

Аналитическое и численное решение задачи методом наименьших квадратов

Евгений Баяк

Октябрь 2024

Введение

Метод наименьших квадратов – это фундаментальный статистический метод, широко используемый в машинном обучении для регрессионного анализа. Он направлен на поиск наиболее подходящей линии или кривой для заданного набора данных путем минимизации суммы квадратов остатков (разниц между наблюдаемыми и прогнозируемыми значениями).

1 Аналитическое решение

1.1 Общий случай

Имеется набор из n значений переменной y (например, результаты наблюдений или экспериментов) и соответствующие им переменные x . Задача состоит в том, чтобы аппроксимировать зависимость между y и x с помощью некоторой функции $f(x, b)$, которая зависит от неизвестных параметров b . Необходимо найти такие значения параметров b , которые наилучшим образом соответствуют предсказанным значениям $f(x, b)$ реальным значениям y . Это можно рассматривать как задачу «решения» переопределенной системы уравнений для параметров b :

$$f(x_t, b) = y_t, \quad t = 1, \dots, n. \quad (1)$$

В регрессионном анализе и в частности в эконометрике используются вероятностные модели зависимости между переменными

$$y_t = f(x_t, b) + \varepsilon_t, \quad (2)$$

где ε_t – случайные ошибки модели.

Соответственно, отклонения наблюдаемых значений y от модельных $f(x, b)$ уже учитываются в самой модели. Оценка параметров \hat{b}_{OLS} (OLS, от английского Ordinary Least Squares), при которых сумма квадратов отклонений ε_t достигает минимального значения выражается формулой:

$$\hat{b}_{OLS} = \arg \min_b RSS(b), \quad (3)$$

где RSS – Residual Sum of Squares – это мера, используемая в регрессионном анализе для оценки качества модели. Она определяется как сумма квадратов

остатков, то есть разностей между наблюдаемыми значениями и предсказанными значениями модели и определяется как:

$$RSS(b) = e^T e = \sum_{t=1}^n e_t^2 = \sum_{t=1}^n (y_t - f(x_t, b))^2. \quad (4)$$

Чтобы минимизировать RSS по параметрам b , необходимо продифференцировать его по b и приравнять к нулю. Аналитическое решение задачи, основанное на методе наименьших квадратов, в общем случае выражается уравнением [1]:

$$\sum_{t=1}^n (y_t - f(x_t, b)) \frac{\partial f(x_t, b)}{\partial b} = 0. \quad (5)$$

1.2 Линейная регрессия

Пусть задана обучающая выборка $S = \{(\bar{x}_1, y_1), \dots, (\bar{x}_l, y_l)\}$, где $\bar{x}_i \in \mathbb{R}^n$ и $y_i \in \mathbb{R}$ для $i = 1, \dots, l$. Задача линейной регрессии заключается в нахождении линейной функции:

$$f(\bar{x}) = (\bar{w} \cdot \bar{x}) + b, \quad (6)$$

которая наилучшим образом интерполирует элементы выборки S . Геометрически данная функция представляет собой гиперплоскость, которая приближает точки y_i на аргументах x_i при $i = 1, \dots, l$.

Данная задача была решена Гауссом и Лежандром еще в XVIII веке при помощи минимизации суммы квадратов разностей значений функции $f(\bar{x}_i)$ и точек y_i при $i = 1, \dots, l$. Теория обобщения для данного метода хорошо представлена в математической статистике для случая линейной модели генерации данных с гауссовским случайным шумом.

1.3 Метод наименьших квадратов

Функция потерь L для модели линейной регрессии определена как сумма квадратов разностей между наблюдаемыми значениями y_i и прогнозируемыми значениями $f(\bar{x}_i)$:

$$L(\bar{w}, b) = \sum_{i=1}^n (y_i - f(\bar{x}_i))^2 = \sum_{i=1}^n (y_i - (\bar{w} \cdot \bar{x}_i + b))^2 \quad (7)$$

Задача метода наименьших квадратов – минимизировать функцию потерь L .

Пусть \tilde{w} – расширенный вектор-столбец весовых коэффициентов и свободного члена:

$$\tilde{w} = \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \\ b \end{pmatrix} \quad (8)$$

Аналогичным образом, \tilde{x} – расширенный вектор-столбец переменных:

$$\tilde{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \\ 1 \end{pmatrix} \quad (9)$$

В новых расширенных переменных функция регрессии имеет однородный вид без свободного члена:

$$f(\tilde{x}) = (\tilde{w} \cdot \tilde{x}). \quad (10)$$

Тогда матрица типа $(l \times (n+1))$, строками которой являются расширенные векторы-строки $\tilde{x}'_i = (\tilde{x}'_i, 1)$ переменных обозначается

$$\tilde{X} = \begin{pmatrix} \tilde{x}'_1 \\ \tilde{x}'_2 \\ \vdots \\ \tilde{x}'_l \end{pmatrix} = \begin{pmatrix} x_{11} & \dots & x_{1n} & 1 \\ x_{21} & \dots & x_{2n} & 1 \\ \vdots & \ddots & \vdots & \vdots \\ x_{l1} & \dots & x_{ln} & 1 \end{pmatrix}. \quad (11)$$

Также введен l -мерный вектор-столбец значений интерполяции

$$\hat{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_l \end{pmatrix}. \quad (12)$$

Разности $|f(\tilde{x}_i) - y_i|$ называются остатками. Вектор-столбец остатков имеет вид $\bar{y} - (\tilde{X}\tilde{w})$, а функционал можно записать в матричном виде как квадрат нормы вектора-столбца остатков:

$$L(\tilde{w}) = \|\tilde{X}\tilde{w} - y\|^2 = (y - \tilde{X}\tilde{w})'(y - \tilde{X}\tilde{w}). \quad (13)$$

Теперь задача регрессии может быть записана в виде задачи минимизации квадрата нормы вектора остатков:

$$L(\tilde{w}) = \|\tilde{X}\tilde{w} - y\|^2 \rightarrow \min. \quad (14)$$

Геометрически это может интерпретироваться как поиск проекции наименьшей длины вектора \bar{y} на подпространство (гиперплоскость), порожденное векторами-столбцами матрицы \tilde{X} .

Для поиска минимума приравниваем частные производные этого функционала (по переменным w_1, \dots, w_n, b) к нулю. Получим систему из $n+1$ уравнений.

$$\frac{\partial L(\tilde{w})}{\partial \tilde{w}} = 2\tilde{X}^T(\tilde{X}\tilde{w} - y)^T = \bar{0}. \quad (15)$$

Система упрощается к виду:

$$\tilde{X}^T\tilde{X}\tilde{w} = \tilde{X}^Ty. \quad (16)$$

Если матрица $\tilde{X}^T\tilde{X}$ обратима, решение получается по формуле [2]:

$$\tilde{w} = (\tilde{X}^T\tilde{X})^{-1}\tilde{X}^Ty. \quad (17)$$

2 Численное решение

Численное решение задачи наименьших квадратов представляет собой важный аспект регрессионного анализа, позволяющий находить параметры модели, которые минимизируют сумму квадратов отклонений между наблюдаемыми и предсказанными значениями. В отличие от линейных моделей, где решение может быть найдено аналитически, в случае нелинейных моделей требуется применение численных методов.

Одним из наиболее распространенных методов для решения задач нелинейных наименьших квадратов является метод Гаусса-Ньютона. Этот метод основан на итеративном решении линейных задач наименьших квадратов, которые возникают при каждом шаге итерации. Основная идея заключается в том, чтобы линейно аппроксимировать функцию потерь в окрестности текущего приближения параметров и затем решать соответствующую линейную задачу.

Другим подходом является метод Левенберга-Маркардта, который сочетает в себе преимущества метода Гаусса-Ньютона и градиентного спуска. Этот метод адаптирует шаг итерации в зависимости от того, насколько хорошо текущая модель описывает данные, что позволяет более эффективно находить минимум функции потерь.

Для успешного применения методов нелинейных наименьших квадратов важно иметь начальные приближения для параметров, так как они могут существенно повлиять на сходимость алгоритма. Численные методы для решения задач нелинейных наименьших квадратов являются мощными инструментами в статистическом моделировании, позволяя эффективно находить параметры для сложных моделей.

2.1 Метод Гаусса-Ньютона

Метод Гаусса-Ньютона является одним из наиболее распространенных численных методов для решения задач нелинейных наименьших квадратов. Как уже упоминалось, задача заключается в минимизации суммы квадратов остатков по формуле 4. Пусть вектор остатков $\mathbf{g}(b)$ определяется как:

$$\mathbf{g}(b) = \begin{bmatrix} y_1 - f(x_1, b) \\ y_2 - f(x_2, b) \\ \vdots \\ y_n - f(x_n, b) \end{bmatrix}. \quad (18)$$

Метод предполагает, что функция $f(x, b)$ является дифференцируемой. Чтобы минимизировать RSS по параметрам b , необходимо продифференцировать его по b и приравнять к нулю. Для этого используется разложение Тейлора для вектора остатков $\mathbf{g}(\hat{b} + h)$ в окрестности текущего приближения \hat{b} :

$$\mathbf{g}(\hat{b} + h) \approx \mathbf{g}(\hat{b}) + J(\hat{b})h, \quad (19)$$

где матрица Якоби $J(\hat{b})$ имеет размерность $n \times k$ (при этом $n \geq k$) и определяется как:

$$J(\hat{b}) = \begin{bmatrix} \frac{\partial g_1}{\partial b_1}(b) & \frac{\partial g_1}{\partial b_2}(b) & \dots & \frac{\partial g_1}{\partial b_k}(b) \\ \frac{\partial g_2}{\partial b_1}(b) & \frac{\partial g_2}{\partial b_2}(b) & \dots & \frac{\partial g_2}{\partial b_k}(b) \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial g_n}{\partial b_1}(b) & \frac{\partial g_n}{\partial b_2}(b) & \dots & \frac{\partial g_n}{\partial b_k}(b) \end{bmatrix}. \quad (20)$$

Можно пренебречь остатком в разложении Тейлора. Подставляя разложение Тейлора в выражение для $RSS(b)$, получается:

$$RSS(\hat{b} + h) \approx L(h) = RSS(\hat{b}) + J(\hat{b})h^T \mathbf{g}(\hat{b}) + \frac{1}{2}h^T J(\hat{b})^T J(\hat{b})h. \quad (21)$$

Относительно h для упрощения можно обозначить $J(\hat{b}) = J$ и $\mathbf{g}(\hat{b}) = \mathbf{g}$. Градиент функции L :

$$L'(h) = J^T \mathbf{g} + J^T J h. \quad (22)$$

Гессиан $L''(h) = J^T J$ является независимым от h и симметричным. Если матрица Якоби J имеет полный ранг (то есть столбцы линейно независимы), то гессиан будет положительно определённым. Это означает, что функция $L(h)$ имеет уникальный минимум.

Чтобы найти параметры b , минимизирующие RSS , необходимо решить уравнение:

$$(J^T J)h = -J^T \mathbf{g}. \quad (23)$$

Этот шаг позволяет вычислить обновление параметров по формуле:

$$\hat{b}_{k+1} = \hat{b}_k - (J^T J)^{-1} J^T \mathbf{g}(\hat{b}_k). \quad (24)$$

где $\mathbf{g}(\hat{b}_k)$ — вектор остатков, вычисленный на текущем шаге. Процесс повторяется до тех пор, пока изменения в параметрах не станут достаточно малыми, что указывает на сходимость алгоритма.

Метод Гаусса-Ньютона эффективен для задач, где функция $f(x, b)$ является гладкой и хорошо аппроксимируется линейной моделью в окрестности решения. Однако, как и любой итеративный метод, он требует хороших начальных приближений для параметров, чтобы обеспечить сходимость к глобальному минимуму [3].

3 Реализация

В Python можно реализовать метод наименьших квадратов используя библиотеки. Ниже приведен простой пример с использованием NumPy:

```
import numpy as np
import matplotlib.pyplot as plt

x = np.array([1, 2, 3, 4, 5])
y = np.array([2, 3, 5, 7, 11])
```

```

A = np.vstack([x, np.ones(len(x))]).T
m, c = np.linalg.lstsq(A, y, rcond=None)[0]

plt.plot(x, y, 'o', label='данные', markersize=10)
plt.plot(x, m*x + c, 'r', label=f'y = {m:.2f}x + {c:.2f}')
plt.legend()
plt.show()

```

В этом коде показано, как с помощью метода наименьших квадратов подогнать линию к набору точек данных и визуализировать результаты. На рисунке 1 изображен график результата работы фрагмента кода. Полученная линия минимизирует сумму квадратов расстояний до точек.

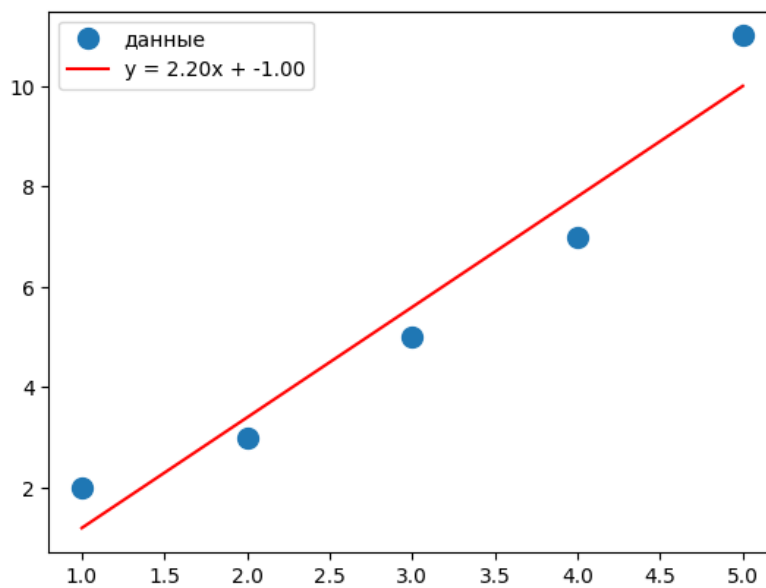


Рис. 1: Линейная регрессия

Ниже приведен пример кода на Python, который реализует метод Гаусса-Ньютона для нахождения параметров экспоненциальной модели.

```

import numpy as np
import matplotlib.pyplot as plt

x_data = np.array([0, 1, 2, 3, 4])
y_data = np.array([1, 2.7, 7.4, 20.1, 54.6])

a = 1.0
b = 0.5

def residuals(params, x, y):
    a, b = params
    return y - a * np.exp(b * x)

```

```

for _ in range(20):
    r = residuals((a, b), x_data, y_data)
    J = np.column_stack(
        (-np.exp(b * x_data), -a * x_data * np.exp(b * x_data))
    )
    delta = np.linalg.inv(J.T @ J) @ J.T @ r
    a -= delta[0]
    b -= delta[1]

x = np.linspace(0, 4, 100)
y = a * np.exp(b * x)
plt.scatter(x_data, y_data, label='данные', color='blue')
plt.plot(x, y, color='red', label=f'y = {a:.2f}*e^{(b:.2f)*x}')
plt.xlabel('x')
plt.ylabel('y')
plt.legend()
plt.show()

```

Метод Гаусса-Ньютона используется для минимизации суммы квадратов отклонений между наблюдаемыми значениями и предсказанными значениями модели, заданной уравнением $y = a \cdot e^{b \cdot x}$.

В процессе работы кода сначала инициализируются параметры a и b , затем вычисляются остатки между фактическими и предсказанными значениями. Далее формируется матрица Якоби, которая используется для обновления параметров в каждой итерации. После нескольких итераций получаются оптимальные значения параметров, которые минимизируют отклонения. На рисунке 2 отображаются исходные точки данных и полученная модель. Полученная кривая демонстрирует, как модель хорошо соответствует данным, минимизируя сумму квадратов расстояний до точек.

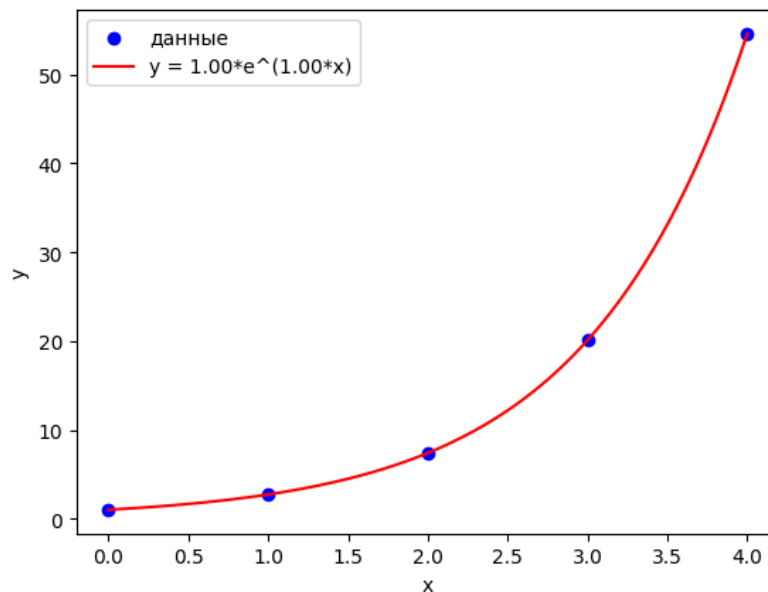


Рис. 2: Нелинейная регрессия

Заключение

Метод наименьших квадратов является мощным инструментом в области машинного обучения, предлагающим простой и эффективный подход к регрессионному анализу. Его основное предназначение заключается в минимизации суммы квадратов отклонений между наблюдаемыми и предсказанными значениями, что позволяет находить оптимальные параметры модели. Этот метод широко используется в различных областях, таких как экономика, физика и социология, где необходимо анализировать данные и делать прогнозы на основе исторической информации.

Преимуществом метода наименьших квадратов является его простота и интуитивная интерпретация, что позволяет выявлять зависимости между переменными. Он также служит основой для более сложных алгоритмов машинного обучения, таких как регрессионные деревья и нейронные сети.

Кроме того, метод легко реализуется с помощью различных программных библиотек, что делает его доступным для специалистов с разным уровнем подготовки. В условиях растущих объемов данных и потребности в точных прогнозах метод наименьших квадратов продолжает оставаться актуальным инструментом для анализа и обработки данных.

Список литературы

- [1] Магнус, Я. Р. Катышев П.К., Пересецкий А.А. Эконометрика. Начальный курс: Учеб. / Я. Р. Магнус, П. К. Катышев, А. А. Пересецкий – 6-е изд., перераб. и доп. – М.: Дело, 2004. – 576 с.
- [2] Вьюгин, В. В. Математические основы теории машинного обучения и прогнозирования / В.В. Вьюгин – М.: 2013. – 387 с.
- [3] Madsen K. – Methods for non-linear least squares problems / K. Madsen, H. B. Nielsen, O. Tingleff. – 2nd ed. – Informatics and Mathematical Modelling, Technical University of Denmark – 2004 – 58 p.