Raport z Projektu nr 2 z Zaawansowanych Metod Uczenia Maszynowego - Selekcja zmiennych

Wojciech Celej

1. Użyte narzędzia

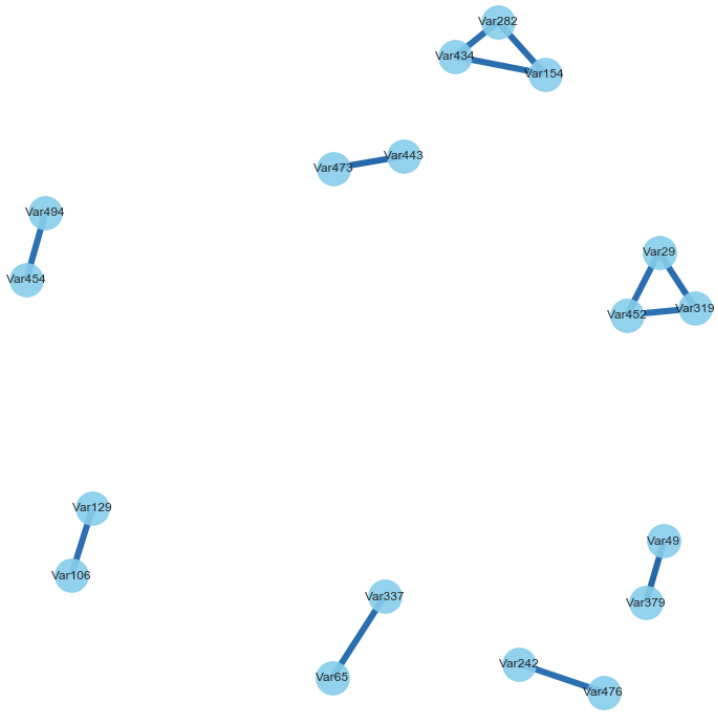
- *sklearn* - implementacja funkcji metryki, selekcja zmiennych oparta o testy chi2 i mutual_info, metoda Lasso, Random Forest używany przez *boruta*
- *boruta* - implementacja nienadzorowanej metody selekcji zmiennych
- *lightgbm* - algorytm boostingowy, implementacja CV
- *shap* - ocena istotności zmiennych
- *matplotlib*, *seaborn*, *networkx*, *tqdm*- wizualizacja wyników

2. Filtrowanie zmiennych

Zbiór treningowy zawiera 2000 obserwacji i 500 zmiennych. Dane są wyłącznie numeryczne, nie występują braki danych, zmienne numeryczne są dodatnie. Główny nacisk w projekcie położony został na metody selekcji zmiennych. W celu uzyskania sensownych wyników konieczne jest wyodrębnienie istotnych cech. Zaimplementowane metody zaczerpnięto z kompletnego artykułu: link. W artykule tym opisano 3 grupy metod selekcji cech (każda z poniższych metod została przetestowana w projekcie):

- *filter methods*
- *wrapper methods*
- *embedded methods*

Dokonano następujących operacji wstępnych:



1. W celu uniknięcia redundancji, usunięto wzajemnie skorelowane zmienne. Jako próg przyjęto korelację >= 0.95. Dla każdej spójnej składowej grafu pozostawiono jednoego reprezentanta. W ten sposób pozbyto się 10 zmiennych.
2. W celu odrzucenia najmniej istotnych cech zastosowano złożoną metodę filtra. Posortowano zmienne wg. ich istotności obliczonej testami *chi2* i *mutual_info_classif* z *sklearn*, następnie wybrano po 400 najistotniejszych cech dla obu testów i obliczono przecięcie tych zbiorów. W ten sposób uzyskano 325 istotnych cech. Zgodnie z opisem dokumentacji z *sklearn* link, obie metody wykrywają również zależności nieliniowe między zmiennymi.

Dalsze kroki różnią się w zależności od wybranego podejścia.

3. Selekcja zmiennych algorytmem *boruta*, trening LightGBM

Poniższe podejście to metoda typu *wrapper*. Opis algorytmu znajduje się na stronie link. Metoda ta jest nienadzorowana. Działa w oparciu o permutację kolumn (*shadow features*) i sprawdzanie, czy po wielu iteracjach (dopasowaniach na modelu lasu losowego) dana prawdziwa zmienna jest bardziej istotna od zaburzonej zmiennej o największej istotności. Taka zmienna w kolejnych iteracjach dostaje punkty, a po przekroczeniu pewnego progu zostaje uznana za istotną.

Zmienne uznane przez model za istotne (wybrano 10 zmiennych): 'Var129', 'Var282', 'Var319', 'Var337', 'Var339', 'Var379', 'Var456', 'Var473', 'Var476', 'Var494'

Następnie dokonano wyboru hiperparametrów dla modelu LightGBM metodą zachłanną, wzorując się na dokumentacji link1 oraz na artykule z TDS link2. Z wyjściowego zbioru treningowego wyodrębniono 10% na zbiór testowy (posłuży do wyznaczenia finalnego rezultatu oraz do narysowania krzywych ROC). Na pozostałych 90% dokonano wyboru optymalnych parametrów stosując 5-krotną CV oraz wczesne zatrzymanie uczenia.

Optymalne parametry: `'objective': 'binary', 'n_jobs': 4, 'learning_rate': 0.04, 'num_iterations': 162 'max_depth': -1, 'min_child_samples': 15, 'min_child_weight': 1, 'num_leaves': 50, 'bagging_fraction': 0.75, 'bagging_freq': 5, 'feature_fraction': 0.75, 'lambda_l2': 0.1, 'max_bin': 200`

4. Lasso

Algorytm Lasso metoda typu *embedded* - zawiera w sobie wbudowany mechanizm regularyzacji l1, polegający na odrzucaniu cech nieistotnych. Jako modelu użyto klasy *LogisticRegressionCV* z *sklearn*, która zawiera w sobie wbudowaną CV (zastosowano również 5-krotną) i automatyczny wybór optymalnego parametru regularyzacji (sprawdzono 20 punktów w skali logarytmicznej pomiędzy [1e-4, 1e4]). Model optymalizowano na tym samym zbiorze treningowym co LightGBM. Wybrana wartość: `C = 0.00026367`

Istotne zmienne wg. Lasso (zbiór ten zawiera się całkowicie w zbiorze zmiennych wybranych poprzednio): 'Var337', 'Var339', 'Var379', 'Var476', 'Var494'

5. Wyniki

Wyniki uzyskane z 5-krotnej CV na zbiorze treningowym:

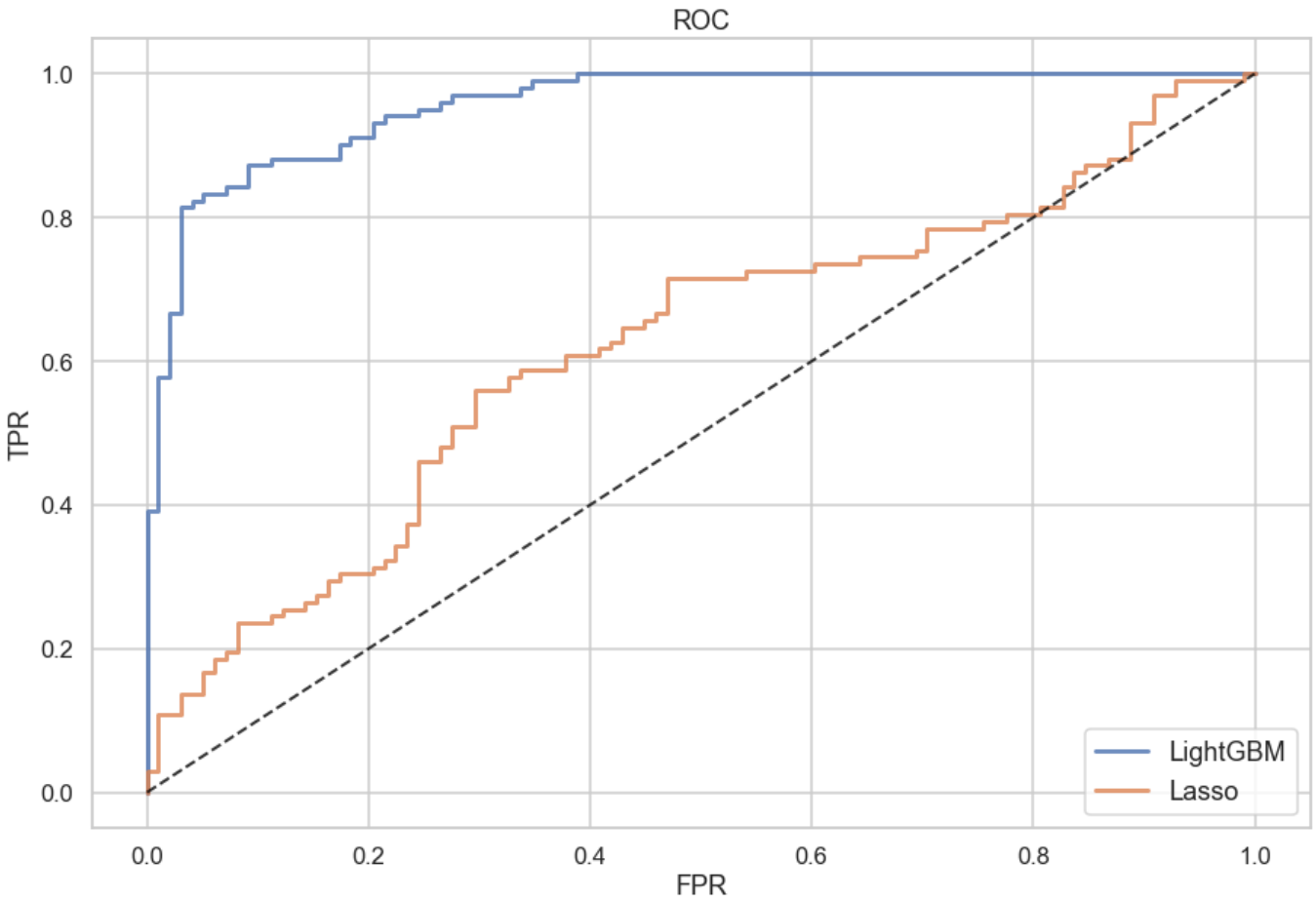
Model	bal_accuracy_mean	bal_accuracy_std
LightGBM	0.8832	0.0153
Lasso	0.6061	0.0279

Wyniki uzyskane na zbiorze testowym:

Model	bal_accuracy	auc
LightGBM	0.8820	0.9569

Model	bal_accuracy	auc
Lasso	0.6269	0.6187

Krzywe ROC:



6. Porównanie ważności zmiennych przed wyborem i po

Częstość wyboru zmiennych przez LightGBM na 10 zmiennych po lewej, na 500 zmiennych po prawej. Widać, że wybór zmiennych jest identyczny. Można z tego wnioskować, że wyboru zmiennych istotnych można by też dokonać jako 10 najistotniejszych zmiennych dla modelu boostingowego wytrenowanego na wszystkich 500 zmiennych (jako próg można by przyjąć 1.25 średniej z ważności zmiennych).