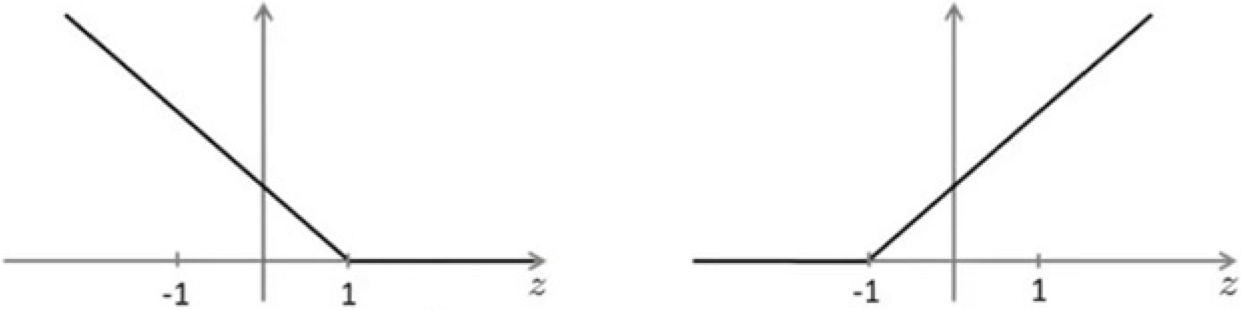
## Large Margin Intuition

Sometimes people refer to SVM as **large margin classifiers**, we'll consider what that means and what an **SVM hypothesis** looks like.

$$\min_{\theta} C \sum_{i=1}^{m} \left[ y^{(i)} cost_1(\theta^T x^{(i)}) + (1 - y^{(i)}) cost_0(\theta^T x^{(i)}) \right] + \frac{1}{2} \sum_{i=1}^{n} \theta_j^2$$

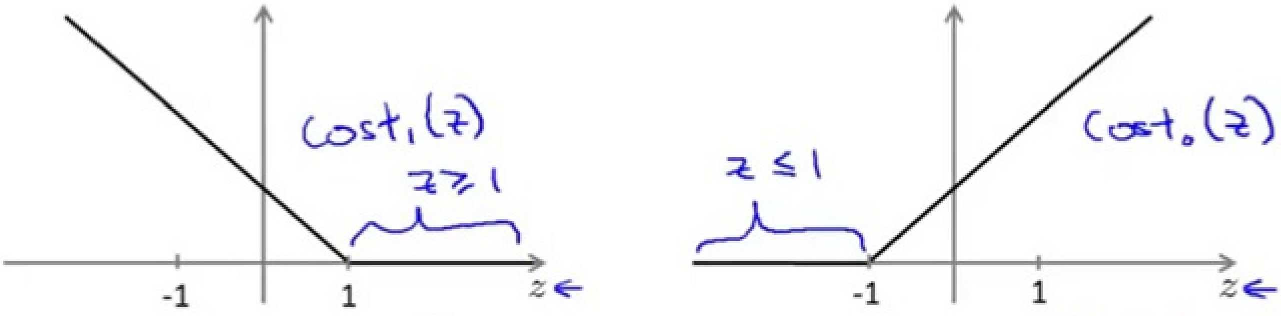


If $y = 1$, we want $\theta^T x \geq 1$ (not just $\geq 0$)
If $y = 0$, we want $\theta^T x \leq -1$ (not just $< 0$)

Here's our cost function for the support vector machine where on the left we've plotted our $\text{cost}_1(z)$ function that we used for positive examples and on the right we've plotted our $\text{cost}_0(z)$ function, where we have $z$ on the horizontal axis:

- If $y = 1$, $\text{cost}_1(z) = 0$ only when $z >= 1$
- If $y = 0$, $\text{cost}_0(z) = 1$ only when $z <= -1$

If we have a positive example, we only really need $z >= 0$, if this is the case then we predict $1$. SVM wants a bit more than that - doesn't want to *just* get it right, but have the value be quite a bit bigger than zero (throws in an extra safety margin factor).



If $\underline{y = 1}$, we want $\underline{\theta^T x \geq 1}$ (not just $\geq 0$) $\theta^T x \geq \cancel{0}\ 1$
If $\underline{y = 0}$, we want $\underline{\theta^T x \leq -1}$ (not just $< 0$) $\theta^T x \leq \cancel{0}\ -1$

Logistic regression does something similar too, let's see what happens or let's see what the consequences of this, in the context of the support vector machine.

## SVM decision boundary

Consider a case where we set $C$ to be huge (i.e. $C = 100,000$), if $C$ is very, very large, then when minimizing the optimization objective ($CA + B$), if $C$ is huge we're going to pick an $A$ value so that $A$ is equal to zero. What is the optimization problem here - how do we make A = 0?

We saw already that whenever we have a training example with a label of $y^{(i)} = 1$ if we want to make that first term ($A$) zero, what we need is to find a value of $\theta$ so that $\theta^T x^{(i)}$ is greater than or equal to $1$. And similarly, whenever we have an example, with label zero $y^{(i)} = 0$, in order to make sure that the cost is zero we need that $\theta^T x^{(i)}$ is less than or equal to $1$.

- Whenever $y^{(i)} = 1$
  - $\theta^T x^{(i)} >= 1$
- Whenever $y^{(i)} = 0$
  - $\theta^T x^{(i)} <= 1$
- So - if we think of our optimization problem a way to ensure that this first "$A$" term is equal to $0$, we re-factor our optimization problem into just minimizing the "$B$" (regularization) term, because
  - When $A = 0$ --> $A * C = 0$
- So we're minimizing $B$, under the constraints shown below

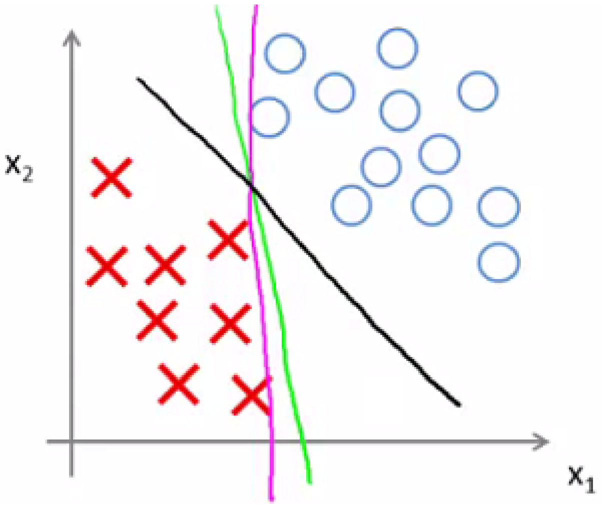
$$\min_{\theta} \cancel{C \times 0} + \frac{1}{2} \sum_{i=1}^{n} \theta_j^2$$

$$\text{s.t.} \quad \theta^T x^{(i)} \geq 1 \quad \text{if } y^{(i)} = 1$$

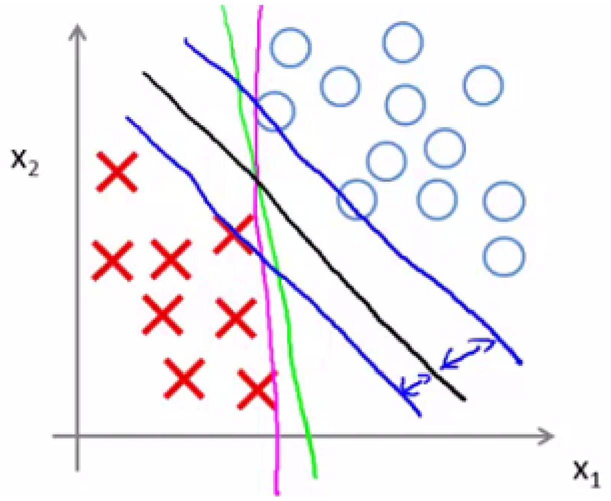
$$\theta^T x^{(i)} \leq -1 \quad \text{if } y^{(i)} = 0$$

**SVM decision boundary: Linearly separable case**

Concretely, if we look at a data set with positive and negative examples, this data is linearly separable which means that there exists a straight line (altough there is many a different straight lines) that can separate the positive and negative examples perfectly.



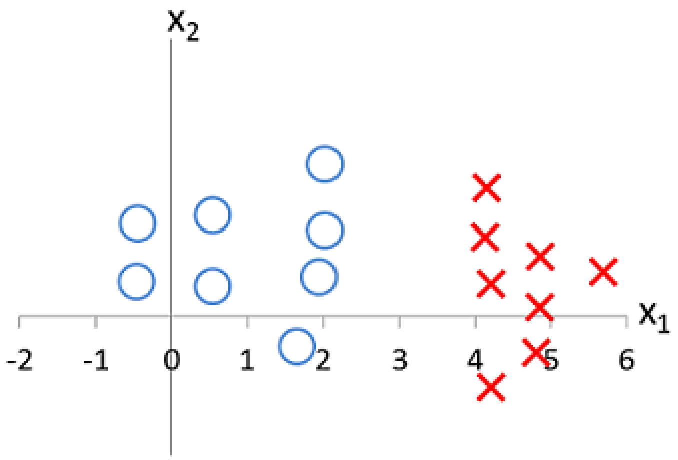
- The green and magenta lines are functional decision boundaries which could be chosen by logistic regression.
  - But they probably don't generalize too well.
- The black line, by contrast is the the chosen by the SVM because of this safety net imposed by the optimization graph.
  - More robust separator.
- Mathematically, that black line has a larger minimum distance (margin) from any of the training examples.



With the previously plot we see that the black decision boundary has some larger minimum distance from any of our training examples, whereas the magenta and the green lines they come awfully close to the training examples, and then that seems to do a less a good job separating the positive and negative classes than our black line.

Concretely by separating with the largest margin we incorporate robustness into our decision making process (the support vector machine is sometimes also called a **large margin classifier** and this is actually a consequence of the optimization problem that we saw previously).

**Video Question:** Consider the training set to the right, where "x" denotes positive examples ($y = 1$) and "o" denotes negative examples ($y = 0$). Suppose you train an SVM (which will predict $1$ when $\theta_0 + \theta_1 x_1 + \theta_2 x_2 \geq 0$). What values might the SVM give for $\theta_0$, $\theta_1$, and $\theta_2$?



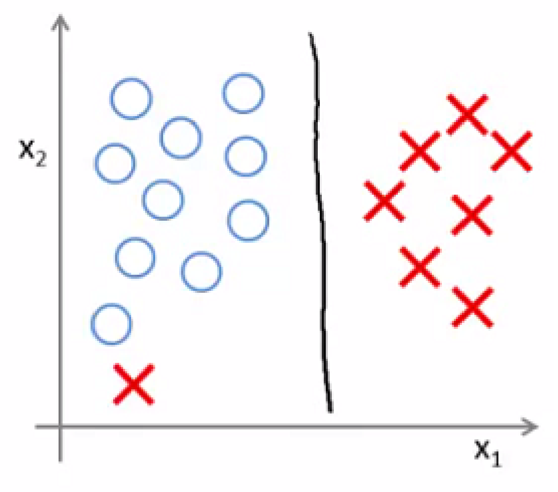
- $\theta_0 = 3, \theta_1 = 1, \theta_2 = 0$

$\theta_0 = -3, \theta_1 = 1, \theta_2 = 0$

- $\theta_0 = 3, \theta_1 = 0, \theta_2 = 1$
- $\theta_0 = -3, \theta_1 = 0, \theta_2 = 1$

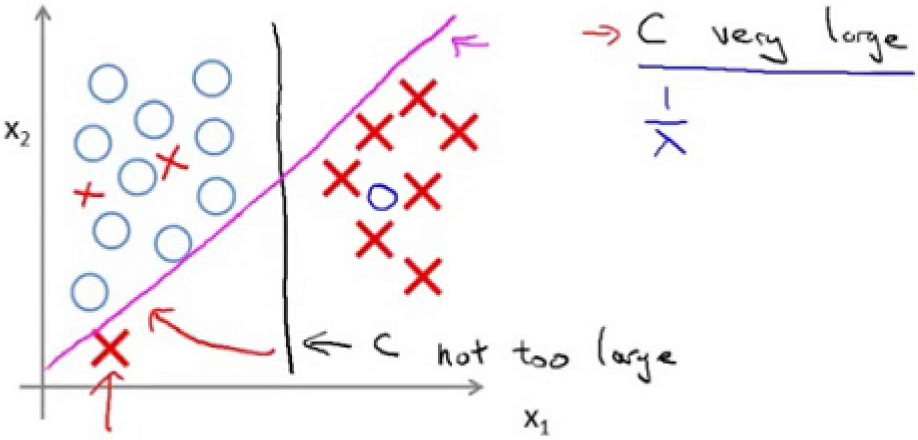
## Large margin classifier in presence of outliers

SVM is more sophisticated than the large margin might look, if we were just using large margin then SVM would be very sensitive to outliers



We would risk making a ridiculous hugely impact our classification boundary, a single example might not represent a good reason to change an algorithm.

- If $C$ is very large then we do use this quite naive maximize the margin approach



- So we'd change the black to the magenta
- But if $C$ is reasonably small, or a not too large, then we stick with the black decision boundary

If the regularization parameter $C$ were very large, then this is actually what SVM will do, it will change the decision boundary from the black to the magenta one but if $C$ were reasonably small if we were to use the $C$, not too large then we still end up with this black decision boundary.

What about non-linearly separable data? Then SVM still does the right thing if we use a normal size $C$. So the idea of SVM being a large margin classifier is only really relevant when we have no outliers and we can easily linearly separable data, means we ignore a few outliers.