БДЗ (ТРАНД)
Корсачев Антон

Датасет: https://archive.ics.uci.edu/ml/datasets/Tic-Tac-Toe+Endgame

Датасет содержит 958 кортежей и он охарактеризован 9-ю признаками:

- значение в верхней левой ячейке игрового поля(крестик, нолик, пусто)
- значение в верхней ячейке игрового поля(крестик, нолик, пусто)
- значение в верхней правой ячейке игрового поля(крестик, нолик, пусто)
- значение в средней левой ячейке игрового поля(крестик, нолик, пусто)
- значение в средней ячейке игрового поля(крестик, нолик, пусто)
- значение в средней правой ячейке игрового поля(крестик, нолик, пусто)
- значение в нижней левой ячейке игрового поля(крестик, нолик, пусто)
- значение в нижней ячейке игрового поля(крестик, нолик, пусто)
- значение в нижней правой ячейке игрового поля(крестик, нолик, пусто)

Целевой признак: выигрышная комбинация для игрока, играющая за крестики(1 - победа, 0 - пройгрыш)

Рассмотрим задачу бинарной классификации. В качестве положительных примеров будем использовать те кортежи, у которых целевой признак = 1(победа), а в качестве отрицательных - 0(пройгрыш).

Шкалирование

Поскольку в кортежах задействованы не бинарные признаки, то произведем шкалирование по принципу:

любая ячейка игрового поля имеет три значения => зададим 3 новых признака для каждой ячейки - крестик в n-й ячейке (0 - нет, 1 - да), нолик в n-ой ячейке (0 - нет, 1 - да), пусто в n-ой ячейки (0 - нет, 1 - да).

Алгоритм 1.

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

$$\text{Pos} = \frac{1}{|G^+|} \sum_{i \in G^+} |g' \cap g_i^+|$$

$$\text{Neg} = \frac{1}{|G^-|} \sum_{i \in G^-} |g' \cap g_i^-|$$

Неизвестный пример относится к тому набору, где эта сумма больше, т.е. если Pos > Neg то положительно классифицируем, иначе отрицательно.

Алгоритм 2.

Пересечение признаков неизвестного примера с положительным и проверка чтобы пересечение не вкладывалось ни в одно отрицательное. Если все так, то начисляем голос в виде "относительной мощности пересечения". То же самое для отрицательных.

$$\text{Pos} = \frac{1}{|G^+|} \sum_{i \in G^+} \begin{cases} \frac{1}{|g'|} |g' \cap g_i^+|, \text{ если } |(g' \cap g_i^+) \cap g_k^-| == 0, k \in G^- \\ 0, \text{ Иначе} \end{cases}$$

$$\text{Neg} = \frac{1}{|G^-|} \sum_{i \in G^-} \begin{cases} \frac{1}{|g'|} |g' \cap g_i^-|, \text{ если } |(g' \cap g_i^-) \cap g_k^+| == 0, k \in G^+ \\ 0, \text{ Иначе} \end{cases}$$

Где сумма накопленных "голосов" больше - туда и классифицируем, т.е. если Pos > Neg то положительно классифицируем, иначе если Neg > Pos то классифицируем отрицательно.

Проверка валидности

В качестве метрик качества были использованы все предлагаемые в руководстве метрики:

- True Positive

- True Negative
- False Positive
- False Negative
- True Positive Rate
- True Negative Rate
- Negative Predictive Value
- False Positive Rate
- False Discovery Rate
- Accuracy
- Precision
- Recall

Сравнение качества производилось по усредненным (средним арифметическим) метрикам на основании использования метода кросс-валидации (K=5).

Результаты:
В таблице приведены значения метрик алгоритмов 1 и 2, а также наиболее популярных алгоритмов - Random forests, k-Nearest Neighbor.

Метрика	Алгоритм 1	Алгоритм 2	k-Nearest Neighbor	Random forests
True Positive	34	60	57	90
True Negative	24	32	165	183
False Positive	8	0	19	1
False Negative	27	1	47	14
True Positive Rate	0.5573770491803278	0.9836065573770492	0.5480769230769231	0.8653846153846154

True Negative Rate	0.75	0.9629629629629629	0.8967391304347826	0.9945652173913043
Negative Predictive Value	0.47058823529411764	0.9696969696969697	0.7783018867924528	0.9289340101522843
False Positive Rate	0.25	0.037037037037037035	0.10326086956521739	0.005434782608695652
False Discovery Rate	0.19047619047619047	0.016129032258064516	0.25	0.010989010989010989
Accuracy	0.659	0.96781	0.7708	0.9479166666666666
Precision	0.7346938775510204	0.9859154929577465	0.7783018867924528	0.9289340101522843
Recall	0.6101694915254238	0.9838709677419355	0.8967391304347826	0.9945652173913043