

Введение в искусственный интеллект.

Машинное обучение

Тема: Машины опорных векторов – SVM

Бабин Д.Н., Иванов И.Е.

кафедра Математической Теории Интеллектуальных Систем



1 Случай линейной разделимости

1. Случай линейной разделимости
2. Случай линейной неразделимости

1. Случай линейной разделимости
2. Случай линейной неразделимости
3. Решение с помощью двойственной задачи

1. Случай линейной разделимости
2. Случай линейной неразделимости
3. Решение с помощью двойственной задачи
4. Обобщение SVM с помощью ядрового трюка

1. Случай линейной разделимости
2. Случай линейной неразделимости
3. Решение с помощью двойственной задачи
4. Обобщение SVM с помощью ядрового трюка
5. Многоклассовый SVM

Вспомним прошлую лекцию

Рассмотрим задачу бинарной классификации: $X \rightarrow Y$, $X = \mathbb{R}^n$, $Y = \{+1, -1\}$ на обучающей выборке $X^m = (x_i, y_i)_{i=1}^m$.

Линейный классификатор $a(x; w, w_0) = \text{sign}(\langle w, x \rangle - w_0)$.



Вспомним прошлую лекцию

Рассмотрим задачу бинарной классификации: $X \rightarrow Y$, $X = \mathbb{R}^n$, $Y = \{+1, -1\}$ на обучающей выборке $X^m = (x_i, y_i)_{i=1}^m$.

Линейный классификатор $a(x; w, w_0) = \text{sign}(\langle w, x \rangle - w_0)$.

Минимизация эмпирического риска в данном случае:

$$R(w, w_0, X^m) = \sum_{i=1}^m [a(x_i; w, w_0) \neq y_i] = \sum_{i=1}^m [(\langle w, x_i \rangle - w_0)y_i < 0],$$



Вспомним прошлую лекцию

Рассмотрим задачу бинарной классификации: $X \rightarrow Y$, $X = \mathbb{R}^n$, $Y = \{+1, -1\}$ на обучающей выборке $X^m = (x_i, y_i)_{i=1}^m$.

Линейный классификатор $a(x; w, w_0) = \text{sign}(\langle w, x \rangle - w_0)$.

Минимизация эмпирического риска в данном случае:

$$R(w, w_0, X^m) = \sum_{i=1}^m [a(x_i; w, w_0) \neq y_i] = \sum_{i=1}^m [(\langle w, x_i \rangle - w_0)y_i < 0],$$

Добавим аппроксимацию и L_2 регуляризацию:

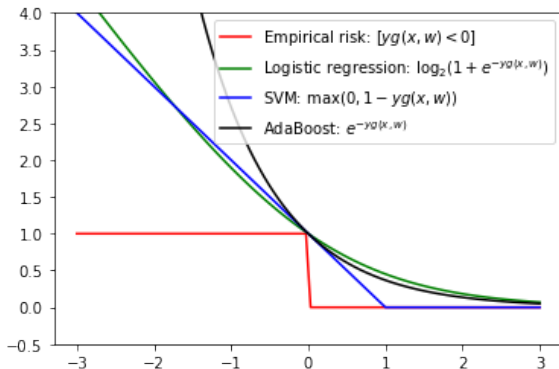
$$R(w, w_0, X^m) \leq \sum_{i=1}^m \max(0, 1 - (\langle w, x_i \rangle - w_0)y_i) + \frac{1}{2C} \|w\|^2$$



- Аппроксимация штрафует за приближение к границе классов: $(\langle w, x_i \rangle - w_0)y_i = 1$

Аппроксимация и регуляризация

- Аппроксимация штрафует за приближение к границе классов: $(\langle w, x_i \rangle - w_0)y_i = 1$
- Регуляризация штрафует неустойчивые решения



Случай линейной разделимости¹

Линейная разделимость

$\exists w, w_0$ т.ч. $(\langle w, x_i \rangle - w_0)y_i > 0$ для всех $i = 1, \dots, m$.

Очевидно, что можно перенормировать вектор w , т.ч. $\min_i (\langle w, x_i \rangle - w_0)y_i = 1$.

Разделяющая полоса: $-1 \leq \langle w, x_i \rangle - w_0 \leq +1$.

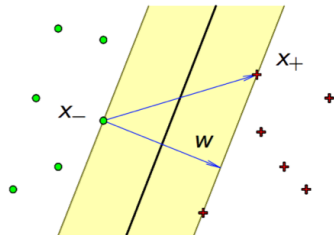
Разделяющая гиперплоскость (посередине):

$$\langle w, x_i \rangle - w_0 = 0.$$

Можем добиться того, что существует по крайней мере одна точка на каждой из границ

(Упражнение: доказать): $\exists x_{\pm} : \langle w, x_{\pm} \rangle - w_0 = \pm 1$.

Ширина полосы: $\frac{\langle x_+ - x_-, w \rangle}{\|w\|} = \frac{2}{\|w\|} \rightarrow \max_w$



¹Вапник, В. Н., Червоненкис, А. Я. (1964). Об одном классе персептронов.

Т.о., в случае линейной разделимости можно оптимизационную задачу записать как:

$$\begin{cases} \frac{1}{2} \|w\|^2 \rightarrow \min_w, \\ y_i(\langle w, x_i \rangle - w_0) \geq 1, \quad i = 1, \dots, m \end{cases}$$

Случай линейной неразделимости

Обобщим² задачу на этот случай: алгоритм может допускать ошибки на обучающих объектах.

²Cortes, C., and Vapnik, V. (1995). Support-vector networks.

Случай линейной неразделимости

Обобщим² задачу на этот случай: алгоритм может допускать ошибки на обучающих объектах.

Ограничение: таких ошибок должно быть поменьше.

²Cortes, C., and Vapnik, V. (1995). Support-vector networks.

Случай линейной неразделимости

Обобщим² задачу на этот случай: алгоритм может допускать ошибки на обучающих объектах.

Ограничение: таких ошибок должно быть поменьше.

Решение: введение дополнительных переменных $\xi_i \geq 0$, характеризующих величину ошибки на объектах x_i .

²Cortes, C., and Vapnik, V. (1995). Support-vector networks.



Случай линейной неразделимости

Обобщим² задачу на этот случай: алгоритм может допускать ошибки на обучающих объектах.

Ограничение: таких ошибок должно быть поменьше.

Решение: введение дополнительных переменных $\xi_i \geq 0$, характеризующих величину ошибки на объектах x_i .

$$\begin{cases} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^m \xi_i \rightarrow \min_{w, w_0, \xi}, \\ y_i(\langle w, x_i \rangle - w_0) \geq 1 - \xi_i, & i = 1, \dots, m, \\ \xi_i \geq 0 & i = 1, \dots, m. \end{cases}$$

²Cortes, C., and Vapnik, V. (1995). Support-vector networks.

Случай линейной неразделимости

Обобщим² задачу на этот случай: алгоритм может допускать ошибки на обучающих объектах.

Ограничение: таких ошибок должно быть поменьше.

Решение: введение дополнительных переменных $\xi_i \geq 0$, характеризующих величину ошибки на объектах x_i .

$$\begin{cases} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^m \xi_i \rightarrow \min_{w, w_0, \xi}, \\ y_i(\langle w, x_i \rangle - w_0) \geq 1 - \xi_i, & i = 1, \dots, m, \\ \xi_i \geq 0 & i = 1, \dots, m. \end{cases}$$

Замечание. Положительная константа C определяет компромисс между максимизацией ширины разделяющей полосы и минимизацией суммарной ошибки.

²Cortes, C., and Vapnik, V. (1995). Support-vector networks.

О константе C

Поставленная выше оптимизационная задача эквивалентна минимизации аппроксимированного э.р. с регуляризатором:

$$\sum_{i=1}^m \max(0, 1 - y_i(\langle w, x_i \rangle - w_0)) + \frac{1}{2C} \|w\|^2 \rightarrow \min_{w, w_0}$$

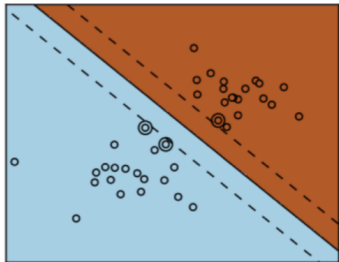


О константе C

Поставленная выше оптимизационная задача эквивалентна минимизации аппроксимированного э.р. с регуляризатором:

$$\sum_{i=1}^m \max(0, 1 - y_i(\langle w, x_i \rangle - w_0)) + \frac{1}{2C} \|w\|^2 \rightarrow \min_{w, w_0}$$

Большое значение C : узкая полоса, мало ошибок

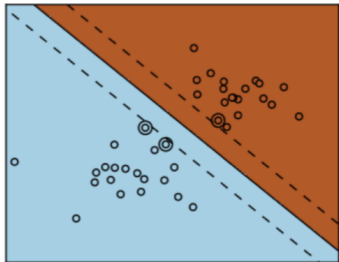


О константе C

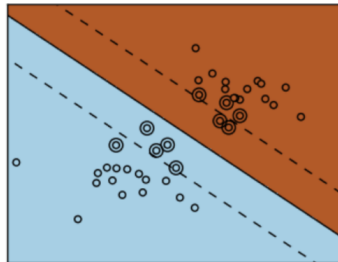
Поставленная выше оптимизационная задача эквивалентна минимизации аппроксимированного э.р. с регуляризатором:

$$\sum_{i=1}^m \max(0, 1 - y_i(\langle w, x_i \rangle - w_0)) + \frac{1}{2C} \|w\|^2 \rightarrow \min_{w, w_0}$$

Большое значение C : узкая полоса, мало ошибок



Маленькое значение C : широкая полоса, много ошибок





Условия Каруша-Куна-Таккера (ККТ)^{3,4}

Условия ККТ – это **необходимые** условия решения задачи нелинейного программирования (обобщение метода множителей Лагранжа).

Задача нелинейного программирования:

$$\begin{cases} f(x) \rightarrow \min_x, \\ g_i(x) \leq 0, & i = 1, \dots, k, \\ h_j(x) = 0, & j = 1, \dots, \ell. \end{cases}$$

³W. Karush (1939). Minima of Functions of Several Variables with Inequalities as Side Constraints.

⁴Kuhn, H. W.; Tucker, A. W. (1951). Nonlinear programming.

Условия Каруша-Куна-Таккера (ККТ)^{3,4}

Условия ККТ – это **необходимые** условия решения задачи нелинейного программирования (обобщение метода множителей Лагранжа).

Задача нелинейного программирования:

$$\begin{cases} f(x) \rightarrow \min_x, \\ g_i(x) \leq 0, & i = 1, \dots, k, \\ h_j(x) = 0, & j = 1, \dots, \ell. \end{cases}$$

Необходимые условия: если x - точка локального минимума, то \exists множители μ_i, λ_j , т.ч.

$$\begin{cases} \frac{\partial L}{\partial x} = 0; L(x, \mu, \lambda) = f(x) + \sum_{i=1}^k \mu_i g_i(x) + \sum_{j=1}^{\ell} \lambda_j h_j(x) & \text{(функция Лагранжа)} \\ g_i(x) \leq 0, h_j(x) = 0 & \text{(исходные ограничения)} \\ \mu_i \geq 0 & \text{(двойственные ограничения)} \\ \mu_i g_i(x) = 0 & \text{(дополняющая нежесткость)} \end{cases}$$

³W. Karush (1939). Minima of Functions of Several Variables with Inequalities as Side Constraints.

⁴Kuhn, H. W.; Tucker, A. W. (1951). Nonlinear programming.

Функция Лагранжа для SVM

$$L(w, w_0, \xi; \lambda, \eta) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^m \lambda_i ((\langle w, x_i \rangle - w_0) y_i - 1) - \sum_{i=1}^m \xi_i (\lambda_i + \eta_i - C),$$
 где:
 λ_i – переменные, двойственные к ограничениям $(\langle w, x_i \rangle - w_0) y_i \geq 1 - \xi_i$,
 η_i – переменные, двойственные к ограничениям $\xi_i \geq 0$.



Функция Лагранжа для SVM

$L(w, w_0, \xi; \lambda, \eta) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^m \lambda_i ((\langle w, x_i \rangle - w_0) y_i - 1) - \sum_{i=1}^m \xi_i (\lambda_i + \eta_i - C)$, где:
 λ_i – переменные, двойственные к ограничениям $(\langle w, x_i \rangle - w_0) y_i \geq 1 - \xi_i$,
 η_i – переменные, двойственные к ограничениям $\xi_i \geq 0$.

Необходимые условия примут вид:

$$\begin{cases} \frac{\partial L}{\partial w} = 0; \frac{\partial L}{\partial w_0} = 0; \frac{\partial L}{\partial \xi} = 0 \\ \xi_i \geq 0, \lambda_i \geq 0, \eta_i \geq 0 & i = 1, \dots, m \\ \lambda_i = 0 \text{ либо } (\langle w, x_i \rangle - w_0) y_i = 1 - \xi_i & i = 1, \dots, m \\ \eta_i = 0 \text{ либо } \xi_i = 0 & i = 1, \dots, m \end{cases}$$



Дифференцируем функцию Лагранжа

$$L(w, w_0, \xi; \lambda, \eta) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^m \lambda_i ((\langle w, x_i \rangle - w_0) y_i - 1) - \sum_{i=1}^m \xi_i (\lambda_i + \eta_i - C).$$

$$\begin{cases} \frac{\partial L}{\partial w} = w - \sum_{i=1}^m \lambda_i y_i x_i = 0 & \Rightarrow w = \sum_{i=1}^m \lambda_i y_i x_i \\ \frac{\partial L}{\partial w_0} = - \sum_{i=1}^m \lambda_i y_i = 0 & \Rightarrow \sum_{i=1}^m \lambda_i y_i = 0 \\ \frac{\partial L}{\partial \xi_i} = -\lambda_i - \eta_i + C = 0 & \Rightarrow \lambda_i + \eta_i = C \end{cases}$$

Дифференцируем функцию Лагранжа

$$L(w, w_0, \xi; \lambda, \eta) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^m \lambda_i ((\langle w, x_i \rangle - w_0) y_i - 1) - \sum_{i=1}^m \xi_i (\lambda_i + \eta_i - C).$$

$$\begin{cases} \frac{\partial L}{\partial w} = w - \sum_{i=1}^m \lambda_i y_i x_i = 0 & \Rightarrow w = \sum_{i=1}^m \lambda_i y_i x_i \\ \frac{\partial L}{\partial w_0} = - \sum_{i=1}^m \lambda_i y_i = 0 & \Rightarrow \sum_{i=1}^m \lambda_i y_i = 0 \\ \frac{\partial L}{\partial \xi_i} = -\lambda_i - \eta_i + C = 0 & \Rightarrow \lambda_i + \eta_i = C \end{cases}$$

❶ $\lambda_i = 0, \eta_i = C, \xi_i = 0, (\langle w, x_i \rangle - w_0) y_i \geq 1$: периферийные объекты

Дифференцируем функцию Лагранжа

$$L(w, w_0, \xi; \lambda, \eta) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^m \lambda_i ((\langle w, x_i \rangle - w_0) y_i - 1) - \sum_{i=1}^m \xi_i (\lambda_i + \eta_i - C).$$

$$\begin{cases} \frac{\partial L}{\partial w} = w - \sum_{i=1}^m \lambda_i y_i x_i = 0 & \Rightarrow w = \sum_{i=1}^m \lambda_i y_i x_i \\ \frac{\partial L}{\partial w_0} = - \sum_{i=1}^m \lambda_i y_i = 0 & \Rightarrow \sum_{i=1}^m \lambda_i y_i = 0 \\ \frac{\partial L}{\partial \xi_i} = -\lambda_i - \eta_i + C = 0 & \Rightarrow \lambda_i + \eta_i = C \end{cases}$$

- ❶ $\lambda_i = 0, \eta_i = C, \xi_i = 0, (\langle w, x_i \rangle - w_0) y_i \geq 1$: периферийные объекты
- ❷ $0 < \lambda_i < C, 0 < \eta_i < C, \xi_i = 0, (\langle w, x_i \rangle - w_0) y_i = 1$: опорные объекты на границе



Дифференцируем функцию Лагранжа

$$L(w, w_0, \xi; \lambda, \eta) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^m \lambda_i ((\langle w, x_i \rangle - w_0) y_i - 1) - \sum_{i=1}^m \xi_i (\lambda_i + \eta_i - C).$$

$$\begin{cases} \frac{\partial L}{\partial w} = w - \sum_{i=1}^m \lambda_i y_i x_i = 0 & \Rightarrow w = \sum_{i=1}^m \lambda_i y_i x_i \\ \frac{\partial L}{\partial w_0} = - \sum_{i=1}^m \lambda_i y_i = 0 & \Rightarrow \sum_{i=1}^m \lambda_i y_i = 0 \\ \frac{\partial L}{\partial \xi_i} = -\lambda_i - \eta_i + C = 0 & \Rightarrow \lambda_i + \eta_i = C \end{cases}$$

- ❶ $\lambda_i = 0, \eta_i = C, \xi_i = 0, (\langle w, x_i \rangle - w_0) y_i \geq 1$: периферийные объекты
- ❷ $0 < \lambda_i < C, 0 < \eta_i < C, \xi_i = 0, (\langle w, x_i \rangle - w_0) y_i = 1$: опорные объекты на границе
- ❸ $\lambda_i = C, \eta_i = 0, \xi_i > 0, (\langle w, x_i \rangle - w_0) y_i < 1$: опорные объекты-ошибки



О двойственных задачах

Прямая задача:

$$\begin{cases} f(x) \rightarrow \min_x, \\ g_i(x) \leq 0, & i = 1, \dots, k, \\ h_j(x) = 0, & j = 1, \dots, \ell. \end{cases}$$

Функция Лагранжа: $L(x, \mu, \lambda) = f(x) + \sum_{i=1}^k \mu_i g_i(x) + \sum_{j=1}^{\ell} \lambda_j h_j(x) \rightarrow \min_x$.

⁵Wolfe, P. (1961). A duality theorem for non-linear programming.

О двойственных задачах

Прямая задача:

$$\begin{cases} f(x) \rightarrow \min_x, \\ g_i(x) \leq 0, & i = 1, \dots, k, \\ h_j(x) = 0, & j = 1, \dots, \ell. \end{cases}$$

Функция Лагранжа: $L(x, \mu, \lambda) = f(x) + \sum_{i=1}^k \mu_i g_i(x) + \sum_{j=1}^{\ell} \lambda_j h_j(x) \rightarrow \min_x$.

Двойственная функция: $Q(\mu, \lambda) = \min_x L(x, \mu, \lambda)$.

⁵Wolfe, P. (1961). A duality theorem for non-linear programming.



О двойственных задачах

Прямая задача:

$$\begin{cases} f(x) \rightarrow \min_x, \\ g_i(x) \leq 0, & i = 1, \dots, k, \\ h_j(x) = 0, & j = 1, \dots, \ell. \end{cases}$$

Функция Лагранжа: $L(x, \mu, \lambda) = f(x) + \sum_{i=1}^k \mu_i g_i(x) + \sum_{j=1}^{\ell} \lambda_j h_j(x) \rightarrow \min_x$.

Двойственная функция: $Q(\mu, \lambda) = \min_x L(x, \mu, \lambda)$.

Двойственная задача:

$$\begin{cases} Q(\mu, \lambda) \rightarrow \max_{\mu, \lambda}, \\ \mu_i \geq 0, & i = 1, \dots, k \end{cases}$$

⁵Wolfe, P. (1961). A duality theorem for non-linear programming.



О двойственных задачах

Прямая задача:

$$\begin{cases} f(x) \rightarrow \min_x, \\ g_i(x) \leq 0, & i = 1, \dots, k, \\ h_j(x) = 0, & j = 1, \dots, \ell. \end{cases}$$

Функция Лагранжа: $L(x, \mu, \lambda) = f(x) + \sum_{i=1}^k \mu_i g_i(x) + \sum_{j=1}^{\ell} \lambda_j h_j(x) \rightarrow \min_x$.

Двойственная функция: $Q(\mu, \lambda) = \min_x L(x, \mu, \lambda)$.

Двойственная задача:

$$\begin{cases} Q(\mu, \lambda) \rightarrow \max_{\mu, \lambda}, \\ \mu_i \geq 0, & i = 1, \dots, k \end{cases}$$

Теорема (Дуальность Вулфа⁵)

Если $f(x)$, $g_i(x)$, $h_j(x)$ – выпуклые функции, x^* – решение прямой задачи, а (μ^*, λ^*) – решение двойственной задачи, то $Q(\mu^*, \lambda^*) = f(x^*)$.

⁵Wolfe, P. (1961). A duality theorem for non-linear programming.

Двойственная задача для SVM

Подставим решения прямой задачи

$$w = \sum_{i=1}^m \lambda_i y_i x_i, \sum_{i=1}^m \lambda_i y_i = 0, \lambda_i + \eta_i = C$$

в функцию Лагранжа

$$L(w, w_0, \xi; \lambda, \eta) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^m \lambda_i (y_i (\langle w, x_i \rangle - w_0) - 1) - \sum_{i=1}^m \xi_i (\lambda_i + \eta_i - C).$$

Двойственная задача для SVM

Подставим решения прямой задачи

$$w = \sum_{i=1}^m \lambda_i y_i x_i, \sum_{i=1}^m \lambda_i y_i = 0, \lambda_i + \eta_i = C$$

в функцию Лагранжа

$$L(w, w_0, \xi; \lambda, \eta) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^m \lambda_i (y_i (\langle w, x_i \rangle - w_0) - 1) - \sum_{i=1}^m \xi_i (\lambda_i + \eta_i - C).$$

Получим:

$$\begin{aligned} Q(\lambda) &= \frac{1}{2} \sum_{i=1}^m \lambda_i y_i x_i \sum_{j=1}^m \lambda_j y_j x_j - \sum_{i=1}^m \lambda_i y_i \left\langle \sum_{j=1}^m \lambda_j y_j x_j, x_i \right\rangle + w_0 \sum_{i=1}^m \lambda_i y_i + \sum_{i=1}^m \lambda_i - \sum_{i=1}^m \xi_i (C - C) = \\ &= -\frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m \lambda_i \lambda_j y_i y_j \langle x_i, x_j \rangle + \sum_{i=1}^m \lambda_i \end{aligned}$$



Двойственная задача для SVM

Объединяя, получаем формулировку двойственной задачи:

$$\begin{cases} -Q(\lambda) = \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m \lambda_i \lambda_j y_i y_j \langle x_i, x_j \rangle - \sum_{i=1}^m \lambda_i \rightarrow \min_{\lambda}, \\ 0 \leq \lambda_i \leq C, \\ \sum_{i=1}^m \lambda_i y_i = 0. \end{cases}$$

⁶https://en.wikipedia.org/wiki/Quadratic_programming



Двойственная задача для SVM

Объединяя, получаем формулировку двойственной задачи:

$$\begin{cases} -Q(\lambda) = \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m \lambda_i \lambda_j y_i y_j \langle x_i, x_j \rangle - \sum_{i=1}^m \lambda_i \rightarrow \min_{\lambda}, \\ 0 \leq \lambda_i \leq C, \\ \sum_{i=1}^m \lambda_i y_i = 0. \end{cases}$$

Минимизируется квадратичный функционал $Q(\lambda)$, имеющий неотрицательно определённую квадратичную форму \Rightarrow этот функционал – выпуклый.

⁶https://en.wikipedia.org/wiki/Quadratic_programming



Двойственная задача для SVM

Объединяя, получаем формулировку двойственной задачи:

$$\begin{cases} -Q(\lambda) = \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m \lambda_i \lambda_j y_i y_j \langle x_i, x_j \rangle - \sum_{i=1}^m \lambda_i \rightarrow \min_{\lambda}, \\ 0 \leq \lambda_i \leq C, \\ \sum_{i=1}^m \lambda_i y_i = 0. \end{cases}$$

Минимизируется квадратичный функционал $Q(\lambda)$, имеющий неотрицательно определённую квадратичную форму \Rightarrow этот функционал – выпуклый.

Область, определяемая линейными ограничениями, также выпуклая.

⁶https://en.wikipedia.org/wiki/Quadratic_programming

Двойственная задача для SVM

Объединяя, получаем формулировку двойственной задачи:

$$\begin{cases} -Q(\lambda) = \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m \lambda_i \lambda_j y_i y_j \langle x_i, x_j \rangle - \sum_{i=1}^m \lambda_i \rightarrow \min_{\lambda}, \\ 0 \leq \lambda_i \leq C, \\ \sum_{i=1}^m \lambda_i y_i = 0. \end{cases}$$

Минимизируется квадратичный функционал $Q(\lambda)$, имеющий неотрицательно определённую квадратичную форму \Rightarrow этот функционал – выпуклый.

Область, определяемая линейными ограничениями, также выпуклая.

Следовательно, данная двойственная задача имеет **единственное** решение.

⁶https://en.wikipedia.org/wiki/Quadratic_programming

Двойственная задача для SVM

Объединяя, получаем формулировку двойственной задачи:

$$\begin{cases} -Q(\lambda) = \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m \lambda_i \lambda_j y_i y_j \langle x_i, x_j \rangle - \sum_{i=1}^m \lambda_i \rightarrow \min_{\lambda}, \\ 0 \leq \lambda_i \leq C, \\ \sum_{i=1}^m \lambda_i y_i = 0. \end{cases}$$

Минимизируется квадратичный функционал $Q(\lambda)$, имеющий неотрицательно определённую квадратичную форму \Rightarrow этот функционал – выпуклый.

Область, определяемая линейными ограничениями, также выпуклая.

Следовательно, данная двойственная задача имеет **единственное** решение.

Способ решения – методами **квадратичного** программирования (например, можно использовать метод внутренней точки⁶).

⁶https://en.wikipedia.org/wiki/Quadratic_programming

Решение прямой задачи для SVM

Пусть единственное решение двойственной задачи $(\lambda_i)_{i=1}^m$. Тогда решение прямой задачи выражается через решение двойственной как:

$$\begin{cases} w = \sum_{\lambda_i \neq 0} \lambda_i y_i x_i, & \text{суммируем только по опорным векторам } \lambda_i \neq 0 \\ w_0 = \langle w, x_j \rangle - y_j & \text{для опорного вектора на границе } 0 < \lambda_j < C \end{cases}$$



Решение прямой задачи для SVM

Пусть единственное решение двойственной задачи $(\lambda_i)_{i=1}^m$. Тогда решение прямой задачи выражается через решение двойственной как:

$$\begin{cases} w = \sum_{\lambda_i \neq 0} \lambda_i y_i x_i, & \text{суммируем только по опорным векторам } \lambda_i \neq 0 \\ w_0 = \langle w, x_j \rangle - y_j & \text{для опорного вектора на границе } 0 < \lambda_j < C \end{cases}$$

При этом сам линейный классификатор примет вид

$$a(x) = \text{sign}\left(\sum_{i=1}^m \lambda_i y_i \langle x_i, x \rangle - w_0\right)$$

что можно понимать как линейность в пространстве \mathbb{R}^m с признаками $f_i = \langle x_i, x \rangle$.





Напомним задачу минимизации аппроксимированного э.р. (в данном случае – Hinge loss) с L_2 -регуляризатором:

$$\sum_{i=1}^m \max(0, 1 - y_i(\langle w, x_i \rangle - w_0)) + \frac{1}{2C} \|w\|^2 \rightarrow \min_{w, w_0}$$

⁷Lopez-Martinez, D. (2017). Regularization approaches for support vector machines with applications to biomedical data.



Напомним задачу минимизации аппроксимированного э.р. (в данном случае – Hinge loss) с L_2 -регуляризатором:

$$\sum_{i=1}^m \max(0, 1 - y_i(\langle w, x_i \rangle - w_0)) + \frac{1}{2C} \|w\|^2 \rightarrow \min_{w, w_0}$$

А что если добавить и/или заменить на L_1 -регуляризацию?

⁷Lopez-Martinez, D. (2017). Regularization approaches for support vector machines with applications to biomedical data.



Напомним задачу минимизации аппроксимированного э.р. (в данном случае – Hinge loss) с L_2 -регуляризатором:

$$\sum_{i=1}^m \max(0, 1 - y_i(\langle w, x_i \rangle - w_0)) + \frac{1}{2C} \|w\|^2 \rightarrow \min_{w, w_0}$$

А что если добавить и/или заменить на L_1 -регуляризацию?

Оказывается, обычная SVM (которая соответствует гребневой регрессии) преобразуется в LASSO либо ElasticNet SVM, и при этом все равно возможно сведение к квадратичной задаче.

⁷Lopez-Martinez, D. (2017). Regularization approaches for support vector machines with applications to biomedical data.



Аппроксимированный э.р. с L_1 регуляризатором:

$$\sum_{i=1}^m \max(0, 1 - y_i(\langle w, x_i \rangle - w_0)) + \frac{1}{C} \sum_{i=1}^n |w_i| \rightarrow \min_{w, w_0}$$



Аппроксимированный э.р. с L_1 регуляризатором:

$$\sum_{i=1}^m \max(0, 1 - y_i(\langle w, x_i \rangle - w_0)) + \frac{1}{C} \sum_{i=1}^n |w_i| \rightarrow \min_{w, w_0}$$

Введем переменные $u_i = \frac{|w_i| + w_i}{2}$, $v_i = \frac{|w_i| - w_i}{2}$, $u_i \geq 0$, $v_i \geq 0$.

Аппроксимированный э.р. с L_1 регуляризатором:

$$\sum_{i=1}^m \max(0, 1 - y_i(\langle w, x_i \rangle - w_0)) + \frac{1}{C} \sum_{i=1}^n |w_i| \rightarrow \min_{w, w_0}$$

Введем переменные $u_i = \frac{|w_i| + w_i}{2}$, $v_i = \frac{|w_i| - w_i}{2}$, $u_i \geq 0$, $v_i \geq 0$.

Тогда замена переменных: $w_i = u_i - v_i$, $|w_i| = u_i + v_i$, и задача выше сводится к:

$$\begin{cases} \sum_{i=1}^n u_i + \sum_{i=1}^n v_i + C \sum_{i=1}^m \xi_i \rightarrow \min_{u, v, w_0, \xi}, \\ y_i(\langle u, x_i \rangle - \langle v, x_i \rangle - w_0) \geq 1 - \xi_i, & i = 1, \dots, m, \\ \xi_i \geq 0 & i = 1, \dots, m, \\ u_i \geq 0, v_i \geq 0 & i = 1, \dots, n. \end{cases}$$

которая аналогична рассмотренной до этого.



Если в исходном пространстве сложно разделить выборку, то попробуем перейти в пространство большей размерности⁸ $\varphi : X \rightarrow H$.

⁸Boser, B. E., Guyon, I. M., and Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers.

Если в исходном пространстве сложно разделить выборку, то попробуем перейти в пространство большей размерности⁸ $\varphi : X \rightarrow H$.

Определение ядра

Ядро – функция $K : X \times X \rightarrow \mathbb{R}$, т.ч. $K(x_1, x_2) = \langle \varphi(x_1), \varphi(x_2) \rangle$ при некотором $\varphi : X \rightarrow H$, где H – гильбертово пространство.

⁸Boser, B. E., Guyon, I. M., and Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers.

Ядро и неотрицательность

Если в исходном пространстве сложно разделить выборку, то попробуем перейти в пространство большей размерности⁸ $\varphi : X \rightarrow H$.

Определение ядра

Ядро – функция $K : X \times X \rightarrow \mathbb{R}$, т.ч. $K(x_1, x_2) = \langle \varphi(x_1), \varphi(x_2) \rangle$ при некотором $\varphi : X \rightarrow H$, где H – гильбертово пространство.

Неотрицательная определенность

Следующие два определения эквивалентны для проверки $K(x_1, x_2)$

- $\int_X \int_X K(x_1, x_2) f(x_1) f(x_2) dx_1 dx_2 \geq 0$ для любой функции $f : X \rightarrow \mathbb{R}$
- Для любой конечной выборки $X^m = (x_1, \dots, x_m)$ из X матрица $K = \|K(x_i, x_j)\|$ размера $m \times m$ неотрицательно определена: $z^T K z \geq 0$ для любого $z \in \mathbb{R}^m$.

⁸Boser, B. E., Guyon, I. M., and Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers.

Теорема Мерсера⁹

Функция $K(x_1, x_2)$ является ядром тогда и только тогда, когда:

- $K(x_1, x_2)$ симметрична: $K(x_1, x_2) = K(x_2, x_1)$, и
- $K(x_1, x_2)$ неотрицательно определена.

⁹Mercer, J. (1909). Functions of positive and negative type, and their connection the theory of integral equations.

Теорема Мерсера⁹

Функция $K(x_1, x_2)$ является ядром тогда и только тогда, когда:

- $K(x_1, x_2)$ симметрична: $K(x_1, x_2) = K(x_2, x_1)$, и
- $K(x_1, x_2)$ неотрицательно определена.

Замечание. Если K не удовлетворяет указанным выше условиям, то минимизируемый функционал для классификатора уже не будет выпуклым, и решение может оказаться не единственным!

⁹Mercer, J. (1909). Functions of positive and negative type, and their connection the theory of integral equations.



- Скалярное произведение $K(x_1, x_2) = \langle x_1, x_2 \rangle$ – ядро

- Скалярное произведение $K(x_1, x_2) = \langle x_1, x_2 \rangle$ – ядро
- Константа $K(x_1, x_2) = c$ – ядро

- Скалярное произведение $K(x_1, x_2) = \langle x_1, x_2 \rangle$ – ядро
- Константа $K(x_1, x_2) = c$ – ядро
- Произведение ядер $K(x_1, x_2) = K_1(x_1, x_2)K_2(x_1, x_2)$ – ядро



- Скалярное произведение $K(x_1, x_2) = \langle x_1, x_2 \rangle$ – ядро
- Константа $K(x_1, x_2) = c$ – ядро
- Произведение ядер $K(x_1, x_2) = K_1(x_1, x_2)K_2(x_1, x_2)$ – ядро



- Скалярное произведение $K(x_1, x_2) = \langle x_1, x_2 \rangle$ – ядро
- Константа $K(x_1, x_2) = c$ – ядро
- Произведение ядер $K(x_1, x_2) = K_1(x_1, x_2)K_2(x_1, x_2)$ – ядро
- Линейная $K(x_1, x_2) = \alpha_1 K_1(x_1, x_2) + \alpha_2 K_2(x_1, x_2)$ – ядро при $\alpha_1, \alpha_2 > 0$, K_1, K_2 – ядрах



- Скалярное произведение $K(x_1, x_2) = \langle x_1, x_2 \rangle$ – ядро
- Константа $K(x_1, x_2) = c$ – ядро
- Произведение ядер $K(x_1, x_2) = K_1(x_1, x_2)K_2(x_1, x_2)$ – ядро
- Линейная $K(x_1, x_2) = \alpha_1 K_1(x_1, x_2) + \alpha_2 K_2(x_1, x_2)$ – ядро при $\alpha_1, \alpha_2 > 0$, K_1, K_2 - ядрах
- Для любой $\varphi : X \rightarrow X$ подстановка $K(x_1, x_2) = K_1(\varphi(x_1), \varphi(x_2))$ – ядро при K_1 - ядро



- Скалярное произведение $K(x_1, x_2) = \langle x_1, x_2 \rangle$ – ядро
- Константа $K(x_1, x_2) = c$ – ядро
- Произведение ядер $K(x_1, x_2) = K_1(x_1, x_2)K_2(x_1, x_2)$ – ядро
- Линейная $K(x_1, x_2) = \alpha_1 K_1(x_1, x_2) + \alpha_2 K_2(x_1, x_2)$ – ядро при $\alpha_1, \alpha_2 > 0$, K_1, K_2 – ядрах
- Для любой $\varphi : X \rightarrow X$ подстановка $K(x_1, x_2) = K_1(\varphi(x_1), \varphi(x_2))$ – ядро при K_1 – ядро
- $K(x_1, x_2) = k(x_1 - x_2)$ – ядро \Leftrightarrow Фурье-образ $F[k](\omega) = (2\pi)^{\frac{n}{2}} \int_X e^{-i\langle \omega, x \rangle} k(x) dx$ неотрицателен



- Скалярное произведение $K(x_1, x_2) = \langle x_1, x_2 \rangle$ – ядро
- Константа $K(x_1, x_2) = c$ – ядро
- Произведение ядер $K(x_1, x_2) = K_1(x_1, x_2)K_2(x_1, x_2)$ – ядро
- Линейная $K(x_1, x_2) = \alpha_1 K_1(x_1, x_2) + \alpha_2 K_2(x_1, x_2)$ – ядро при $\alpha_1, \alpha_2 > 0$, K_1, K_2 – ядрах
- Для любой $\varphi : X \rightarrow X$ подстановка $K(x_1, x_2) = K_1(\varphi(x_1), \varphi(x_2))$ – ядро при K_1 – ядро
- $K(x_1, x_2) = k(x_1 - x_2)$ – ядро \Leftrightarrow Фурье-образ $F[k](\omega) = (2\pi)^{\frac{n}{2}} \int_X e^{-i\langle \omega, x \rangle} k(x) dx$ неотрицателен
- Композиция произвольного ядра K_1 и произвольной функции $f : \mathbb{R} \rightarrow \mathbb{R}$, представимой в виде сходящегося степенного ряда с неотрицательными коэффициентами $K(x_1, x_2) = f(K_1(x_1, x_2))$ – ядро



Изначально наша двойственная задача была сформулирована в терминах линейного ядра:

$$\begin{cases} -Q(\lambda) = \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m \lambda_i \lambda_j y_i y_j \langle x_i, x_j \rangle - \sum_{i=1}^m \lambda_i \rightarrow \min_{\lambda}, \\ 0 \leq \lambda_i \leq C, \\ \sum_{i=1}^m \lambda_i y_i = 0. \end{cases}$$



SVM с другими ядрами

Изначально наша двойственная задача была сформулирована в терминах линейного ядра:

$$\begin{cases} -Q(\lambda) = \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m \lambda_i \lambda_j y_i y_j \langle x_i, x_j \rangle - \sum_{i=1}^m \lambda_i \rightarrow \min_{\lambda}, \\ 0 \leq \lambda_i \leq C, \\ \sum_{i=1}^m \lambda_i y_i = 0. \end{cases}$$

Когда мы меняем ядро с $\langle x_i, x_j \rangle$ на $K(x_i, x_j)$, задача преобразуется в:

$$\begin{cases} -Q(\lambda) = \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m \lambda_i \lambda_j y_i y_j K(x_i, x_j) - \sum_{i=1}^m \lambda_i \rightarrow \min_{\lambda}, \\ 0 \leq \lambda_i \leq C, \\ \sum_{i=1}^m \lambda_i y_i = 0. \end{cases}$$



Изначально наша двойственная задача была сформулирована в терминах линейного ядра:

$$\begin{cases} -Q(\lambda) = \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m \lambda_i \lambda_j y_i y_j \langle x_i, x_j \rangle - \sum_{i=1}^m \lambda_i \rightarrow \min_{\lambda}, \\ 0 \leq \lambda_i \leq C, \\ \sum_{i=1}^m \lambda_i y_i = 0. \end{cases}$$

Когда мы меняем ядро с $\langle x_i, x_j \rangle$ на $K(x_i, x_j)$, задача преобразуется в:

$$\begin{cases} -Q(\lambda) = \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m \lambda_i \lambda_j y_i y_j K(x_i, x_j) - \sum_{i=1}^m \lambda_i \rightarrow \min_{\lambda}, \\ 0 \leq \lambda_i \leq C, \\ \sum_{i=1}^m \lambda_i y_i = 0. \end{cases}$$

При этом сам линейный классификатор принимает вид (x_j - опорный вектор на границе):

$$a(x) = \text{sign}\left(\sum_{i=1}^m \lambda_i y_i K(x_i, x) - w_0\right), w_0 = \sum_{i=1}^m \lambda_i y_i K(x_i, x_j) - y_j$$



Пример нахождения пространства H

Пусть $X = \mathbb{R}^2$, $K(u, v) = \langle u, v \rangle^2$ при $u = (u_1, u_2)$, $v = (v_1, v_2)$. Хотим найти H и $\varphi : X \rightarrow H$, т.ч. $K(x_1, x_2) = \langle \varphi(x_1), \varphi(x_2) \rangle$.



Пример нахождения пространства H

Пусть $X = \mathbb{R}^2$, $K(u, v) = \langle u, v \rangle^2$ при $u = (u_1, u_2)$, $v = (v_1, v_2)$.

Хотим найти H и $\varphi : X \rightarrow H$, т.ч. $K(x_1, x_2) = \langle \varphi(x_1), \varphi(x_2) \rangle$.

Сделаем эквивалентные преобразования:

$$K(u, v) = \langle u, v \rangle^2 = \langle (u_1, u_2), (v_1, v_2) \rangle^2 = (u_1 v_1 + u_2 v_2)^2 = u_1^2 v_1^2 + u_2^2 v_2^2 + 2u_1 v_1 u_2 v_2$$



Пример нахождения пространства H

Пусть $X = \mathbb{R}^2$, $K(u, v) = \langle u, v \rangle^2$ при $u = (u_1, u_2)$, $v = (v_1, v_2)$.

Хотим найти H и $\varphi : X \rightarrow H$, т.ч. $K(x_1, x_2) = \langle \varphi(x_1), \varphi(x_2) \rangle$.

Сделаем эквивалентные преобразования:

$$K(u, v) = \langle u, v \rangle^2 = \langle (u_1, u_2), (v_1, v_2) \rangle^2 = (u_1 v_1 + u_2 v_2)^2 = u_1^2 v_1^2 + u_2^2 v_2^2 + 2u_1 v_1 u_2 v_2$$

$$K(u, v) = \langle (u_1^2, u_2^2, \sqrt{2}u_1 u_2), (v_1^2, v_2^2, \sqrt{2}v_1 v_2) \rangle$$

Пример нахождения пространства H

Пусть $X = \mathbb{R}^2$, $K(u, v) = \langle u, v \rangle^2$ при $u = (u_1, u_2)$, $v = (v_1, v_2)$.

Хотим найти H и $\varphi : X \rightarrow H$, т.ч. $K(x_1, x_2) = \langle \varphi(x_1), \varphi(x_2) \rangle$.

Сделаем эквивалентные преобразования:

$$K(u, v) = \langle u, v \rangle^2 = \langle (u_1, u_2), (v_1, v_2) \rangle^2 = (u_1 v_1 + u_2 v_2)^2 = u_1^2 v_1^2 + u_2^2 v_2^2 + 2u_1 v_1 u_2 v_2$$

$$K(u, v) = \langle (u_1^2, u_2^2, \sqrt{2}u_1 u_2), (v_1^2, v_2^2, \sqrt{2}v_1 v_2) \rangle$$

$$\text{Т.о., } H = \mathbb{R}^3 \text{ и } \varphi : (u_1, u_2) \mapsto (u_1^2, u_2^2, \sqrt{2}u_1 u_2).$$



Пример нахождения пространства H

Пусть $X = \mathbb{R}^2$, $K(u, v) = \langle u, v \rangle^2$ при $u = (u_1, u_2)$, $v = (v_1, v_2)$.

Хотим найти H и $\varphi : X \rightarrow H$, т.ч. $K(x_1, x_2) = \langle \varphi(x_1), \varphi(x_2) \rangle$.

Сделаем эквивалентные преобразования:

$$K(u, v) = \langle u, v \rangle^2 = \langle (u_1, u_2), (v_1, v_2) \rangle^2 = (u_1 v_1 + u_2 v_2)^2 = u_1^2 v_1^2 + u_2^2 v_2^2 + 2u_1 v_1 u_2 v_2$$

$$K(u, v) = \langle (u_1^2, u_2^2, \sqrt{2}u_1 u_2), (v_1^2, v_2^2, \sqrt{2}v_1 v_2) \rangle$$

$$\text{Т.о., } H = \mathbb{R}^3 \text{ и } \varphi : (u_1, u_2) \mapsto (u_1^2, u_2^2, \sqrt{2}u_1 u_2).$$

Линейной поверхности в H будет соответствовать квадратичная поверхность в X .



- Полиномиальное ядро с мономами степени d : $K(x_1, x_2) = \langle x_1, x_2 \rangle^d$

Примеры ядер для SVM

- Полиномиальное ядро с мономами степени d : $K(x_1, x_2) = \langle x_1, x_2 \rangle^d$
- Полиномиальное ядро с мономами степени $\leq d$: $K(x_1, x_2) = (\langle x_1, x_2 \rangle + 1)^d$



Примеры ядер для SVM

- Полиномиальное ядро с мономами степени d : $K(x_1, x_2) = \langle x_1, x_2 \rangle^d$
- Полиномиальное ядро с мономами степени $\leq d$: $K(x_1, x_2) = (\langle x_1, x_2 \rangle + 1)^d$
- Радиальное ядро (RBF): $K(x_1, x_2) = \exp(-\gamma \|x_1 - x_2\|^2)$ (наиболее универсальное)

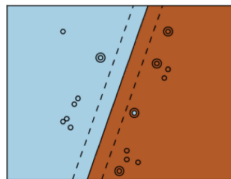


Примеры ядер для SVM

- Полиномиальное ядро с мономами степени d : $K(x_1, x_2) = \langle x_1, x_2 \rangle^d$
- Полиномиальное ядро с мономами степени $\leq d$: $K(x_1, x_2) = (\langle x_1, x_2 \rangle + 1)^d$
- Радиальное ядро (RBF): $K(x_1, x_2) = \exp(-\gamma \|x_1 - x_2\|^2)$ (наиболее универсальное)

Линейное ядро

$$K(x_1, x_2) = \langle x_1, x_2 \rangle$$

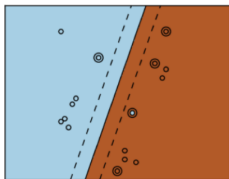


Примеры ядер для SVM

- Полиномиальное ядро с мономами степени d : $K(x_1, x_2) = \langle x_1, x_2 \rangle^d$
- Полиномиальное ядро с мономами степени $\leq d$: $K(x_1, x_2) = (\langle x_1, x_2 \rangle + 1)^d$
- Радиальное ядро (RBF): $K(x_1, x_2) = \exp(-\gamma \|x_1 - x_2\|^2)$ (наиболее универсальное)

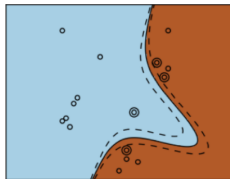
Линейное ядро

$$K(x_1, x_2) = \langle x_1, x_2 \rangle$$



Полиномиальное ядро

$$K(x_1, x_2) = (\langle x_1, x_2 \rangle + 1)^3$$

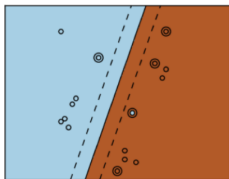


Примеры ядер для SVM

- Полиномиальное ядро с мономами степени d : $K(x_1, x_2) = \langle x_1, x_2 \rangle^d$
- Полиномиальное ядро с мономами степени $\leq d$: $K(x_1, x_2) = (\langle x_1, x_2 \rangle + 1)^d$
- Радиальное ядро (RBF): $K(x_1, x_2) = \exp(-\gamma \|x_1 - x_2\|^2)$ (наиболее универсальное)

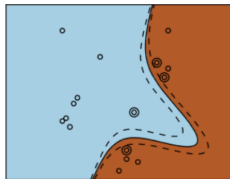
Линейное ядро

$$K(x_1, x_2) = \langle x_1, x_2 \rangle$$



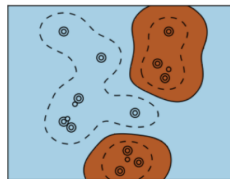
Полиномиальное ядро

$$K(x_1, x_2) = (\langle x_1, x_2 \rangle + 1)^3$$



Радиальное ядро

$$K(x_1, x_2) = \exp(-\|x_1 - x_2\|^2)$$



Предположим, что нужно построить классификатор методом опорных векторов для задачи классификации с количеством классов $|Y| = N > 2$. Тогда возможны 2 варианта:

Предположим, что нужно построить классификатор методом опорных векторов для задачи классификации с количеством классов $|Y| = N > 2$. Тогда возможны 2 варианта:

Стратегия “один-против-всех”

Обучаем N бинарных SVM-классификаторов, каждый из которых отделяет некоторый класс от остальных $N - 1$ классов.

Затем в качестве класса берем

$$a(x, w, w_0) = \arg \max_{c \in Y} (\langle w^c, x \rangle - w_0^c)$$



Многоклассовый SVM

Предположим, что нужно построить классификатор методом опорных векторов для задачи классификации с количеством классов $|Y| = N > 2$. Тогда возможны 2 варианта:

Стратегия “один-против-всех”

Обучаем N бинарных SVM-классификаторов, каждый из которых отделяет некоторый класс от остальных $N - 1$ классов.

Затем в качестве класса берем

$$a(x, w, w_0) = \arg \max_{c \in Y} (\langle w^c, x \rangle - w_0^c)$$

Стратегия “каждый-против-каждого”

Обучаем $\frac{N(N-1)}{2}$ бинарных SVM-классификаторов, каждый из которых отделяет между собой некоторую пару классов.

В результате применения классификаторов получаем $\frac{N(N-1)}{2}$ доминирующих классов. Итоговый класс выбирается большинством голосов.



Плюсы

- Наглядная оптимизационная модель
- Задача имеет единственное решение
- Легко обобщается для нелинейной классификации

Плюсы и минусы SVM

Плюсы

- Наглядная оптимизационная модель
- Задача имеет единственное решение
- Легко обобщается для нелинейной классификации

Минусы

- Непонятно, как подбирать ядро в конкретном случае
- Подбор константы C
- Решение задачи квадратичного программирования, особенно с экзотическими ядрами, может занять много времени

