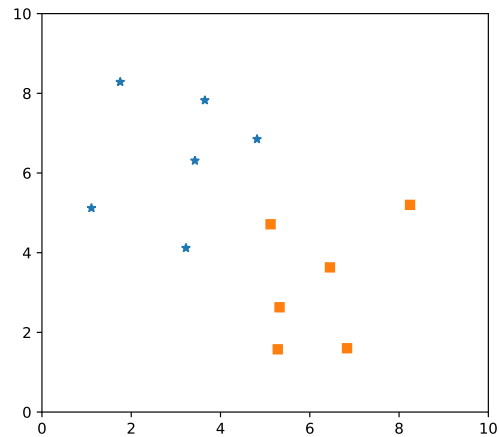


SVM Quiz

1. Support Vectors and Margins

Consider the two-class dataset below:



- Draw the decision boundary that would be found by a linear support vector machine for this dataset.
- Circle the support vectors.
- Imagine that a new square data point is added to this dataset at position (2,6), (in the middle of the blue stars). How would this affect the decision boundary if,
 - $C = 1$
 - $C = \infty$

2. Kernel Functions

Support vector machines are able to produce non-linear decision boundaries by, in a sense, transforming low-dimensional inputs into a high-dimensional space, then performing classification in that high-dimensional space. This usually works because high-dimensional data is much more likely to be linearly separable than low-dimensional data.

As an example, consider the following two-dimensional data points: $A = (1, 2)$, $B = (2, 4)$. Use the following function to map these two datapoints into the indicated six-dimensional space:

$$\Phi(\mathbf{x}) = (x_1^2, x_2^2, \sqrt{2} x_1 x_2, \sqrt{2} x_1, \sqrt{2} x_2, 1)$$

(a) $\Phi(A) =$

(b) $\Phi(B) =$

Now calculate the dot product between these two points in the space defined by Φ :

(c) $\Phi(A)^T \Phi(B) =$

The kernel function corresponding to the projection above is,

$$K(\mathbf{x}, \mathbf{y}) = (\mathbf{x}^T \mathbf{y} + 1)^2$$

(This is degree-two polynomial kernel. More generally, we could choose a polynomial kernel of arbitrary degree: $K(\mathbf{x}, \mathbf{y}) = (\mathbf{x}^T \mathbf{y} + 1)^d$.)

Use this kernel function to calculate,

(d) $K(A, B) =$

If all went well, your answers for (c) and (d) should be the same. Obviously, using the kernel function required significantly fewer calculations. The difference would be even more dramatic if A and B were three-dimensional data points, or if we were to select a higher degree polynomial.

The beauty of non-linear support vector machines is that training and classification only require the dot products between data points. The data points themselves don't show up anywhere in the calculations. This means that the work you did in (a), (b) and (c) is *never actually performed by the SVM*. It "cheats" by using the Kernel function instead. This cheating is referred to as the "kernel trick".

3. Support Vector Machines and Random Forests are both popular classifiers that are widely used in practice. Each has a different mechanism for avoiding overfitting. What are the two approaches?