Машинное обучение

Лекция 5. Линейные методы классификации.

Катя Тузова

Разбор летучки

Какие проблемы могут возникнуть при использовании метода k-means? Как их избежать?

Разбор летучки

– Чувствительность к начальному выбору μ_c
– Необходимость задавать k

Разбор летучки

В чем заключается идея алгоритма Xmeans?

Разбор летучки

Идея:

- Получать на вход не k, а диапазон, в котором может находиться k.
- Запустить k-means на самом маленьком значении из диапазона.
- Разбивать пополам полученные кластеры и проверять, не улучшилась ли кластеризация.

Разбор летучки

- Несколько случайных кластеризаций
- Постепенное наращивание числа k
- Использование k-means++

Разбор летучки

Как можно использовать минимальное остовное дерево для разбиения выборки на кластеры? Как найти минимальное остовное дерево?

Разбор летучки

Построить минимальное остовное дерево, а потом выкидывать из него ребра максимального веса.

Сколько ребер выбросим – столько кластеров получим.

Разбор летучки

Для чего требуется свойство монотонности при иерархической кластеризации?

Разбор летучки

Чтобы избежать самопересечений в постоенном дереве.

Разбор летучки

Каким образом выбираются центры кластеров в алгоритме k-means++?

Разбор летучки

- Выбрать первый центроид случайным образом
- Для каждой точки найти значение квадрата расстояния до ближайшего центроида.
- Выбрать из этих точек следующий центроид так, чтобы вероятность выбора точки была пропорциональна вычисленному для неё квадрату расстояния

Разбор летучки

Что такое отступ обобщенного метрического классификатора?

Разбор летучки

Отступ показывает степень "типичности объекта".

Отступом объекта $x_i \in X^l$ относительно классификатора a называется величина
$M(x_i) = \Gamma_{y_i}(x_i) - \max\limits_{y \in Y \setminus y_i} \Gamma_y(x_i)$

Разбор летучки

$\Gamma_y(u)$ - оценка близости объекта u к классу y

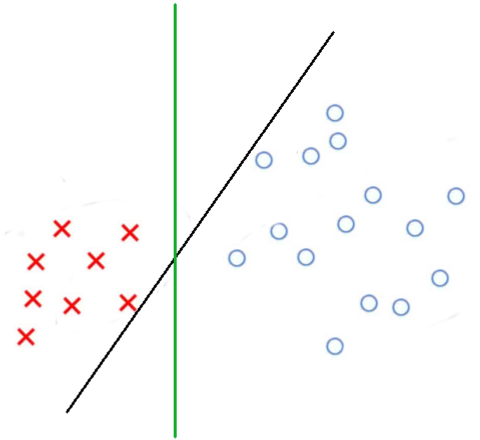
Постановка задачи

$X = \mathbb{R}^n$, $Y = \{-1, +1\}$
$X^l = (x_i, y_i)_{i=1}^l$ – обучающая выборка

Найти:
$(n-1)$-мерную гиперплоскость, которая разделяет данные как можно лучше.

Как можно лучше – это как?

Пример



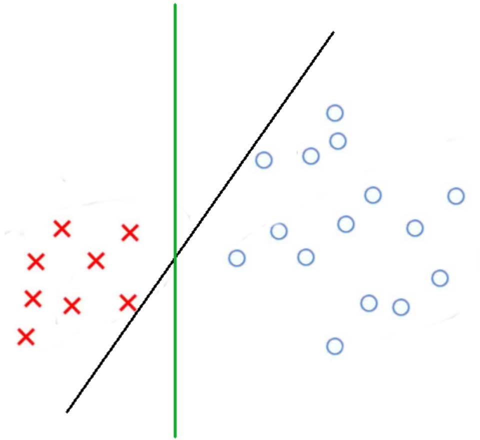
Постановка задачи

Как можно лучше:
Два разделенных класса должны лежать как можно дальше от разделяющей гиперплоскости.

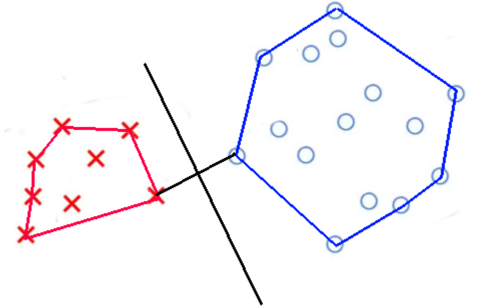
Вопрос

Как построить такую гиперплоскость?

Пример



Выпуклая оболочка

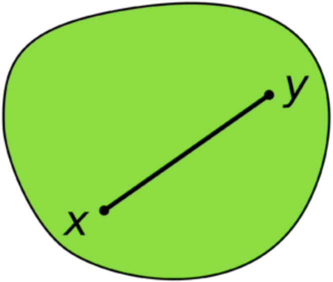


Выпуклая оболочка

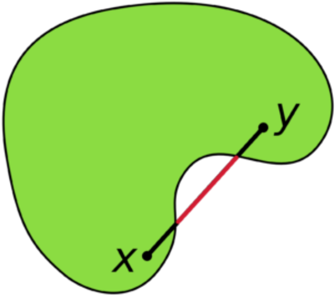
Выпуклой оболочкой множества X называется наименьшее выпуклое множество, содержащее X.

Выпуклое множество — множество, содержащее вместе с любыми двумя точками соединяющий их отрезок.

Выпуклая оболочка



Выпуклая оболочка



Невыпуклая оболочка

Выпуклая оболочка

Найти две ближайшие точки в выпуклых оболочках данных, а затем провести разделяющую гиперплоскость через середину отрезка.

Выпуклая оболочка

$\min\limits_w \|c - d\|^2$, где $c = \sum\limits_{y_i=1} w_i x_i$, $d = \sum\limits_{y_i=-1} w_i x_i$

$$\sum_{y_i=1} w_i = \sum_{y_i=-1} w_i = 1,\ w_i \geq 0$$

Задачу можно решать общими оптимизационными алгоритмами.

Построение разделяющей поверхности

Построение разделяющей поверхности

$X^l = (x_i, y_i)_{i=1}^l$ – обучающая выборка
$Y = \{-1, +1\}$

Задача:
Построить алгоритм классификации $a(x, w) = \operatorname{sign} g(x, \theta)$

$g(x, \theta)$ – разделяющая функция
θ – вектор параметров, $\theta \in \mathbb{R}^l$

Линейный классификатор

$f_j : X \to \mathbb{R}$, $j = 1, \dots, n \qquad w_j \in \mathbb{R}$
$g(x, w) = w_j f_j(x) - w_0$

$$a(x, w) = sign(\sum_{j=1}^{n} w_j f_j(x) - w_0)$$

Введём константный признак $f_0 \equiv -1$:

$$a(x, w) = sign(\sum_{j=0}^{n} w_j f_j(x))$$

Линейный классификатор

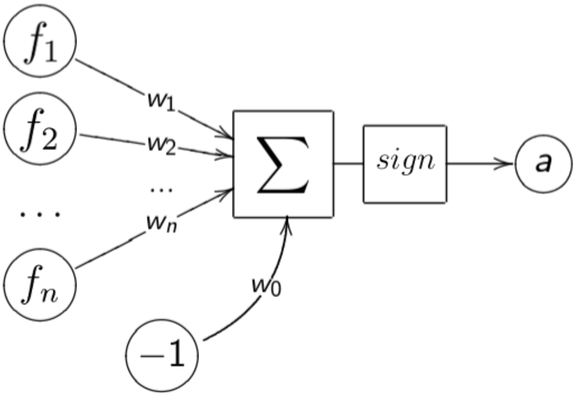
$$a(x, w) = sign(\sum_{j=1}^{n} w_j f_j(x))$$

Перейдём к векторной записи:

$\mathbf{x} = (f_1, f_2, \ldots, f_n)$

$$a(\mathbf{x}, \mathbf{w}) = sign(\langle \mathbf{w}, \mathbf{x} \rangle)$$

Линейный классификатор



Линейный классификатор

Как подбирать w_j?

Построение разделяющей поверхности

Как оценить качество классификации?

Определение отступа

$g = \langle \mathbf{w}, \mathbf{x} \rangle = 0$ – разделяющая поверхность
$M_i(\mathbf{w}) = \langle \mathbf{w}, \mathbf{x_i} \rangle y_i$ – отступ объекта x_i
$M_i(\mathbf{w}) < 0 \Rightarrow$ алгоритм $a(\mathbf{x}, \mathbf{w})$ ошибается на x_i

Минимизация эмпирического риска

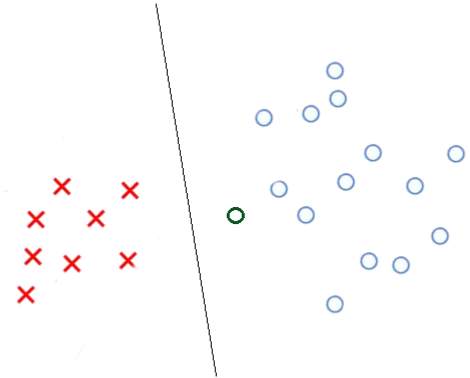
Число ошибок на обучающей выборке:

$$Q(\mathbf{w}) = \sum_{i=1}^{l} [M_i(\mathbf{w}) < 0] \to \min_{\mathbf{w}}$$

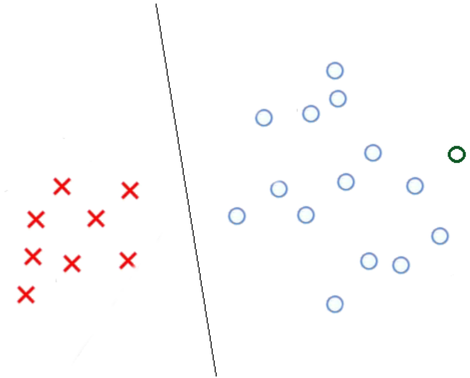
$Q(\mathbf{w})$ – функционал качества

- Индикаторную функцию сложно оптимизировать
- Теряем информацию насколько i-й объект был надежен

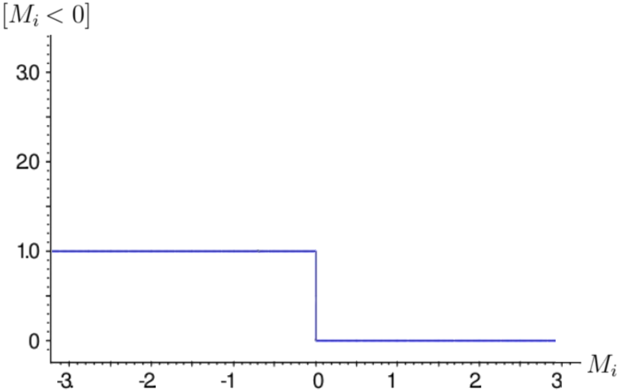
Отступ



Отступ



Функция $[M < 0]$



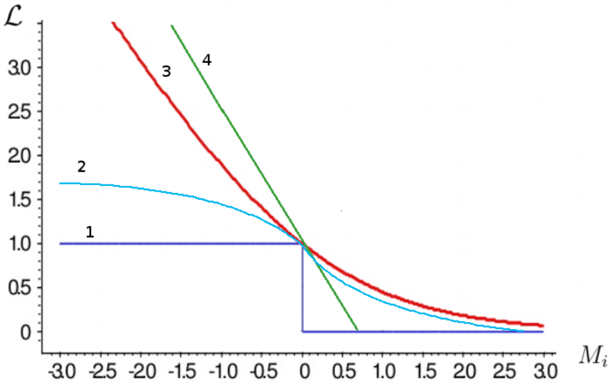
Минимизация эмпирического риска

$$Q(\mathbf{w}) = \sum_{i=1}^{l} [M_i(\mathbf{w}) < 0] \leq$$

$$\leq \sum_{i=1}^{l} \mathcal{L}(M_i(\mathbf{w})) \to \min_{\mathbf{w}}$$

$\mathcal{L}$ – функция потерь, невозрастающая, неотрицательная.
$\mathcal{L}$ должна мажорировать $[M_i(\mathbf{w}) < 0]$

Примеры $\mathcal{L}$

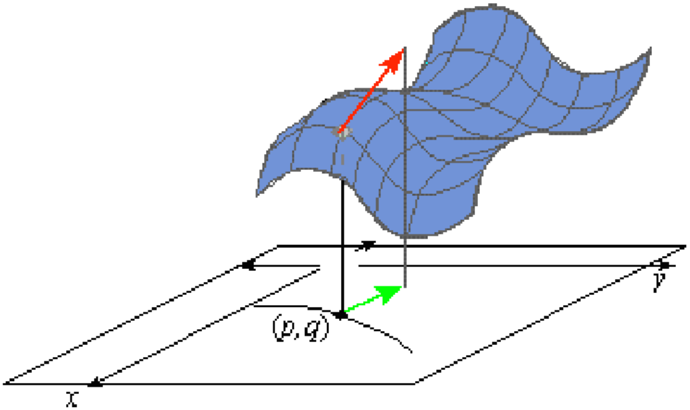


1. $[M_i(\mathbf{w}) < 0]$
2. $L(M) = \log_2(1 + e^{-M})$ – логарифмическая
3. $S(M) = 2(1 + e^{M})^{-1}$ – сигмоидная
4. $V(M) = (1 - M)_+$ – кусочно-линейная

Градиент

Что такое градиент?

Градиент



Метод градиентного спуска

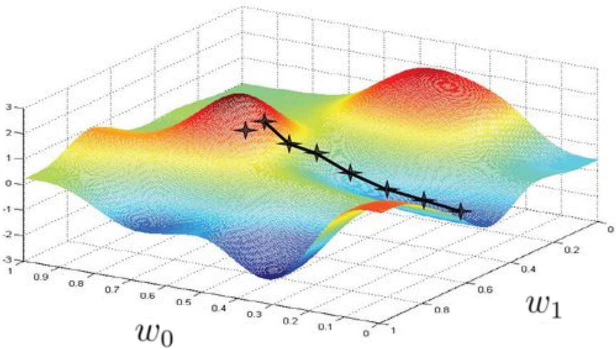
Input: α – градиентный шаг (темп обучения)
Output: $w_0, w_1, \ldots, w_n$

Инициализировать: w_j, $j = 0, \ldots, n$
Повторить пока $\mathbf{w}$ не стабилизируются:

$$\mathbf{w} = \mathbf{w} - \alpha \bigtriangledown Q(\mathbf{w})$$

$$\bigtriangledown Q(\mathbf{w}) = (\tfrac{\partial Q(\mathbf{w})}{\partial w_j})_{j=0}^{n}$$

Градиентный спуск



Метод градиентного спуска

Случай двух признаков.
Input: α – градиентный шаг (темп обучения)
Output: w_0, w_1

Инициализировать: w_0, w_1
Повторить пока w_0 и w_1 не стабилизируются:

$$tmp_0 = w_0 - \alpha \frac{\partial Q(w)}{\partial w_0}$$
$$tmp_1 = w_1 - \alpha \frac{\partial Q(w)}{\partial w_1}$$
$$w_0 = tmp_0$$
$$w_1 = tmp_1$$

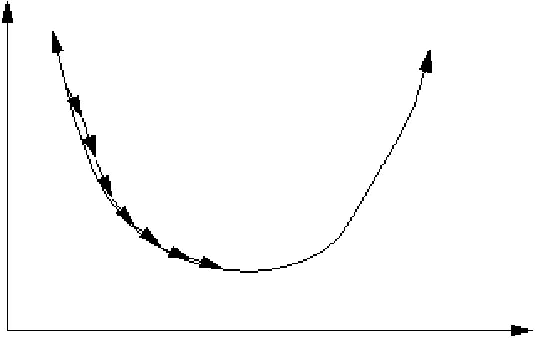
Метод градиентного спуска

Почему важно обновить w_0 и w_1 одновременно?

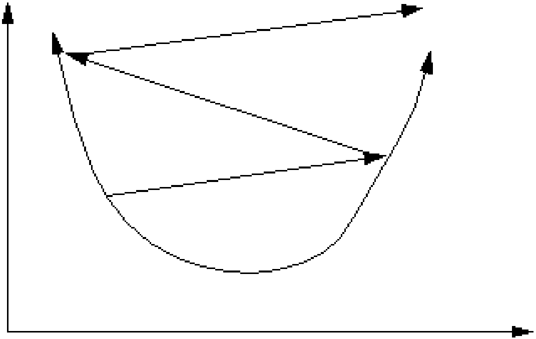
Градиент функционала качества Q

$$\bigtriangledown Q(\mathbf{w}) = (\tfrac{\partial Q(\mathbf{w})}{\partial w_j})_{j=0}^{n} = \sum_{i=1}^{l} \mathcal{L}'(\langle \mathbf{w}, \mathbf{x_i} \rangle y_i) \mathbf{x_i} y_i$$

Маленький градиентный шаг



Большой градиентный шаг



Выбор величины шага

- $\alpha_t \to 0$
- Метод скорейшего градиентного спуска:
  $Q(w - \alpha \bigtriangledown Q(w)) \to \min\limits_{\alpha}$
- Пробные случайные шаги

В чем проблема?

В чем проблема?

$$\mathbf{w} = \mathbf{w} - \alpha \sum_{i=1}^{l} \mathcal{L}'(\langle \mathbf{w}, \mathbf{x_i} \rangle y_i) \mathbf{x_i} y_i$$

Метод стохастического градиента

$$\mathbf{w} = \mathbf{w} - \alpha \sum_{i=1}^{l} \mathcal{L}'(\langle \mathbf{w}, \mathbf{x_i} \rangle y_i)\mathbf{x_i} y_i$$

Давайте брать (x_i, y_i) по одному и сразу обновлять вектор весов

Алгоритм

Input: X^l, α, η
Output: $w_0, w_1, \ldots, w_n$

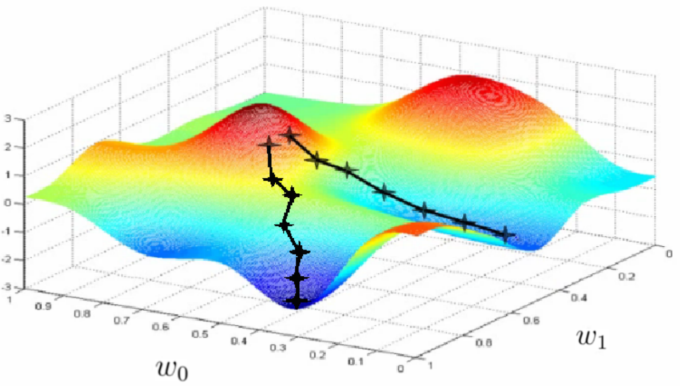
Перемешать данные в X^l
Инициализировать: w_j, $j = 0, \ldots, n$

$$Q(\mathbf{w}) = \sum_{i=1}^{l} \mathcal{L}(\langle \mathbf{w}, \mathbf{x_i} \rangle y_i)$$

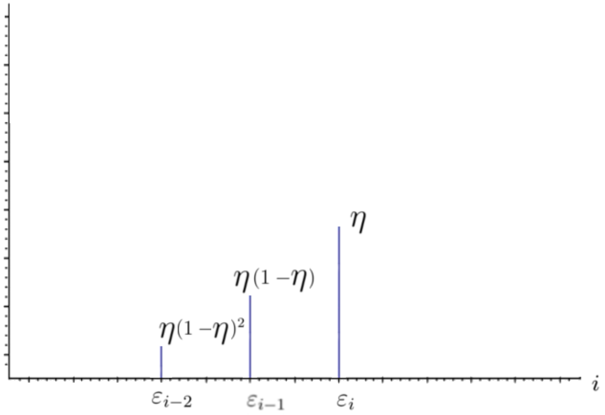
Повторить пока Q и/или w не стабилизируются:

Взять x_i из X^l
Потеря: $\varepsilon_i = \mathcal{L}(\langle \mathbf{w}, \mathbf{x_i} \rangle y_i)$
Градиентный шаг: $w = w - \alpha \mathcal{L}'(\langle \mathbf{w}, \mathbf{x_i} \rangle y_i) \mathbf{x_i} y_i$
Оценить $Q = (1 - \eta) Q + \eta \varepsilon_i$

Градиентный спуск



Учет ошибки ε_i в алгоритме



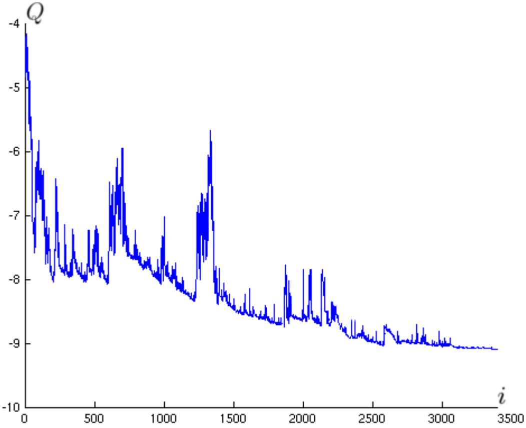
Вопрос

Чего не хватает?

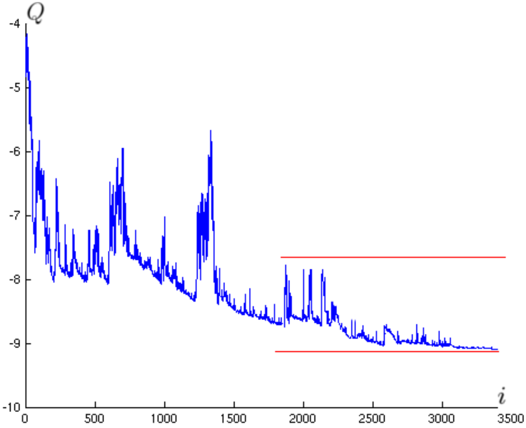
Вопрос

Что значит – "пока Q и/или w не стабилизируются"?

Зависимость Q от номера итерации



Зависимость Q от номера итерации



Актуальные вопросы

- Инициализация весов
- Порядок предъявления объектов

Инициализация весов

Инициализация весов

- $w_j = 0$, j = 1, . . . , n
- $w_j = random(-\frac{1}{2n}, \frac{1}{2n})$ – небольшие случайные значения
- Обучение по небольшой случайной подвыборке объектов
- Многократный запуск из разных случайных начальных приближений

Порядок предъявления объектов

Порядок предъявления объектов

- Попеременно брать объекты из разных классов
- Чаще брать те объекты, на которых была допущена большая ошибка
- Вообще не брать "хорошие"объекты с $M_i > \mu_+$
- Вообще не брать выбросы с $M_i < \mu_-$

Плюсы и минусы

+ Легко реализовать
+ Легко обобщить на разные g, $\mathcal{L}$
+ Не обязательно брать все элементы выборки для обучения

- Возможна медленная сходимость
- Застревание в локальных минимумах
- Подбор параметров
- Проблема переобучения

Проблема переобучения

Почему случается переобучение?

Проблема переобучения

$a(\mathbf{x}, \mathbf{w}) = sign(\langle \mathbf{w}, \mathbf{x} \rangle)$
Линейная зависимость признаков:
$\forall \mathbf{x} \exists \mathbf{u} : \langle \mathbf{u}, \mathbf{x} \rangle = 0$
$\Rightarrow \forall \gamma : a(\mathbf{x}, \mathbf{w}) = sign(\langle \mathbf{w} + \gamma \mathbf{u}, \mathbf{x} \rangle)$

Алгоритм a' работает точно также как исходный a.
А значит мы можем получить любое решение из семейства $\mathbf{w} + \gamma \mathbf{u}$

Проблема переобучения

- Слишком мало объектов
- Слишком много признаков
- Линейная зависимость признаков

Как заподозрить?

- Слишком большие веса $\|\mathbf{w}\|$
- Неустойчивость $a(\mathbf{x}, \mathbf{w})$
- Плохое качество классификации на контрольных данных

Решение

- Сокращение весов
- Ранняя остановка алгоритма

Сокращение весов

Как можно сокращать веса?

Сокращение весов

Штраф за увеличение нормы вектора весов:
$Q_\tau = Q + \frac{\tau}{2}\|\mathbf{w}\|^2 \to \min\limits_{\mathbf{w}}$

Градиент:
$\triangledown Q_\tau = \triangledown Q + \tau \mathbf{w}$

Градиентный шаг:
$\mathbf{w} = \mathbf{w}(1 - \alpha\tau) - \alpha \triangledown Q(\mathbf{w})$
τ – параметр регуляризации

Вопрос

Как подобрать τ?

Как подобрать τ?

- Скользящий контроль

Степени свободы

Степени свободы

- Вид разделяющей поверхности
- Вид аппроксимации функционала качества $Q(\mathbf{w})$
- Вид регуляризатора

Резюме лекции

- Построение выпуклых оболочек
- Определение линейного классификатора
- Минимизация эмпирического риска
- Метод градиентного спуска
- Метод стохастического градиентного спуска

На следующей лекции

– Метод опорных векторов