***класс*sklearn.svm.SVC (***\****,*C****= 1.0***,*ядро****​​= 'rbf'***,*степень****= 3***,*гамма****= 'масштаб'***,*coef0****= 0.0***,*сокращение****= Истина***,*вероятность* =***Ложь***,*tol****= 0.001***, размер *кэша****= 200***,*class\_weight****= Нет***,*verbose****= Ложь***,***max\_iter =****-1* ,*decision\_function\_shape****= 'ovr'***,*break\_ties****= Ложь***,*random\_state****= Нет )*[**[источник]**](https://github.com/scikit-learn/scikit-learn/blob/98ed9dc73/sklearn/svm/_classes.py#L614)

Классификация опорных векторов C.

Реализация основана на libsvm. Время подгонки масштабируется как минимум квадратично с числом образцов и может быть непрактичным за пределами десятков тысяч образцов. Для больших наборов данных рассмотрите возможность использования **[LinearSVC](https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html" \l "sklearn.svm.LinearSVC" \o "sklearn.svm.ЛинейныйSVC)**или **[SGDClassifier](https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.SGDClassifier.html" \l "sklearn.linear_model.SGDClassifier" \o "sklearn.linear_model.SGDClassifier)**вместо этого, возможно, после **[Nystroem](https://scikit-learn.org/stable/modules/generated/sklearn.kernel_approximation.Nystroem.html" \l "sklearn.kernel_approximation.Nystroem" \o "sklearn.kernel_approximation.Nystroem)**трансформатора или другого [приближения ядра](https://scikit-learn.org/stable/modules/kernel_approximation.html#kernel-approximation) .

Поддержка нескольких классов осуществляется по схеме «один к одному».

Подробную информацию о точной математической формулировке предоставляемых функций ядра и о том, как gamma, coef0и degreeвлияют друг на друга, см. в соответствующем разделе в описательной документации: [Функции ядра](https://scikit-learn.org/stable/modules/svm.html#svm-kernels) .

Чтобы узнать, как настраивать гиперпараметры SVC, см. следующий пример: [Вложенная и невложенная перекрестная проверка](https://scikit-learn.org/stable/auto_examples/model_selection/plot_nested_cross_validation_iris.html#sphx-glr-auto-examples-model-selection-plot-nested-cross-validation-iris-py)

Подробнее читайте в [Руководстве пользователя](https://scikit-learn.org/stable/modules/svm.html#svm-classification) .

**Параметры :**

**C *float, по умолчанию=1.0***

Параметр регуляризации. Сила регуляризации обратно пропорциональна C. Должна быть строго положительной. Штраф — это квадрат штрафа l2. Для наглядной визуализации эффектов масштабирования параметра регуляризации C см. [Масштабирование параметра регуляризации для SVC](https://scikit-learn.org/stable/auto_examples/svm/plot_svm_scale_c.html#sphx-glr-auto-examples-svm-plot-svm-scale-c-py) .

**ядро *{'linear', 'poly', 'rbf', 'sigmoid', 'precomputed'} или вызываемое, по умолчанию = 'rbf'***

Указывает тип ядра, который будет использоваться в алгоритме. Если ничего не указано, будет использоваться 'rbf'. Если указан вызываемый объект, он используется для предварительного вычисления матрицы ядра из матриц данных; эта матрица должна быть массивом shape . Для наглядной визуализации различных типов ядра см. [Построение границ классификации с различными ядрами SVM](https://scikit-learn.org/stable/auto_examples/svm/plot_svm_kernels.html#sphx-glr-auto-examples-svm-plot-svm-kernels-py) .(n\_samples, n\_samples)

**степень *int, по умолчанию=3***

Степень функции полиномиального ядра ('poly'). Должна быть неотрицательной. Игнорируется всеми другими ядрами.

**гамма *{'scale', 'auto'} или float, по умолчанию='scale'***

Коэффициент ядра для «rbf», «poly» и «sigmoid».

* если gamma='scale'передано (default), то в качестве значения гаммы используется 1 / (n\_features \* X.var()),
* если «авто», то используется 1 / n\_features
* если float, должно быть неотрицательным.

***Изменено в версии 0.22:*** значение по умолчанию gammaизменено с «авто» на «масштаб».

**coef0 *float, по умолчанию=0.0***

Независимый член в функции ядра. Имеет значение только в 'поли' и 'сигмоида'.

**сокращение *bool, по умолчанию = True***

Использовать ли эвристику сжатия. См. [Руководство пользователя](https://scikit-learn.org/stable/modules/svm.html#shrinking-svm) .

**вероятность *bool, по умолчанию=False***

Следует ли включить оценки вероятности. Это необходимо включить до вызова fit, замедлит этот метод, поскольку он внутренне использует 5-кратную перекрестную проверку, и predict\_probaможет быть несовместимым с predict. Подробнее читайте в [Руководстве пользователя](https://scikit-learn.org/stable/modules/svm.html#scores-probabilities) .

**tol *float, по умолчанию = 1e-3***

Допустимость критерия остановки.

**размер\_кеша *с плавающей точкой, по умолчанию=200***

Укажите размер кэша ядра (в МБ).

**class\_weight *dict или «сбалансированный», по умолчанию=Нет***

Установите параметр C класса i на class\_weight[i]\*C для SVC. Если не указано, предполагается, что все классы имеют вес один. «Сбалансированный» режим использует значения y для автоматической корректировки весов обратно пропорционально частотам классов во входных данных как .n\_samples / (n\_classes \* np.bincount(y))

**подробный *логический, по умолчанию = Ложь***

Включить подробный вывод. Обратите внимание, что этот параметр использует преимущество настройки времени выполнения для каждого процесса в libsvm, которая, если включена, может работать некорректно в многопоточном контексте.

**max\_iter *int, по умолчанию=-1***

Жесткое ограничение на итерации в решателе или -1 для отсутствия ограничений.

**decision\_function\_shape *{'ovo', 'ovr'}, по умолчанию = 'ovr'***

Возвращать ли функцию принятия решения один против одного ('ovr') с формой (n\_samples, n\_classes), как и все остальные классификаторы, или исходную функцию принятия решения один против одного ('ovo') libsvm с формой (n\_samples, n\_classes \* (n\_classes - 1) / 2). Однако следует отметить, что внутренне один против одного ('ovo') всегда используется как многоклассовая стратегия для обучения моделей; матрица ovr создается только из матрицы ovo. Для бинарной классификации этот параметр игнорируется.

***Изменено в версии 0.19:*** decision\_function\_shape по умолчанию имеет значение «ovr».

***Добавлено в версии 0.17:*** рекомендуется *decision\_function\_shape='ovr' .*

***Изменено в версии 0.17:*** Устаревшие *decision\_function\_shape='ovo' и None* .

**break\_ties *bool, по умолчанию = Ложь***

Если true, decision\_function\_shape='ovr'и число классов > 2, [predict](https://scikit-learn.org/stable/glossary.html" \l "term-predict) разорвет связи в соответствии со значениями достоверности [decision\_function](https://scikit-learn.org/stable/glossary.html" \l "term-decision_function) ; в противном случае возвращается первый класс среди связанных классов. Обратите внимание, что разрыв связей требует относительно высоких вычислительных затрат по сравнению с простым predict. См. [SVM Tie Breaking Example](https://scikit-learn.org/stable/auto_examples/svm/plot_svm_tie_breaking.html#sphx-glr-auto-examples-svm-plot-svm-tie-breaking-py) для примера его использования с decision\_function\_shape='ovr'.

***Добавлено в версии 0.22.***

**random\_state *int, экземпляр RandomState или None, по умолчанию=None***

Управляет генерацией псевдослучайных чисел для перемешивания данных для оценки вероятности. Игнорируется, если probabilityFalse. Передайте int для воспроизводимого вывода между несколькими вызовами функций. См. [Глоссарий](https://scikit-learn.org/stable/glossary.html#term-random_state) .

**Атрибуты :**

**class\_weight\_ *ndarray формы (n\_classes,)***

Множители параметра C для каждого класса. Рассчитываются на основе class\_weightпараметра.

**классы\_ *ndarray формы (n\_classes,)***

Метки классов.

[**coef\_**](https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html#sklearn.svm.SVC.coef_)***ndarray формы (n\_classes \* (n\_classes - 1) / 2, n\_features)***

Веса, назначенные признакам, когда kernel="linear".

**Dual\_coef\_ *ndarray формы (n\_classes -1, n\_SV)***

Двойственные коэффициенты опорного вектора в решающей функции (см. [Математическая формулировка](https://scikit-learn.org/stable/modules/sgd.html#sgd-mathematical-formulation) ), умноженные на их цели. Для мультикласса коэффициент для всех классификаторов 1-на-1. Расположение коэффициентов в мультиклассовом случае несколько нетривиально. Подробности см. [в разделе «Мультикласс» Руководства пользователя .](https://scikit-learn.org/stable/modules/svm.html#svm-multi-class)

**fit\_status\_ *int***

0, если установлено правильно, 1 в противном случае (выдаст предупреждение)

**intercept\_ *ndarray формы (n\_classes \* (n\_classes - 1) / 2,)***

Константы в решающей функции.

**n\_features\_in\_ *int***

Количество признаков, наблюдаемых во время [подгонки](https://scikit-learn.org/stable/glossary.html#term-fit) .

***Добавлено в версии 0.24.***

**feature\_names\_in\_ *ndarray формы ( n\_features\_in\_,)***

Имена функций, видимых во время [fit](https://scikit-learn.org/stable/glossary.html" \l "term-fit) . Определяется только тогда, когда X имеет имена функций, которые являются строками.

***Добавлено в версии 1.0.***

**n\_iter\_ *ndarray формы (n\_classes \* (n\_classes - 1) // 2,)***

Количество итераций, выполняемых процедурой оптимизации для подгонки модели. Форма этого атрибута зависит от количества оптимизированных моделей, которое, в свою очередь, зависит от количества классов.

***Добавлено в версии 1.1.***

**support\_ *ndarray формы (n\_SV)***

Индексы опорных векторов.

**support\_vectors\_ *ndarray формы (n\_SV, n\_features)***

Опорные векторы. Пустой массив, если ядро ​​предварительно вычислено.

[**n\_support\_**](https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html#sklearn.svm.SVC.n_support_)***ndarray формы (n\_classes,), dtype=int32***

Количество опорных векторов для каждого класса.

[**probA\_**](https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html#sklearn.svm.SVC.probA_)***ndarray формы (n\_classes \* (n\_classes - 1)/2)***

Параметр, полученный при масштабировании по Платту, когда probability=True.

[**probB\_**](https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html#sklearn.svm.SVC.probB_)***ndarray формы (n\_classes \* (n\_classes - 1)/2)***

Параметр, полученный при масштабировании по Платту, когда probability=True.

**shape\_fit\_ *кортеж целого числа формы (n\_dimensions\_of\_X,)***

Размеры массива обучающего вектора X.