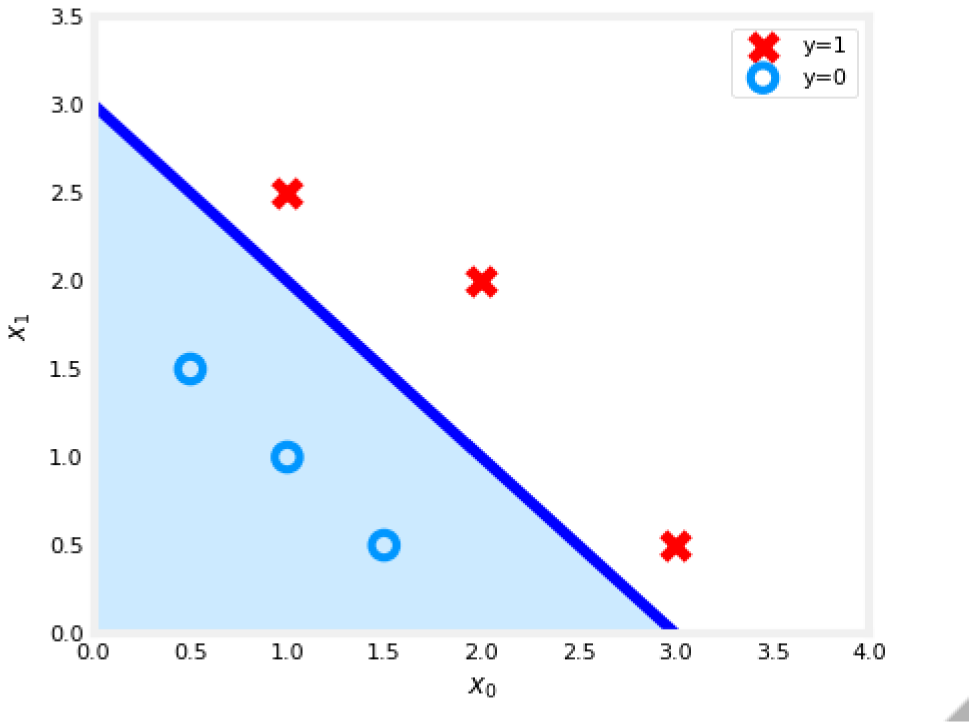
Let's see what this looks like graphically. We'll start by plotting $-3 + x_0 + x_1 = 0$, which is equivalent to $x_1 = 3 - x_0$.

In [5]:
```
# Choose values between 0 and 6
x0 = np.arange(0,6)

x1 = 3 - x0
fig,ax = plt.subplots(1,1,figsize=(5,4))
# Plot the decision boundary
ax.plot(x0,x1, c="b")
ax.axis([0, 4, 0, 3.5])

# Fill the region below the line
ax.fill_between(x0,x1, alpha=0.2)

# Plot the original data
plot_data(X,y,ax)
ax.set_ylabel(r'$x_1$')
ax.set_xlabel(r'$x_0$')
plt.show()
```



- In the plot above, the blue line represents the line $x_0 + x_1 - 3 = 0$ and it should intersect the x1 axis at 3 (if we set x_1 = 3, x_0 = 0) and the x0 axis at 3 (if we set x_1 = 0, x_0 = 3).
- The shaded region represents $-3 + x_0 + x_1 < 0$. The region above the line is $-3 + x_0 + x_1 > 0$.
- Any point in the shaded region (under the line) is classified as $y = 0$. Any point on or above the line is classified as $y = 1$. This line is known as the "decision boundary".

As we've seen in the lectures, by using higher order polynomial terms (eg: $f(x) = g(x_0^2 + x_1 - 1)$, we can come up with more complex non-linear boundaries.

Congratulations!

You have explored the decision boundary in the context of logistic regression.

In [ ]:

```

```

In [ ]:

```

```