RAPORT

Projekt nr 2

Piotr Olesiejuk

7 czerwca 2019

1 Cel projektu

Celem projektu było zastosowanie wybranych metod selekcji cech, które zostaną wykorzystane w procesie konstrukcji klasyfikatora.

2 Miara oceny klasyfikatora

Miarą przyjętą do oceny jakości klasyfikacji jest balanced accuracy. Sposób jej obliczania jest następujący:

$$BA = \frac{1}{2}(\frac{true\ positive}{positive} + \frac{true\ negative}{negative}) \tag{1}$$

3 Zbiór danych

Zbiór danych podzielony jest na część treningową oraz walidacyjną. Część treningowa posiada zmienną odpowiedzi, natomiast walidacyjna nie. W poniższej tabeli zebrano ilość rekordów oraz zmiennych z poszczególnych zbiorach:

Zbiór	Ilość rekordów	Ilość cech
treningowy	2000	500
walidacyjny	600	500

4 Testowane algorytmy klasyfikacyjne

Do konstrukcji klasyfikatorów oraz w procesie selekcji cech wykorzystano następujący zestaw klasyfikatorów:

1. las losowy (randomForest)
2. regresja logistyczna (glm)
3. xgboost
4. drzewo decyzyjne (rpart)

5 Selekcja cech

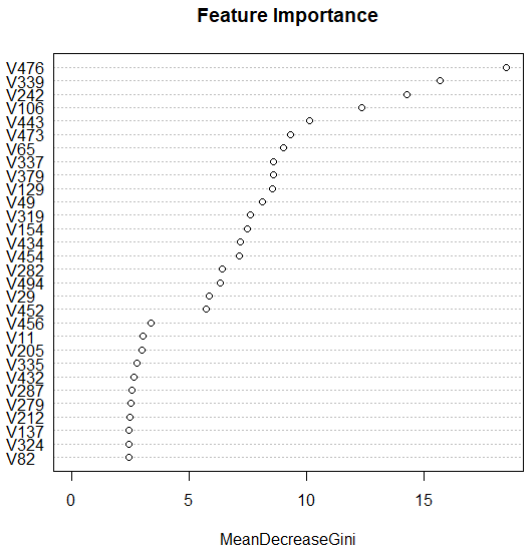
5.1 Pierwsze kroki

W pierwszym podejściu do danych zastosowano algorytm xgboost bez wykorzystania selekcji cech w celu sprawdzenia jakie balanced accuracy jest osiągalne na całym zbiorze. Jeżeli selekcja cech w dalszej części projektu dałaby gorsze wyniki niż takie podejście, wtedy byłby to punkt wyjścia do stwierdzenia, że nie jest ona przeprowadzona poprawnie.
Do testu podzielono zbiór danych na część treningową oraz testową (80/20) i przeprowadzono trening xgboost z wykorzystaniem 5-cio krotnej kroswalidacji. Balanced accuracy na takim zbiorze wyniósł 79,69% co oznacza, że każdy gorszy wynik po selekcji cech powinien być oceniony krytycznie.

5.2 Selekcja cech z wykorzystaniem randomForest

Jako pierwszą metodę selekcji wybrano wybór cech na podstawie ich istotności w algorytmie randomForest. W tym celu wytrenowano randomForest na całym zbiorze treningowym i funkcją importance() dokonano sprawdzenia istotności poszczególnych zmiennych. Następnie istotności te zostały posortowane w kolejności malejącej i wybrano z nich 40 pierwszych zmiennych do dalszej części selekcji.

W kolejnym kroku przeprowadzono trening i test dla algorytmu xgboost dla różnych zestawów zmiennych objaśniających w taki sposób, że pierwszy trening przeprowadzono dla dwóch najbardziej istotnych zmiennych, a każdy kolejny trening był przeprowadzony na starym zestawie zmiennych objaśniających powiększonym o kolejną, najbardziej istotną zmienną. Przy takim podejściu zostało wybrane 15 najistotniejszych cech. Dokładanie kolejnych cech nie przynosiło znaczącej poprawy BA, powtarzanie testu wykazało, że dalsza poprawa mocno zależy od wstępnego podziału na próbę testową i treningową w kroswalidacji i może wynieść maksymalnie ok. 1 %, co zdecydowało o pozostawieniu 15 cech. Wytrenowany na tych zmiennych model wykazał wartość BA = 85,71%. Poniżej znajduje się tabela podsumowująca opisane podejście wyboru zmiennych.



Rysunek 1: Istotność cech w selekcji z wykorzystaniem randomForest

Jak widać istotnymi zmiennymi z punktu widzenia modelu najbardziej okazały się zmienne "V476", "V339", "V242" oraz "V106". Poniżej znajduje się tabela podsumowująca opisane podejście wyboru zmiennych.

Podstawa selekcji	istotność cech w modelu randomForest
Klasyfikator	xgboost
Ilość wybranych istotnych cech	15
Balanced accuracy	85,71%

5.3 Information Gain

Ciekawą sprawą jest, że information gain, które de facto jest opisane jako algorytm do selekcji cech przy ciągłych zmiennych objaśnianych dał dużo lepsze wyniki niż następna metoda. Information gain wybrał 7 cech i dla kolejnych algorytmów dał następujące wyniki:

Model	Balancced accuracy
xgboost	74.8%
rpart	66.5%
glm	57,21%

5.4 Selekcja cech z wykorzystaniem filtrowania oraz glm

Jako drugą metodę przetestowano filtrowanie dostępne w pakiecie bounceR oraz varImp na modelu glm. W tym celu metodą featureSelection wybrano wstępny zestaw predyktorów, który następnie postłużył do budowy modelu logistycznego. Model logistyczny ma wbudowaną variable importance, którą zostały wybrane cechy najbardziej istotne w modelu. Natępnie ten zestaw cech został wykorzystany do wytrenowania modelu xgboost. Metoda ta nie dała jednak dobrych wyników, bardzo prawdopodobne, że metoda featureSelection ucięła istotne predyktory, ponieważ skuteczność xgboost na kroswalidacji wyniosła zaledwie nieco ponad 53% dla kilkunastu predyktorów co by wskazywało, że zostały wybrane głównie nieistotne predyktory.

6 Pliki programu

1. functions.R - zawiera wszystkie zdefiniowane funkcje wykorzystane w projekcie
2. projekt.R - zawiera kod, który wykonuje selekcję cech oraz trenowanie modeli
3. evaluate.R - plik w którym trenowany jest finalny model i dokonywana predykcja na zbiorze walidacyjnym