Санкт-Петербургский политехнический университет Петра Великого

Институт прикладной математики и механики

Кафедра прикладной математики

Машина опорных векторов
Доклад на тему:
“Асимптотическое поведение LOO-ошибки SVM-классификатора с гауссовым ядром”

Выполнил:

Студент: Дамаскинский Константин
Группа: 3630102/70201

2021 г.

Содержание

Список иллюстраций

Список таблиц

1. Постановка задачи

1.1. Общая формулировка

Пусть дана тренировочная последовательность $\{x_i, y_i\}, x_i \in \mathbb{R}^n,\ y_i \in \{1, -1\},\ i \in \overline{1, l}$. Тогда настройка машины опорных векторов состоит в решении следующей задачи оптимизации:

$$\min_{w,b,\xi} \frac{1}{2} w^T w + C \sum_{i=1}^{l} \xi_i \tag{1.1.1}$$

при условии

$$y_i(w^T z_i + b) \geq 1 - \xi_i \tag{1.1.2}$$

$$\xi_i \geq 0, i \in \overline{1, l} \tag{1.1.3}$$

где $z_i = \varphi(x_i)$ – результат отображения тренировочного вектора в пространство размерности $\dim w$, $C > 0$ – штрафной параметр.

К данной задаче квадратичного программирования строится двойственная задача:

$$\min_{\alpha} F(\alpha) = \frac{1}{2} \alpha^T Q \alpha - e^T \alpha \tag{1.1.4}$$

при условии

$$0 \leq \alpha_i \leq C \tag{1.1.5}$$

$$y^T \alpha = 0, i \in \overline{1, l} \tag{1.1.6}$$

где $e = (1, \ldots, 1)^T$, Q – положительно полуопределённая матрица размера $l \times l$, задаваемая по формуле: $Q_{ij} = y_i K(x_i, x_j) y_j; K(x_i, x_j) = \varphi^T(x_i)\varphi(x_j)$ – *ядро*. Тогда $w = \sum_{i=1}^{l} \alpha_i y_i \varphi(x_i)$.

В данном докладе мы рассмотрим настройку машины опорных векторов с гауссовым ядром:

$$K(\tilde{x}, \overline{x}) = \exp\left(\frac{-\|\tilde{x} - \overline{x}\|^2}{2\sigma^2}\right) \tag{1.1.7}$$

1.2. LOO-ошибка

Оценить ошибку обобщения, получаемую при обучении SVM, можно вычислив loo-ошибку (leave-one-out): машину обучают на всей тренировочной последовательности без i-го элемента, затем подают на вход i-й элемент и проверяют, правильно ли машина его классифицировала. Данную операцию проделывают для всех элементов тренировочной выборки. Доля неверно классифицированных элементов ТП и называется loo-ошибкой.

2. Построение оптимального SVM-классификатора

В данном разделе будет показана методика выбора параметров σ и C, позволяющая построить хорошо обученный SVM-классификатор.

2.1. Асимптотический анализ поведения SVM

Будем оценивать поведение машины в крайних возможных значениях штрафного параметра C ($C \to 0, C \to \infty$) и параметра рассеяния σ ($\sigma^2 \to 0, \sigma^2 \to \infty$).

Ниже приведены результаты, полученные в [1] . Они справедливы при следующих предположениях:

1. $l_1 > l_2 + 1 > 2$, где l_i – мощность i-ой тренировочной последовательности
2. $\forall i \neq j : \ x_i \neq x_j$ – все элементы обучающей выборки различны.

В таком случае имеем следующие **критерии обучаемости**:

1. Сильное недообучение происходит, когда:
   (a) параметр σ конечный и $C \to 0$
   (b) $\sigma \to 0$ и C – конечный и достаточно маленький
   (c) $\sigma \to \infty$ и C – конечный
2. Сильное переобучение происходит, когда $\sigma \to 0$ и C очень большой
3. Если σ конечный и $C \to \infty$, то классификатор хорошо разделяет обучающую выборку на два класса, следовательно, в случае сильно зашумлённой обучающей выборки мы получим переобучение
4. Если $C = \tilde{C}\sigma^2$ и $\sigma \to \infty$, где $\tilde{C} = const$, то такой классификатор вырождается в линейный SVM-классификатор с штрафным параметром C.

Полученные результаты можно представить графически следующим образом:

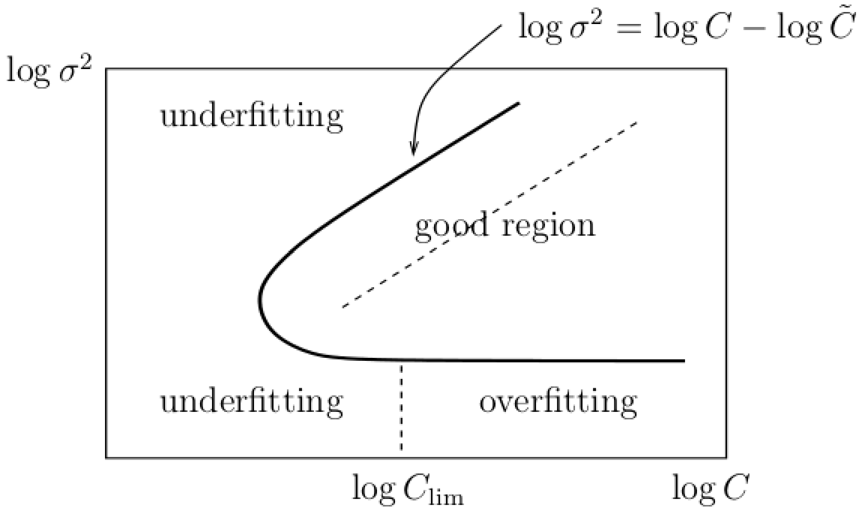

Рис. 1. График обучаемости SVM с гауссовым ядром. По вертикальной оси отложен $\log \sigma^2$, по горизонтальной – $\log C$

2.2. Комментарии к приведённой иллюстрации

Underfitting – область, в которой происходит недообучение модели, overfitting – переобучение. Любая прямая, описываемая уравнением $\log \sigma^2 = \log C - \log \tilde{C}$, делает классификатор линейным. Пунктирная прямая соответствует выбору $\tilde{C}$, обеспечивающему оптимальную обученность.

Область, отмеченная верхней надписью “underfitting”, соответствует пункту 1c критериев обучаемости.

Область с напдисью “good region” соответствует пункту 4.

Область, отмеченная нижней надписью “underfitting”, соответствует пунктам 1a и 1b.

Область, отмеченная нижней надписью “overfitting”, соответствует пункту 2. При этом параметр C_{lim} соответствует значению C, левее которого действуют предположения пункта 1b, а правее – пункта 2.

Горизонтальная прямая, разделяющая “good region ” и “overfitting”, соответствует пункту 3.

Стоит отметить, что данный рисунок, как, впрочем, и описание характеристик C и σ не являются математически строгими: понятия “очень большой” или “достаточно маленький” в каждой конкретной ситуации могут означать совершенно разные числовые значения. Однако приведённое графическое разделение пространства гиперпараметров на данные части позволяет получить представление о том, как строить эффективный эвристический алгоритм настройки машины опорных векторов, речь о котором пойдёт в следующем разделе.

2.3. Подбор σ и C

Самым простым методом подбора является метод грубой силы: в логарифмических координатах строится равномерная сетка $r \times r$ и по *всем узлам* ищется комбинация, дающая наименьшую общую ошибку обучения. Однако данный метод, очевидно, является очень непроизводительным, поскольку требует r^2 итераций.

Гораздо более рациональный способ подбора оптимальных σ и C вытекает из результатов, показанных на рисунке 2.1: найдём сначала уравнение пунктирной прямой, а затем – оптимальную комбинацию на ней.

Таким образом, предлагается следующий алгоритм:

1. Найдём оптимальный C для **линейного классификатора** (при $\sigma \to \infty$) и назовём его $\tilde{C}$
2. Найдём оптимальную пару, лежащую на прямой, задаваемой уравнением:

$$\log \sigma^2 = \log C - \log \tilde{C} \tag{2.3.1}$$

Поскольку при $\sigma \to \infty$ SVM с гауссовым ядром ведёт себя как линейный классификатор, подобранный $\tilde{C}$ будет гарантировать хорошую обученность. Во многих задачах распознавания шаблонов линейный классификатор сам по себе даёт удовлетворительный результат. Поэтому второй шаг можно рассматривать как этап улучшения точности результатов, выдаваемых классификатором.

Рассматривая предложенный двухстадийный подбор параметров на сетке $r \times r$, получим, что процедура поиска в таком случае потребует только $2r$ итераций [1], вместо r^2 итераций, необходимых для полного перебора всех узлов сетки.

3. Тестирование эвристического метода настройки SVM-классификатора

Название	Число признаков	Мощность ТП	Размерность тестовой выборки	loo. Метод грубой силы	loo. Эвристика
banana	2	400	4900	0.1235	0.1178
diabets	8	468	300	0.2433	0.2433
image	18	1300	1010	0.02475	0.02475
splice	60	1000	2175	0.09701	0.10110
ringnorm	20	400	7000	0.01429	0.01800
twonorm	20	400	7000	0.03100	0.02914
tree	18	700	11692	0.1132	0.1246
adult	123	1605	29589	0.1614	0.1614
web	300	2427	38994	0.02223	0.02223

Таблица 1. Сравнение поведения SVM при двух методах настройки

[1]"Двойка" обусловлена тем, что в уравнении 2.3.1 стоит $\log \sigma^2 \equiv 2 \log \sigma$, то есть для перебора на сетке с шагом по $\log C$ равным r понадобится $2r$ операций

Для настройки SVM методом грубой силы была использована равномерная сетка $[-10; 10] \times [-10; 10]$ с шагом 1 (всего рассматривалась 441 точка пространства гиперпараметров).

При настройке SVM эвристическим методом поиск $\tilde{C}$ осуществлялся путём пятикратной валидации на линейной SVM с использованием равномерной сетки $[-8; 2]$ для $\log C$ и $[-8; 8]$ для $\log \sigma^2$ с шагом 0.5. Всего в совокупности (учитывая поиск $\tilde{C}$) было рассмотрено 54 точки – в 8 раз меньше, чем в методе грубой силы!

В семи из десяти задач эвристический метод показал loo-ошибку не хуже, чем метод грубой силы. Принимая также во внимание лучшую производительность данного метода по сравнению с методом грубой силы, можно сделать вывод, что данный метод хорошо подходит для применения на практике.

Список литературы

[1] Chih-Jen Lin S. Sathiya Keerthi. *Asymptotic Behaviors of Support Vector Machines with Gaussian Kernel.* англ. 2003.