Обучение с учителем

Санкт-Петербургский государственный университет
Кафедра статистического моделирования

16 сентября 2025, Санкт-Петербург

Введение

Машинное обучение — это раздел искусственного интеллекта, в котором разрабатываются методы и алгоритмы, позволяющие компьютерам обнаруживать закономерности в данных и делать прогнозы без явных инструкций.

Обучение с учителем — один из способов машинного обучения, в ходе которого для каждого примера в обучающем наборе известно, какой результат является правильным.

Пример задач:

- Регрессия: предсказание стоимости недвижимости, количества продаж некоторого товара, погоды.
- Классификация: предсказание ценовой категории товара, типа изображения, болеет ли человек или нет.

Постановка задачи

Дано:

1. Пространство объектов X — множество описаний объектов (например, фотографии, тексты, таблицы с признаками).
2. Пространство ответов Y — множество меток или значений, которые нужно предсказывать (например, классы «кот»/«собака», цена товара).
3. Обучающая выборка $D = \{(x_i, y_i)\}_{i=1}^{n}$, где $x_i \in X$, $y_i \in Y$.

Модель:

$$y = f(x) + \varepsilon,$$

где $f(x)$ — некоторая фиксированная (но неизвестная) функция, ε — шум, $\mathsf{E}\varepsilon = 0$ и ε не зависит от x.

Предположение: $f(x)$ лежит в некотором классе функций (например, в классе линейных функций).

Задача: по обучающей выборке D построить оценку $\hat{f}(x)$ функции $f(x)$ в выбранном классе функций.

Функция потерь и ее минимизация

Чтобы оценить, насколько хорошо модель предсказывает ответы, используется **функция потерь** $L(y, \hat{y})$. Она показывает, насколько велико расхождение между истинными значениями y и его предсказаниями $\hat{y}$.

Тогда задача машинного обучения — минимизация выбранной функции потерь:

$$L(y, \hat{y}) \longrightarrow \min .$$

В большинстве случаев вычислить точку минимума функции потерь аналитически не представляется возможным, поэтому для его нахождения прибегают к методам **детерменированной и стохастической оптимизации** (например, перебор значений по сетке, метод Ньютона и квазиньютоновские методы, (стохастический) градиентный спуск, случайный поиск). Наиболее распространенный — градиентный спуск.

Градиентный спуск

Пусть $f(x)$ — некоторая гладкая функция, у которой необходимо найти минимум. Обозначим $p_n = -\nabla f(x_n)$ — направление антиградиента в точке x_n. Тогда

$$x_{n+1} = x_n + \alpha p_n,$$

где α — гиперпараметр, отвечающий за скорость обучения.

Условия сходимости: выпуклость $f(x)$, липшицевость $\nabla f(x)$, ...

В машинном обучении:

$$f(x) = \frac{1}{n} \sum_{i=1}^{n} f_i(x).$$

Вместо того, чтобы на каждой итерации оптимизировать f, можно случайным образом разыгрывать индекс i и вычислять шаг с $p_n = -\nabla f_i(x_n)$ — стохастический градиентный спуск.

Процесс обучения и проверка качества модели

Процесс обучения любого алгоритма машинного обучения выглядит следующим образом:

1. Выборка D предварительно разбивается на тренировочную и тестовую: $D = D_{\text{train}} \sqcup D_{\text{test}}$ (также часто присутствует и валидационная выборка D_{val}, с помощью которой подбираются гиперпараметры модели).
2. На тренировочных данных модель обучается: минимизируется выбранная функция потерь $L(y, \hat{y})$.
3. После обучения проверяется качество/обобщающая способность модели — на тестовых данных вычисляются различные метрики (например, MSE и MAE в задаче регрессии, accuracy и ROC AUC в задаче классификации).
4. Также имеет смысл сравнить полученные результаты с baseline предсказаниями (например, среднее в задаче регрессии и наиболее распространенная метка в задаче классификации).

Линейная классификация

Пусть целевая переменная y принимает значения $\{-1, 1\}$. Хотим обучить линейную модель так, чтобы плоскость, которую она задает, как можно лучше отделяла объекты одного класса от другого.

Линейный классификатор:

$$\hat{y} = \hat{a}(x; w) = \operatorname{sign}\langle x, w \rangle.$$

Функция потерь:

$$L(y, \hat{y}) = \sum_{i=1}^{n} \mathbb{I}[y_i \langle x_i, w \rangle < 0] \longrightarrow \min_{w}.$$

Линейная классификация. Отступ

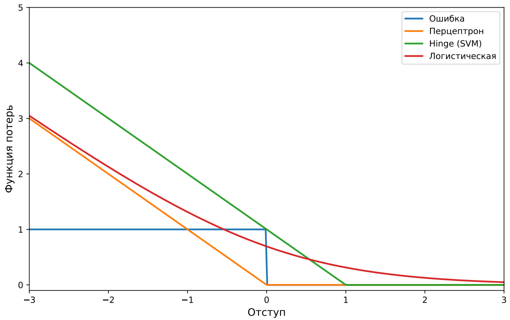
Величина $M_i = y_i\langle x_i, w\rangle$ называется **отступом** (margin) классификатора. Абсолютная величина отступа говорит о степени уверенности классификатора.

Проблема: функция $\mathbb{I}[M < 0]$ кусочно-постоянная, следовательно функцию потерь невозможно оптимизировать градиентными методами, поскольку во всех точках производная равна нулю.

Решение: можно мажорировать эту функцию более гладкой функцией и минимизировать функцию потерь с этой мажорирующей функцией с помощью методов численной оптимизации.

Линейная классификация. Функции потерь

1. Перцептрон: $L(M) = \max(0, -M)$ — отступы учитываются только для неправильно классифицированных объектах пропорционально величине отступа.
2. Hinge (SVM): $L(M) = \max(0, 1 - M)$ — объекты, которые классифицированы правильно, но не очень «уверенно», продолжают вносить свой вклад в градиент.
3. Логистическая: $L(M) = \ln\left(1 + e^{-M}\right)$.



Логистическая регрессия

Посмотрим на задачу классификации как на задачу предсказания вероятностей (например, предсказание «кликабельности» рекламного баннера).

Принцип работы: научить линейную модель предсказывать значения $z \in \mathbb{R}$ (логиты), а затем преобразовывать их в вероятности с помощью сигмоиды:

$$z_i = \langle x_i, w \rangle = \ln \frac{p_i}{1 - p_i}, \quad p_i = \frac{1}{1 + e^{-\langle x_i, w \rangle}} = \sigma(\langle x_i, w \rangle).$$

Функция правдоподобия для распределения Бернулли:

$$p(y \mid \mathbf{X}, w) = \prod_{i=1}^{n} p_i^{y_i} (1 - p_i)^{1 - y_i}.$$

Прологарифмируем:

$$\sum_{i=1}^{n} \Big[y_i \ln(\sigma(\langle x_i, w \rangle)) + (1 - y_i) \ln(1 - \sigma(\langle x_i, w \rangle)) \Big].$$

Теперь пусть $y \in \{-1, 1\}$. Тогда, поскольку $\sigma(z) = 1 - \sigma(-z)$, логарифм правдоподобия можно представить в следующем виде:

$$\begin{aligned}
\ln p(y \mid \mathbf{X}, w) &= -\sum_{i=1}^{n} \Big[\mathbb{I}[y_i = 1]\sigma(z_i) + \mathbb{I}[y_i = -1] \left(1 - \sigma(z_i)\right) \Big] \\
&= -\sum_{i=1}^{n} \ln \sigma(y_i \langle x_i, w \rangle) \\
&= \sum_{i=1}^{n} \ln \left(1 + e^{-M}\right)
\end{aligned}$$

Таким образом, функцию потерь в логистической регрессии можно представить в виде функции от отступа.