Las losowy - raport

Metodologia

Do analizy wykorzystano zbiór danych Breast Cancer z biblioteki `sklearn.datasets`. Przeprowadzono szereg testów dla różnych ustawień parametrów modeli RandomForest i XGBoost, modyfikując parametry takie jak `max_depth`, `min_samples_split` dla RandomForest oraz `max_depth` i `subsample` dla XGBoost.

Wyniki

RandomForest

Max Depth	Min samples split	Accuracy	Time (s)
None	2	0.97	0.17
None	4	0.96	0.16
None	6	0.96	0.13
None	8	0.96	0.14
10	2	0.97	0.18
10	4	0.96	0.17
10	6	0.96	0.16
10	8	0.96	0.17
20	2	0.97	0.17
20	4	0.96	0.16
20	6	0.96	0.16
20	8	0.96	0.16
30	2	0.97	0.19
30	4	0.96	0.18
30	6	0.96	0.17
30	8	0.96	0.16

XGBoost

Max Depth	Subsample	Accuracy	Time (s)
3	0.7	0.98	0.05
3	0.8	0.96	0.05
3	0.9	0.96	0.04
5	0.7	0.98	0.06
5	0.8	0.96	0.07
5	0.9	0.96	0.06
7	0.7	0.98	0.05
7	0.8	0.96	0.06
7	0.9	0.96	0.07

Wnioski

RandomForest:

Najwyższą dokładność (97%) uzyskano przy kilku konfiguracjach, szczególnie gdy min samples split wynosiło 2. Wynika to prawdopodobnie z większej zdolności modelu do nauki z bardziej szczegółowych danych bez znacznego przycinania drzew.

Dokładność 96% osiągano regularnie dla pozostałych ustawień, co sugeruje, że model jest odporny na zmiany w parametrze min_samples_split przy większych wartościach i różnych głębokościach drzew.

XGBoost:

XGBoost osiągnął najwyższą dokładność 98% przy konfiguracji max_depth=3 i subsample=0.7, co wskazuje, że mniejsza głębokość drzewa w połączeniu z odpowiednio dużą próbką podczas treningu może skutecznie zapobiegać przeuczeniu.

Pozostałe konfiguracje XGBoost stabilnie osiągały dokładność 96%, co pokazuje, że algorytm jest stabilny przy różnych ustawieniach głębokości i subsample.

RandomForest vs XGBoost:

Obie techniki wykazały się wysoką dokładnością, jednak subtelna różnica na korzyść XGBoost przy określonej konfiguracji sugeruje, że przy odpowiednim doborze parametrów, techniki wzmacniania gradientowego mogą osiągnąć lepsze wyniki w niektórych scenariuszach.

Mimo że RandomForest osiągnął najwyższą dokładność 97% w kilku ustawieniach, najlepszy wynik uzyskany za pomocą XGBoost był wyższy (98%), co może sugerować większą skuteczność tego modelu przy odpowiedniej kalibracji parametrów.

Analiza czasu wykonania pokazuje, że model XGBoost, mimo osiągnięcia wyższej dokładności w niektórych konfiguracjach, jest szybszy w porównaniu z RandomForest. Ta wydajność czyni go preferowanym wyborem w zastosowaniach wymagających szybkiego przetwarzania dużych zestawów danych.

Podczas gdy oba modele wykazują wysoką dokładność, XGBoost oferuje lepszą równowagę między dokładnością a szybkością wykonania, co może być kluczowe w dynamicznych środowiskach operacyjnych.