Доклад

Линейные модели

Шалыгин Георгий Эдуардович

Содержание

Список иллюстраций

Список таблиц

1 Цели и задачи

- Изучить линейные модели.
- Задачи:
  - Рассмотреть задачи, решаемые линейными моделями.
  - Изучить линейные модели и варианты их применения в задачах.
  - Исследовать методы улучшения линейных моделей.

2 Задачи

2.1 Регрессия

2.1.1 Постановка задачи

Задача контролируемого машинного обучения, которая прогнозирует значение метки по набору связанных компонентов. Метка здесь может принимать любое значение, а не просто выбирается из конечного набора значений, как в задачах классификации. Алгоритмы регрессии моделируют зависимость меток от связанных компонентов, чтобы определить закономерности изменения меток при разных значениях компонентов. На вход алгоритма регрессии подается набор примеров с метками известных значений. Результатом работы алгоритма регрессии является функция, которая умеет прогнозировать значения метки для любого нового набора входных компонентов. Вот несколько примеров для сценария регрессии: - прогнозирование цен на дома по таким атрибутам, как количество комнат, расположение и размер; - прогнозирование будущей цены акций на основе исторических данных и текущих тенденций рынка; - прогнозирование продаж товара в зависимости от рекламного бюджета. В итоге, рассматривается регрессионная модель зависимости одной (объясняемой, зависимой) переменной y от другой или нескольких других переменных (факторов, регрессоров, независимых переменных) x_i с линейной функцией зависимости: $M : \mathbb{X} \to \mathbb{R}$. Подробнее в [1].

2.1.2 Модель для задачи регрессии

По набору $x_1, x_2, ..., x_d$ предсказываем y. Модель описывается формулоой

$$\tilde{y} = w_1 x_1 + ... + w_d x_d + w_0$$

для какого-то фиксированного набора w_i. Изначально, предполагаем, что $\forall y_i$: $y_i = \tilde{y} + \epsilon$, где ошибка ϵ описывается нормальным распределением. Пример регрессии для двух переменных приведен на fig. 2.1.

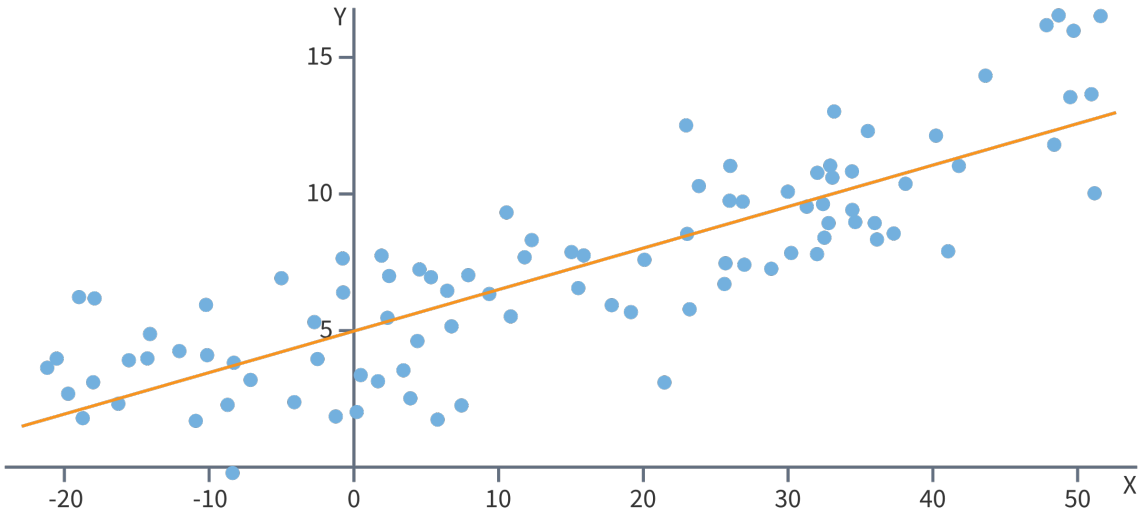

Рис. 2.1: График регрессии

2.2 Классификация

2.2.1 Постановка задачи

Задача контролируемого машинного обучения, которая прогнозирует распределение элементов данных по двум классам (категориям). На вход алгоритма классификации подается набор примеров с метками, каждая из которых представляет собой целое число 0 или 1. Результатом работы алгоритма двоичной

классификации является классификатор, который умеет прогнозировать класс для новых экземпляров без метки. Вот несколько примеров для сценария двоичной классификации:

- Распределение комментариев Twitter по тональности — позитивные или негативные.
- Диагностика пациента на наличие определенной болезни.
- Принятие решений о присвоении отметки “спам” сообщению электронной почты.
- Определение того, содержит ли фотография определенный элемент, например изображение собаки или фрукта.

Для получения наилучших результатов обучения двоичной классификации обучающие данные должны быть сбалансированы (т. е. число положительных и отрицательных обучающих данных должно быть одинаковым). Отсутствующие значения необходимо обработать до обучения. Дополнительные сведения см. в [2]. ### Модель для задачи классификации

Формально, строится отображение $M : \mathbb{X} \to 1, 2, \ldots, k$, k – количество классов. Рассмотрим самый распространенный случай $k = 2$. Для решения проблемы задача регрессии может быть сформулирована иначе: вместо предсказания бинарной переменной мы предсказываем непрерывную переменную со значениями на отрезке $[0, 1]$ при любых значениях независимых переменных. Это достигается применением следующего регрессионного уравнения $sigm(\tilde{y})$, где $sigm(t) = \dfrac{1}{1 + e^{-t}}$. Функция приведена на fig. 2.2.

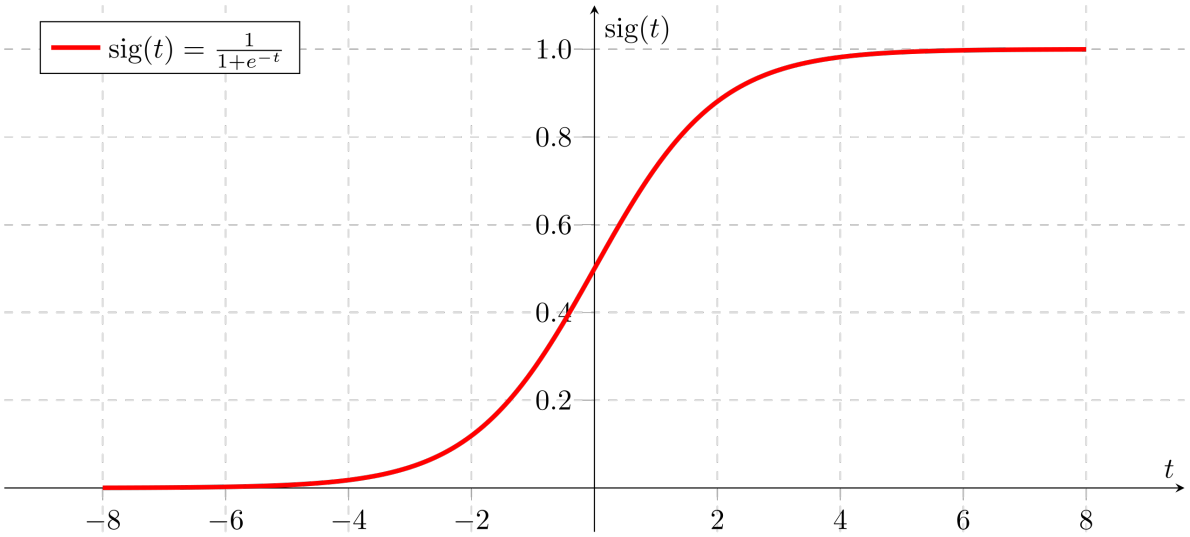

Рис. 2.2: Функция sigm

Данный вид классификации называется логистической регрессией.

3 Обучение линейных моделей.

3.1 Метод наименьших квадратов для линейной регрессии

Метод наименьших квадратов (МНК) — математический метод, применяемый для решения различных задач, основанный на минимизации суммы квадратов отклонений некоторых функций от экспериментальных входных данных. Он может использоваться для «решения» переопределенных систем уравнений (когда количество уравнений превышает количество неизвестных), для поиска решения в случае обычных (не переопределенных) нелинейных систем уравнений, для аппроксимации точечных значений некоторой функции. МНК является одним из базовых методов регрессионного анализа для оценки неизвестных параметров регрессионных моделей по выборочным данным. Сущность МНК (обычного, классического) заключается в том, чтобы найти такие параметры w_i, при которых сумма квадратов отклонений (ошибок, для регрессионных моделей их часто называют остатками регрессии) $MSE = (\tilde{y} - y)^2$ будет минимальной.

3.1.1 Аналитический подход

Минимум функции находится с помощью градиента. Получившееся решение имеет вид:

$$w = (X^T X)^{-1} X^T y$$

3.1.2 Градиентный спуск

Градиентный спуск — это алгоритм оптимизации, используемый для минимизации ошибок в модели машинного обучения. Он работает путем итеративной корректировки параметров модели в направлении отрицательного градиента функции потерь (которая представляет ошибку), чтобы уменьшить ошибку и найти оптимальные параметры, которые дают наилучшие результаты прогнозирования. Алгоритм продолжает этот процесс до тех пор, пока он не достигнет минимума или не будет выполнен заранее определенный критерий остановки. Описывается формулой:

$$w_j = w_j - \alpha \frac{d}{dw_j} MSE$$

Код алгоритма:

```
# Batch Gradient Descent
import numpy as np
eta = 0.1 # learning rate
n_iterations = 1000
m = 100

X = 2 * np.random.rand(100, 1)
y = 4 + 3 * X + np.random.randn(100, 1)
X_b = np.c_[np.ones((100, 1)), X]  # add x0 = 1 to each instance
theta = np.random.randn(2,1) #random initilization
for iteration in range(n_iterations):
    gradients = 2/m * X_b.T.dot(X_b.dot(theta)-y)
    theta = theta - eta * gradients
print(theta)
```

Подробнее о методах обучения и реализации моделей в [3].

4 Регуляриза

4.1 Обучение логистической регрессии для бинарной классификации

Задача обучения линейного классификатора заключается в том, чтобы по выборке X настроить вектор весов w. В логистической регрессии для этого решается задача минимизации эмпирического риска с функцией потерь специального вида:

$$Q(w) = \sum_{i=1}^{m} \ln(1 + exp(-y_i \langle x_i, w \rangle)) \to min_w$$

Задача минимизации решается любыми вычислительными методами, напрмер, описанным выше градиентным спуском.

Подробнее о методах обучения и реализации моделей в [3].

5 Регуляризация

5.1 Описание

Регуляризация (англ. regularization) в статистике, машинном обучении, теории обратных задач — метод добавления некоторых дополнительных ограничений к условию с целью решить некорректно поставленную задачу или предотвратить переобучение. Чаще всего эта информация имеет вид штрафа за сложность модели. Однин из способов бороться с негативным эффектом излишнего подстраивания под данные — использование регуляризации, т. е. добавление некоторого штрафа за большие значения коэффициентов у линейной модели. Тем самым запрещаются слишком “резкие” изгибы, и предотвращается переобучение. ## Виды регуляризации Переобучение в большинстве случаев проявляется в том, что итоговые модели имеют слишком большие значения параметров. Соответственно, необходимо добавить в целевую функцию штраф за это. Наиболее часто используемые виды регуляризации — L_1 и L_2, а также их линейная комбинация — эластичная сеть. - L_1:

$$Q(w, X) = \lambda \sum_{j=1}^{n} w_j^2$$

Минимизация регуляризованного соответствующим образом эмпирического риска приводит к выбору такого вектора параметров w, которое не слишком сильно отклоняется от нуля. В линейных классификаторах это позволяет избежать

проблем мультиколлинеарности и переобучения. - L_2:

$$Q(w, X) = \lambda \sum_{j=1}^{n} |w_j|$$

Данный вид регуляризации также позволяет ограничить значения вектора w. Однако, к тому же он обладает интересным и полезным на практике свойством — обнуляет значения некоторых параметров, что в случае с линейными моделями приводит к отбору признаков. Для задачи линейной регрессии в этом случае существует аналитическое решение в виде

$$w = (X^T X + \lambda I)^{-1} X^T y$$

Дальнейшие сведения можно получить в [4].

6 Выводы

В итоге были рассмотрены линейные модели, имеющие простой вид, высокий уровень интерпретируемости, множество алгоритмов построения и применяющиеся в огромном числе задачах, сводимых к классификации и регрессии.

Список литературы

1. Демиденко Е.З. Линейная и нелинейная регрессия. М.: Финансы и статистика, 1981.
2. Ng A. Stanford CS229 Lecture Notes. 2010. 324 с.
3. Винниченко М.Ю. ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ ЯДЕРНЫХ МОДЕЛЕЙ ГЛУБОКОГО ОБУЧЕНИЯ. НИУ ВШЭ, 2021.
4. Воронцов К.В. Математические методы обучения по прецедентам. ВЦ им. А.А. Дородницына РАН ФИЦ «Информатика и управление», 2004.