Chapter 5

Support Vector Machine
Метод опорных векторов

Метод опорных векторов (Support Vector Machine - SVM) -это очень мощная и универсальная модель машинного обучения, способная выполнять линейную или нелинейную классификацию, регрессию и даже выявление выбросов. Она является одной из самых популярных моделей в МО, и любой интересующийся МО обязан иметь ее в своем инструментальном комплекте. Методы SVM особенно хорошо подходят для классификации сложных, но небольших или средних наборов данных.

Линейная классификация SVM

Методы SVM чувствительны к масштабам признаков, как можно видеть на рис. 5.2: график слева имеет масштаб по вертикали, намного превышающий масштаб по горизонтали, поэтому самая широкая полоса близка к горизонтали. После масштабирования признаков (например, с использованием класса StandardScaler из Scikit-Learn) граница решений выглядит гораздо лучше (на графике справа).

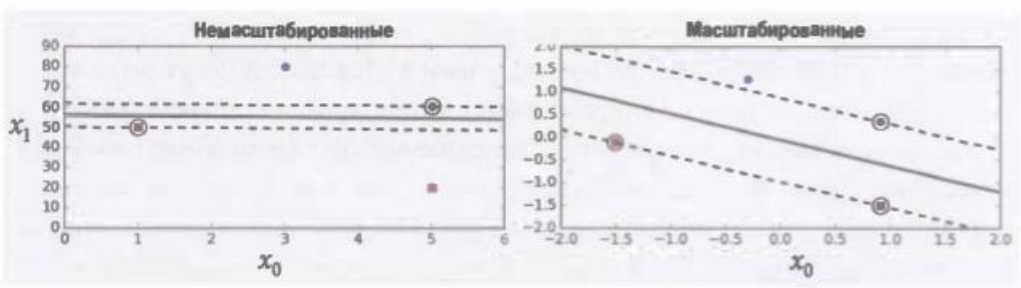

Рис. 5.2. Чувствительность к масштабам признаков

Классификация с мягким зазором

Цель заключается в том, чтобы отыскать хороший баланс между удержанием полосы как можно более широкой и ограничением количества нарушений зазора (т.е. появления экземпляров, которые оказываются посредине полосы или даже на неправильной стороне). Это называется **классификацией с мягким зазором (soft margin classification).**

В классах SVM библиотеки Scikit-Learn вы можете управлять упомянутым балансом, используя **гиперпараметр C**: *меньшее значение C ведет к более широкой полосе, но большему числу нарушений зазора*

Если ваша модель SVM переобучается, тогда можете попробовать ее регуляризировать путем сокращения ***C*.**

Имея на выбор так много ядер, как принять решение, какое ядро использовать? Примите в качестве эмпирического правила: вы должны всегда первым пробовать линейное ядро (помните, что LinearSVC гораздо быстрее SVC (kernel=" linear")), особенно если обучающий набор очень большой либо изобилует признаками. Если обучающий набор не слишком большой, тогда вы должны испытать также гауссово ядро RBF; оно работает хорошо в большинстве случаев. При наличии свободного времени и вычислительной мощности вы также можете поэкспериментировать с рядом других ядер, применяя перекрестную проверку и решетчатый поиск, в особенности, когда существуют ядра, которые приспособлены к структуре данных вашего обучающего набора.

Таблица 5.1. Сравнение классов Scikit-Learn для классификации SVM

Класс	Сложность времени обучения	Поддержка внешнего обучения	Требуется ли масштаби-рование	Ядерный трюк
`LinearSVC`	$O(m \times n)$	Нет	Да	Нет
`SGDClassifier`	$O(m \times n)$	Да	Да	Нет
`SVC`	от $O(m^2 \times n)$ до $O(m^3 \times n)$	Нет	Да	Да

Регрессия SVM

`LinearSVR`

Как упоминалось ранее, алгоритм SVM довольно универсален: он поддерживает не только линейную и нелинейную классификацию, но также линейную и нелинейную регрессию.

Регрессия SVM пробует уместить как можно больше образцов вне полосы.

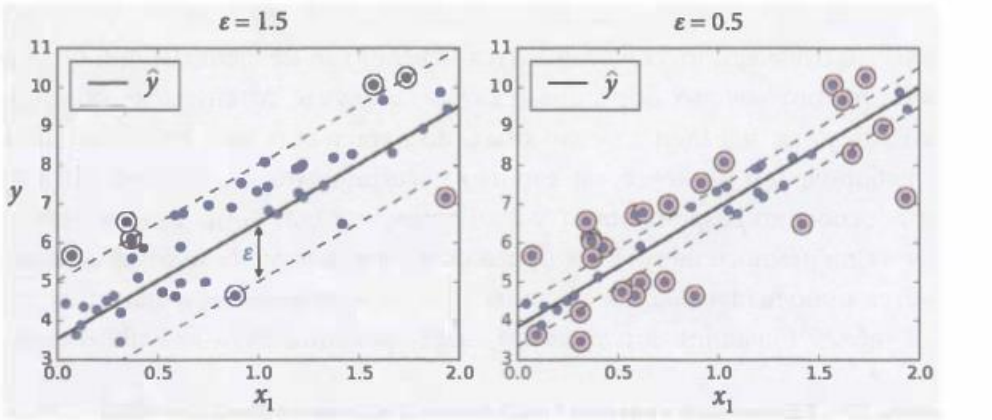

Рис. 5.10. Регрессия SVM

Добавление дополнительных обучающих образцов внутри зазора не влияет на прогнозы модели; соответственно, говорят, что модель нечувствительна к ε.

Внутренняя кухня

Уравнение 5.2. Прогноз линейного классификатора SVM

$$\hat{y} = \begin{cases} 0, \text{ если } \mathbf{w}^T \cdot \mathbf{x} + b < 0, \\ 1, \text{ если } \mathbf{w}^T \cdot \mathbf{x} + b \geq 0 \end{cases}$$

Ответы на вопросы:

1. Какая фундаментальная идея лежит в основе методов опорных векторов?

Фундаментальная идея, лежащая в основе методов опорных векторов, состоит в том, чтобы обеспечить самую широкую, какую только возможно, полосу между классами. Другими словами, целью является наличие как можно более широкого зазора между границей решений, которая разделяет два класса и обучающие образцы. При выполнении классификации с мягким зазором классификатора SVM ищет компромисс между идеальным разделением двух классов и самой широкой из возможных полосой (т.е. несколько образцов могут оказаться на самой полосе). Еще одна ключевая идея в том, чтобы использовать ядра при обучении на нелинейных наборах данных.

2. Что такое опорный вектор?

После обучения классификатора SVM опорный вектор - это любой образец, расположенный на полосе (см. предыдущий ответ), включая границу. Граница решений полностью определяется опорными векторами. Образцы, которые не являются опорными векторами (т.е. находятся вне полосы), не оказывают никакого влияния; вы могли бы удалить такие образцы, добавить дополнительные образцы или переместить их, и до тех пор, пока образцы остаются вне полосы, они не будут влиять на границу решений. При вычислении прогнозов задействуются только опорные векторы, а не полный обучающий набор.

3. Почему важно масштабировать входные образцы при использовании методов SVM?

Методы SVM пытаются обеспечить самую широкую, какую только возможно, полосу между классами (см. первый ответ), так что если

обучающий набор не масштабирован, то методы SVM будут иметь тенденцию игнорировать небольшие признаки.

4. Может ли классификатор SVM выдать меру доверия, когда он классифицирует образец? Как насчет вероятности?

Классификатор SVM способен выдавать расстояние между испытательным образцом и границей решений, которое вы можете применять в качестве меры доверия. Тем не менее, эту меру невозможно прямо преобразовать в оценку вероятности класса. Если вы установите `probability=True` при создании классификатора SVM в Scikit-Learn, тогда после обучения он будет калибровать вероятности с использованием логистической регрессии по мерам SVM (обученных посредством дополнительной перекрестной проверки с контролем по пяти блокам на обучающих данных). Это добавит к SVM методы `predict_proba()` и `predict_log_proba()`

5. Какую форму задачи SVM -прямую или двойственную вы должны применять для обучения модели на обучающем образце с миллионом образцов и сотнями признаков?

Данный вопрос применим только к линейным SVM, поскольку ядерные SVM могут применять только двойственную форму. Вычислительная сложность прямой формы задачи SVM пропорциональна количеству обучающих образцов m, в то время как вычислительная сложность двойственной формы пропорциональна числу между m^2 и m^3. Таким образом, при наличии миллионов образцов вы определенно должны использовать прямую форму, потому что двойственная форма будет гораздо более медленной.

6. Пусть вы обучаете классификатор SVM с ядром RBF. Кажется, он недообучается на обучающем наборе: вам следует увеличить или уменьшить γ? Что скажете о C?

Если классификатор SVM с ядром RBF недообучается на обучающем наборе, то возможно регуляризации слишком много. Чтобы уменьшить ее, вам понадобится увеличить гиперпараметр γ либо C (или оба).