Иваницкий Илья

Решение БДЗ содержит следующие методы:

Алгоритмы:

Алгоритм 1 intersection_classif(plus, minus, x_test, y_test, threshold,pow_)

Алгоритм основан на нормированной сумме мощности пересечения признаков неизвестного примера с примерами-(+) и примерами-(-).

То же самое для отрицательных. (В таблице пример 1)

В модификации этого алгоритма при сравнении поступившего примера с каждым положительным или отрицательным объектом, вес пересечения возводятся в определенную степень pow_ (в таблице пример 3)

Неизвестный пример относится к тому ко множеству положительных примеров, если:
neg - pos <= threshold
К отрицательным, если:
neg - pos > -threshold

Если ни одно из условий не выполняется, то смотрим по поддержке, как в алгоритме 3 (с порогом 4) – (в таблице пример 5)

Алгоритм 2 intersection_with_contra_classif(plus, minus, x_test, y_test, threshold, pow_)

Пересекаем с положительным и проверяем чтобы пересечение не вкладывалось ни в одно отрицательное. если все так, то начисляем голос в виде "относительной мощности пересечения".

То же самое для отрицательных. (В таблице пример 2)

В модификации этого алгоритма относительная мощность пересечения возводится в определенную степень pow_. (В таблице пример 4)

Неизвестный пример относится к тому ко множеству положительных примеров, если:
neg - pos <= threshold
К отрицательным, если:
neg - pos > -threshold

Если ни одно из условий не выполняется, то смотрим по поддержке, как в алгоритме 3 (с порогом 4.) – (В таблице пример 6)

Алгоритм 3 Как у Амира Сафиуллина – взято только для спорных случаев в Алгоритме 1 и в Алгоритме 2.

Лучший порог = 4;
Лучший порог искался с помощью скользящего контроля максимизуруя accuracy
«Для неизвестного примера начисляем голоса пропорционально поддержке в положительных примерах и мощности пересечения с положительным примером, если она больше порога.

Для минуса то же самое.

Пример классифицируется туда, где сумма «голосов» больше.»

В нашем случае этот алгоритм использовался для того, чтобы решить спорные ситуации (примеры в таблице 5 и 6) – когда *pos* и *neg* объекта различаются на малое значение (их разность содержится в некоторой окрестности 0)

Также к отрицательным свойствам этой модификации стоит отнести и резкое увеличение времени работы алгоритма.

UPD
Как для Алгоритма 1, так и для Алгоритма 2, я попробовал возводить не в определенную степень, а в степень, соответствующую в Алгоритме 1 весу пересечения, а в Алгоритме 2 – мощности пересечения. Однако прироста в accuracy это не дало.

Сравнение:

Ниже приведена таблица со средними значениями accuracy для каждого алгоритма (1-4), и для трех наиболее популярных — SVC, Random forests, k-Nearest Neighbor.

7-folder cross-validation для датасета cars.data

	Алгоритмы	**Среднее значение по accuracy (на)**
1	1 (pow = 1)	0.804
2	2 (pow = 1)	0.981
3	1 (optimal pow = 200 – далее улучшение незначительно)	0.927
4	2 (optimal pow = 1 – метод не улучшает)	0.981
5	1 (threshold = 0.12)	0.954
6	2 (threshold = 0.03)	0.993
7	SVC	0.995
8	Random Forests	0.987
9	kNN	0.978