Машинное обучение в экономике

Логистическая регрессия и метод опорных векторов

Потанин Богдан Станиславович

доцент, научный сотрудник, кандидат экономических наук

2023–2024

Логистическая регрессия

Интуиция

- 123

- Предположим, что условные вероятности могут быть оценены как линейные комбинации признаков (регрессоров):

$$P(Y_i = 1|X_i = x_i) = g\left(X_i\beta\right)$$

- Коэффициенты β часто называют **весами**, а функция $g()$ принимает значения от 0 до 1, поскольку отражает условные вероятности.
- В качестве функции $g()$ удобно взять функцию распределения некоторого распределения с носителем на R. Наиболее популярным в машинном обучении является логистическое распределение, при котором мы получаем **логит модель**:

$$g(t) = \frac{1}{1 + e^{-t}}$$

- В эконометрике не менее популярной является функция распределения стандартного нормального распределения $g(t) = \Phi(t)$, при которой мы получаем **пробит модель**.

- Для оценивания условных вероятностей достаточно оценить параметры β методом максимального правдоподобия:

$$L(\beta; X, Y) = \prod_{i:y_i=1} P(Y_i = 1|X_i) \prod_{i:y_i=0} P(Y_i = 0|X_i) = \prod_{i:y_i=1} g(X_i\beta) \prod_{i:y_i=0} 1 - g(X_i\beta) =$$

$$\prod_{i:y_i=1} \frac{1}{1 + e^{-X_i\beta}} \prod_{i:y_i=0} 1 - \frac{1}{1 + e^{-X_i\beta}}$$

$$\ln L(\beta; X, Y) = \sum_{i:y_i=1} -\ln(1 + e^{-X_i\beta}) + \sum_{i:y_i=0} X_i\beta - \ln(1 + e^{-X_i\beta})$$

- Можно показать, что логарифм функции правдоподобия является вогнутой функцией по β при любых x_i, а значит ее максимум является единственным.
- В отличие от линейного МНК, в данном случае не существует аналитического выражения для $\hat{\beta}$, что мотивирует **максимизацию численными методами**.

Численная оптимизация

Мотивация и классификация

Численная оптимизация позволяет находить приблизительный максимум или минимум функции без необходимости искать аналитическое решение.

- Методы **локальной** оптимизации (BFGS, градиентный спуск) как правило работают достаточно быстро, но позволяют находить лишь локальные экстремумы. Методы **глобальной** оптимизации (генетический алгоритм, метод отжига – SA) позволяют найти несколько экстремумов, один из которых может оказаться глобальным. Однако, глобальная оптимизация обычно крайне затратна по времени.
- Методы локальной оптимизации часто опираются на Градиент (градиентный спуск, ADAM) или Гессиан функции (BFGS, BHHH). В последнем случае число итераций алгоритма, как правило, оказывается меньше, но время каждой итерации – больше, особенно, при большом числе оцениваемых параметров.

Поскольку число оцениваемых параметров в эконометрических моделях, как правило, относительно невелико (в сравнении с моделями машинного обучения), то чаще используются алгоритмы, использующие информацию о Гессиане (BFGS, BHHH).

Алгоритм **градиентного спуска** является одним из простейших численных методов нахождения минимума функции.

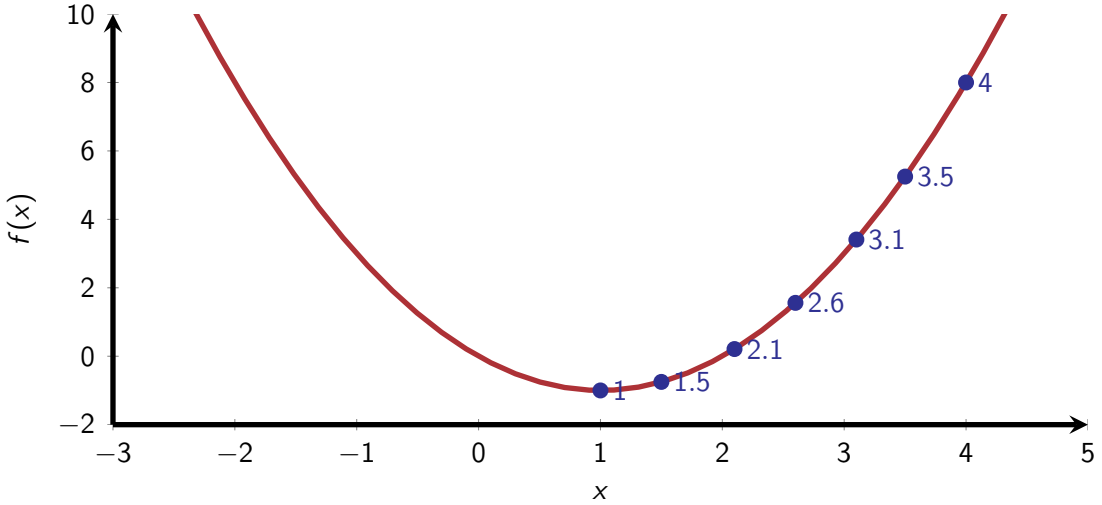
- Выбираем произвольную начальную точку x_0.
- Считаем градиент функции в этой точке $\nabla f(x_0)$.
- Переходим в новую точку $x_1 = x_0 - \alpha \nabla f(x_0)$, где α – малая положительная константа.
- Повторяем процедуру до тех пор, пока не будут соблюдены **условия остановки** (termination conditions), например, о том, что $|\nabla f(x_0)| < \varepsilon$, где ε – маленькое положительное число.

Нетрудно показать аналитически, что функция $f(x) = x^2 - 2x$ достигает минимума в точке $x^* = 1$. В качестве альтернативы аналитическому решению попробуем приблизиться к минимуму с помощью 10 итераций описанного алгоритма, произвольным образом полагая $x_0 = 3$ и $\alpha = 0.2$.

i	0	1	2	3	4	5	6	7	8	9	10
x_i	3	2.20	1.72	1.43	1.26	1.16	1.09	1.06	1.03	1.02	**1.01**
$\nabla f(x_i)$	4	2.40	1.44	0.86	0.52	0.31	0.19	0.11	0.07	0.04	0.02
$f(x_i)$	3	0.44	-0.48	-0.81	-0.93	-0.98	-0.99	-1.00	-1.00	-1.00	-1.00

Численная оптимизация

Упрощенная графическая иллюстрация локальной численной оптимизации



- **Проблема** – на первый взгляд линейная форма условных вероятностей $X_i\beta$ снижает гибкость модели.
- **Решение** – можно добавить нелинейность в модель, например, взяв не только сами признаки, но и некоторые нелинейные функции от них.
- Например, вместо возраста age_i можно взять его полином третьей степени просто добавив в число признаков age_i^2 и age_i^3.
- Другой пример: чтобы учесть взаимодействие между возрастом age_i и доходом income_i можно добавить в число признаков их произведение $\text{age}_i \times \text{income}_i$.
- Оптимальная спецификация логистической регрессии может быть подобрана, например, с помощью кросс-валидации.

Мультиномиальная логистическая регрессия

Определение

- Мультиномиальная логистическая регрессия используется в случае, когда необходимо предсказать значение одной из K взаимоисключающих альтернатив.
- Например, необходимо спрогнозировать, какое мороженое предпочтет индивид: шоколадное, ванильное или эскимо. В данном случае $K = 3$.
- Влияние признаков x_i на полезность (склонность к выбору) k-й альтернативы описывается как:

$$u(k, x_i) = X_i\beta_k + \varepsilon_{ki}, \qquad \varepsilon_{ki} \sim \text{extreme error value distribution, i.i.d.}$$

- Индивид выбирает альтернативу, приносящую ему наибольшую полезность.
- Можно показать, что условные вероятности выбора k-й альтернативы записываются при помощи **softmax** функции:

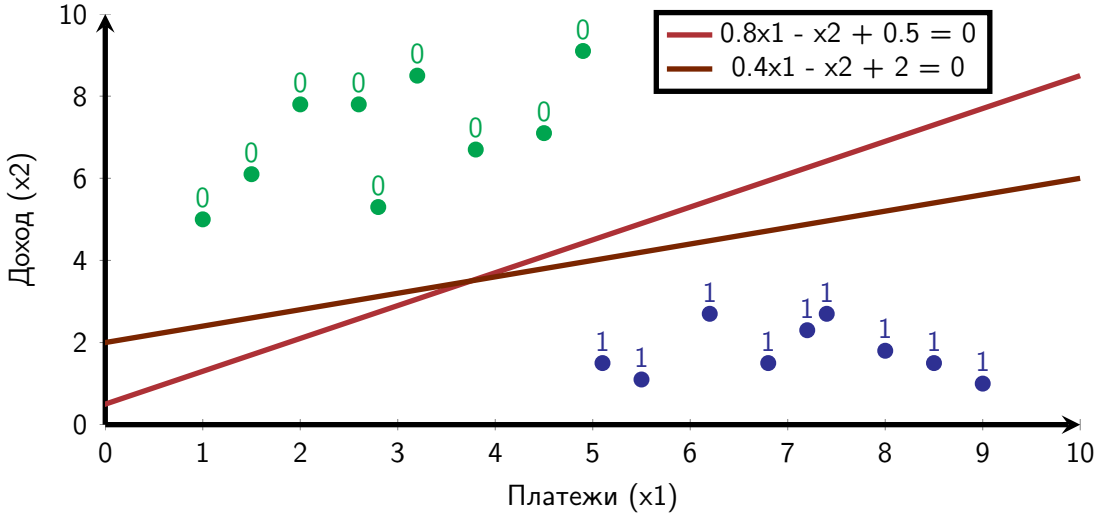
$$P(Y_i = k|X_i = x_i) = e^{-x_i\beta_k} / \sum_{j=1}^{K} e^{-x_j\beta_j} \qquad k \in \{1, ..., K\}, \beta_1 = (0, ...0).$$

- Параметры β_j оцениваются с помощью метода максимального правдоподобия.

- В случае с категориальными переменными, принимающими более, чем два значения, цены ошибок прогнозов могут различаться в зависимости от того, какую с какой категорией мы перепутали.
- Например, в задаче кридитного скоринга мы можем рассматривать три случая: дефолт,

Метод опорных векторов

Графическая иллюстрация идеи на примере дефолта



Метод опорных векторов SVM

Случай с наличием разделяющей гиперплоскости

- **Предположение** – существуют линии, которые можгут полностью разграничить два класса в зависимости от значений признаков.
- **Проблема** – какую из линий выбрать, если их бесконечно много?
- **Решение** – выбираем линию таким образом, чтобы максимизировать **перпендикулярное** расстояние от нее до ближайшего наблюдения. Этой расстояние именуется **отступом** (margin).
- После того, как мы выбрали линию, наблюдения, имеющие наименьшее перпендикулярное расстояние до этой линии, именуются **опорными векторами**.
- Разделяющая линия задается уравнением $\beta_0 + x\beta = 0$, где β и β_0 подбираются из соображений минимизации отступа.
- Следуя традиции и для удобства класс 0 будем обозначать как -1.

- Рассмотрим β_0 и β такие, что $\beta_0 + X_i\beta = 0$ задает разделяющую линию с максимальным отступом.
- Определим классификатор следующим образом:

$$\hat{y}(x) = \begin{cases} 1, \text{ если } \beta_0 + X_i\beta \geq c_1 \text{ (точки над отступом разделяющей линии)} \\ -1, \text{ если } \beta_0 + X_i\beta \leq -(c_{-1}) \text{ (точки под отступом разделяющей линии)} \end{cases}$$

- Если $c_1 > c_{-1}$, то геометрически очевидно, что отступ не является максимальным, поскольку сдвинув линию вниз мы сможем его увеличить. По аналогии невозможен случай $c_1 < c_{-1}$, а значит $c_1 = c_{-1}$.
- Поскольку умножении равенства $\beta_0 + X_i\beta = 0$ на константу его не изменяет, то без потери общности можно положить $c_1 = c_{-1} = 1$, откуда получаем классификатор:

$$\hat{y}(x) = \begin{cases} 1, \text{ если } \beta_0 + X_i\beta \geq 1 \\ -1, \text{ если } \beta_0 + X_i\beta \leq -1 \end{cases}$$

Метод опорных векторов SVM

Оптимизационная задача

- Определим перпендикулярное расстояние от наблюдения X_i до линии $\beta_0 + X_i\beta = 0$:

$$d(X_i; \beta_0, \beta) = \frac{|\beta_0 + X_i\beta|}{\sqrt{\beta_1^2 + ... + \beta_m^2}} = \frac{|\beta_0 + X_i\beta|}{||\beta||}$$

- Обозначим через q произвольный опорный вектор, то есть наблюдение, находящееся на расстоянии отступа от разграничивающей линии $\beta_0 + X_i\beta = 0$.
- Из введенного ранее определения классификатора следует, что $|\beta_0 + \beta q| = 1$, а значит $d(q, \beta_0, \beta_1) = 1/||\beta||$.
- Поскольку отступ определяется через опорный вектор q, то задача максимизации отступа может быть сведена к задаче максимизации $d(q, \beta_0, \beta_1)$, что эквивалентно минимизации $||\beta||$.
- При решении этой задачи важно гарантировать, что найденные β_0 и β соответствуют линии, являющейся разграничивающей линией, то есть наблюдения различных классов должны лежать по разные стороны от нее:

$$\begin{cases} \beta_0 + X_i\beta \geq 1, \text{ если } Y_i = 1 \\ \beta_0 + X_i\beta \leq -1, \text{ если } Y_i = -1 \end{cases} \iff Y_i(\beta_0 + X_i\beta) \geq 1$$

- Таким образом, для удобства избавляясь от квадратного корня (строго возрастающая функция) задачу максимизации отступа можно сформулировать как следующую задачу условной минимизации (**квадратичное программировнаие**):

$$(\hat{\beta}_0, \hat{\beta}) = \underset{(\beta_0,\beta)}{\text{argmin}}\ \beta_1^2 + ... + \beta_m^2 \quad \text{при ограничении} \quad Y_i\left(\beta_0 + X_i\beta\right) \geq 1$$

- Эта оптимизационная задача не имеет аналитического решения, однако минимум может быть найден численными методами.
- Напомним, что при этом классификатор определяется следующим образом:

$$\hat{y}(x) = \begin{cases} 1, \text{ если } \beta_0 + X_i\beta \geq 1 \\ -1, \text{ если } \beta_0 + X_i\beta \leq -1 \end{cases}$$

- Очевидно, что на практике разделяющая линия существует очень редко, а значит имеется такое наблюдение X_i, что при любых β_0 и β_1 неравенство $Y_i(\beta_0 + X_i\beta) \geq 1$ не соблюдается.
- В таком случае необхоидмо допустить возможность нарушения абсолютного разграничения, то есть наблюдения из класса 1 могут попадать в область наблюдений -1 и наоборот.
- Тогда оптимизационную задачу можно привести к виду:

$$\underset{(\beta_0,\beta,\xi_1,\ldots,\xi_n)}{\operatorname{argmin}} \sum_{j=1}^{m} \beta_j^2 + C\sum_{i=1}^{n} \xi_i, \text{ при ограничении } Y_i(\beta_0 + X_i\beta) \geq 1 - \xi_i$$

- Константа C определяет вес штрафов ξ_i в оптимизационной задаче и подбирается с помощью кросс-валидации.

- В качестве разграничивающей классы функции можно рассмотреть не линию, а более сложную функцию.
- **Идея** – вместо самих признаков рассмотрим базисные функции от них.

- В качестве унифицированного способа репрезентации моделей машинного обучения часто используют **loss function** (функция потерь).
- Рассмотрим классификатор $\hat{y}(x)$.