[HW2] - Przygotowanie modelu o największej zdolności predykcyjnej

Grzegorz Zakrzewski, 313555

Styczeń 2024

1 Ręczne przygotowanie modelu

1.1 Szybkie spojrzenie na zbiór danych

Zadany zbiór danych obejmuje 500 zmiennych objaśniających. Zbiór treningowy zawiera 2000 obserwacji, a zbiór testowy 600 obserwacji. Nie występują żadne braki danych. Klasyfikacja następuje do dwóch klas: `1` oraz `-1`, które są równoliczne w zbiorze treningowym. Wszystkie zmienne objaśniające przyjmują wartości ze zbioru liczb rzeczywistych.

Na zmiennych objaśniających zostały policzone wartości statystyk takich jak średnia, odchylenie standardowe, wartości trzech kwartyli oraz maksimum. Wizualizacja rozkładów tych statystyk znajduje się na Rysunku 1. Na pierwszy rzut oka, wszystkie zmienne objaśniające zdawały się pochodzić z podobnego rozkładu, ze średnią około 500. Spojrzenie na rozkłady statystyk wyprowadza nas jednak z błędu. Zmienne objaśniające różnią się między sobą wariancją, niektóre zmienne przyjmują znacznie szerszy zakres wartości niż inne.

Na podstawie charakterystyki zadania oraz właściwości zbioru danych nasuwają się dwa ważne problemy. Przede wszystkim, przygotowany model może nie wykryć istotnych zmiennych ze zbioru danych. Dlatego też ważnym punktem będzie ograniczenie zbioru zmiennych. Drugim problemem może być to, że model zbyt mocno dopasuje się do danych treningowych i wynik na zbiorze testowym będzie słaby. Remedium na to może być zastosowanie cross-walidacji, również ze względu, że zbiór nie jest zbyt duży.

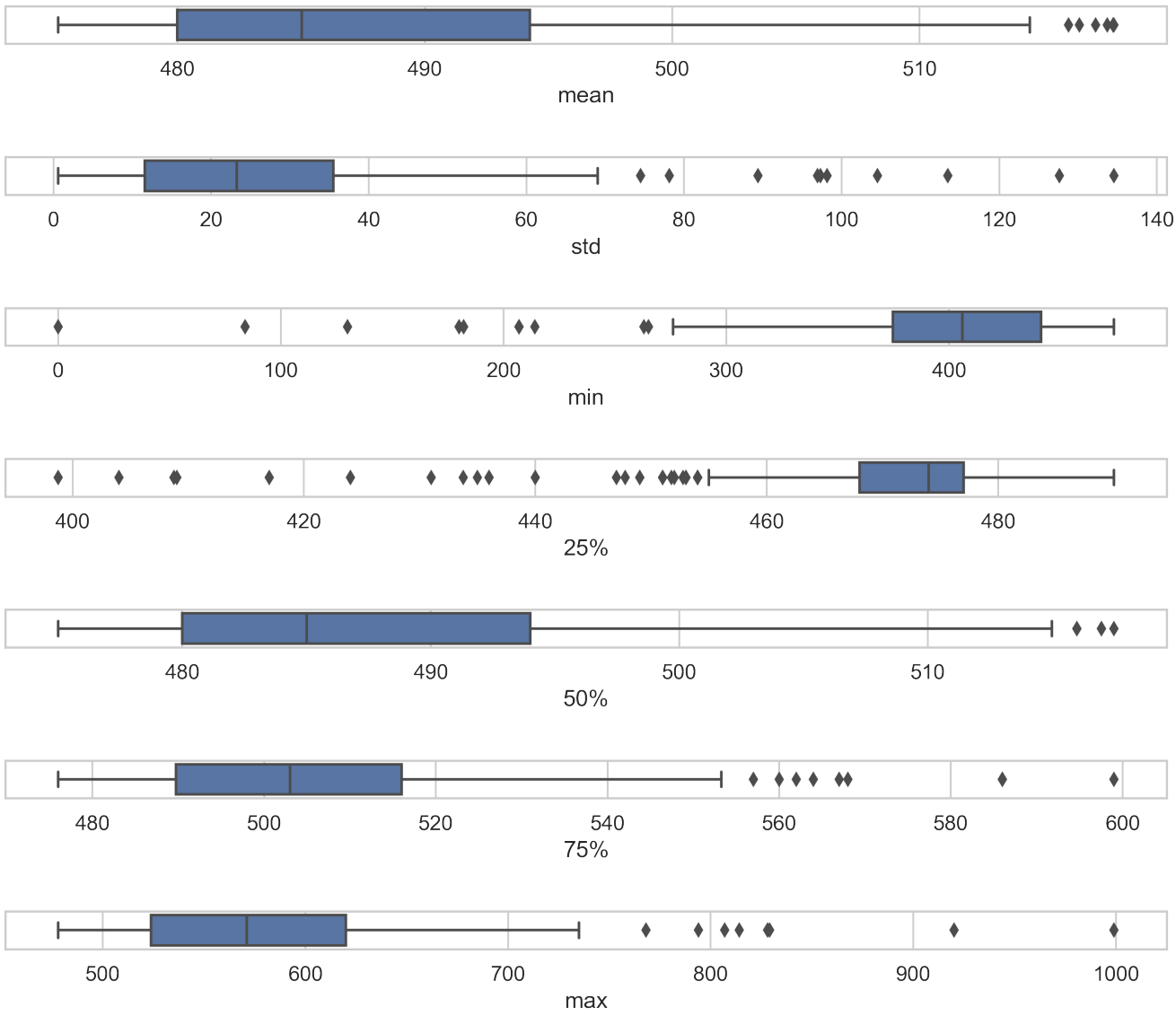
1.2 Szczegóły przeprowadzenia eksperymentu

Wytrenowane zostaną modele regresji logistycznej z regularyzacją LASSO, lasu losowego oraz wzmacniania gradientowego (xgboost). Wykorzystane będą implementacje tych modeli z paczek *scikit-learn* oraz *xgboost*. Hiperparametry tych modeli będą wybrane z arbitralnie określonych zakresów przedstawionych w Tabeli 1, przetestowane zostaną wszystkie możliwe wartości (regresja logistyczna) lub 300 losowo wybranych kombinacji (las losowy, xgboost). Miarą do oceny jakości modeli będzie zadane w poleceniu *balanced accuracy*. Zastosowana będzie 4-zbiorowa cross-walidacja. Przed rozpoczęciem uczenia dane będą standaryzowane przez odjęcie średniej i podzielenie przez odchylenie standardowe.

model	hiperparametr	zbiór wartości
regresja logistyczna LASSO	`C`	[0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1.0]
las losowy	`n_estimators`	100
	`max_depth`	[2, 6, 10, 16, 20]
	`min_samples_split`	[2, 4, 6, 10, 15, 20]
	`min_samples_leaf`	[2, 5, 10, 15, 20]
	`max_features`	[2, 4, 6, 10, "auto"]
	`max_leaf_nodes`	[25, 50, 75, 100]
xgboost	`n_estimators`	100
	`max_depth`	[2, 4, 6, 8]
	`gamma`	[0.2, 0.4, 0.6, 0.8]
	`min_child_weight`	[2, 5, 8, 11, 14, 17]
	`min_subsample`	[0.2, 0.4, 0.6, 0.8]
	`colsample_bytree`	[0.2, 0.4, 0.6, 0.8]

Tabela 1: Arbitralnie wybrane zakresy hiperparametrów modeli w eksperymentach

Modele nauczone na pełnym zbiorze zmiennych objaśniających posłużą w drugiej iteracji. Po pierwsze, regresja z regularyzacją LASSO ogólnie charakteryzuje się tym, że zeruje współczynniki nieistotnych zmiennych. Istotne



Rysunek 1: Rozkłady wartości wybranych statystyk obliczonych na treningowym zbiorze danych

według regresji zmienne posłużą do przygotowania w drugiej turze modeli lasu losowego oraz xgboost, przy niezmienionych zasadach z poprzedniego paragrafu. Podobnie, wytrenowany model lasu losowego umożliwia podgląd tzw. *feature importances*. Modele regresji, lasu losowego oraz xgboost zostaną wytrenowane w drugiej turze na najbardziej istotnych zmiennych.

Tak samo jak regularyzacja LASSO oraz *feature importances*, do redukcji zbioru zmiennych posłuży analiza głównych składowych (PCA). Modele regresji, lasu losowego oraz xgboost zostaną wytrenowane na wybranym podzbiorze czynników obliczonych za pomocą PCA.

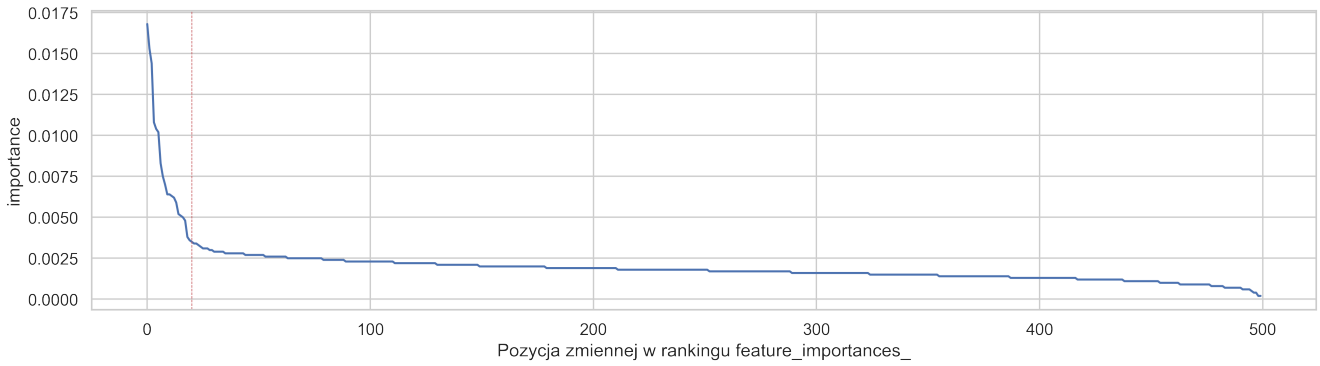
1.2.1 Ograniczenie zbioru zmiennych na podstawie regularyzacji LASSO - szczegóły

Najlepszy wynik (średnią wartość metryki *balanced accuracy* przy poczwórnej cross-walidacji) osiągnął model regresji logistycznej dla wartości parametru regularyzacji `C = 0.01`. W tym modelu tylko dwóm zmiennym została przypisana niezerowa wartość współczynnika. Były to zmienne o numerach 475 (wartość współczynnika 0.241) oraz 48 (wartość współczynnika 0.031).

1.2.2 Ograniczenie zbioru zmiennych na podstawie *feature importances* - szczegóły

Wartości *feature importances* z modelu lasu losowego posortowane malejąco zostały przedstawione na Rysunku 2. Na wykresie widać wyraźny łokieć około 20. zmiennej. Właśnie te pierwsze 20. zmiennych o najwyższych wartościach *feature importances* zostały wybrane do trenowania modeli w drugiej turze. Warto dodać, że wśród tych 20

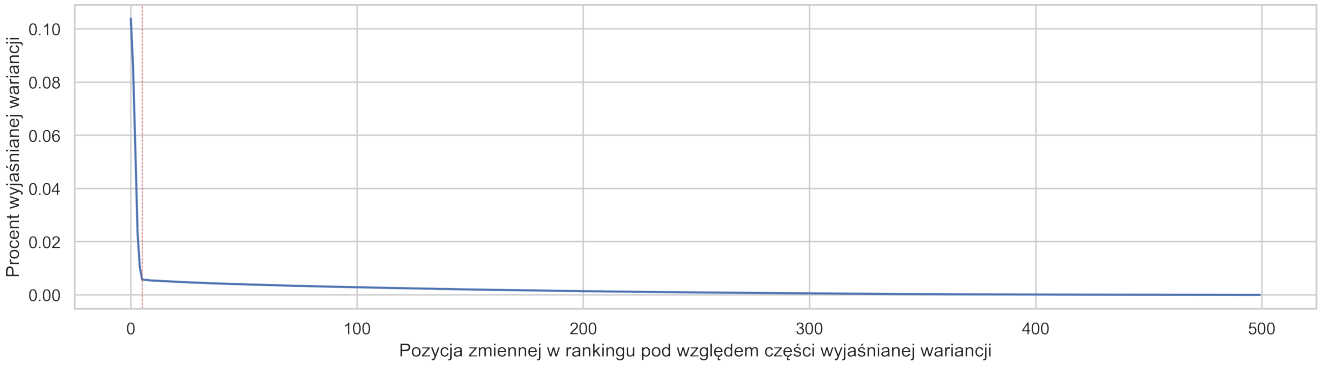
zmiennych znajdują się również te wybrane przez regresję LASSO.



Rysunek 2: Wartości *feature importances* z modelu lasu losowego posortowane malejąco

1.2.3 Ograniczenie zbioru zmiennych za pomocą PCA - szczegóły

Procentowe wartości wyjaśnianej wariancji (*explained variance ratio*) otrzymane za pomocą analizy głównych składowych zostały przedstawione posortowane malejąco na Rysunku 3. Na wykresie widać bardzo wyraźny łokieć na 5. czynniku. Te pierwsze 5. czynników otrzymanych przez transformację PCA zostało wybrane do trenowania modeli.



Rysunek 3: Wartości `explained_variance_ratio_` otrzymane z transformacji PCA posortowane malejąco

1.3 Wyniki oraz wybór najlepszego modelu

Wyniki eksperymentów, to jest średnią wartości metryki *balanced accuracy* przy poczwórnej cross-walidacji, umieszczone są w Tabeli 2. Ograniczanie zbioru zmiennych nie poprawiło wyników regresji logistycznej, prawdopodobne ze względu na "wbudowaną" w ten model selekcję. Wybór zmiennych znacznie poprawił wyniki modeli las losowy oraz xgboost. Najlepszy wynik osiągnął las losowy na podzbiorze zmiennych wybranym za pomocą *feature importances*. Hiperparametry najlepszego modelu zostały jeszcze odrobinę poprawione ręcznie tak, aby zmniejszyć ryzyko zbyt mocnego dopasowania do zbioru treningowego. Te hiperparametry przedstawione są w Tabeli 3. Najlepszy model posłużył do wygenerowania predykcji na zbiorze testowym.

zbiór danych	liczba zmiennych	regresja logistyczna	las losowy	xgboost
wszystkie zmienne	500	0.615	0.647	0.768
LASSO	2	-	0.617	0.605
feature importances	20	0.615	**0.874**	0.867
PCA	5	0.612	0.849	0.851

Tabela 2: Wyniki eksperymentów "ręcznych" - wartości metryki *balanced accuracy*

hiperparametr	wartość
n_estimators	1000
max_depth	12
min_samples_split	8
min_samples_leaf	4
max_features	6
max_leaf_nodes	50

Tabela 3: Hiperparametry najlepszego modelu przygotowanego ręcznie, którym jest model lasu losowego

2 Przygotowanie modelu z wykorzystaniem biblioteki AutoMLowej

Do przygotowania modelu w tej części została wybrana AutoMLowa biblioteka `flaml`. Wybór ten został podyktowany zasłyszaną opinią, że ta właśnie biblioteka nie sprawia problemów w instalacji i użytkowaniu, co zdaje się być ważnym walorem.

Ta opinia okazała się być prawdziwą. Bez żadnych trudności udało się zmusić moduł `flaml.AutoML` do poszukiwania najlepszego modelu uczenia maszynowego na zadanych danych. Wystarczyło określić typ zadania jako klasyfikację, zdefiniować funkcję do minimalizacji (1 - *balanced accuracy*) oraz określić maksymalną liczbę iteracji. Po kilku testowych próbach ustawiłem tą maksymalną liczbę iteracji na 500, ponadto zaznaczyłem, że ewaluacja modeli ma wykorzystywać cross-walidację. Warto dodać, że moduł `flaml.AutoML` zapewnia wiele elastyczności w wyborze testowanych estymatorów oraz przestrzeni poszukiwań hiperparametrów. Te funkcje nie zostały wykorzystane, uznawszy, że w ramach tego zadania wybór estymatora i hiperparametrów powinien zależeć całkowicie od biblioteki autoMLowej.

Zaskoczeniem okazał się wynik modelu wybranego przez bibliotekę `flaml`. Biblioteka testowała modele takie jak lgbm, las losowy, xgboost, inne pochodne modeli drzewiastych oraz regresję logistyczną. Najlepszym modelem okazał się `xgb_limitdepth` i został wybrany, osiągnąwszy wynik `balanced_accuracy=0.43`. Wybrane hiperparametry tego modelu znajdują się w Tabeli 4. Ten wynik jest o tyle zastanawiający, że ręcznie przygotowywane modele na pełnym zbiorze danych osiągały zauważalnie lepsze wyniki.

hiperparametr	wartość
n_estimators	140
max_depth	8
min_child_weight	0.0017
learning_rate	0.0411
subsample	1
colsample_bylevel	0.415
colsample_bytree	0.873
reg_alpha	0.0083
reg_lambda	0.0010

Tabela 4: Hiperparametry najlepszego modelu wybranego `flaml`, którym jest model `xgb_limitdepth`

3 Podsumowanie

Model lasu losowego wytrenowany na wybranym podzbiorze zmiennych to najlepszy ręcznie przygotowany model. Model wybrany przez bibliotekę autoMLową to model `xgb_limitdepth`. Na zbiorze treningowym (przy zastosowaniu cross-walidacji) ręcznie przygotowany model osiągnął *balanced accuracy* na poziomie 0.87, a model autoMLowy uzyskał wynik około 0.43. Zastanawiająca jest dysproporcja otrzymanych wyników. Być może ręcznie przygotowany model jest zbyt mocno dopasowany do danych treningowych, lub też autoMLowa biblioteka `flaml` nie została poprawnie użyta. Prawdę o jakości przygotowanych modeli może powiedzieć tylko wynik na zbiorze testowym.