Алгоритмы ленивой классификации

Абрамов Дмитрий

м15НоД ИССА

1 Введение

В данной работе было реализовано 4 алгоритма ленивой классификации объектов, представленных бинарными признаками. Для небинарных данных производится шкалирование. Тестирование алгоритмов проводилось на данных **Tic-Tac-Toe**.

2 Алгоритм 1

В данном алгоритме вычисляется пересечение описания классифицируемого объекта и объектов из плюс- и минус-контекста. Затем проверяется фальсифицируемость гипотезы:

- каждый из объектов из плюс-контекста голосует за положительный результат, если его пересечение с классифицируемым объектом не вкладывается в описание из минус-контекста.

- каждый из объектов из минус-контекста голосует за отрицательный результат, если его пересечение с классифицируемым объектом не вкладывается в описание из плюс-контекста

Фальсифицированные гипотезы не голосуют.

В итоге, решение принимается методом простого большинства.

3 Алгоритм 2

В данном алгоритме, в отличие от предыдущего, для каждого примера считается его поддержка в плюс- и минус-контекстах.

В итоге, выбирается класс, соответствующий контексту с большей поддержкой.

4 Алгоритм 3

В данном алгоритме, в отличие от первого, фальсифицируемые гипотезы могут учавствовать в голосовании, но только при условии, что их поддержка больше замыкания в противоположном контексте. Также здесь вводится ограничение на мощность пересечения - пересечение классифицируемого объекта и плюс- или минус-контекста должно включать сожержать на менее чем $C * 100\%$ признаков.

5 Алгоритм 4

Данный алгоритм является компиляцией второго и третьего алгоритмов, помимо проверки того, что поддержка больше, чем замыкание в противоположном контексте, при голосовании учитываетяся еще и разность между ними. Также проверяется то, что можность пересечения должна быть больше константы C.

6 Нахождение оптимального значения параметра

Для алгоритмов 3 и 4 была запущена кросс-валидация для того, чтобы определить оптимальное значение константы C. Для 3-го алгоритма:

C	Precision	Recall	F1	Accuracy
0.5	0.9870	0.9952	0.9907	0.9272
0.55	0.9875	1.0	0.9934	0.9288
0.6	0.9668	0.9644	0.9624	0.8336
0.65	0.9087	0.9477	0.9239	0.8110
0.7	0.8852	0.9469	0.9118	0.8115

Таблица 1: Выбор оптимального параметра C

Для 4-го алгоритма результаты практически идентичны. В итоге, опираясь на такие метрики, как F1 и Accuracy было принято решение взять константу $C = 0.55$.

7 Результаты

В результате, для данных train1 и test1 из Tic-Tac-Toe (итоговое тестирование проводилось только на них из-за вычислительной сложности алгоритмов) получились следующие результаты:

Алгоритм	Precision	Recall	F1	Accuracy
1	0.7922	1.0	0.8841	0.8280
2	0.9686	1.0	0.9839	0.0.9692
3	0.8906	0.9661	0.9268	0.8205
4	0.6761	0.8571	0.7559	0.6396

Таблица 2: Результаты

Алгоритм	False Discovery Rate	FN Rate	TN Predicted Rate	FP Rate
1	0.2078	0.0	1.0	0.5
2	0.3297	0.0	1.0	0.9375
3	0.2785	0.0339	0.7777	0.7586
4	0.0943	0.1429	0.7419	0.1786

Таблица 3: Результаты

Как видим, можно сделать вывол, что лучший результат показал 2-й алгоритм.