WUM projekt 1

Kraszewski Konstanty, Niewiadowski Paweł

April 2021

1 Wstęp

W tym raporcie opisany jest proces tworzenia optymalnego modelu klasyfikującego, objaśniającego zmienną binarną na podstawie 16 zmiennych kategorycznych o trzech stanach. Jest to pierwszy projekt z przedmiotu *Wstęp do Uczenia Maszynowego* na wydziale Matematyki i Nauk Informacyjnych.

2 Opis problemu

Naszym zadaniem było stworzenie modelu będącego w stanie poprawnie przewidzieć przynależność danego kongresmena do partii politycznej (Demokrata lub Republikanin) na podstawie jego głosowań w wybranych zagadnieniach.

3 Opis zbioru danych

Dane pochodzą ze strony: *apispreadsheets.com/datasets/121.*

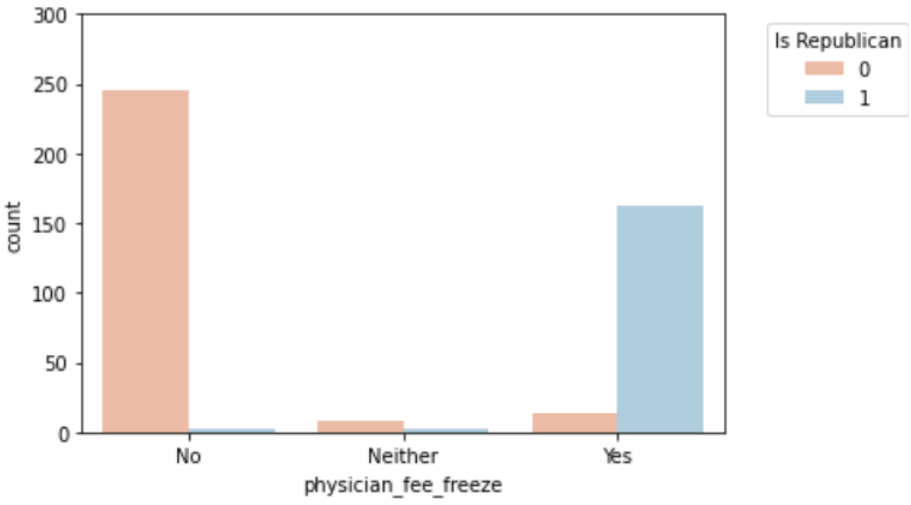
Tabela zawiera 16 kolumn kategorycznych reprezentujących poszczególne głosowania oraz jedną kolumnę reprezentującą przynależność polityczną, czyli zmienną objaśnianą. Wartości głosowań mają następującą strukturę:
y - głos za,
n - głos przeciw,
? - brak głosu lub nie opowiedzenie się po żadnej ze stron.

Kolumna *political_party* przyjmuje wartości *republican* lub *democrat*
Głosowania poruszały następujące tematy:

- Niepełnosprawne dzieci,
- Współdzielenie kosztów projektu wodnego,
- Przyjęcie postanowienia w sprawie budżetu,
- Zamrożenie opłat lekarzy,

- Pomoc humanitarna w El Salvador,
- Wspólnoty religijne w szkołach,
- Zakaz testu antysatelitowego,
- Pomoc partyzantom w Nikaragui,
- Rozbrojenie rakiet MX,
- Imigracja,
- Ograniczenia dla firm syntetycznych paliw,
- Wydatki na edukację,
- Finansowanie procesów sądowych,
- Przestępczość,
- Export bezcłowy,
- Export z południowej Afryki.

Głosowania te zostały wybrane przez CQA jako najważniejsze w 1986 roku. Liczba wierszy odpowiada ówczesnej liczbie kongresmenów. W zbiorze znajduje się 267 demokratów i 168 republikanów, przy czym jeden republikan nie oddał żadnego głosu. Pomiędzy poszczególnymi głosowaniami a partią istnieje sporo silnych zależności, wiele osób również głosowało tak samo we wszystkich głosowaniach. Model powinien być dzięki tym zależnościom dość dokładny.



Rysunek 1: Oddane głosy w sprawie zamrożenia opłat lekarzy z podziałem na partię.

4 Preprocessing

W ramach wstępnego przetwarzania dane tekstowe (y, ? oraz n) zostały zamienione na liczby (odpowiednio 1, 0, oraz -1). Podobnie nazwy partii zostały zastąpione wartościami 0 i 1, aby ujednolicić format oraz ułatwić modelom pracę z danymi.

Zastosowanie na zbiorze danych kodowania (Onehot Encoding) nie poprawiło osiąganych wyników, więc ostatecznie nie zostało użyte. Biorąc pod uwagę fakt, że wszystkie zmienne mają tę samą formę (zmienne kategoryczne o trzech kategoriach), żadne inne modyfikacje nie zostały uznane za konieczne.

5 Model

Na początku na danych testowych wytrenowanych zostało jedenaście modeli. Następnie pięć najlepszych użyto w modelach *VotingClassifier* i *StackingClassifier*. Ich wyniki sprawdzane były metodą walidacji krzyżowej biorąc pod uwagę pięć miar: pole pod wykresem krzywej ROC, F1, dokładność, precyzję i czułość. Otrzymane zostały następujące wyniki:

	ROC AUC	f1	accuracy	precision	recall	mean
VotingClassifier	0.992714	0.951193	0.960215	0.947308	0.958333	0.961953
RandomForestClassifier	0.985877	0.951561	0.960430	0.947802	0.958333	0.960801
BaggingClassifier(AdaBoost)	0.993807	0.946481	0.956989	0.954945	0.942308	0.958906
StackingClassifier	0.992714	0.946860	0.956882	0.946667	0.950000	0.958625
LogisticRegression	0.990936	0.946845	0.956989	0.947308	0.950000	0.958416
AdaBoostClassifier	0.988347	0.946842	0.957097	0.947070	0.950000	0.957871
BaggingClassifier(LogisticRegression)	0.992763	0.943823	0.953763	0.933114	0.958333	0.956359
BaggingClassifier(KNeighbors)	0.977230	0.930241	0.940538	0.893040	0.975000	0.943210
BaggingClassifier(DecisionTree)	0.979602	0.924783	0.937312	0.911777	0.942308	0.939156
KNeighborsClassifier	0.967836	0.926342	0.937204	0.891832	0.966667	0.937976
XGBClassifier	0.982807	0.920062	0.934194	0.919927	0.926282	0.936654
GradientBoostingClassifier	0.977924	0.918805	0.934194	0.924231	0.917308	0.934492
DecisionTreeClassifier	0.967037	0.912555	0.930753	0.921758	0.908974	0.928215

Tablica 1: Pierwotne wyniki modeli.

Wybrane wcześniej pięć najlepszych modeli zostało następnie poddanych strojeniu hiperparametrów z użyciem *GridSearchCV*. Dostrojone modele ponownie zostały wykorzystane do stworzenia komitetów *VotingClassifier* i *StackingClassifier*. Okazało się jednak, że żaden z nowych modeli nie osiągnął lepszego wyniku niż przed strojeniem hiperparametrów, co widać w kolejnej tabeli:

	roc_auc	f1	accuracy	precision	recall	mean
VotingClassifier	0.992714	0.951193	0.960215	0.947308	0.958333	0.961953
RandomForestClassifier	0.985877	0.951561	0.960430	0.947802	0.958333	0.960801
BaggingClassifier(AdaBoost)	0.993807	0.946481	0.956989	0.954945	0.942308	0.958906
StackingClassifier	0.992714	0.946860	0.956882	0.946667	0.950000	0.958625
LogisticRegression	0.990936	0.946845	0.956989	0.947308	0.950000	0.958416
tunedLogisticRegression	0.990936	0.946845	0.956989	0.947308	0.950000	0.958416
AdaBoostClassifier	0.988347	0.946842	0.957097	0.947070	0.950000	0.957871
tunedAdaBoostClassifier	0.987777	0.946198	0.956989	0.954212	0.941667	0.957369
tunedBaggingClassifier(LogisticRegression)	0.992763	0.943823	0.953763	0.933114	0.958333	0.956359
BaggingClassifier(LogisticRegression)	0.992763	0.943823	0.953763	0.933114	0.958333	0.956359
tunedVotingClassifier	0.990911	0.943193	0.953656	0.939615	0.950000	0.955475
tunedBaggingClassifier(AdaBoost)	0.991886	0.942546	0.953763	0.946520	0.941667	0.955276
tunedStackingClassifier	0.991350	0.942512	0.953656	0.946667	0.941667	0.955170
tunedRandomForestClassifier	0.987266	0.943475	0.953763	0.940806	0.950000	0.955062
BaggingClassifier(KNeighbors)	0.977230	0.930241	0.940538	0.893040	0.975000	0.943210
BaggingClassifier(DecisionTree)	0.979602	0.924783	0.937312	0.911777	0.942308	0.939156
KNeighborsClassifier	0.967836	0.926342	0.937204	0.891832	0.966667	0.937976
XGBClassifier	0.982807	0.920062	0.934194	0.919927	0.926282	0.936654
GradientBoostingClassifier	0.977924	0.918805	0.934194	0.924231	0.917308	0.934492
DecisionTreeClassifier	0.967037	0.912555	0.930753	0.921758	0.908974	0.928215

Tablica 2: Wyniki modeli po strojeniu hiperparametrów.

6 Podsumowanie

Ostatecznie wybranych zostało pięć najlepszych modeli, czyli *VotingClassifier*, *RandomForestClassifier*, *BaggingClassifier(AdaBoost)*, *StackingClassifier* oraz *LogisticRegression*. Ich wyniki na zbiorze testowym okazały się następujące:

	score
VotingClassifier	0.984733
StackingClassifier	0.984733
LogisticRegression	0.984733
RandomForestClassifier	0.961832
BaggingClassifier(AdaBoost)	0.961832

Tablica 3: Ostateczne wyniki modeli na zbiorze testowym.

Jak widać najlepsze wyniki osiągnęły dwa modele będące komitetami oraz regresja logistyczna.