Функция потерь

Задача машинного обучения не будет полной без *функции потерь* (loss function) или *метрики качества* – эти понятия обычно взаимозаменяемы. Для каждого объекта и для каждого допустимого ответа функция потерь говорит, насколько этот ответ далёк от истины. Для каждой задачи машинного обучения можно придумать очень много разных метрик качества, в этом разделе мы приведём только несколько наиболее распространённых.

Регрессия: оценка стоимости квартир

Модуль отклонения. Допустим, реальная арендная плата за квартиру 40 тысяч рублей, а наш алгоритм предсказывает, что 30 тысяч. Насколько этот ответ далёк от правильного? Иными словами, какой штраф назначить за такой ответ? Один из разумных ответов – 10 тысяч, а в общем случае – модуль разности реального ответа и ответа алгоритма. То есть, если a – предсказанная арендная плата, а y – реальная арендная плата, то функция потерь равняется $L(y, a) = |y - a|$.

Квадрат отклонения. Как правило, в задачах регрессии функция потерь -- это некоторая функция от отклонения от реального ответа. Например, часто используется квадратичная функция потерь: $L(y, a) = (y - a)^2$. Её можно сравнить со шкалой прогрессивного налогообложения: если отклонение от правильного ответа вырастет в 2 раза, то штраф вырастет в 4 раза, если отклонение вырастет в 3 раза, то штраф вырастет в 9 раз и т.д. Квадрат отклонения в прикладных задачах используется даже чаще, чем просто отклонение, поскольку он всюду *дифференцируем* – а это как раз то свойство, которое позволяет нам считать производные и находить минимум / максимум.

Представьте, что вы разрабатываете алгоритм, который определяет возраст человека по фотографии. На вход вашему алгоритму подаётся фотография 30-летнего человека, предсказание вашего алгоритма – 26 лет. Каким будет штраф для такого ответа, если вы используете квадратичную функцию потерь?

Введите численный ответ

Введите число

Функция потерь

Бинарный классификатор: пёсики и кексики

Индикаторная функция потерь. Самый простой способ задать функцию потерь в задаче бинарной классификации – это индикатор того, что класс определён правильно. Напомним, что метки 0 и 1 соответствуют пёсикам и кексикам. Тогда положим $L(0,0) = L(1,1) = 1$ и для всех остальных аргументов $L(y,a) = 0$. Однако, такая функция не будет даже *непрерывной,* что усложняет задачу оптимизации, поскольку это условие необходимо для дифференцируемости. С понятием непрерывности мы познакомимся в части программы, посвящённой математическому анализу.

Обозначение. Для такой функции есть удобное обозначение $\mathbf{1}\{y\ = a\}$ – индикатор события. Индикаторная функция от некоторого логического высказывания равна 1, если это высказывание истинно, и 0, если ложно. В нашем случае $\mathbf{1}\{y\ = a\}$ равно 1 если равенство $y = a$ истинно, и $\mathbf{1}\{y\ = a\}$ равно 0 если равенство $y = a$ ложно, то есть если $y \neq a$.

Предсказание вероятности. Удобный способ перейти от разрывной к непрерывной функции потерь в задачах бинарной классификации это исправить целевую функцию: будем предсказывать не класс, а *вероятность* того, что на фотографии изображён кексик. Из предсказания вероятности несложно сделать предсказание конкретного класса: например, если алгоритм предсказывает, что с вероятностью больше 50% на фотографии изображён кексик, то мы будем возвращать 1, а в противном случае 0. Таким образом, мы свели задачу классификации к задаче регрессии, и в качестве функции потерь можем использовать, например, квадрат отклонения. На практике обычно используются другие функции, специфичные именно для предсказания вероятности, но их мы трогать пока не будем.

Роль данных в машинном обучении

Как правило, машинное обучение применяется в тех случаях, когда явно задать целевую функцию достаточно сложно: мы сможем отличить пёсика от кексика почти на любой конкретной фотографии, но объяснить инопланетянину, как отличить фотографию кексика от фотографии собачки будет проблематично: скорее всего, если нам пришлось бы чем-то таким заниматься, то мы бы делали это при помощи примеров.

Обучающая выборка – это набор размеченных данных, то есть набор объектов, для которых известно значение целевой функции. Размеченные данные можно предствлять себе как способ описания целевой функции.

Функция потерь для всей выборки целиком

Как правило, функцию потерь вычисляют для всей выборки целиком: точность предсказания на одном объекте мало о чём говорит. Получить функцию потерь для выборки целиком можно, например, усреднив штрафы на каждом объекте по отдельности.

Регрессия

Пронумеруем наши объекты числами от 1 до n, пусть для этих объектов значения целевой функции это $y_1, y_2, \dots, y_n$ соответственно, а предсказание нашего алгоритма это $a_1, a_2, \dots, a_n$ соответственно.

Пример. Пусть наши объекты это квартиры, а целевая функция – цена аренды квартиры. Пусть 13-ая квартира из нашего списка имеет цену аренды 42.500. А наш алгоритм для этой квартиры предсказал стоимость аренды равную 38.000. Тогда $y_{13} = 42.500$ и $a_{13} = 38.000$.

Mean absolute error (MAE). Среднее отклонение по модулю – это просто среднее арифметическое модулей отклонений $MAE(y_1, y_2, \dots, y_n, a_1, a_2, \dots, a_n) := \frac{1}{n}(|y_1 - a_1| + |y_2 - a_2| + \dots + |y_n - a_n|)$.

Mean squared error (MSE). Аналогично, среднеквадратичная ошибка $MSE(y_1, y_2, \dots, y_n, a_1, a_2, \dots, a_n) := \frac{1}{n}((y_1 - a_1)^2 + (y_2 - a_2)^2 + \dots + (y_n - a_n)^2)$.

Комментарий. Символ ":=" обозначает "по определению равно". То есть выше мы определили, что мы называем MAE и MSE.

Предположим, в нашей обучающей выборке 4 фотографии, на которых изображены люди 20, 25, 30 и 40 лет. Пусть для этих фотографий наш алгоритм предсказывает возраст 21, 25, 27 и 45 лет соответственно. Вычислите среднеквадратичную ошибку такого предсказания.

(Выборка из 4 объектов — это смешно для человека знакомого с машинным обучением, но мы намеренно ограничиваемся игрушечными примерами, чтобы их можно было пощупать руками)

Введите численный ответ

Введите число

Бинарная классификация

Пусть в нашей обучающей выборке n объектов и $y_1, y_2, \ldots, y_n$ – их классы. Пусть наш классификатор предсказал для этих объектов классы $a_1, a_2, \ldots, a_n$ соответственно.

Точность (accuracy). Одна из классических метрик в задаче бинарной классификации – доля правильных ответов:
$Acc(y_1, y_2, \ldots, y_n, a_1, a_2, \ldots, a_n) = \frac{1}{n}(\mathbf{1}\{y_1 = a_1\} + \ \mathbf{1}\{y_2 = a_2\ \} + \cdots + \ \mathbf{1}\{y_n = a_n\ \})$.

Напоминание. Выражение $\mathbf{1}\{y_i = a_i\}$ равно 1, если равенство $y_i = a_i$ истинно (то есть, когда ответ классификатора на i-ом объекте верный). И $\mathbf{1}\{y_i = a_i\}$ равно 0, если равенство $y_i = a_i$ ложно, то есть если $y_i \neq a_i$ (ответ классификатора на i-ом объекте неверный).

Тем самым, число $\mathbf{1}\{y_1 = a_1\} \ + \cdots + \ \mathbf{1}\{y_n = a_n\ \}$ это количество правильных ответов классификатора. А точность — это отношение количества правильных ответов классификатора к размеру выборки.

Задача-капча. Допустим, в нашей обучающей выборке 5 фотографий:

Источник: гитхаб датасета

И наш классификатор для этих пяти фотографий выдает метки классов $1, 1, 1, 0, 0$. Напомним, метка 0 соответствует пёсику, а 1 – кексику. Вычислите точность такого классификатора в процентах.

Введите численный ответ

Введите число