Алгоритмы, вычисляющие вероятность принадлежности классам
Логистическая регрессия

Лекция 9

Алгоритм выдает не метку класса, а вероятность принадлежности классам

Пример предсказания

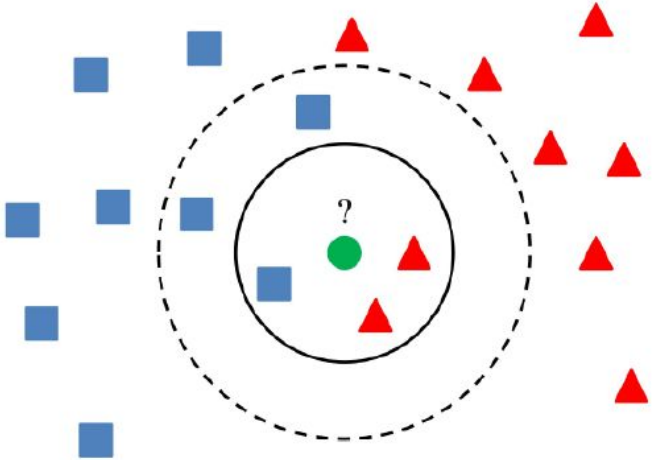
Объект	Рост	Вес	Пол (0-ж, 1-м) (предсказанный)
A	180	70	0.75
B	150	45	0.1

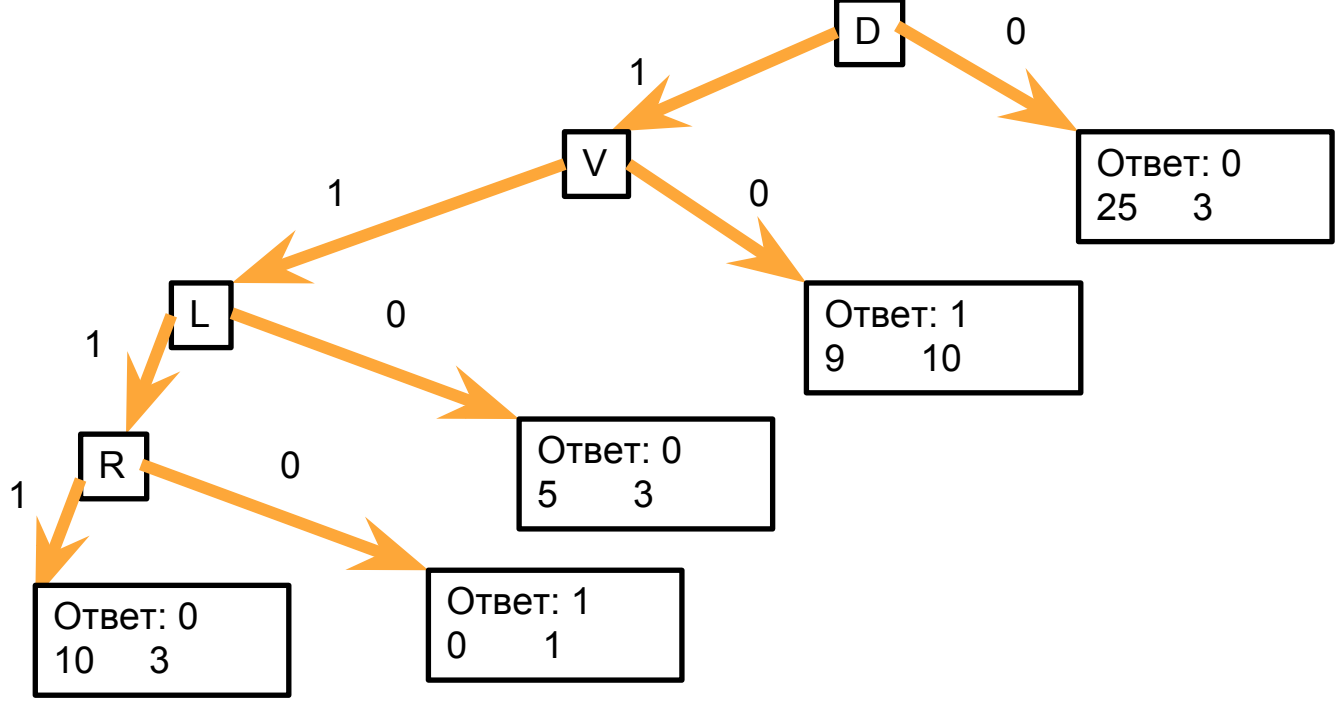
Примеры таких алгоритмов

1. Наивный Байес
2. Логистическая регрессия (см. ниже)
3. модификация kNN (подумайте, какая?)
4. модификация дерева (подумайте, какая?)
5. …

Как модифицировать kNN?

Ваши предложения?





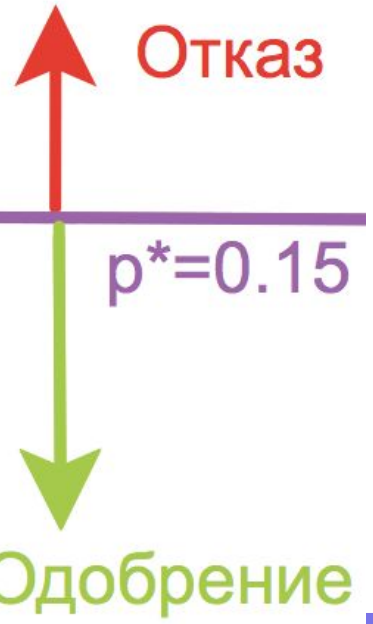
Как модифицировать дерево? Ваши предложения...

Зачем нужны такие алгоритмы предсказания?

1. Когда данные будут подаваться на вход другим алгоритмам (вероятность несёт в себе дополнительную информацию: нашу уверенность в классификации).
2. Возможность переложить ответственность на заказчика)))))

Классический пример (кредитный скоринг)

Клиент	Вероятность невозврата
Mike	0.78
Jack	0.45
Larry	0.13
Kate	0.06
William	0.03
Jessica	0.02



Показатели качества алгоритма, выдающего вероятности

Самый простой способ: округлить

Округляем ответы алгоритма до целых чисел

Объект	Рост	Вес	Пол (0-ж, 1-м) (предсказанный)	Ответ (округл)
A	180	70	0.75	1
B	150	45	0.1	0

А потом считать для них precision, recall … (см. соотв. тему)

Но:

такой способ не является оптимальным, особенно когда

1)классы не сбалансированы (число объектов одного класса много больше объектов другого класса);

2)цены ошибок за неправильные классификации объектов класса 1 и класса 0 различны.

Для вероятностей используют свои показатели качества

1. ROC-AUC
2. log-loss
3. ….

На чем основаны эти величины?

Качество классификации в первом приближении

Пусть алгоритм выдал оценки, как показано в табл. 1. Упорядочим строки табл. 1 по убыванию ответов алгоритма – получим табл. 2. Ясно, что в идеале её столбец «класс» тоже станет упорядочен (сначала идут 1, потом 0); в случае «слепого угадывания» будет случайное распределение 0 и 1.

Так вот: число нарушений порядка в табл. 2 определяет качество алгоритма.

id	оценка	класс
1	0.5	0
2	0.1	0
3	0.2	0
4	0.6	1
5	0.2	1
6	0.3	1
7	0.0	0

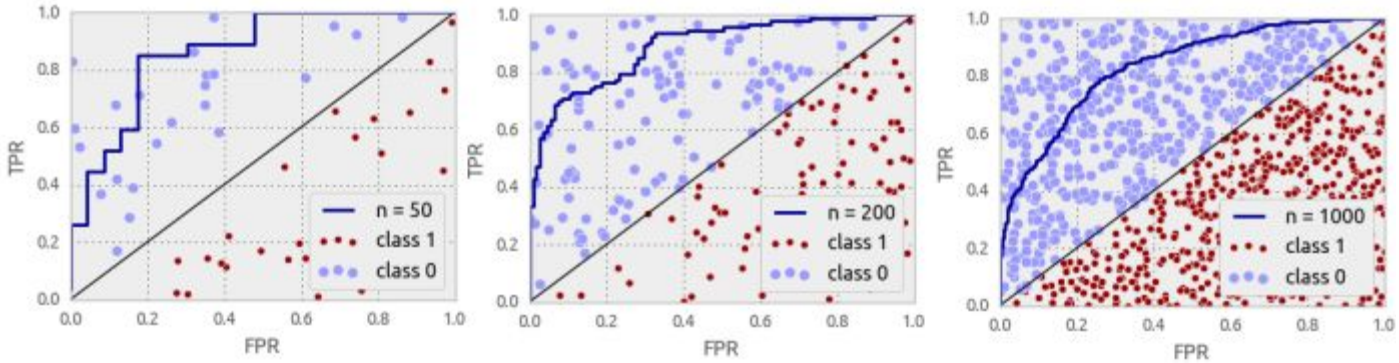
Табл. 1

id	оценка	класс
4	0.6	1
1	0.5	0
6	0.3	1
3	0.2	0
5	0.2	1
2	0.1	0
7	0.0	0

Табл. 2

ROC-кривая

Выглядит примерно так (синяя синяя):

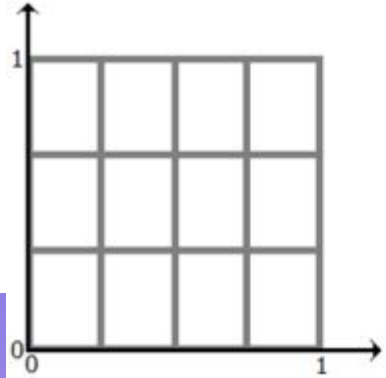


Чем больше площадь (AUC=area under curve) под ней – тем лучше.

А как строить ROC-кривую?

Строим ROC-кривую

Как и все показатели качества ROC-кривая строится по тестовой выборке.
Пусть n –число нулей в тестовой выборке, m – число единиц. Надо взять единичный квадрат на координатной плоскости, разбить его на m равных частей горизонтальными линиями и на n – вертикальными

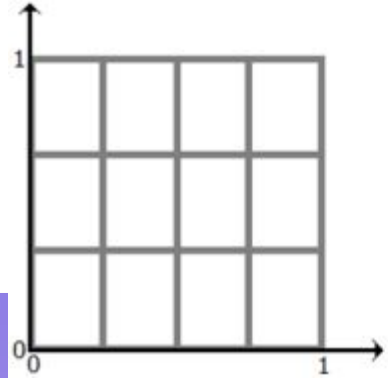


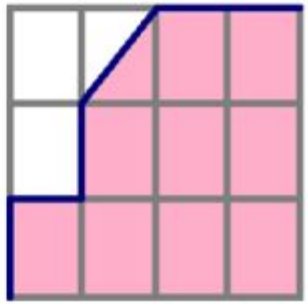
id	оценка	класс
4	0.6	1
1	0.5	0
6	0.3	1
3	0.2	0
5	0.2	1
2	0.1	0
7	0.0	0

Табл. 2

Строим ROC-кривую

Теперь будем просматривать строки в таблице сверху вниз и прорисовывать на сетке линии, переходя их одного узла в другой. Стартуем из точки (0, 0). Если значение метки класса в просматриваемой строке 1, то делаем шаг вверх; если 0, то делаем шаг вправо.



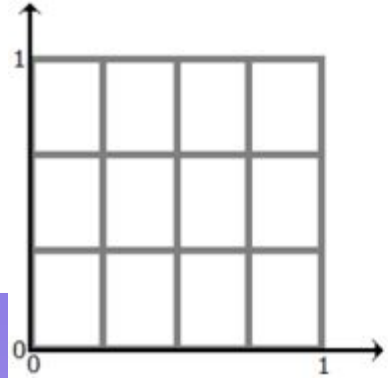


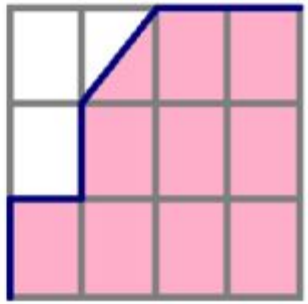
id	оценка	класс
4	0.6	1
1	0.5	0
6	0.3	1
3	0.2	0
5	0.2	1
2	0.1	0
7	0.0	0

Табл. 2

Важное правило:

если у нескольких объектов значения меток равны, то мы делаем шаг в точку, которая на a блоков выше и b блоков правее,
где a – число единиц в группе объектов с одним значением метки, b – число нулей в ней.





id	оценка	класс
4	0.6	1
1	0.5	0
6	0.3	1
3	0.2	0
5	0.2	1
2	0.1	0
7	0.0	0

Табл. 2

Как выбрать порог p для округления с помощью ROC-кривой?

Смысл порога p: если вероятность для объекта выше p, то объект относят к классу 1.
Для этого нужно взять точку с прямой. Порог p для нее будет равен вероятности объекта, который использовался при построении кривой в эту точку.
Точка на рисунке соотв. порогу 0.3

id	оценка	класс
4	0.6	1
1	0.5	0
6	0.3	1
3	0.2	0
5	0.2	1
2	0.1	0
7	0.0	0

Табл. 2

А что значат координаты точки на ROC-прямой?

Значение по горизонтали: FPR (false positive rate), доля объектов класса 0, которые неверно классифицированы нашим алгоритмом
Значение по вертикали: TPR (true positive rate, recall), доля объектов класса 1, которые верно классифицированы нашим алгоритмом.

id	оценка	класс
4	0.6	1
1	0.5	0
6	0.3	1
3	0.2	0
5	0.2	1
2	0.1	0
7	0.0	0

Табл. 2

Логистическая регрессия (ЛР)

Несмотря на название (регрессия),

…ЛР предсказывает вероятности принадлежности классам, то есть решает задачу классификации.

Ради упрощения формул метки классов будут далее обозначаться {-1,1} , а не {0,1}.

Объект	X1	X2	Y
A	1	2	1
B	3	4	-1

Сразу на примере

Нужно найти оптимальные (в некотором смысле) числа w1,w2,w0 которые определят результат классификации (линейная модель):

если w1*X1+w2*X2+w0>0, то объект с признаками (X1,X2) относим к классу 1 (иначе к классу -1).

Поиск чисел wi

Числа w1,w2,w0 ищутся как минимум функции

$$\sum_{i=1}^{m} \ln\left(1 + e^{-y_i(w_1 x_1 + w_2 x_2 + w_0)}\right)$$

m – объем тренировочной выборки.
Почему нужно рассматривать именно такую функцию – см. ниже
В нашем примере нужно минимизировать

$$\ln\left(1 + e^{-1(w_1 1 + w_2 2 + w_0)}\right) + \ln\left(1 + e^{1(w_1 3 + w_2 4 + w_0)}\right)$$

Объект	X1	X2	Y
A	1	2	1
B	3	4	-1

Находим wi

С помощью разных численных методов мы находим значения wi, на которых функция принимает наименьшее значение.

На этом можно остановиться (алгоритм умеет предсказывать класс объекта), но можно его допилить до предсказания вероятности.

Чтобы перейти к вероятностям, нужно

старое правило:
если w1*X1+w2*X2+w0>0, то объект с признаками (X1,X2) относим к классу 1 (иначе к классу -1).

заменить на новое правило:

выдать вероятность принадлежности классу 1 равную σ(w1*X1+w2*X2+w0),
где σ(z) – сигмоидная функция (сигмоид)

Сигмоид

$$\sigma(z) = \frac{e^z}{e^z + 1}$$

Остался вопрос: откуда взялась эта функция

$$\sum_{i=1}^{m} \ln\left(1 + e^{-y_i(w_1 x_1 + w_2 x_2 + w_0)}\right)$$

Когда модель ошибается?

Вспомним правило классификации:

если w1*X1+w2*X2+w0>0, то объект с признаками (X1,X2) относим к классу 1 (иначе к классу -1).

Когда модель допускает ошибку на тренировочной выборке?
1) когда w1*X1+w2*X2+w0>0, но объект из класса -1
2) когда w1*X1+w2*X2+w0<0, но объект из класса 1

Когда модель ошибается?

Иными словами, модель ошибается на объекте тренировочной выборки, если

$$M=Y(w1*X1+w2*X2+w0)<0$$

здесь Y – значение целевого признака объекта.

Величина M называется отступом.

Что минимизировать?

Введем обозначение

$$[M < 0] = \begin{cases} 1, \text{если } M < 0 \\ 0, \text{если } M > 0 \end{cases}$$

тогда для объектов тренировочной выборки нужно минимизировать функцию

$$\sum_{i=1}^{m} [\mathrm{M_i} < 0] = \sum_{\mathrm{i}=1}^{\mathrm{m}} [y_i(w_1 x_1 + w_2 x_2 + w_0) < 0]$$

где Mi – отступ i-го объекта тренировочной выборки.

Что минимизировать?

Есть проблема: эта функция

$$\sum_{i=1}^{m}[\mathrm{M_i} < 0] = \sum_{\mathrm{i}=1}^{\mathrm{m}}[y_i(w_1 x_1 + w_2 x_2 + w_0) < 0]$$

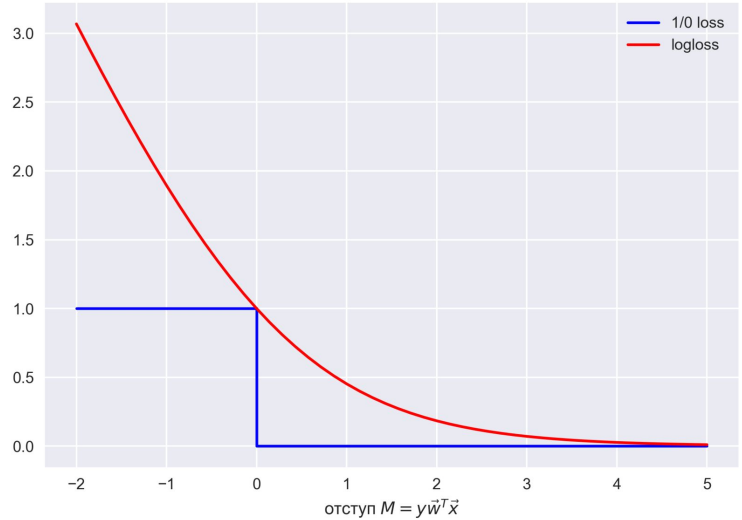
не дифференцируемая (нельзя взять производную). Искать минимум таких функций – это мазохизм.

Идея: найти дифференцируемую функцию, которая больше выражения [M<0].

Можно взять такую функцию

$$\ln(1 + e^{-M}) = \ln\left(1 + e^{-y_i(w_1 x_1 + w_2 x_2 + w_0)}\right)$$

Она действительно больше функции [M<0]



Ну теперь понятно, почему

… в логистической регрессии ищут минимум выражения

$$\sum_{i=1}^{m} \ln\left(1 + e^{-y_i(w_1 x_1 + w_2 x_2 + w_0)}\right) = \sum_{i=1}^{m} \ln\left(1 + e^{-M_i}\right)$$

есть и другие соображения (основанные на поиске максимума правдоподобия [1])

Использованная литература

1. https://habrahabr.ru/company/ods/blog/323890/#1-lineynaya-regressiya (про ЛР, сигмоид и отступ)
2. https://alexanderdyakonov.wordpress.com/2017/07/28/auc-roc-площадь-под-кривой-ошибок (про ROC-AUC)