Wstęp do Uczenia Maszynowego
Projekt Klasyfikacji

Kacper Grzymkowski, Jan Gąska

Zbiór danych

Dane składają się z 16 kolumn, które opisują jak dan
polityk głosował nad pewnymi kluczowymi ustawan
w Kongresie Stanów Zjednoczonych:

- “y” - polityk głosował za ustawą
- “n” - polityk głosował przeciw ustawie
- “?” - polityk wstrzymał się od głosu lub nie głosował

Politycy w Stanach Zjednoczonych należą do jednej z dwóch partii (nasza zmienna celu):

- “republican” - Partia republikańska
- “democrat” - Partia demokratyczna

	handicapped_infants	water_project_cost_sharing	adoption_of_the_budget_resolution	_exports	export_administration_act_south_africa	political_party
0	n	y	n	n	y	republican
1	n	y	n	n	?	republican
2	?	y	y	n	n	democrat
3	n	y	y	n	y	democrat
4	y	y	y	y	y	democrat
...	...	...	...	...	...	...
430				n	y	republican
431				n	y	democrat
432				n	y	republican
433				n	y	republican
434				?	n	republican

435 rows × 17 columns

Zbiór danych

Zbiór danych jest dosyć nietypowy - wszystkie zmienne są ciągami znaków, ale przedstawiają pewne binarne zachowania - jeżeli polityk zagłosował za ustawą, to nie zagłosował przeciwko.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 435 entries, 0 to 434
Data columns (total 17 columns):
 #   Column                                  Non-Null Count  Dtype 
---  ------                                  --------------  ----- 
 0   handicapped_infants                     435 non-null    object
 1   water_project_cost_sharing              435 non-null    object
 2   adoption_of_the_budget_resolution       435 non-null    object
 3   physician_fee_freeze                    435 non-null    object
 4   el_salvador_aid                         435 non-null    object
 5   religious_groups_in_schools             435 non-null    object
 6   anti_satellite_test_ban                 435 non-null    object
 7   aid_to_nicaraguan_contras               435 non-null    object
 8   mx_missile                              435 non-null    object
 9   immigration                             435 non-null    object
 10  synfuels_corporation_cutback            435 non-null    object
 11  education_spending                      435 non-null    object
 12  superfund_right_to_sue                  435 non-null    object
 13  crime                                   435 non-null    object
 14  duty_free_exports                       435 non-null    object
 15  export_administration_act_south_africa  435 non-null    object
 16  political_party                         435 non-null    object
dtypes: object(17)
memory usage: 57.9+ KB
```

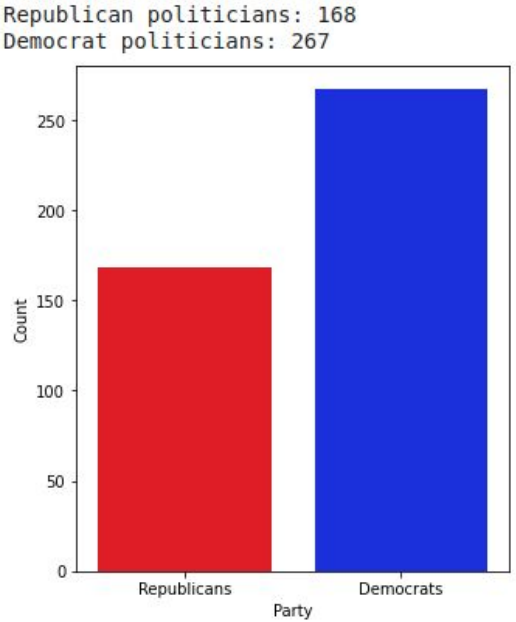
Zarys historyczny

- Dane pochodzą z 1984 r. z Izby Reprezentantów USA z 98 posiedzenia Kongresu.
- W tym czasie demokraci posiadali przewagę w Kongresie w liczebności 269 do 164 republikanów.
- Republikanie za to mieli przewagę w Senacie.
- W Afryce Południowej miała niedawno miejsce kontrowersyjna zmiana konstytucji
- W Ameryce Środkowej rebelianckie grupy wspierane przez USA walczą przeciwko domniemanym pro-komunistycznym rządom

EDA - liczności

Naturalnym punktem startowym jest policzenie ile polityków należy do poszczególnych partii. Demokraci są u władzy, a Republikanie w opozycji. Zbiór nie jest zbalansowany, ale jest bliski.

Liczba polityków w każdej z partii pozwala nam identyfikować kadencję, z której są dane - pozwoli to nam postawić te dane w kontekście historycznym.

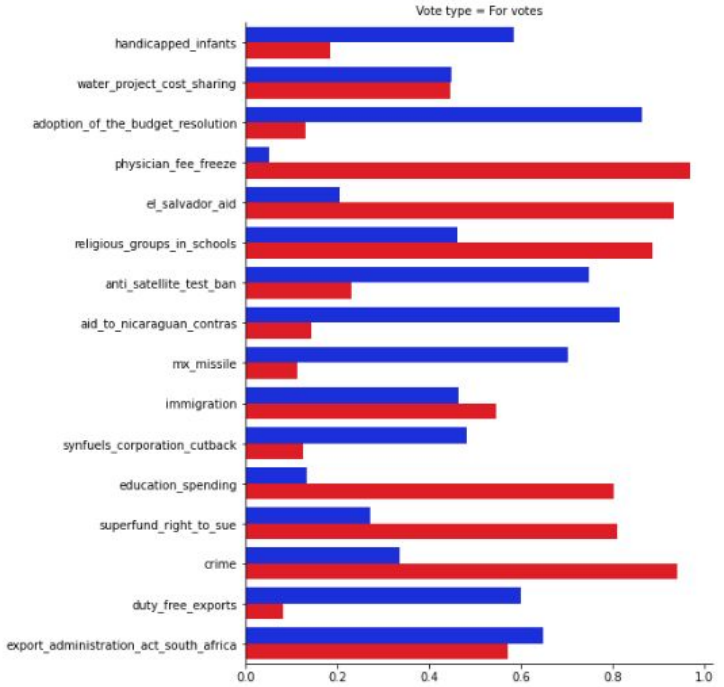


EDA - głosowania

Duże różnice pomiędzy ustawami są obiecujące dla budowania modeli.

Można się spodziewać, że jeżeli jedna partia zagłosuje za ustawą, to druga zagłosuje przeciwko, ale nie w przypadku:

- “export_administration_act_south_africa”
- “immigration”
- “water_project_cost_sharing”



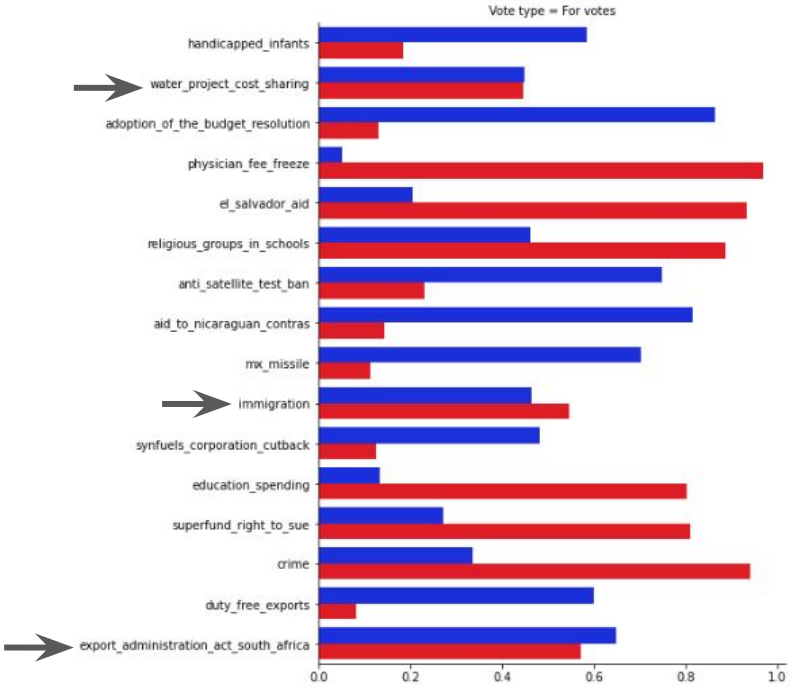
Wykres znormalizowany do rozmiaru partii

EDA - głosowania

Duże różnice pomiędzy ustawami są obiecujące dla budowania modeli.

Można się spodziewać, że jeżeli jedna partia zagłosuje za ustawą, to druga zagłosuje przeciwko, ale nie w przypadku:

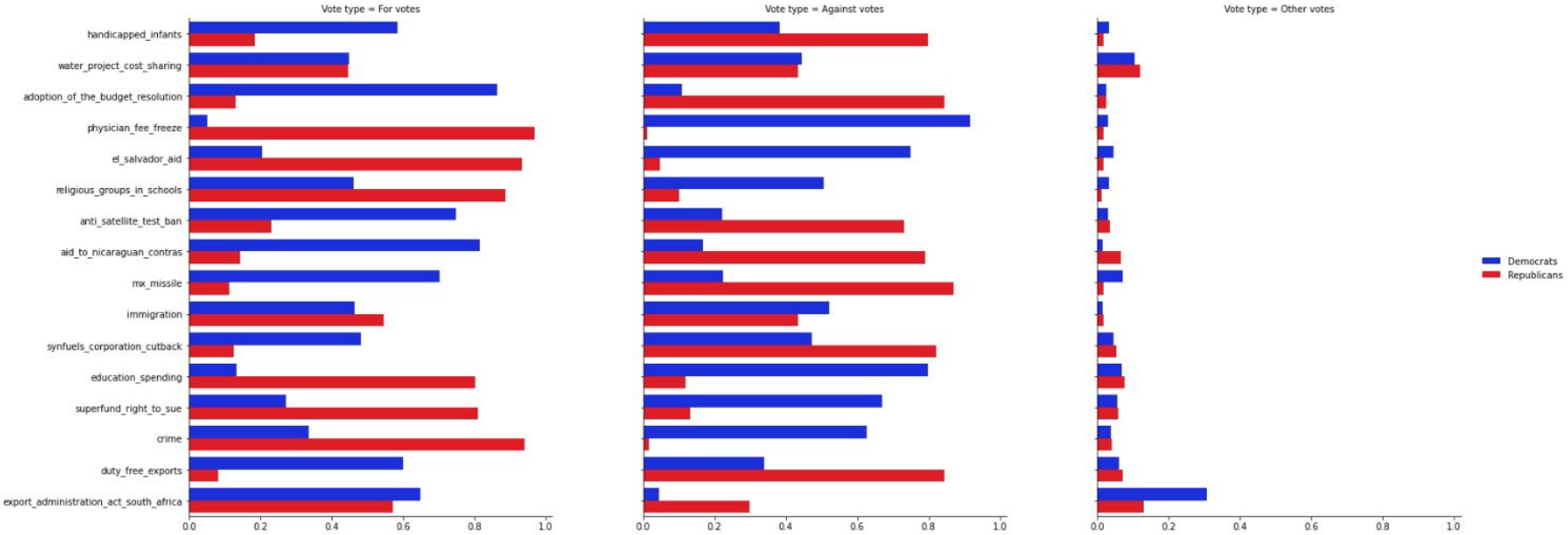
- “export_administration_act_south_africa”
- “immigration”
- “water_project_cost_sharing”



Wykres znormalizowany do rozmiaru partii

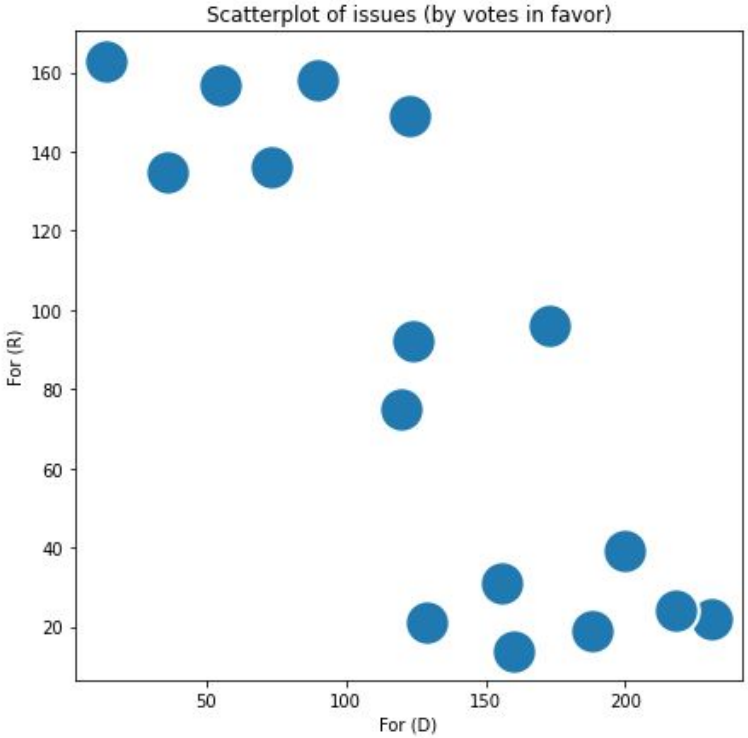
EDA - głosowania

Wykres znormalizowany do rozmiaru partii



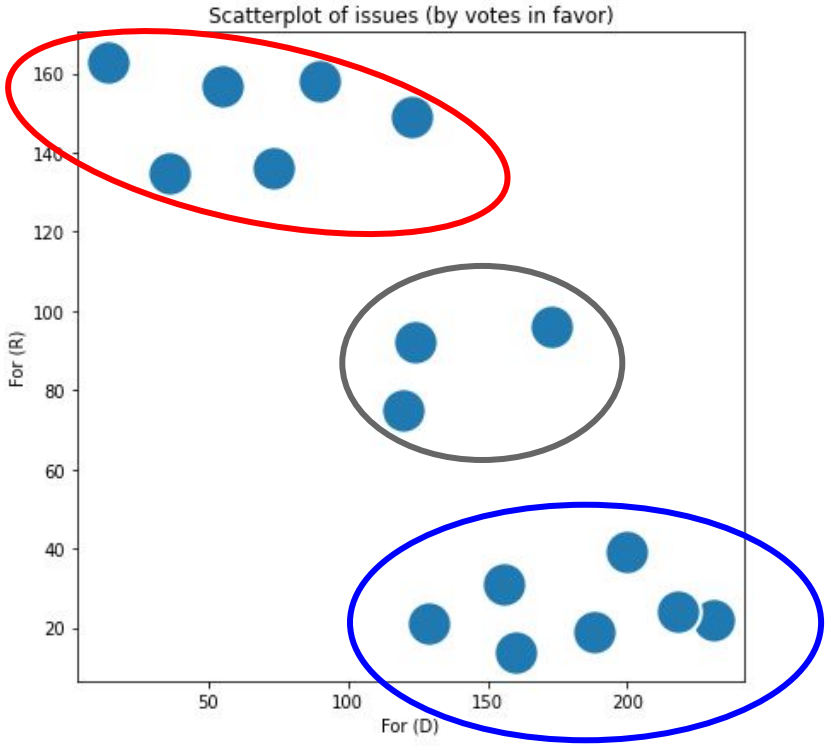
EDA - Scatterplot

Umieszczając ustawy na wykres punktowy, gdzie każdy punkt jest ustawą, a na osiach x i y jest liczba głosów za tą ustawą z każdej z partii politycznych, dostajemy trzy wyspy - “republikańską”, “centrystyczną” i “demokratyczną”



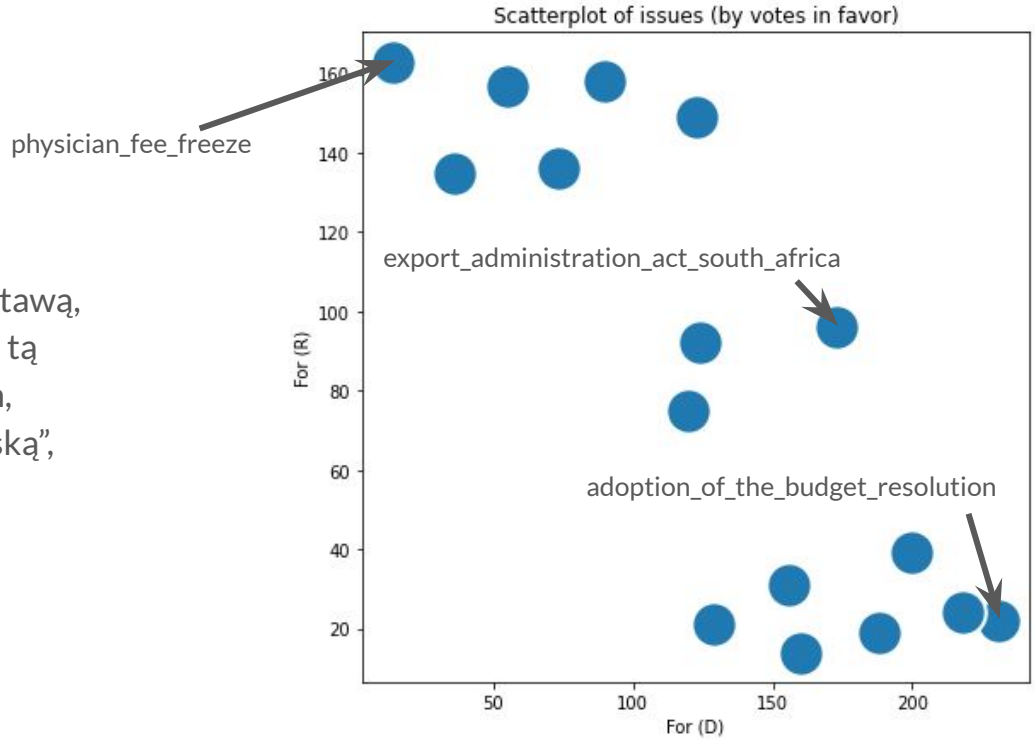
EDA - Scatterplot

Umieszczając ustawy na wykres punktowy, gdzie każdy punkt jest ustawą, a na osiach x i y jest liczba głosów za tą ustawą z każdej z partii politycznych, dostajemy trzy wyspy - “republikańską”, “centrystyczną” i “demokratyczną”



EDA - Scatterplot

Umieszczając ustawy na wykres punktowy, gdzie każdy punkt jest ustawą, a na osiach x i y jest liczba głosów za tą ustawą z każdej z partii politycznych, dostajemy trzy wyspy - “republikańską”, “centrystyczną” i “demokratyczną”



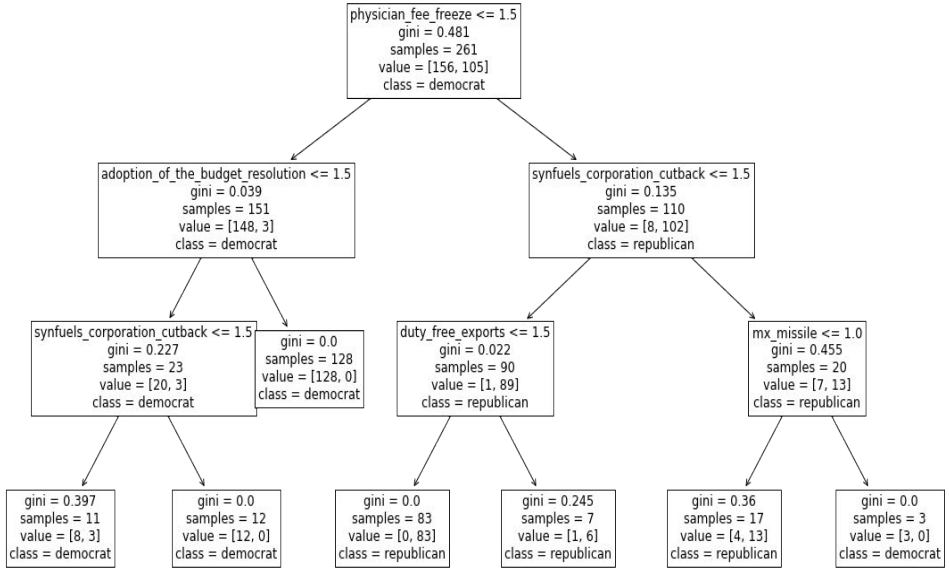
Modele

Drzewa, lasy i naiwny Bayes

Podejście pierwsze - drzewo smol

Model niedużego drzewa, nie przeuczającego się, lecz ciągle charakteryzującym się wysoką accuracy na zbiorze testowym oraz treningowym.

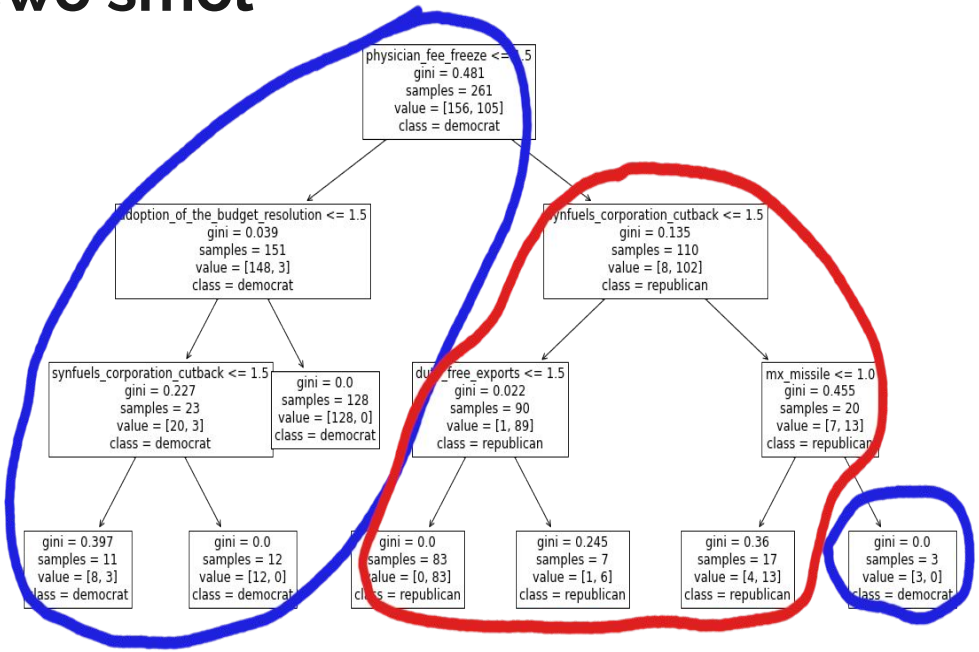
Był to pierwszy model przez nas zastosowany model optymalizowany, jego wysokość oraz ilość liści nie jest duża jak na ilość cech w naszej ramce, lecz jeszcze nie minimalna.



Podejście pierwsze - drzewo smol

Widzimy jak drzewo dokonuje szybkiego podziału na demokratów oraz republikanów.

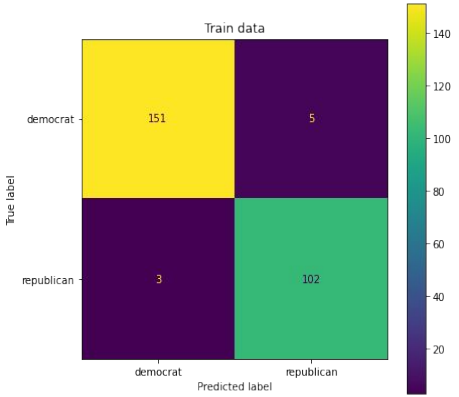
Także, warto zwrócić uwagę, iż z góry przypisywana jest klasa demokraty, tudzież model zakłada a priopri przynależność do partii demokratycznej, którą obala lub zachowuje.

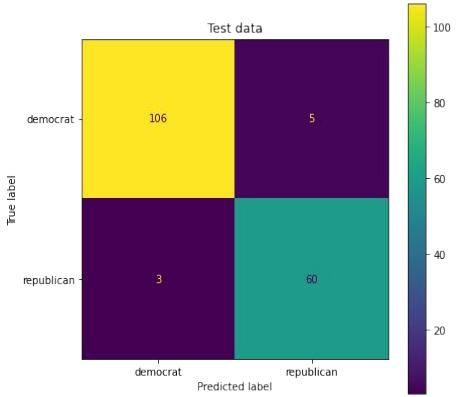


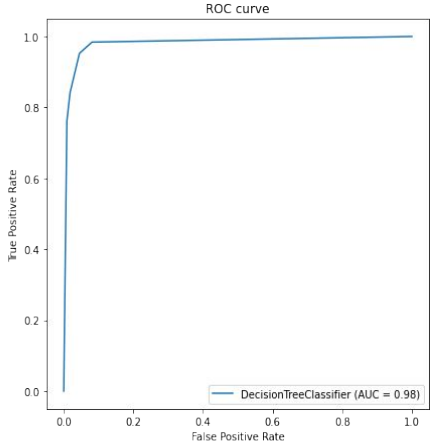
Wyniki drzewa smol

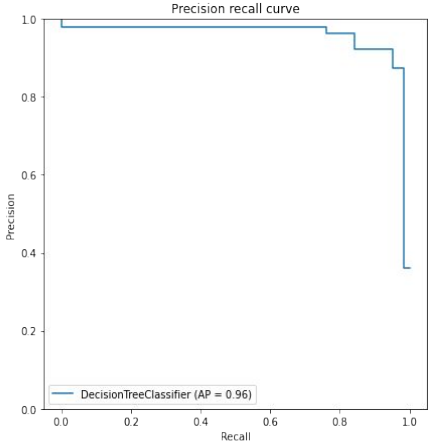
Wyniki drzewa plasowały się na poziomie 97% accuracy na zbiorze treningowym oraz 95% na zbiorze treningowym. Parametrami drzewa były następujące wartości:

```
(ccp_alpha=0.0, class_weight=None, criterion='gini',
 max_depth=3, max_features=None, max_leaf_nodes=10,
 min_impurity_decrease=0.0, min_impurity_split=None,
 min_samples_leaf=1, min_samples_split=2,
 min_weight_fraction_leaf=0.0, presort='deprecated',
 random_state=420, splitter='best')
```





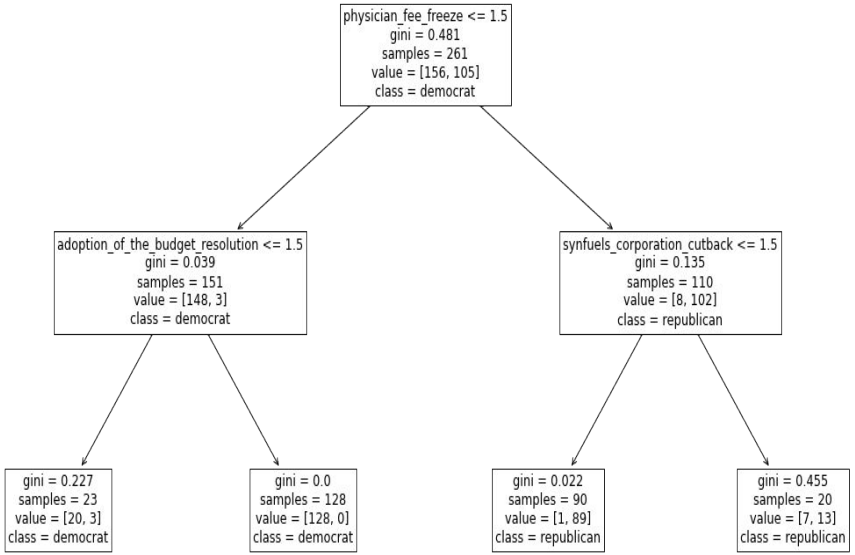




Drzewo smoler

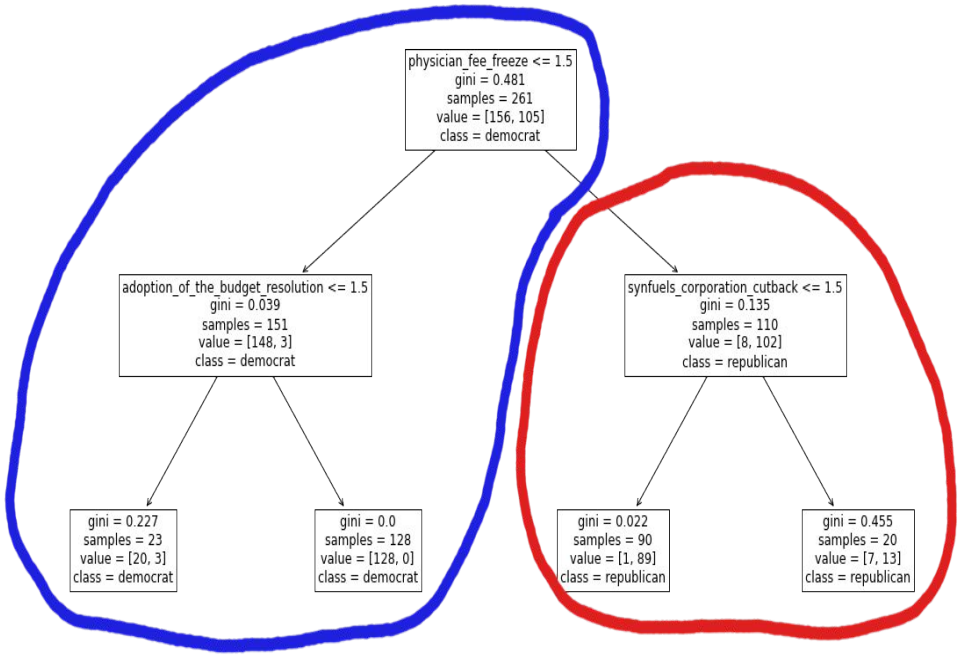
Pomimo wysokiego wyniku już modelu drzewa smol, zdecydowaliśmy się na jeszcze bardziej stanowczą minimalizaję do najniższego optymalnego drzewa, **smoler**.

Liczba liści została zmniejszona do 4, a głębokość do 3.



Drzewo smoler

Pewną istotną obserwacją oraz wnioskiem wynikającym z pracy drzewa smoler jest wręcz idealna separacja po pierwszym podziale. Model korzystając wyłącznie z cechy “physician_fee_freeze” posegregował binarnie na klasy pomiędzy republikanów i demokratów.

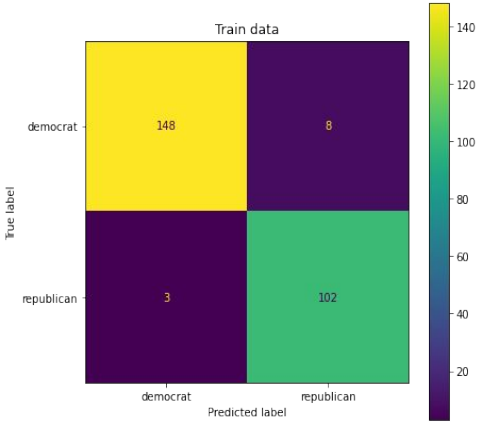


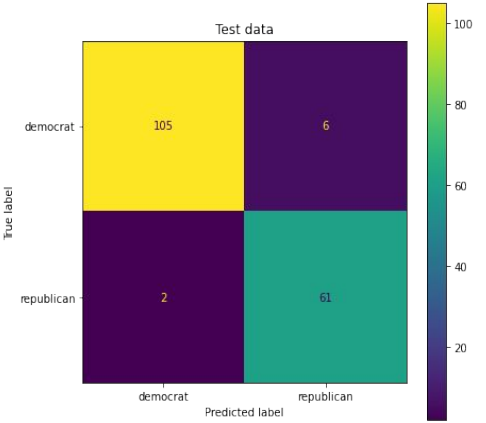
Drzewo smoler - wyniki

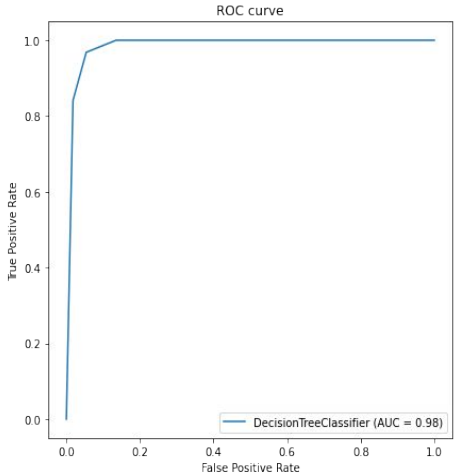
Pomimo mniejszego stopnia złożenia, drzewo smoler uzyskało odpowiednio 96% oraz 95% accuracy na zbiorach treningowym oraz testowym. Świadczy to o bardzo wysokim współczynniku importance wcześniej omawianej cechy.

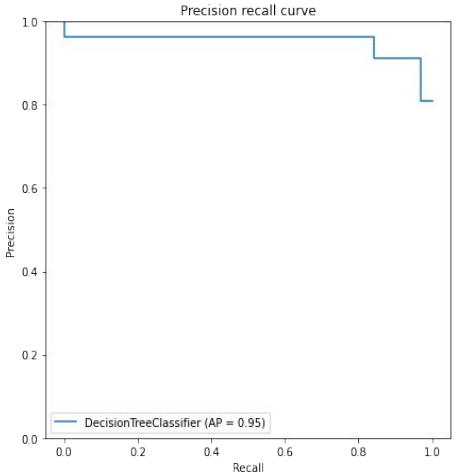
Parametry drzewa smoler:

```
(ccp_alpha=0.0, class_weight=None, criterion='gini',
 max_depth=2, max_features=None, max_leaf_nodes=None,
 min_impurity_decrease=0.0, min_impurity_split=None,
 min_samples_leaf=1, min_samples_split=2,
 min_weight_fraction_leaf=0.0, presort='deprecated',
 random_state=420, splitter='best')
```





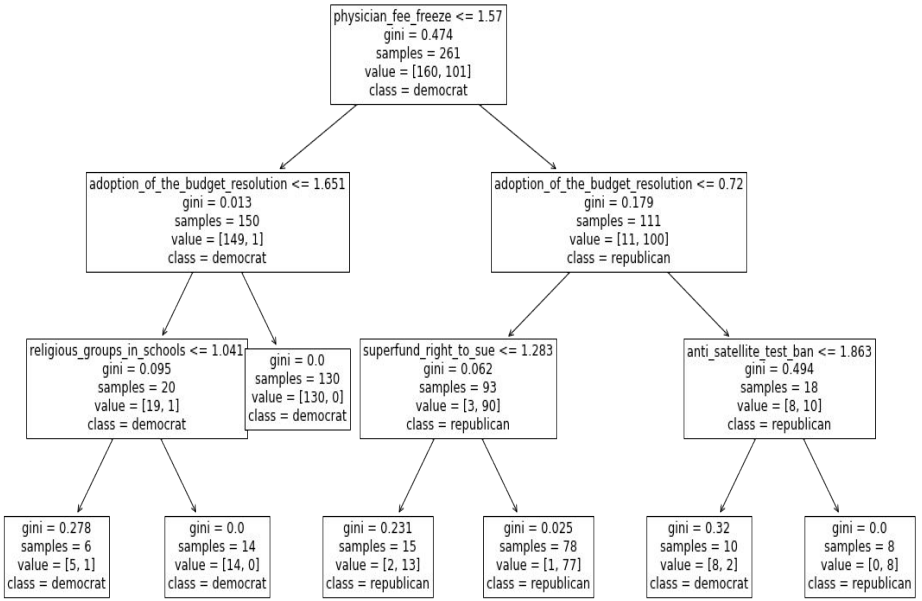




Drzewa a tuning

Powyższe modele były dziełem naszej intuicji, jak stoją w porównaniu z automatycznym strojeniem parametrów?

Oto najoptymalniejsze drzewo znalezione dzięki operacji GridSearchCV.

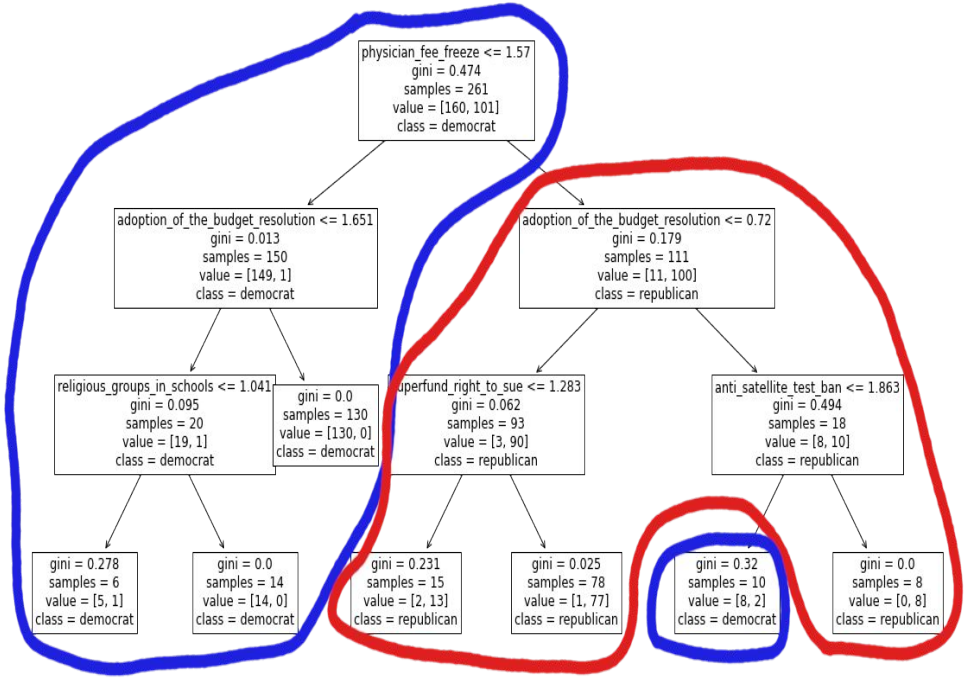


Drzewa a tuning

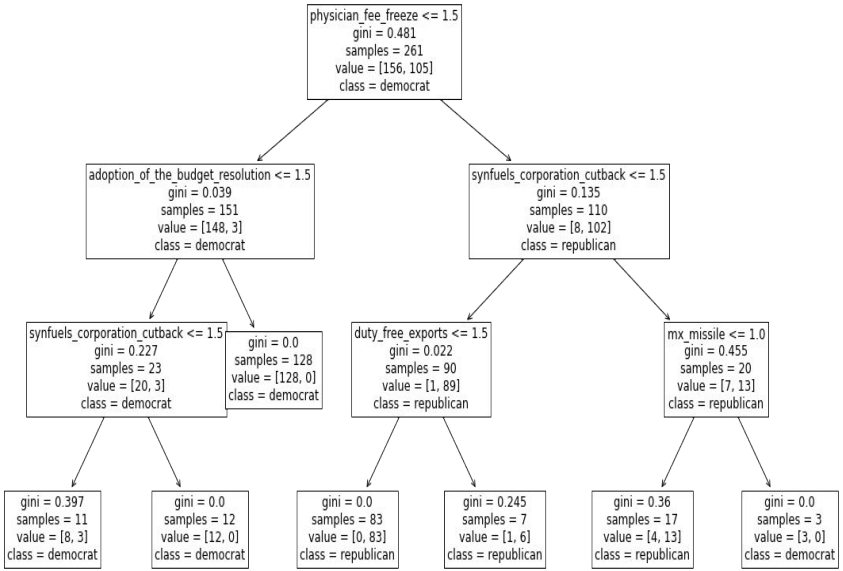
Jak widzimy, drzewo to ma duże podobieństwo z drzewem **smol.** Wykazało się accuracy na poziomie 97% i 94% na odpowiednio zbiorach treningowym oraz testowym.

Oto jej parametry:

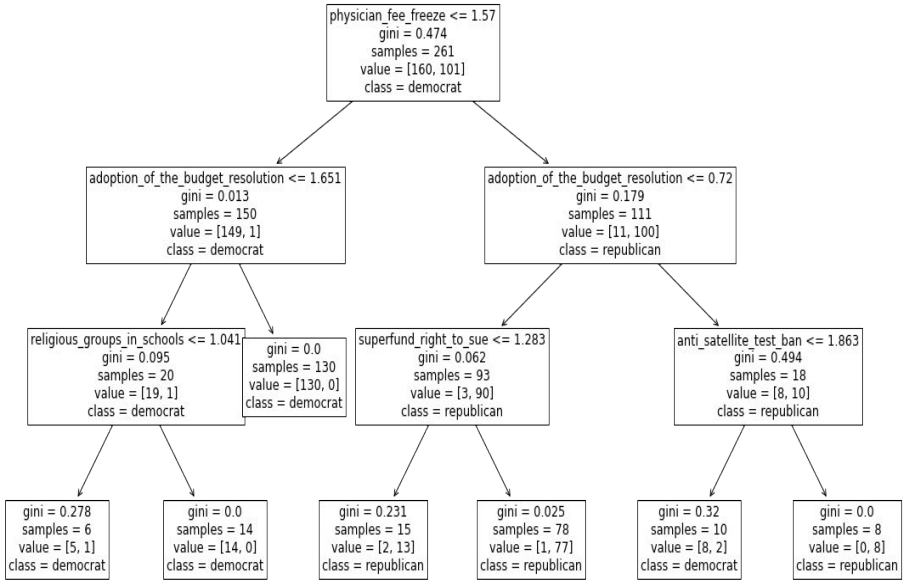
```
(ccp_alpha=0.0, class_weight=None, criterion='gini',
 max_depth=3, max_features=None, max_leaf_nodes=None,
 min_impurity_decrease=0.0, min_impurity_split=None,
 min_samples_leaf=1, min_samples_split=2,
 min_weight_fraction_leaf=0.0, presort='deprecated',
 random_state=420, splitter='random')
```



Drzewo tuningowane vs smol



Drzewo **smol**



Drzewo tuningowane

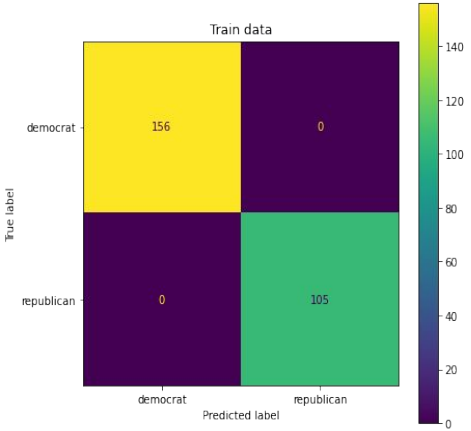
Lasy

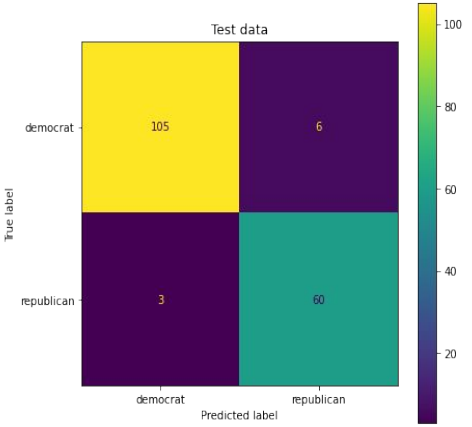
Las losowy “ręczny”

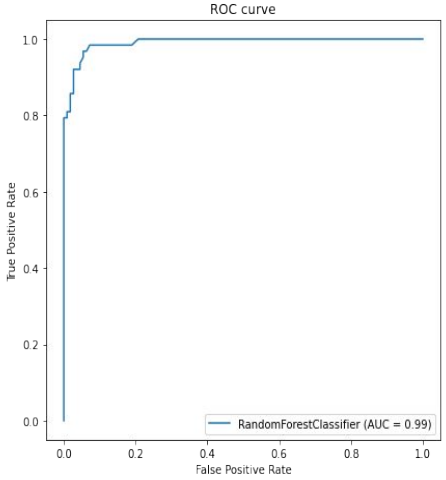
Początkowy model lasu bez wspomagania tuningiem parametrów wygląda następująco:

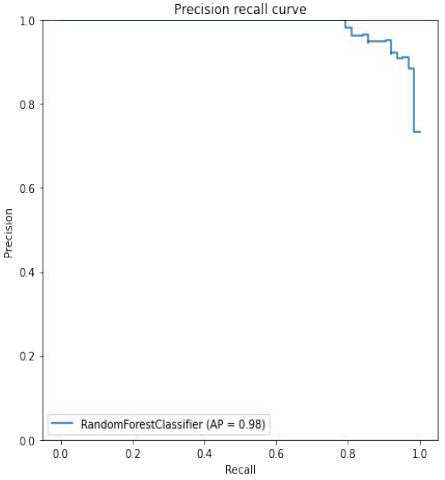
```
(bootstrap=True, ccp_alpha=0.0, class_weight=None,
 criterion='gini', max_depth=1, max_features='auto',
 max_leaf_nodes=None, max_samples=None,
 min_impurity_decrease=0.0, min_impurity_split=None,
 min_samples_leaf=1, min_samples_split=2,
 min_weight_fraction_leaf=0.0, n_estimators=100,
 n_jobs=None, oob_score=False, random_state=None,
 verbose=0, warm_start=False)
```

Uzyskał wynik 94% i 93% odpowiednio na zbiorach treningowym oraz testowym.









Las losowy tuningowany

W opozycji do “ręcznego” lasu losowego, tuningowany dzielił pewne zależności pomiędzy parametrami. Przedstawimy pięć estymatorów najgorszych oraz pięć najlepszych.

Las losowy tuningowany

W przypadku pięciu najlepszych estymatorów zauważyć można maksymalną głębokość na poziomie 6, taką samą jak we wcześniej omawianych drzewach, w analizie wykazano, iż liczba estymatorów nie miała większego znaczenia.

	param_n_estimators	param_criterion	param_max_depth	param_min_samples_split	mean_test_score	std_test_score	rank_test_score
57	5	entropy	6	4	0.977208	0.034823	1
22	50	gini	6	4	0.973219	0.029988	2
23	100	gini	6	4	0.973219	0.029988	2
26	100	gini	6	6	0.973219	0.029988	2
28	50	gini	10	2	0.973219	0.029988	2

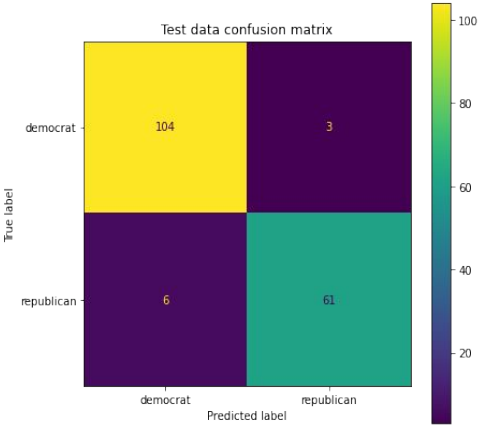
Las losowy tuningowany

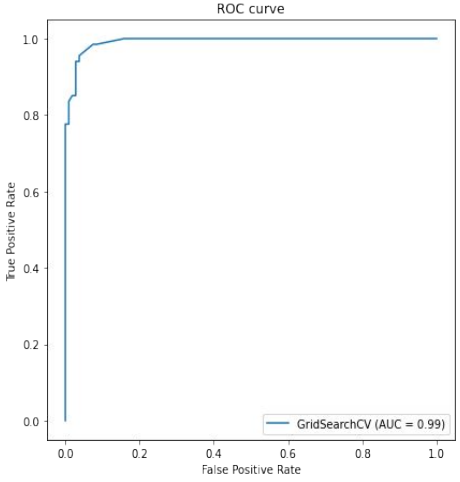
W kwestii najgorszych estymatorów widzimy, iż są to same drzewa o wysokości 1.
W ogólności ilość estymatorów oraz kryterium podziału nie były tak istotne jak max depth oraz minimalna liczba podziału.

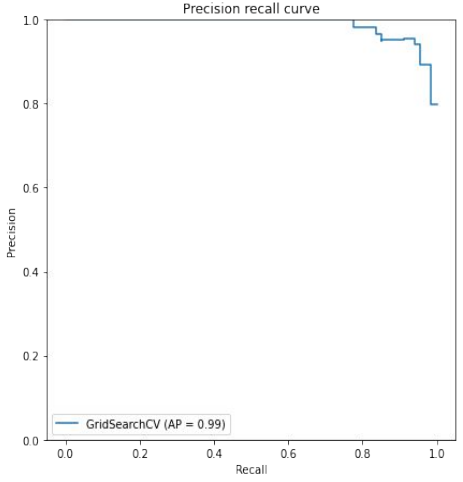
	param_n_estimators	param_criterion	param_max_depth	param_min_samples_split	mean_test_score	std_test_score	rank_test_score
38	100	entropy	1	2	0.904274	0.025512	68
4	50	gini	1	4	0.904131	0.035624	69
2	100	gini	1	2	0.904131	0.031197	69
7	50	gini	1	6	0.900427	0.035087	71
43	50	entropy	1	6	0.896581	0.017391	72

Las losowy tuningowany

Oto macierze konfuzji oraz wyniki najlepszego lasu losowego z naszego tuningu.





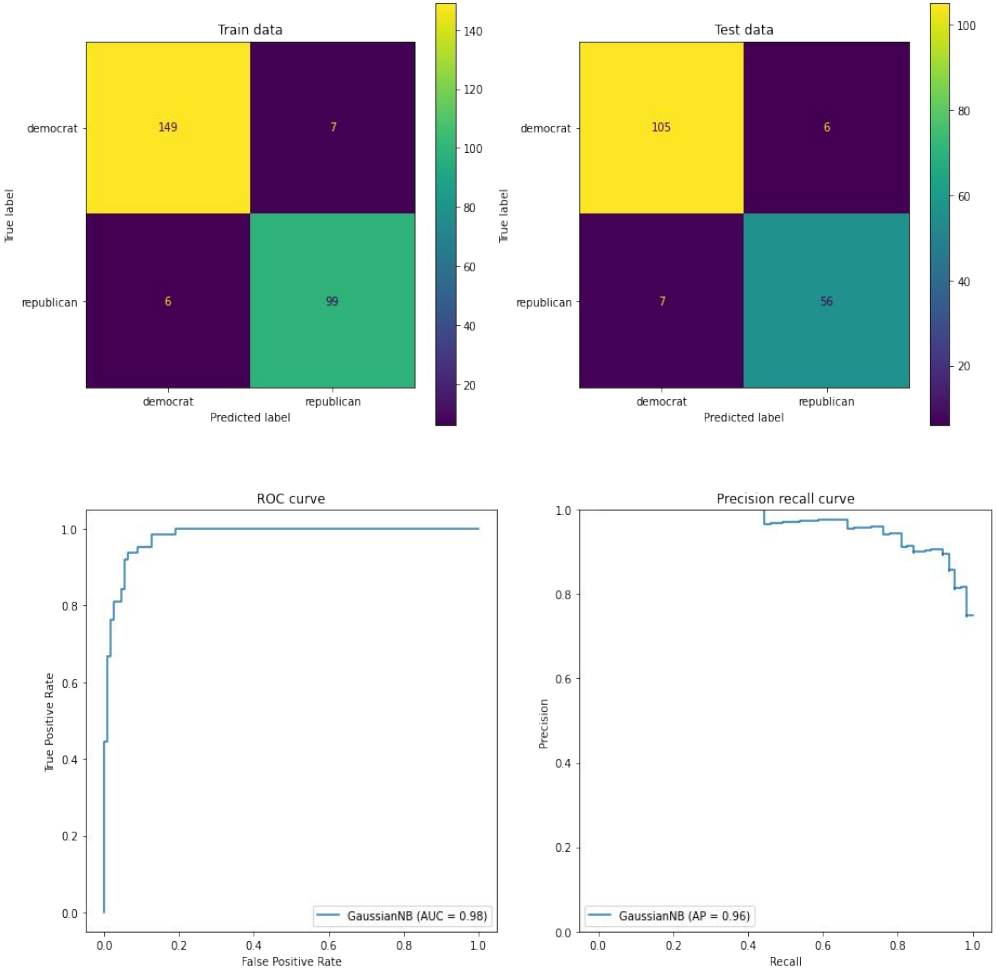


Naiwny Bayes

“Ręczny” Naiwny Bayes

W naszym modelowaniu, algorytm Naiwnego Bayesa okazał się najgorszym modelem.

Jego accuracy wynosiły odpowiednio 90% i 83% na zbiorach treningowym oraz testowym.

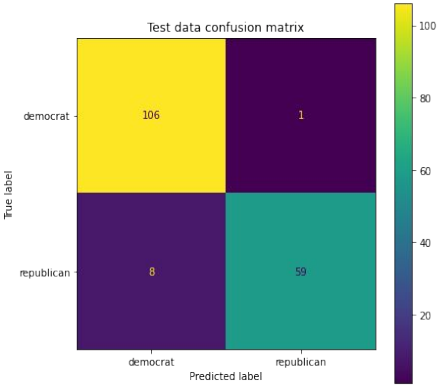


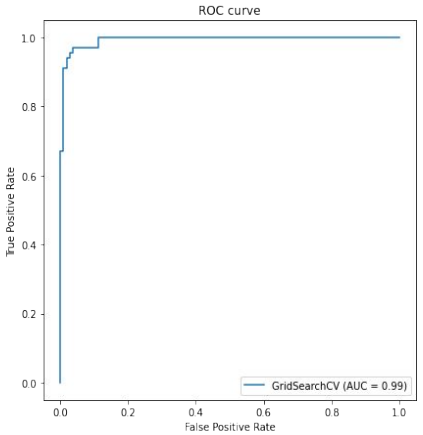
Tuningowany Naiwny Bayes

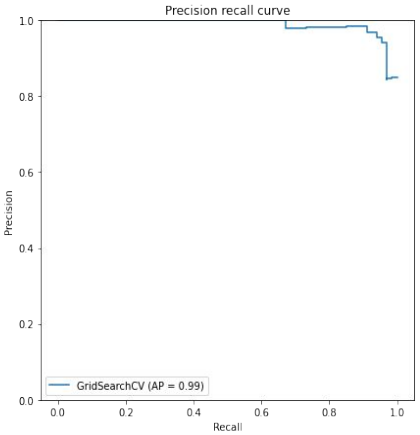
W tuningowaniu Naiwnego Bayesa otrzymaliśmy dość ciekawe rezultaty, które przedstawimy na kolejnych slajdach.

Jak w przypadku lasu losowego, przedstawimy pięć najgorszych oraz pięć najlepszych estymatorów.

W szczególności parametr prior, który oznacza odgórne przypisanie prawdopodobieństwa występowania klasy, okazał się bardzo ciekawy.







Tuningowy Naiwny Bayes

Najlepsze estymatory osiągane są dla priors, gdzie prawdopodobieństwa dla demokratów wynosi 0.1 oraz republikanów 0.9. Przewaga demokratów okazuje się wpływać nieliniowo na model, co potwierdza odgórne założenie przynależności do demokratów przez modele.

	param_var_smoothing	param_priors	mean_test_score	std_test_score	rank_test_score
24	1e-07	[0.1, 0.9]	0.934758	0.03017	1
26	1e-09	[0.1, 0.9]	0.934758	0.03017	1
27	1e-10	[0.1, 0.9]	0.934758	0.03017	1
28	1e-11	[0.1, 0.9]	0.934758	0.03017	1
29	1e-12	[0.1, 0.9]	0.934758	0.03017	1

Tuningowany Naiwny Bayes

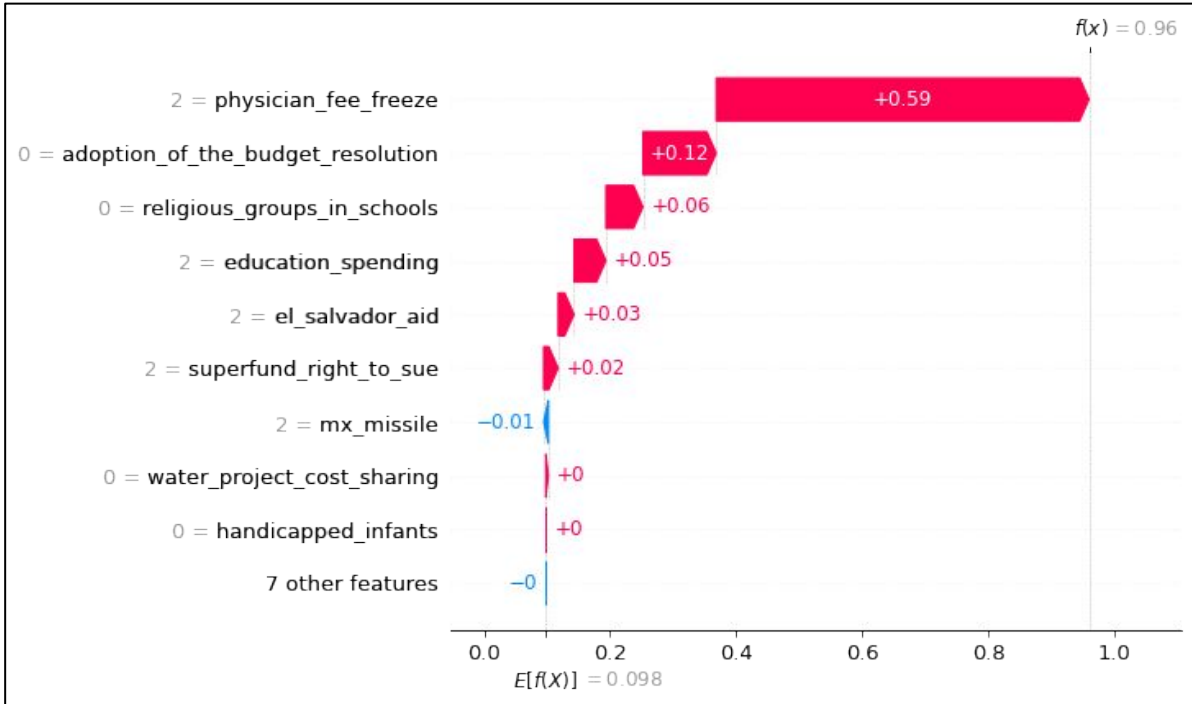
Najgorsze estymatory okazuję się być dla odwróconych wartości dla parametru priors. Okazuje się, że var smothing nie ma, aż takiego znaczenia.

	param_var_smoothing	param_priors	mean_test_score	std_test_score	rank_test_score
30	1e-07	[0.9999, 0.0001]	0.919516	0.043717	38
33	1e-10	[0.9999, 0.0001]	0.919516	0.043717	38
34	1e-11	[0.9999, 0.0001]	0.919516	0.043717	38
31	1e-08	[0.9999, 0.0001]	0.919516	0.043717	38
32	1e-09	[0.9999, 0.0001]	0.919516	0.043717	38

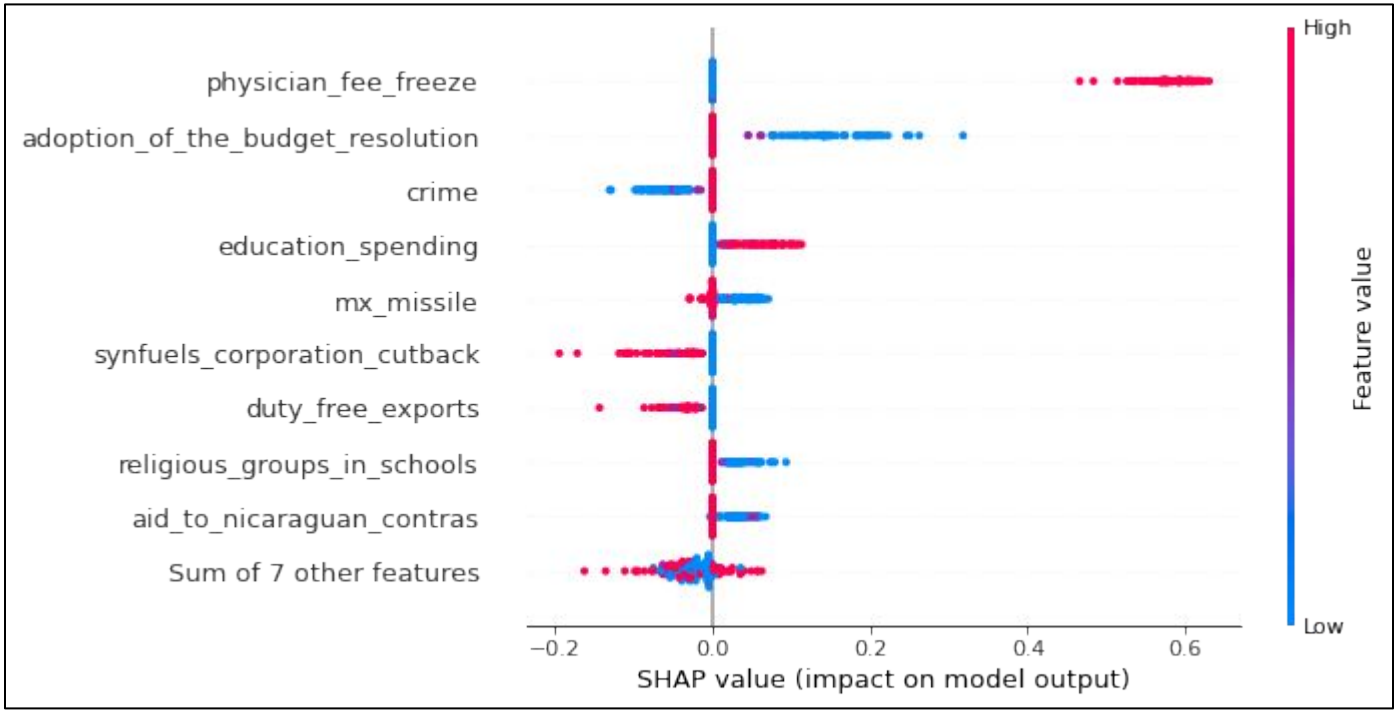
Importance, SHAP

Importance

- Model z góry zakłada przynależność do demokratów
- “Physician fee free” okazała się kluczową cechą w determinacji republikanów
- W rzeczywistości 13 dolnych cech jest praktycznie tak samo istotna jak czwarta od góry



Importance


physician_fee_freeze
adoption_of_the_budget_resolution
crime
education_spending
mx_missile
synfuels_corporation_cutback
duty_free_exports
religious_groups_in_schools
aid_to_nicaraguan_contras
Sum of 7 other features
−0.2
0.0
0.2
0.4
0.6
SHAP value (impact on model output)
High
Feature value
Low

Zakończenie, podsumowanie

Zakończenie, podsumowanie

- Ostatecznie tuningowany las losowy wysublimował się jako najlepszy estymator
- Nasze pierwotne modele były dość “trafione” z parametrami względem tuningowanych modeli
- Największa różnica w wydajności pomiędzy oryginalnym modelem, a modelem tuningowanym przypadła dla Naiwnego Bayesa i wyniosła średnio 10 punktów procentowych.

- Importance cech był dość nierównomiernie rozdystrybuowany, z physician fee freeze wręcz klasyfikującą własnoręcznie z około 40% współczynnikiem
- Siedem dolnych cech jest tak samo “ważna” jak druga od góry względem importance
- Projekt zakończył się sukcesem z utworzeniem optymalnego modelu wykonującego zadanie klasyfikacji

Koniec

Dziękujemy za uwagę i poświęcony czas