ML Homework-Theory

isadrtdinov

November 2019

1 Linear Regression

1. Как известно, след матрицы не зависит от базиса, в котором матрица записана. При этом в жордановом базисе след матрицы равен сумме ее собственных значений. Таким образом:

$$\frac{\partial}{\partial A}\sum_{i=1}^{n}\lambda_i = \frac{\partial}{\partial A}\operatorname{Tr}(A) = E$$

2. $$\frac{\partial}{\partial A}\log\det A = \frac{1}{\det A}\frac{\partial}{\partial A}\det A = \frac{1}{\det A}\det A\left(A^{-1}\right)^T = \left(A^{-1}\right)^T$$

3. $$\frac{\partial}{\partial a}\Big(a^T\exp(aa^T)a\Big) = \frac{\partial}{\partial a}\left(a^T\left(\sum_{k=0}^{\infty}\frac{\left(aa^T\right)^k}{k!}\right)a\right) =$$
$$= \frac{\partial}{\partial a}\left(\sum_{k=0}^{\infty}\frac{\left(a^Ta\right)^{k+1}}{k!}\right) = \frac{\partial}{\partial a}\Big((a^Ta)\exp(a^Ta)\Big) =$$
$$= \frac{\partial a^Ta}{\partial a}\exp(a^Ta) + a^Ta\frac{\partial\exp(a^Ta)}{\partial a} = 2a\exp(a^Ta) + a^Ta\exp(a^Ta)2a =$$
$$= 2a\exp(a^Ta)(1 + a^Ta)$$

4. Для начала вспомним, как записывается градиент для MSE:

$$\nabla_w Q(w) = 2X^T(Xw - y)$$

Тогда мы хотим минимизировать по α функционал:

$$Q(w^{(k-1)} - \alpha\nabla_w Q(w^{(k-1)})) = Q(w^{(k)}) = (Xw^{(k)} - y)^T(Xw^{(k)} - y)$$

Дифференцируем по α учитывая, что только $w^{(k)}$ зависит от α:

$$\frac{\partial w^{(k)}}{\partial\alpha} = \frac{\partial\Big(w^{(k-1)} - \alpha\nabla_w Q(w^{(k-1)})\Big)}{\partial\alpha} = -\nabla_w Q(w^{(k-1)})$$

$$\frac{\partial Q(w^{(k)})}{\partial \alpha} = \frac{\partial}{\partial \alpha}\left({(w^{(k)})}^T X^T X w^{(k)} - {(w^{(k)})}^T X^T y - y^T X w^{(k)} - y^T y\right) =$$
$$= \frac{\partial}{\partial \alpha}\left({(w^{(k)})}^T X^T X w^{(k)}\right) + \nabla_w Q{(w^{(k-1)})}^T X^T y + y^T X \nabla_w Q(w^{(k-1)}) =$$

$$= \frac{\partial}{\partial \alpha}\left({(w^{(k-1)})}^T - \alpha \nabla_w Q{(w^{(k-1)})}^T\right) X^T X \left(w^{(k-1)} - \alpha \nabla_w Q(w^{(k-1)})\right) +$$
$$+\nabla_w Q{(w^{(k-1)})}^T X^T y + y^T X \nabla_w Q(w^{(k-1)}) =$$

$$= 2\alpha \nabla_w Q{(w^{(k-1)})}^T X^T X \nabla_w Q(w^{(k-1)}) - \nabla_w Q{(w^{(k-1)})}^T X^T X w^{(k-1)} -$$
$$-{(w^{(k-1)})}^T X^T X \nabla_w Q(w^{(k-1)}) + \nabla_w Q{(w^{(k-1)})}^T X^T y + y^T X \nabla_w Q(w^{(k-1)})$$

Отсюда получаем единственный нуль производной (обратите внимание, что в знаменателе записано число, так что на него можно разделить):

$$\alpha = \frac{\nabla_w Q{(w^{(k-1)})}^T X^T (X w^{(k-1)} - y) + ({(w^{(k-1)})}^T X^T - y^T) X \nabla_w Q(w^{(k-1)})}{2 \nabla_w Q{(w^{(k-1)})}^T X^T X \nabla_w Q(w^{(k-1)})}$$

$$\alpha = \frac{4({(w^{(k-1)})}^T X^T - y^T) X X^T (X w^{(k-1)} - y)}{2 \nabla_w Q{(w^{(k-1)})}^T X^T X \nabla_w Q(w^{(k-1)})}$$

$$\alpha = \frac{({(w^{(k-1)})}^T X^T - y^T) X X^T (X w^{(k-1)} - y)}{2({(w^{(k-1)})}^T X^T - y^T) X X^T X X^T (X w^{(k-1)} - y)}$$

Поскольку $\frac{\partial Q(w^{(k)})}{\partial \alpha} = a\alpha + b$, где $a > 0$ (так как старший коэффициент - это норма вектора $X \nabla_w Q(w^{(k-1)})$), то исходный функционал - это парабола с ветвями вверх (по α), то полученный нуль производной - это единственная точка минимума функционала.

5. Покажем, что минимум квантильной регрессии достигается при $C =$ τ-квантиль выборки y, обозначим его за q_τ. По аналогии задачи с минимумом MAE с семинара рассмотрим три случая (для случая $C < q_\tau$, обратный рассматривается аналогично):

$$\rho_\tau(y_i - C) - \rho_\tau(y_i - q_\tau) = \begin{cases} (\tau - 1)(y_i - C) - (\tau - 1)(y_i - q_\tau), & y_i < C \\ \tau(y_i - C) - (\tau - 1)(y_i - q_\tau), & C \le y_i \le q_\tau \\ \tau(y_i - C) - \tau(y_i - q_\tau), & q_\tau < y_i \end{cases}$$

$$\rho_\tau(y_i - C) - \rho_\tau(y_i - q_\tau) = \begin{cases} \tau(q_\tau - C) - (q_\tau - C), & y_i < C \\ \tau(q_\tau - C) - (q_\tau - y_i), & C \le y_i \le q_\tau \\ \tau(q_\tau - C), & q_\tau < y_i \end{cases}$$

$$\rho_\tau(y_i - C) - \rho_\tau(y_i - q_\tau) \ge \tau(q_\tau - C) - (q_\tau - C)\,[y_i \le q_\tau]$$

Просуммируем по всей выборке y (здесь Q_τ - это средняя ошибка квантильной регрессии):

$$lQ_\tau(C) - lQ_\tau(q_\tau) \ge l\tau(q_\tau - C) - (q_\tau - C)\sum_{i=1}^{l}[y_i \le q_\tau] \ge$$
$$\ge l\tau(q_\tau - C) - (q_\tau - C)l\tau = 0$$

Таким образом, q_τ - оптимальная константа функционала Q_τ

6. Распишем ошибку алгоритма на одном объекте:

$$L(a(x_i), y_i) = \left(y_i - \frac{\sum_{j \in B_{r(x_i)} \setminus \{i\}} y_j}{n_{r(x_i)} - 1}\right)^2 \left[n_{r(x_i)} > 1\right] + y_i^2 \left[n_{r(x_i)} = 1\right] =$$
$$= \left(\frac{y_i(n_{r(x_i)} - 1) - ((\sum_{j \in B_{r(x_i)}} y_j) - y_i)}{n_{r(x_i)} - 1}\right)^2 \left[n_{r(x_i)} > 1\right] + y_i^2 \left[n_{r(x_i)} = 1\right] =$$
$$= \left(\frac{n_{r(x_i)}\left(y_i - \overline{y}_{r(x_i)}\right)}{n_{r(x_i)} - 1}\right)^2 \left[n_{r(x_i)} > 1\right] + y_i^2 \left[n_{r(x_i)} = 1\right]$$

Здесь под $n_{r(x_i)}$ имеется в виду общее число объектов выборки в ячейке $r(x_i)$ (включая x_i), а под $\overline{y}_{r(x_i)}$ - средний таргет по соответсвующей ячейке. Очевидным образом, обе эти величины могут быть подсчитаны по всем ячейкам за $O(l)$. Эти значения фигурируют в финальной формуле, которая получается усреднением ошибки на одном объекте по всем объектам

$$L = \frac{1}{l}\sum_{i=1}^{l}\left(\left(\frac{n_{r(x_i)}\left(y_i - \overline{y}_{r(x_i)}\right)}{n_{r(x_i)} - 1}\right)^2 \left[n_{r(x_i)} > 1\right] + y_i^2 \left[n_{r(x_i)} = 1\right]\right)$$

Таким образом, вся формула вычисляется за $O(l)$.

2 Linear Classification

1.

2.

3.

4. Для начала представим ранги элементов в аналитическом виде. Ранг элемента - это число элементов, не больших данного (включая его самого). Для удобства обозначений будем считать, что индексация ведется по возрастанию ответов классификатора. Тогда можно записать следующее:

$$r_{(i)} = \sum_{j \le i} 1 = \sum_{j \le i} \left[y_{(j)} = -1 \right] + \sum_{j \le i} \left[y_{(j)} = +1 \right]$$

Теперь посчитаем U_+ статистику:

$$\begin{aligned}
U_+ = \sum_{i: y_{(i)} = +1} r_{(i)} - \frac{l_+(l_+ + 1)}{2} = \sum_{i=1}^{l} \left[y_{(i)} = +1 \right] r_{(i)} - \frac{l_+(l_+ + 1)}{2} = \\
= \sum_{i=1}^{l} \left(\left[y_{(i)} = +1 \right] \sum_{j < i} \left[y_{(j)} = -1 \right] \right) + \\
+ \sum_{i=1}^{l} \left(\left[y_{(i)} = +1 \right] \sum_{j \le i} \left[y_{(j)} = +1 \right] \right) - \frac{l_+(l_+ + 1)}{2} = \\
= \sum_{j < i} \left[y_{(j)} < y_{(i)} \right] + \sum_{i=1}^{l_+} i - \frac{l_+(l_+ + 1)}{2} = \\
= \sum_{j < i} \left[y_{(j)} < y_{(i)} \right]
\end{aligned}$$

Таким образом, $\frac{U_+}{l_+ l_-} = \frac{\sum_{j<i} \left[y_{(j)} < y_{(i)} \right]}{l_+ l_-}$, а это и есть не что иное, как AUC-ROC, то есть число пар объектов из разных классов, верно разделенных классификатором, к общему числу пар.

5. Посчитаем $\arg\min_{b \in \mathbb{R}} \mathbb{E} \left[L(y, b) | x \right]$:

$$\begin{aligned}
\mathbb{E} \left[L(y, b) | x \right] = p(y = +1|x) e^{-b} + (1 - p(y = +1|x)) e^{b} \\
\frac{\partial \mathbb{E} \left[L(y, b) | x \right]}{\partial b} = -p(y = +1|x) e^{-b} + (1 - p(y = +1|x)) e^{b} = 0 \\
(1 - p(y = +1|x)) e^{2b} = p(y = +1|x) \\
b = \frac{1}{2} \log \left(\frac{p(y = +1|x)}{1 - p(y = +1|x)} \right)
\end{aligned}$$

Очевидно, что это минимум функционала, так как:

$$\lim_{|b|\to\infty} \mathbb{E}\left[L(y,b)|x\right] = +\infty,$$

а полученный экстремум единственный. Мы получили функцию, которая не равна тождественно $p(y=+1|x)$. Таким образом, экспоненциальная функция потерь не способна предсказывать истинные вероятности.

6. Пусть w_1, b_1 - оптимальное решение первой оптимизационной задачи (из лекции известно, что оно существует и единственно). Покажем, что tw_1, tb_1 - это оптимальное решение второй задачи. Для начала проверим, что эти значения подходят под ограничения задачи:

$$y_i(< w_1, x_i > + b_1) \geq 1 \Rightarrow$$
$$\Rightarrow y_i(< tw_1, x_i > + tb_1) = ty_i(< w_1, x_i > + b_1) \geq t$$

Это действительно так. Предположим теперь, что это решение не является оптимальным, то есть:

$$\exists w_2 \in \mathbb{R}^d, b_2 \in \mathbb{R} : \begin{cases} ||w_2||^2 < ||tw_1||^2 \\ y_i(< w_2, x_i > + b_2) \geq t \end{cases}$$

Но тогда $\frac{w_2}{t}, \frac{b_2}{t}$ - это допустимое решение первой задачи, лучшее, чем оптимальное w_1, b_1:

$$\begin{cases} ||\frac{w_2}{t}||^2 < ||w_1||^2 \\ y_i(< \frac{w_2}{t}, x_i > + \frac{b_2}{t}) \geq 1 \end{cases}$$

Получили противоречие с оптимальностью решения w_1, b_1. Таким образом, tw_1, tb_1 - это оптимальное решение второй задачи, а поскольку коэффициенты разделяющих гиперплоскостей пропорциональны, то сами гиперплоскости совпадают.

7. Посчитаем производную сигмоиды:

$$\sigma'(z) = -\frac{1}{(1+e^{-z})^2}\left(-e^{-z}\right) = \frac{e^{-z}}{(1+e^{-z})^2} =$$
$$= \frac{1}{1+e^{-z}}\left(1 - \frac{1}{1+e^{-z}}\right) = \sigma(z)(1-\sigma(z))$$

Немного преобразуем функцию потерь и посчитаем градиент:

$$L(x,y,w) = \log\left(1 + \exp\left(-y < w, x >\right)\right) = -\log\left(\sigma(-y < w, x >)\right)$$

$$\frac{\partial L}{\partial w} = -\frac{1}{\sigma(-y < w, x >)}\sigma(-y < w, x >)) \left(1 - \sigma(-y < w, x >)\right) (-yx) =$$
$$= yx \left(1 - \sigma(-y < w, x >)\right)$$

3 Trees

1. (a) Для случая $L(y, c) = (y - c)^2$ в качестве критерия информативности мы получаем MSE, а как известно, среднее - это оптимальная константа для MSE. Таким образом:

$$c = \frac{1}{|R|} \sum_{(x_i, y_i) \in R} y_i$$

$$H(R) = \frac{1}{|R|} \sum_{(x_i, y_i) \in R} \left(y_i - \frac{\sum_{(x_i, y_i) \in R} y_i}{|R|} \right) = \mathbb{D}\left[y\right]$$

   (b) Далее, считая, что $c \in \mathbb{R}^K$, найдем оптимальный вектор для критерия информативности:

$$H(R, c) = \frac{1}{|R|} \sum_{(x_i, y_i) \in R} \sum_{k=1}^{K} \left(c_k - [y_i = k]\right)^2 =$$

$$= \frac{1}{|R|} \sum_{(x_i, y_i) \in R} \sum_{k=1}^{K} (c_k^2 - 2c_k \left[y_i = k\right] + [y_i = k]) =$$

$$= \frac{1}{|R|} \sum_{k=1}^{K} (|R| c_k^2 - 2c_k n_k + n_k) = \sum_{k=1}^{K} (c_k^2 - 2c_k p_k + p_k)$$

Здесь n_k - число объектов класса k, p_k - соответственно, их доля. Поскольку получилась сумма по всем классам, то прооптимизируем функционал отдельно по каждому c_k:

$$\frac{\partial H}{\partial c_k} = 2c_k - 2p_k = 0 \Rightarrow c_k = p_k$$

Очевидно, при $c = (p_1, ..., p_k)$ достигается минимум функционала. Подставим это значение и посчитаем критерий информативности:

$$H(R) = \sum_{k=1}^{K} (p_k^2 - 2p_k^2 + p_k) = \sum_{k=1}^{K} (p_k - p_k^2) = 1 - \sum_{k=1}^{K} p_k^2$$

Это и есть не что иное, как критерий Джини.

   (c) В условии забыто очень важное ограничение на c: $\sum_{k=1}^{K} c_k = 1$ (иначе можно взять c_k сколь угодно большими и получим $L(y, c)$ сколь угодно маленьким, даже отрицательным). Но для начала подставим функцию потерь в критерий информативности и запишем оптимизационную задачу. Первый шаг аналогичен первому пункту:

$$H(R,c) = -\frac{1}{|R|} \sum_{(x_i,y_i)\in R} \sum_{k=1}^{K} [y_i = k] \log c_k = -\sum_{k=1}^{K} p_k \log c_k$$

Задача оптимизации:

$$\begin{cases} -\sum_{k=1}^{K} p_k \log c_k \to \min_{c_k} \\ \sum_{k=1}^{K} c_k = 1 \\ c_k > 0, k = 1...K \end{cases}$$

Решим ее с помощью метода Лагранжа для условного экстремума:

$$L(c) = -\sum_{k=1}^{K} p_k \log c_k + \lambda \left(\sum_{k=1}^{K} c_k - 1 \right)$$

$$\begin{cases} \frac{\partial L}{\partial c_k} = -\frac{p_k}{c_k} + \lambda = 0 \\ \phi(c) = \sum_{k=1}^{K} c_k - 1 = 0 \end{cases}$$

$$\begin{cases} c_k = \frac{p_k}{\lambda} \\ \sum_{k=1}^{K} \frac{p_k}{\lambda} = 1 \end{cases}$$

$$\begin{cases} \lambda = \sum_{k=1}^{K} p_k = 1 \\ c_k = p_k \end{cases}$$

Таким образом:

$$H(R) = -\sum_{k=1}^{K} p_k \log p_k$$

А это и есть энтропийный критерий.

2. При построении дерева глубины D нам необходимо выбрать предикат для $\leq 2^D - 1$ узла. Для выбора предиката в узле дерева нужно перебрать все d признаков, для каждого признака есть $\leq l - 1$ возможных пороговых значений, для всех порогов критерий информативности считается за константное время. Таким образом, общее время построения дерева: $O((2^D - 1)d(l - 1)) = O(2^D ld)$.

4 BVD

1. Найдем смещение напрямую через формулу:

$$bias = \mathbb{E}_x \left[(\mathbb{E}_X \left[\mu(X)(x) \right] - \mathbb{E} \left[y|x \right])^2 \right] = \mathbb{E}_x \left[\left(C - x^T x \right)^2 \right] =$$
$$= C^2 - 2C\mathbb{E}_x \left[x^T x \right] + \mathbb{E}_x \left[(x^T x)^2 \right]$$

Поскольку известно распределение x, найдем матожидание напрямую:

$$\mathbb{E}_x \left[x^T x \right] = \int_{[0,1]^d} (x_1^2 + ... + x_d^2) dx_1 ... dx_d = d \int_0^1 x_1^2 dx_1 = \frac{d}{3}$$

$$\mathbb{E}_x \left[(x^T x)^2 \right] = \int_{[0,1]^d} (x_1^2 + ... + x_d^2)^2 dx_1 ... dx_d =$$
$$\int_{[0,1]^d} \left(\sum_i x_i^4 + \sum_{i \neq j} x_i^2 x_j^2 \right) dx_1 ... dx_d =$$
$$= d \int_0^1 x_1^4 dx_1 + d(d-1) \int_0^1 x_1^2 dx_1 \int_0^1 x_2^2 dx_2 = \frac{d}{5} + \frac{d(d-1)}{9}$$

Получаем ответ:

$$bias = C^2 - \frac{2Cd}{3} + \frac{d}{5} + \frac{d(d-1)}{9}$$

2. Начнем с $\mathbb{E} \left[y|x \right]$, тут все просто:

$$\mathbb{E} \left[y|x \right] = \mathbb{E}[f_x + \varepsilon | x] = f_x$$

Теперь посчитаем $\mathbb{E}_X \left[\mu(X)(x) \right]$ (обратите внимание, что здесь f_x - это константа, не зависящая от обучающей выборки, а потому она выносится за матожидание, ошибка также входит в выборку X, поэтому по ней берется ожидание, и она независима с распределением X):

$$\mathbb{E}_X \left[\mu(X)(x) \right] = \mathbb{E}_X \left[\hat{f}_x \right] = \mathbb{E}_X \left[\frac{1}{l} \sum_{i=1}^{l} \left[X_i = x \right] (f_x + \varepsilon) \right] =$$
$$= \frac{f_x}{l} \mathbb{E}_X \left[\sum_{i=1}^{l} \left[X_i = x \right] \right] + \frac{1}{l} \mathbb{E}_X \left[\sum_{i=1}^{l} \left[X_i = x \right] \right] \mathbb{E}_X \left[\varepsilon \right] =$$
$$= \frac{f_x}{l} \sum_{i=1}^{l} \mathbb{P}_X (X_i = x) = \frac{f_x}{l} \sum_{i=1}^{l} \frac{1}{K} = \frac{f_x}{K}$$

Отсюда находим смещение:

$$bias = \mathbb{E}_x \left[\left(\mathbb{E}_X \left[\mu(X)(x) \right] - \mathbb{E} \left[y | x \right] \right)^2 \right] = \mathbb{E}_x \left[\left(\frac{f_x}{K} - f_x \right)^2 \right] =$$
$$= \left(\frac{K-1}{K} \right)^2 \mathbb{E}_x \left[f_x^2 \right] = \left(\frac{K-1}{K} \right)^2 \sum_{x=1}^{K} \frac{f_x^2}{K} = \frac{(K-1)^2}{K^3} \sum_{x=1}^{K} f_x^2$$

3. (a) Заметим, что какие бы два веса мы не взяли, их сумма будет больше третьего, таким образом, для верной классификации необходимо и достаточно, чтобы хотя бы какие-то два алгоритма выдали правильный результат. Пусть $\xi \sim Bin(3, 1-p)$ - число алгоритмов, получивших верный ответ. Тогда вероятность ошибки композиции:

$$p_0 = \mathbb{P}(\xi \leq 1) = p^3 + 3p^2(1-p) = p^2(3-2p)$$

   (b) Тут все куда проще: поскольку $w_2 > w_1 + w_3$, то ответ композиции тождественно равен ответу второго алгоритма. Поэтому вероятность ошибки равна вероятности ошибки второго алгоритма:

$$p_0 = p$$