**Support Vector Machines in sklearn**

In this section, you'll use support vector machines to fit a given sample dataset.

Before you do that, let's go over the tools required to build this model.

For your support vector machine model, you'll be using scikit-learn's [SVC](http://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html) class. This class provides the functions to define and fit the model to your data.

>>> **from** sklearn.svm **import** SVC

>>> model = SVC()

>>> model.fit(x\_values, y\_values)

In the example above, the model variable is a support vector machine model that has been fitted to the data x\_values and y\_values. Fitting the model means finding the best boundary that fits the training data. Let's make two predictions using the model's predict() function.

>>> print(model.predict([ [0.2, 0.8], [0.5, 0.4] ]))

[[ 0., 1.]]

The model returned an array of predictions, one prediction for each input array. The first input, [0.2, 0.8], got a prediction of 0.. The second input, [0.5, 0.4], got a prediction of 1..

### Hyperparameters

When we define the model, we can specify the hyperparameters. As we've seen in this section, the most common ones are

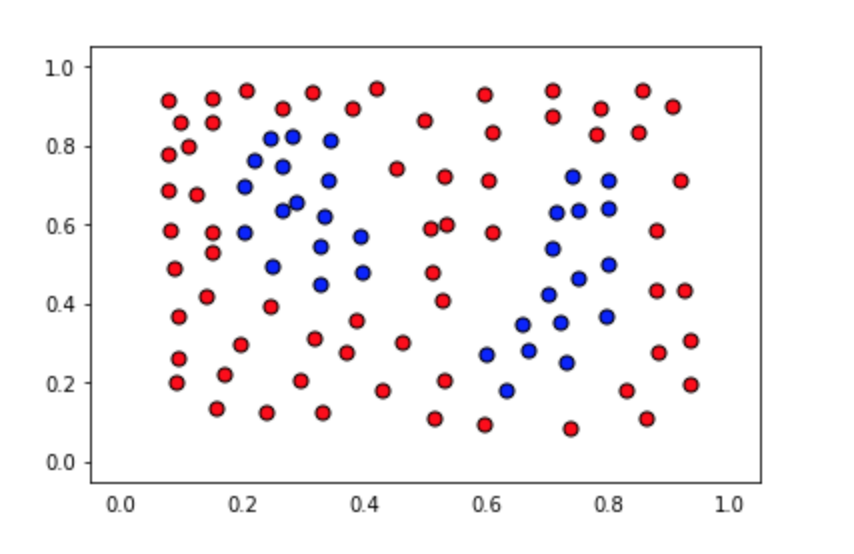
* C: The C parameter.
* kernel: The kernel. The most common ones are 'linear', 'poly', and 'rbf'.
* degree: If the kernel is polynomial, this is the maximum degree of the monomials in the kernel.
* gamma : If the kernel is rbf, this is the gamma parameter.

For example, here we define a model with a polynomial kernel of degree 4, and a C parameter of 0.1.

>>> model = SVC(kernel='poly', degree=4, C=0.1)

## Support Vector Machines Quiz

In this quiz, you'll be given with the following sample dataset, and your goal is to define a model that gives 100% accuracy on it.



The data file can be found under the "data.csv" tab in the quiz below. It includes three columns, the first 2 comprising of the coordinates of the points, and the third one of the label.

The data will be loaded for you, and split into features X and labels y.

### You'll need to complete each of the following steps:

**1. Build a support vector machine model**

* Create a support vector machine classification model using scikit-learn's [SVC](http://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html) and assign it to the variablemodel.

**2. Fit the model to the data**

* If necessary, specify some of the hyperparameters. The goal is to obtain an accuracy of 100% in the dataset. Hint: Not every kernel will work well.

**3. Predict using the model**

* Predict the labels for the training set, and assign this list to the variable y\_pred.

**4. Calculate the accuracy of the model**

* For this, use the function sklearn function [accuracy\_score](http://scikit-learn.org/stable/modules/generated/sklearn.metrics.accuracy_score.html).

When you hit **Test Run**, you'll be able to see the boundary region of your model, which will help you tune the correct parameters, in case you need them.

**Note:** This quiz requires you to find an accuracy of 100% on the training set. Of course, this screams overfitting! If you pick very large values for your parameters, you will fit the training set very well, but it may not be the best model. Try to find the smallest possible parameters that do the job, which has less chance of overfitting, although this part won't be graded.