

Логистическая регрессия

Ранее мы изучили общий подход к обучению линейных классификаторов, основанный на минимизации верхней оценки. При этом мы привели примеры нескольких верхних оценок $\tilde{L}(M)$, среди которых была и *логистическая функция потерь*.

Метод обучения, который получается при использовании логистической функции потерь, называется *логистической регрессией*.

Несмотря на то, что в названии метода используется слово регрессия, этот алгоритм предназначен для решения задач *классификации*.

Основным его свойством является тот факт, что он корректно оценивает вероятность принадлежности объекта к каждому из классов.

Пусть в каждой точке пространства объектов $x \in X$ задана вероятность $p(y = +1|x)$ того, что объект x будет принадлежать классу $+1$. Это означает, что мы допускаем наличие в выборке нескольких объектов с одинаковым признаковым описанием, но с разными значениями целевой переменной; причём если устремить количество объекта x в выборке к бесконечности, то доля положительных объектов среди них будет стремиться к $p(y = +1|x)$.

Примером может служить задача предсказания кликов по рекламным баннерам. При посещении одного и того же сайта один и тот же пользователь может как кликнуть, так и не кликнуть по одному и тому же баннеру, из-за чего в выборке могут появиться одинаковые объекты с разными ответами. При этом важно, чтобы классификатор предсказывал именно вероятности классов — если домножить вероятность первого класса на сумму, которую заплатит заказчик в случае клика, то мы получим матожидание прибыли при показе этого баннера. На основе таких матожиданий можно построить алгоритм, выбирающий баннеры для показа пользователю

Итак, рассмотрим точку x пространства объектов. Как мы договорились, в ней имеется распределение на ответах $p(y = +1|x)$. Допустим, алгоритм $b(x)$ возвращает числа из отрезка $[0, 1]$. Наша задача — выбрать для него такую процедуру обучения, что в точке x ему будет оптимально выдавать число $p(y = +1|x)$. Если в выборке объект x встречается n раз с ответами $\{y_1, \dots, y_n\}$, то получаем следующее требование:

$$\arg \min_{b \in \mathbb{R}} \frac{1}{n} \sum_{i=1}^n L(y_i, b) \approx p(y = +1 | x).$$

При стремлении n к бесконечности получим, что функционал стремится к матожиданию ошибки:

$$\arg \min_{b \in \mathbb{R}} \mathbb{E} [L(y, b) | x] = p(y = +1 | x).$$

Этим свойством обладает, например, квадратичная функция потерь $L(y, x) = (y - z)^2$, если в ней для положительных объектов использовать истинную метку $y = 1$, а для отрицательных брать $y = 0$. Примером функции потерь, которая не позволяет оценивать вероятности, является модуль отклонения $L(y, x) = |y - z|$. Можно показать, что с точки зрения данной функции оптимальным ответом всегда будет либо ноль, либо единица. Это требование можно воспринимать более просто. Пусть один и тот же объект встречается в выборке 1000 раз, из которых 100 раз он относится к классу +1, и 900 раз — к классу -1. Поскольку это один и тот же объект, классификатор должен выдавать один ответ для каждого из тысячи случаев. Можно оценить матожидание функции потерь в данной точке по 1000 примеров при прогнозе b :

$$\mathbb{E}[L(y, b) | x] \approx \frac{100}{1000}L(1, b) + \frac{900}{1000}L(-1, b).$$

Наше требование, по сути, означает, что оптимальный ответ с точки зрения этой оценки должен быть равен 1/10:

$$\arg \min_{b \in \mathbb{R}} \left(\frac{100}{1000}L(1, b) + \frac{900}{1000}L(-1, b) \right) = \frac{1}{10}.$$

Хотя квадратичная функция потерь и приводит к корректному оцениванию вероятностей, она не очень хорошо подходит для решения задачи классификации. Причиной этому в том числе являются и слишком низкие штрафы за ошибку — так, если объект положительный, а модель выдаёт для него вероятность первого класса $b(x) = 0$, то штраф за это равен всего лишь единице: $(1 - 0)^2 = 1$.

Попробуем сконструировать функцию потерь из других соображений. Если алгоритм $b(x) \in [0, 1]$ действительно выдает вероятности, то они должны согласовываться с выборкой. С точки зрения алгоритма вероятность того, что в выборке встретится объект x_i с классом y_i , равна $b(x_i)^{[y_i=+1]}(1 - b(x_i))^{[y_i=-1]}$.

Исходя из этого, можно записать правдоподобие выборки (т.е. вероятность получить такую выборку с точки зрения алгоритма):

$$(a, X) = \prod_{i=1}^{\ell} b(x_i)^{[y_i=+1]}(1 - b(x_i))^{[y_i=-1]}.$$

Данное правдоподобие можно использовать как функционал для обучения алгоритма — с той лишь оговоркой, что удобнее оптимизировать его логарифм:

$$-\sum_{i=1}^{\ell} ([y_i = +1] \log b(x_i) + [y_i = -1] \log(1 - b(x_i))) \rightarrow \min$$

Везде ранее мы требовали, чтобы алгоритм $b(x)$ возвращал числа из отрезка $[0, 1]$. Этого легко достичь, если положить $b(x) = \sigma(\langle \omega, x \rangle)$, где в качестве σ может выступать любая монотонно неубывающая функция с областью значений $[0, 1]$.

Мы будем использовать **сигмоидную функцию**: $\sigma(z) = \frac{1}{1+e^{-z}}$. Таким образом, чем больше скалярное произведение $\langle \omega, x \rangle$, тем больше будет предсказанная вероятность. Как при этом можно интерпретировать данное скалярное произведение? Чтобы ответить на этот вопрос, преобразуем уравнение

$$p(y = 1 | x) = \frac{1}{1 + \exp(-\langle w, x \rangle)}.$$

Выражая из него скалярное произведение

$$\langle w, x \rangle = \log \frac{p(y = +1 | x)}{p(y = -1 | x)}.$$

Получим, что скалярное произведение будет равно логарифму *отношения вероятностей классов* (log-odds).

Как уже упоминалось выше, при использовании квадратичной функции потерь алгоритм будет пытаться предсказывать вероятности, но данная функция потерь является далеко не самой лучшей, поскольку слабо штрафует за грубые ошибки. Логарифмическая функция потерь подходит гораздо лучше, поскольку не позволяет алгоритму сильно ошибаться в вероятностях. Подставим трансформированный ответ линейной модели в логарифмическую функцию потерь:

$$\begin{aligned} & - \sum_{i=1}^{\ell} \left([y_i = +1] \log \frac{1}{1 + \exp(-\langle w, x_i \rangle)} + [y_i = -1] \log \frac{\exp(-\langle w, x_i \rangle)}{1 + \exp(-\langle w, x_i \rangle)} \right) = \\ & = - \sum_{i=1}^{\ell} \left([y_i = +1] \log \frac{1}{1 + \exp(-\langle w, x_i \rangle)} + [y_i = -1] \log \frac{1}{1 + \exp(\langle w, x_i \rangle)} \right) = \\ & = \sum_{i=1}^{\ell} \log (1 + \exp(-y_i \langle w, x_i \rangle)). \end{aligned}$$

Полученная функция в точности представляет собой логистические потери. Линейная модель классификации, настроенная путем минимизации данного функционала, называется *логистической регрессией*. Как видно из приведенных рассуждений, она оптимизирует правдоподобие выборки и дает корректные оценки вероятности принадлежности к положительному классу.