УДК 519.248:[33+301+159.9]

# ЛОГИСТИЧЕСКАЯ РЕГРЕССИЯ

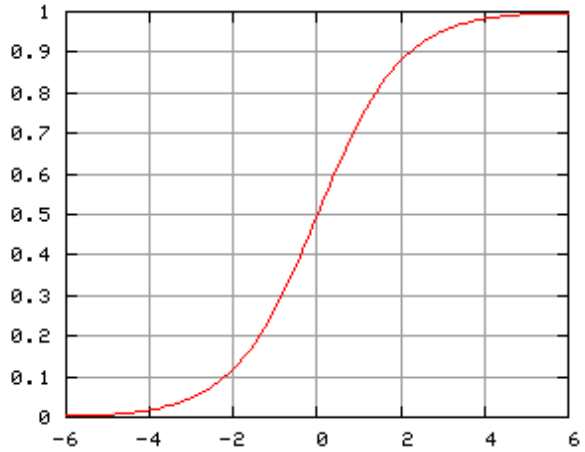
**Романенко А.В.,**
**научный руководитель:**
**профессор, доктор физико-математических наук Воробьев О.Ю.**
***Сибирский федеральный университет,***
***Институт математики***

**Цели:**

Привести примеры использования логистической регрессии.

**Логистическая регрессия** или **логит регрессия** (англ. *logit model*) — это статистическая модель, используемая для предсказания вероятности возникновения некоторого события путём подгонки данных к логистической кривой.

**Описание**




Логистическая функция: $f(x) = \frac{1}{1 + e^{-x}}$.

Логистическая регрессия применяется для предсказания вероятности возникновения некоторого события по значениям множества признаков. Для этого вводится так называемая *зависимая переменная y*, принимающая лишь одно из двух значений — как правило, это числа 0 (событие не произошло) и 1 (событие произошло), и множество *независимых переменных* (также называемых признаками, предикторами или регрессорами) — вещественных $x_1,x_2,...,x_n$, на основе значений которых требуется вычислить вероятность принятия того или иного значения зависимой переменной.

Делается предположение о том, что вероятность наступления события $y = 1$ равна:

$$\Pr\{y = 1|x\} = f(z),$$

где $z = \theta^T x = \theta_1 x_1 + \ldots + \theta_n x_n$, $x$ и $\theta$ — вектора-столбцы значений независимых переменных $x_1,...,x_n$ и параметров (коэффициентов регрессии) — вещественных чисел $\theta_1,...,\theta_n$, соответственно, а $f(z)$ — так называемая *логистическая функция* (иногда также называемая сигмоидом или логит-функцией):

$$f(z) = \frac{1}{1 + e^{-z}}.$$

Так как *y* принимает лишь значения 0 и 1, то вероятность второго возможного значения равна:

$$\Pr\{y = 0|x\} = 1 - f(z) = 1 - f(\theta^T x).$$

Для краткости, функцию распределения *y* при заданном *x* можно записать в таком виде:

$$\Pr\{y|x\} = f(\theta^T x)^y (1 - f(\theta^T x))^{1-y}, \quad y \in \{0, 1\}.$$

Фактически, это есть распределение Бернулли с параметром, равным $f(\theta^T x)$.

**Подбор параметров**

Для подбора параметров $\theta_1,...,\theta_n$ необходимо составить обучающую выборку, состоящую из наборов значений независимых переменных и соответствующих им значений зависимой переменной *y*. Формально, это множество пар $(x^{(1)}, y^{(1)}), ..., (x^{(m)}, y^{(m)})$, где $x^{(i)} \in \mathbb{R}^n$ — вектор значений независимых переменных, а $y^{(i)} \in \{0, 1\}$ — соответствующее им значение *y*. Каждая такая пара называется обучающим примером.

Обычно используется метод максимального правдоподобия, согласно которому выбираются параметры θ, максимизирующие значение функции правдоподобия на обучающей выборке:

$$\hat{\theta} = \operatorname{argmax}_\theta L(\theta) = \operatorname{argmax}_\theta \prod_{i=1}^{m} \Pr\{y = y^{(i)}|x = x^{(i)}\}.$$

Максимизация функции правдоподобия эквивалентна максимизации её логарифма:

$$\log L(\theta) = \sum_{i=1}^{m} \log \Pr\{y = y^{(i)}|x = x^{(i)}\} = \sum_{i=1}^{m} y^{(i)} \log f(\theta^T x^{(i)}) + (1 - y^{(i)}) \log($$

Для максимизации этой функции может быть применён, например, метод градиентного спуска. Он заключается в выполнении следующих итераций, начиная с некоторого начального значения параметров θ:

$$\theta := \theta + \alpha \nabla \log L(\theta) = \theta + \alpha \sum_{i=1}^{m} (y^{(i)} - f(\theta^T x^{(i)})) x^{(i)}, \quad \alpha > 0.$$

На практике также применяют метод Ньютона и стохастический градиентный спуск.

**Регуляризация**

Для улучшения обобщающей способности получающейся модели, то есть уменьшения эффекта переобучения, на практике часто рассматривается логистическая регрессия с регуляризацией.

Регуляризация заключается в том, что вектор параметров θ рассматривается как случайный вектор с некоторой заданной априорной плотностью распределения $p(\theta)$. Для обучения модели вместо метода наибольшего правдоподобия при этом используется метод максимизации апостериорной оценки, то есть ищутся параметры θ, максимизирующие величину:

$$\prod_{i=1}^{m} \Pr\{y^{(i)}|x^{(i)},\theta\} \cdot p(\theta).$$

В качестве априорного распределения часто выступает многомерное нормальное распределение $\mathcal{N}(0,\sigma^2 I)$с нулевым средним и матрицей ковариации $\sigma^2 I$, соответствующее априорному убеждению о том, что все коэффициенты регрессии должны быть небольшими числами, идеально — многие малозначимые коэффициенты должны быть нулями. Подставив плотность этого априорного распределения в формулу выше, и прологарифмировав, получим следующую оптимизационную задачу:

$$\sum_{i=1}^{m} \log \Pr\{y^{(i)}|x^{(i)},\theta\} - \lambda||\theta||^2 \rightarrow \max,$$

где $\lambda = \text{const} / \sigma^2$ — параметр регуляризации. Этот метод известен как L2-регуляризованная логистическая регрессия, так как в целевую функцию входит L2-норма вектора параметров для регуляризации.

Если вместо L2-нормы использовать L1-норму, что эквивалентно использованию распределения Лапласа, как априорного, вместо нормального, то получится другой распространённый вариант метода — L1-регуляризованная логистическая регрессия:

$$\sum_{i=1}^{m} \log \Pr\{y^{(i)}|x^{(i)},\theta\} - \lambda||\theta||_1 \rightarrow \max.$$

**Применение**

Эта модель часто применяется для решения задач классификации — объект *x* можно отнести к классу $y = 1$, если предсказанная моделью вероятность $\Pr\{y=1|x\} > 0.5$, и к классу $y = 0$ в противном случае. Получающиеся при этом правила классификации являются линейными классификаторами.

**Примеры использования:**

**Реализация в *STATISTICA***

Система *STATISTICA* позволяет решать задачи с бинарным откликом в том числе и с помощью логистической регрессии.

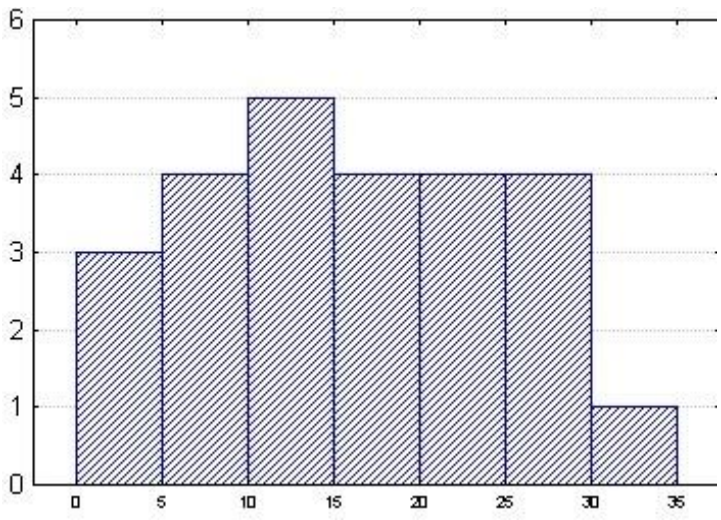
**Задача о программистах**

Приведем пример такого анализа. Предположим, что вы хотите проверить, правда ли, что стаж работы помогает программистам в написании сложных программ, если на написание отпущен ограниченный промежуток времени. Для исследования были выбраны двадцать пять программистов с различным стажем работы (выраженным в месяцах). Их попросили написать сложную компьютерную программу за определенный промежуток времени. Бинарная переменная отклика принимала значение 1, если программист справился с поставленной задачей, и 0, если нет.

Эти исходные данные выглядят следующим образом:

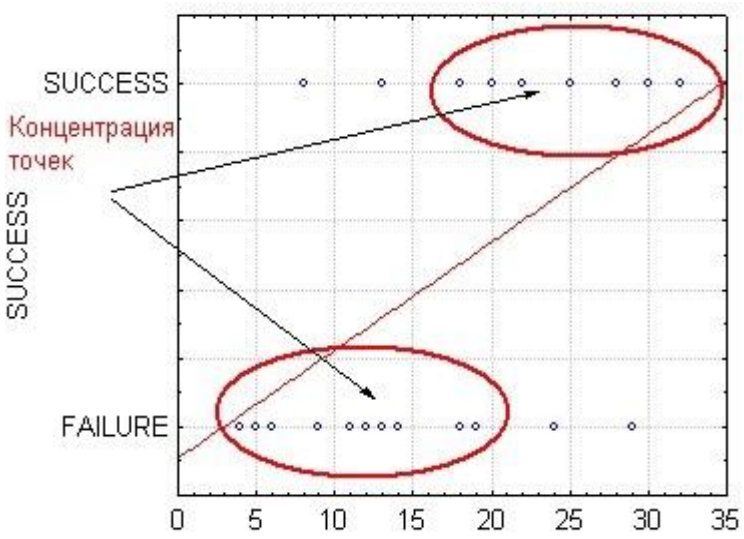
| | Success in a programing | |
|---|---|---|
| | 1 EXPERNCE | 2 SUCCESS |
| **Frank** | 14 | FAILURE |
| Henry | 29 | FAILURE |
| Tom | 6 | FAILURE |
| Beth | 25 | SUCCESS |
| Susan | 18 | SUCCESS |
| Harry | 4 | FAILURE |
| Paul | 18 | FAILURE |
| Pete | 12 | FAILURE |
| Diana | 22 | SUCCESS |
| Louise | 6 | FAILURE |
| Fred | 30 | SUCCESS |
| Hank | 11 | FAILURE |
| Steven | 30 | SUCCESS |
| Tod | 5 | FAILURE |

Первым шагом для любого анализа является осознание структуры представленных данных. У нас есть таблица с двумя переменными. Для начала посмотрим, как распределен стаж работы кандидатов - построим гистограмму для переменной EXPERENCE.




Мы видим, что опыт работы для программистов распределен довольно равномерно. Представлены как опытные, так и неопытные кандидаты и их примерно одинаковое число.

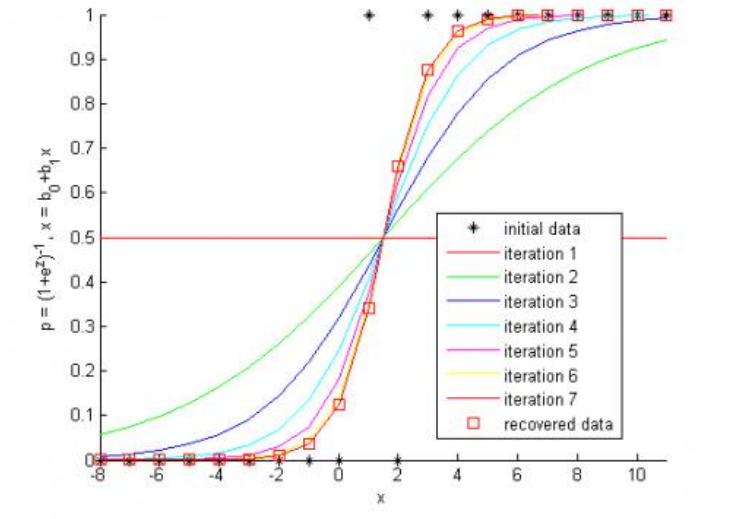
Насколько эффективно программисты справлялись с заданием? Построим диаграмму рассеяния.




**Результат:** На диаграмме рассеяния выделяются два облака точек. Одно - вблизи программистов с небольшим опытом и проваливших задание, второе - вблизи программистов с обширным опытом и выполнивших задание.

**Реализация в MatLab**

Перед началом работы алгоритма задаются начальные значения параметров. Вычисление параметров логистической регрессии происходит итерационно.




**Результат:** На графике показаны исходные данные. По оси абсцисс отложены значения единственного признака, а по оси ординат -- метки класса объектов. Объекты обозначены звездочками. Линии логистической кривой показывают последовательные итерации настройки параметров модели. Значения вероятности принадлежности объектов классам показаны квадратиками на кривой, которая соответствует последней итерации.

Результатом вычислительного эксперимента является иллюстрация работы алгоритма Ньютона-Рафсона для задачи восстановления логистической регрессии. Найден параметр модели, соответствующей минимуму заданной функции невязок. Получена классификация объектов, описанных единственным признаком.

**Литература**


• *Andrew Ng.* Stanford CS229 Lecture Notes

• Материалы с сайтов: www.statsoft.ru