**SVC classifier** (given from the scikit-learn library) - used for implementing Support Vector Machines (SVMs) for classification tasks. Let's break down the parameters in the call:

1. C: This parameter represents the regularization parameter of the SVM. It controls the trade-off between achieving a low training error and a low testing error. A smaller C value allows for a larger margin (soft margin) and permits more training points to be misclassified. A larger C value aims for a narrower margin (hard margin) and enforces a stricter fit to the training data. In other words, a smaller C value emphasizes generalization, while a larger C value emphasizes fitting the training data more accurately.
2. decision\_function\_shape: This parameter determines how the decision function for multiclass classification is computed. 'ovr' stands for "one-vs-rest," where a binary classification problem is created for each class against all other classes. Alternatively, you can use 'ovo' for "one-vs-one," where a binary classifier is trained for each pair of classes. 'ovr' is generally more efficient for problems with a large number of classes.
3. degree: This parameter is used in polynomial kernels. It represents the degree of the polynomial used to compute the decision function. The default is 3, meaning a cubic polynomial is used. Higher degrees can capture more complex relationships in the data, but they also increase the risk of overfitting.
4. gamma: This parameter is specific to certain kernel functions, such as the Radial Basis Function (RBF) kernel. It defines the influence of a single training example. A low gamma means each point has a far-reaching influence, resulting in a smoother decision boundary. A high gamma means each point has a narrow influence, causing the decision boundary to be more wiggly, fitting the training data closely.
5. kernel: This parameter specifies the type of kernel function to be used in the SVM. Common options include 'linear', 'poly' (polynomial), 'rbf' (Radial Basis Function), and others. Each kernel defines how the data in the feature space is transformed to a higher-dimensional space where linear separation might be easier.

These parameters allow you to customize the behavior and performance of the SVM algorithm according to your specific dataset and problem. Keep in mind that parameter tuning can significantly impact the model's performance, so it's often a good practice to perform hyperparameter optimization to find the best combination of parameter values for your particular task.