Warsztaty badawcze - projekt indywidualny

Mateusz Urbański

16 października 2017

Wybór zmiennych do modelu

W celu znalezienia najodpowiedniejszych zmiennych które mają wpływ na zmienną wynikową posłużono się dwiema metodami:

- (1) dla każdej objaśniającej obliczono test Chi-Kwadrat w porównaniu z zmienną objaśnianą Y.
- (2) przy użyciu funkcji GLM obliczono istotność każdej ze zmiennych objaśniających

Dla każdej z metod skonstruowano listę zmiennych objaśniających od najważniejszych do najmniej ważnych. Poniżej dla każdej z metod wypisano 13 najważniejszych zmiennych

```
##  [1] "M1" "W1" "X2" "E1" "P2" "U2" "F2" "J1" "Q1" "T2" "N2" "F1" "A1"
```

```
##  [1] "W1" "E1" "P2" "U2" "Q1" "T2" "M1" "F1" "B1" "P1" "O2" "A1" "Z1"
```

Dane treningowe/testowe

Zbiór 50 000 rekordów podzielono na dwa zbiory po 25 000 elementów będące nowym zbiorem treningowym oraz testowym.

Po obliczeniach algorytmu na zbiorze testowym brano 5000(20%) rekordów które mają największe prawdopodobieństwo przynależności do "klasy +". Sprawdzano ile z nich tak naprawdę należy do tej klasy.

Wybór parametrów algorytmu XGBoost

Podczas laboratoriów jednym z najskuteczniejszych algorytmów okazał się XGBoost. Przetestowano więc go w różnych konfiguracjach aby znaleźć najlepsze możliwie parametry. Poddane testowniu były parametry:

- metoda wybory zmiennych (pierwsza lub druga z opisanych wyżej)
- zmienne - ilość zmiennych objaśniających na podstawie których budowany był model. Brano 5, 8, 11, 12, 13 najlepszych zmiennych wyznaczonych wcześniej opisanymi metodami.
- rundy - parametr rund uczenia algorytmu XGBoost. Brano 5,15,50,70,100 rund
- głębokość - parametr głębokości algorytmu XGBoost. Brano 1,2,5,10,20
- parametr ETA - parametr szybkości uczenia. Dla niskich wartości może dłużej liczyć, dla wysokich może być mało dokładny. Brano 0.001,0.01,0.05,0.1,0.3

Wynik dla XGBoost

Najwyższy wynik na zbiorze testowym uzyskano dla parametrów:

- zmienne(11 zmiennych wybranych metodą Chi-Kwadrat)

```
##  [1] "M1" "W1" "X2" "E1" "P2" "U2" "F2" "J1" "Q1" "T2" "N2"
```

- rundy = 50
- glebokosc = 1

- eta = 0.1

Wynik dokładności: 84.26%

Wynik dla XGBoost - Generalized Linear Model

Testy z poprzedniego punktu przeprowadzono także dla XGBoost z ustawionym parametrem booster="gblinear" któy powoduje użycie regularyzowanego modelu liniowego.

Dla tego algorytmu najlepsze okazały się parametry:

- zmienne(8 zmiennych wybranych metodą Chi-Kwadrat)

```
## [1] "M1" "W1" "X2" "E1" "P2" "U2" "F2" "J1"
```

- rundy = 70
- glebokosc = 1

Dokładność: 83%

Metoda AdaBoost

Sprawdzono także działanie algorytmu AdaBoost. Ze względu na o wiele wolniejsze działanie tego algorytmu niż XGBoost testy zostały przeprowadzone na wersji tego algorytmu zmieniając jedynie ilość iteracji. Algorytm ten został przetestowany na zmiennych, które okazały się być najlepsze dla XGBoost tzn.

```
##  [1] "M1" "W1" "X2" "E1" "P2" "U2" "F2" "J1" "Q1" "T2" "N2"
```

Parametry:

- iteracje = 70

Dokładność: 80%

Wybrany model

Ostatecznie wybrano model XGBoost z 11 zmiennymi ze względu na najwyższą dokłądność wynoszącą 84.26%