Тестовое задание. Яндекс такси

Урванов Егор

11 апреля 2018 г.

1 Задача

Условие. Какая стратегия поведения в листьях решающего дерева приводит к меньшей вероятности ошибки: выдавать в ответе тот класс, который преобладает в листе, или отвечать случайно с тем же распределением классов, что и в листе? Считайте, что рассматривается задача многоклассовой классификации.

Решение. Для того, чтобы понять, как выгоднее всего поступать в листах деревьев, можно поступить несколькими способами. Первый - оценить число ошибок в произвольном листе при одной технике и при другой. Другим вариантом является численное моделирование числа ошибок в листах в каждом из случаев.

Рассмотрим случай, при котором результирующий класс в листах будет совпадать с преобладающим классом. Как частный случай, будем рассматривать два класса. По индукции это можно обобщить на случай m классов. Решающая функция будет выглядеть так:

$$a(x) = \arg \max_{0 \le i < m} (n_i)$$

n_i - количество объектов в $i-$ м классе для рассматриваемого листа. Всего объектов в листе n. Без потери общности, будем считать, что максимум указанной функции единственный.

Введем естественную функцию потерь, которая будет возвращать долю ошибок:

$$\text{Loss} = \frac{1}{n} \sum [y(x) - a(x) \neq 0] = \frac{1}{n}(n - \max_{0 \le i \le m} (n_i)) \tag{1}$$

Если рассматривается 2 класса, то получим:

$$\text{Loss} = \frac{1}{n} \sum [y(x) - a(x) \neq 0] = \frac{1}{n} \sum |y(x) - a(x)| = \frac{1}{n} \min(n_0, n_1)$$

Пример. Пусть в некотором листе соотношение объектов двух классов равно 25 к 75 для классов 0 и 1 соответственно. Тогда, очевидно, доля ошибок будет равна $\frac{25}{75+25} = \frac{1}{4}$.

Рассмотрим вторую ситуацию, когда в листе принимается решение об отнесетии к случайному классу с тем же распределением классов, что задают объекты, находящиеся в листе. Рассмотрим ситуацию для двух классов и по аналогии распространим на случай m классов. Пусть есть случайная величина, которая задает распределение Бернулли с параметром $p = \frac{n_1}{n_0 + n_1}$. Тогда:

$$a(x) \sim \text{Bi}(1, p)$$

В общем случае, при наличии m классов, будем иметь распределение:

$$\mathbb{P}(X = 0) = p_0$$

$$\mathbb{P}(X = 1) = p_1$$

$$\dots$$

$$\mathbb{P}(X = m - 1) = p_{m-1}$$

$$\sum_{i=0}^{m-1} p_i = 1$$

Для оценки доли ошибок, посчитаем матожидание:

$$E_n = \mathbb{E}_x \left[\text{loss(a)}\right] = \mathbb{E}_x \left[[y(x) - a(x) \neq 0]\right]$$

Для простоты записи, напишем:

$$\xi = \xi(x, y, a) = \begin{cases} 0 & y(x) = a(x) \\ 1 & y(x) \neq a(x) \end{cases}$$

Продолжая записи, сделанные выше, получим:

$$E_n = 1 \cdot \mathbb{P}(\xi = 1) + 0 \cdot \mathbb{P}(\xi = 0) = \mathbb{P}(\xi = 1) = \mathbb{P}(a(x) \neq y(x)) = \sum_{i=0}^{m-1} p_i(1 - p_i) = 1 - \mathbb{P}(\xi = 0) = 1 - \sum_{i=0}^{m-1} p_i^2$$

Для того, чтобы показать равенство:

$$\mathbb{P}(a(x) \neq y(x)) = \sum_{i=0}^{m-1} p_i(1 - p_i) = 1 - \sum_{i=0}^{m-1} p_i^2 \tag{2}$$

необходимо заметить, что коль скоро $a(x)$ принимает значение i, $y(x)$ не должен принимать значение i. По правилу умножения вероятностей, получаем:

$$p_i \cdot (p_0 + p_1 + \dots + p_{i-1} + p_{i+1} + \dots + p_{m-2} + p_{m-1}) = p_i \cdot (1 - p_i)$$

В силу несовместности событий, можем записать (2).

Сравнивая (2) и (1), легко заметить, что (1) - частный случай (2). Это будет так, коль скоро все, кроме одного максимального p_{max} положить равными нулю, а p_{max} положить равным 1.

Отсюда сразу же следует, что в случае (2) мы имеем большую долю ошибок, по сравнению с (1). А (1) является оптимальным.

Ответ: стратегия поведения в листьях решающего дерева приводит к меньшей вероятности ошибки, если выдавать в ответе тот класс, который преобладает в листе.

Для случае двух классов, можно провести численное моделирования доли ошибок. Результаты представлены выше.

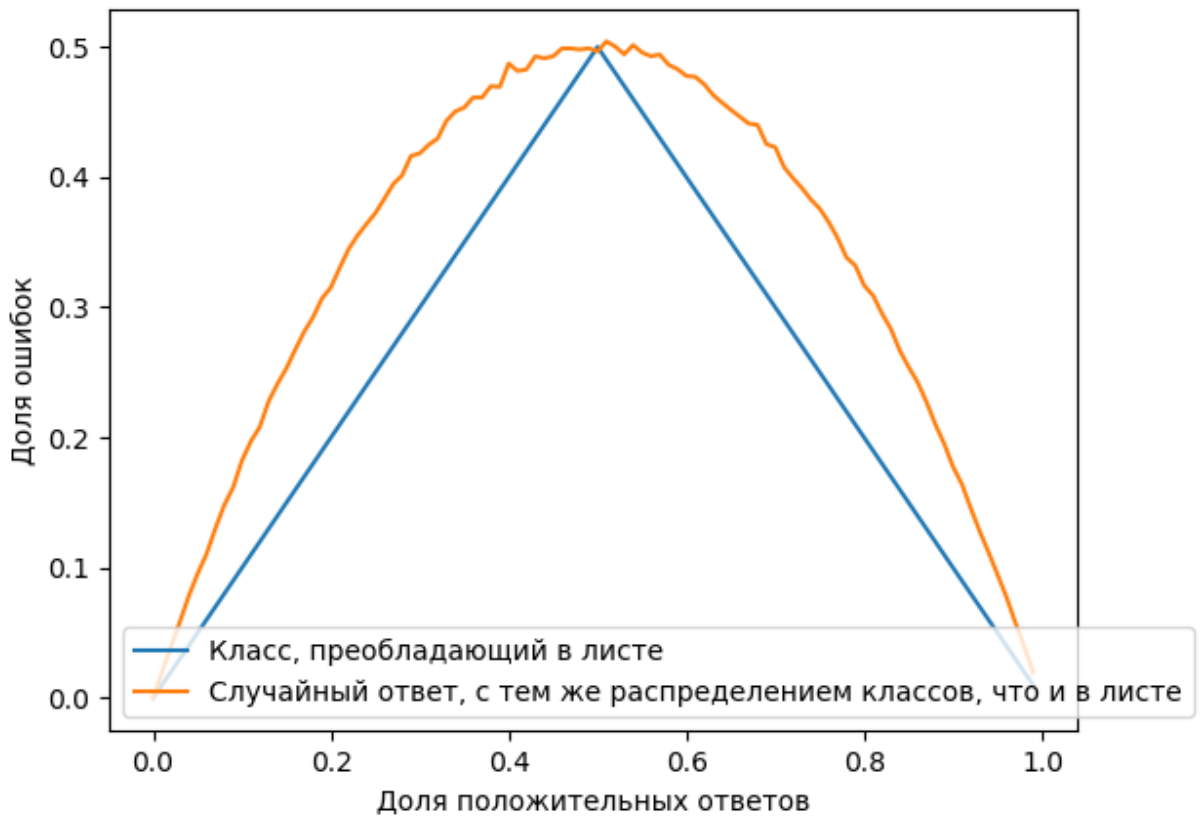

Рис. 1: Численное моделирование

Список литературы

[1] ЧЕМУ НЕ УЧАТ В АНАЛИЗЕ ДАННЫХ И МАШИННОМ ОБУЧЕНИИ. Дьяконов А.

[2] Репозиторий

2 Задача

Условие. При обучении SVM с линейным ядром на наборе данных с очень большим числом разреженных признаков точность многоклассовой классификации на тестовой выборке получилась равной 95%, а при обучении SVM с радиальным ядром — 34%. Почему SVM с более сложным ядром показал менее высокое качество? Когда можно ожидать от радиального ядра улучшения качества по сравнению с линейным?

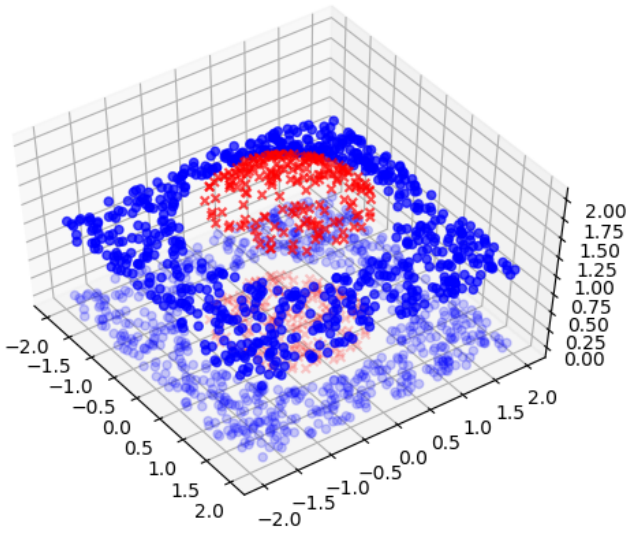

Рис. 2: Принцип работы SVM

Решение. Приведём классическую картинку, на которой показана основная идея работы SVM (рис. 2).

Выявить проблему однозначно по описанию, конечно, невозможно. Но следует предположить несколько фактов, которые с большой вероятностью являются причинными.

Одним из таких фактов является вычислительная сложность для SVM-RBF. В таком случае, есть вероятность, что алгоритм по просту не доучили ввиду долгого процесса обучения. Результатом того явилось сильное падение точности.

Можно увязать SVM-RBF с KNN-RBF и указать [1] связь с проклятием размерности. Теперь, рассматривая исходный алгоритм как KNN, можно сказать с некоторыми допущениями, что в пространствах достаточно большой размерности, для KNN все объекты расположены на большом расстоянии друг от друга [2], поэтому классификация может оказать затруднительной, а ответ далёким от реальности.

Другой возможной причиной является наличие большого количество выбросов. А при больших размерностях и разреженных признаках факт понижения точности лишь подогревается.

Приведём простой пример [4]. В рамках этого примера становится понятно, что в результате построения одной разделяющей поверхности в пространстве высокой размерности и попытке подогнаться частью ядер под шумы, достигается сильное понижение качества (рис. 2).

На рисунке выше чёрной точкой обозначен шумовой объект. Вокруг него сформировано ядро (здесь не совсем корректно оно показано, поскольку все ядра должны суммироваться; на текущей же картинке показана лишь область его действия), которая

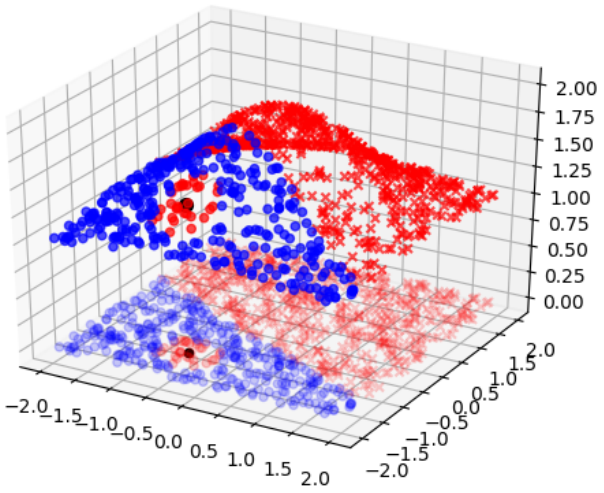

Рис. 3: Шумовая точка в SVM-RBF

заметает большое количество объектов другого класса. В то же время, при линейной классификации, такого эффекта наблюдаться не будет.

В данном случае совершенно неважно, рассматриваем задачу классификации на 2 класса или на m классов. Строя m бинарных классификаторов и используя принцип "один против всех сведём задачу мультиклассификации к задаче бинарной классификации.

Список литературы

[1] Ядра и их применение в машинном обучении. Евгений Соколов. Часть 5. Семинар 8. 14 ноября 2014 г.

[2] Семинары по метрическим методам. Евгений Соколов. 16 сентября 2016 г.

[3] Математические методы обучения по прецедентам (теория обучения машин). Воронцов К. В. Часть. 4.5

[4] Asymptotic Behaviors of Support Vector Machines with Gaussian Kernel. S. Sathiya Keerthi. Chih-Jen Lin

[5] Репозиторий

3 Задача

Условие. В одном проекте заказчик очень хотел, чтобы исследователь решал не поставленную задачу классификации на классы 0 и 1, а задачу регрессии на тех же метках с модулем отклонения в качестве функции потерь. Замысел заказчика был в том, что оцененные числа получатся дробными, и это будет приближением для вероятности класса 1. Считая, что алгоритм старается минимизировать математическое ожидание потерь при условии известного объекта x, выясните, какими будут получаться прогнозы при такой функции потерь и насколько они будут соответствовать замыслу заказчика. *Решение.* Данный подход некорректен, как минимум потому что мы будем иметь оценки, выдаваемые за вероятности, значения которых могут быть больше 1 и меньше 0. В этой задаче речь идёт о том, чтобы минимизировать сумму модулей всех невязок. В случае линейной регрессии, происходит минимизация функционала:

$$E(y|x) \to \min$$

В этом же случае, происходит минимизация функционала медианы:

$$\mathrm{Med}(y|x) \to \min$$

В нашем случае, можно показать, что в отличие от классической регрессии, где вводится предположение о нормальном законе распределения ошибок, здесь имеем ошибки, распределённые по закону Лапласса. Для демонстрации этого факта стоит воспользоваться методом максимального правдоподобия.

4 Задача

Условие. Какой должна быть функция потерь в предыдущей задаче, чтобы действительно оценивались вероятности? Покажите, что это так. *Решение.*

Для того, чтобы получать, действительно, вероятности, необходимо рассмотреть метод максимального правдоподобия и попробовать оценить вероятность y_i

$$p(y|X,w) = \prod_i p(y|x_i,w) = a^{y_i}(1-a)^{1-y_i} \to \max$$

Прологарифмируем функцию правдоподобия:

$$-y_i \log a_i - (1-y_i)\log(1-a_i) \to \min$$

Взяв производную и приравняв к нулю, получим, что в нашей формуле
$a_i = y_i$,
что и требовалос доказать.
Таким образом нужная функция потерь - logloss

5 Задача

Условие. Когда в задаче бинарной классификации предпочтительней использовать ROC-AUC, а не accuracy? Как изменится ROC-AUC решения задачи бинарной классификации, если умножить все прогнозы на 2? *Решение.* ROC-AUC, в отличие от accuracy устойчив к дисбалансу классов. В таком случае, если в задаче наблюдается сильный дисбаланс классов, следует относиться с осторожностью к accuracy.

AUC ROC равен доле пар объектов вида (объект класса 1, объект класса 0), которые алгоритм верно упорядочи. Записав формулу, легко заметить, что от умножение на 2 ничего не измениться.

$$\frac{\sum_{i=0}^{n-1}\sum_{j=0}^{n-1}[a_i < a_j]\cdot[y_i < y_j]}{\sum_{i=0}^{n-1}\sum_{j=0}^{n-1}[y_i < y_j]}$$

Домножив обе части равенства в формуле с переменной a_i, получим ту же самую формулу.

Список литературы

[1] Репозиторий

6 Задача

Условие. Имеется файл log.txt размером 1Tb, содержащий лог в следующем формате:

номер записи, тип запроса, время отклика.

Пример начала лога: 1,/index,0.06 2,/test,0.03 3,/home,0.561 4,/home,0.87 5,/index,1.02 Напишите на Python 2 программу, которая для каждого типа запроса подсчитывает среднее время отклика и 95% доверительный интервал для этой величины. Реализуйте также проверку гипотезы о равенстве средних времен отклика для типов запроса /index и /test на уровне значимости 5%. *Решение.* Предполагаем, что во входном файле есть ограниченное число видо запросов, такое, что они помещаются в память. В противном случае придётся использовать базу данных. Для этого сделан специальный объект ProcessorBuffer, подменяя который можно записывать данные в БД.

Список литературы

[1] Репозиторий