Maciej Zaborek

SPDB - Dokumentacja końcowa

Implementacja wybranego algorytmu eksploracji danych - Koperski, Stefanovic

1. Opis algorytmu

Źródło algorytmu: http://hanj.cs.illinois.edu/pdf/sdh98.pdf

Wynikiem działania algorytmu jest powstanie binarnego drzewa decyzyjnego. W pierwszej kolejności przetwarzane są przestrzenne dane. Każdy z klasyfikowanych obiektów zostaje opisany szeregiem predykatów przestrzennych opisujących jego otoczenie.

Następnie dla zbioru predykatów stosowany jest algorytm *RELIEF*. Dla każdego klasyfikowanego obiektu znajdowani są dwaj najbliżsi sąsiedzi - jeden należący do tej samej klasy co rozpatrywany obiekt i jeden nienależący do niej. Jeśli obiekt tej samej klasy ma taką samą wartość predykatu, to jego waga zostaje zwiększona. Analogicznie, jeśli dla najbliższego obiektu należącego do innej klasy wartość predykatu jest inna, waga zostaje zwiększona. W przeciwnych przypadkach waga zostaje zmniejszona.

Po rozpatrzeniu wszystkich obiektów odrzucone zostają te predykaty przestrzenne, których waga jest poniżej przyjętego progu.

W następnym kroku każdy obiekt zostaje opisany pozostałymi predykatami, których określenie nie wymaga już wyznaczania przestrzennych zależności.

Otrzymany zbiór predykatów zostaje wykorzystany do zbudowania binarnego drzewa decyzyjnego.

2. Dane

W projekcie wykorzystane zostały dane dostępne na stronie przedmiotu pochodzące ze spisu powszechnego z Nowego Jorku. Ich opis można znaleźć pod adresem: http://workshops.boundlessgeo.com/postgis-intro/about_data.html.
Dane te zostały zaimportowane do bazy danych (*Postgresql* + *Postgis*). Zawierają one tabele dotyczące danych demograficznych i społeczno-ekonomicznych mieszkańców poszczególnych obszarów miasta oraz dane dotyczące stacji metra, ulic oraz zabójstw. Dla potrzeb implementacji utworzona została tabela *nyc_census_tracts*, która składa się z połączonych bloków *nyc_census_blocks*. Każdemu obszarowi (tract) odpowiada wpis w tabeli *nyc_census_sociodata*, która zawiera dane społeczno-ekonomiczne opisujące dany tract. To opisane przez nią obszary, których jest 2166 zostały sklasyfikowane.

3. Implementacja

Powyższy algorytm został zaimplementowany w języku python. Klasyfikacja dotyczy opisanych w bazie danych obszarów (tract). Obszary są klasyfikowane pod względem

bezpieczeństwa (czy liczba zabójstw rocznie na hektar przekracza 4 (średnia dla Nowego Jorku to ponad 5) lub pod względem dominacji danego środku transportu wśród pracowników na danym obszarze.

Sposób uruchomienia programu:
python3 spdb.py -n 5 -c transit_other -t 100

Parametry:

- n - liczba przeprowadzonych testów. Otrzymany wynik jest uśredniony.
- c - rodzaj klasyfikacji. Możliwe wartości, to:
    - “homicides” - klasyfikacja obszarów pod względem bezpieczeństwa
    - transit_private, transit_public, transit_walk, transit_walk, transit_other - dominacji wśród pracowników na danym obszarze jednego ze środków transportu
- t - próg wykorzystany w algorytmie RELIEF (domyślnie 100)

Pojedynczy test, na który składa się: budowa klasyfikatora i jego testowanie dla danych losowo podzielonych na zbiory trenujące i testowe oraz ocena jakości klasyfikacji jest wykonywany w funkcji *runTest*.

W pierwszej kolejności dane zostają podzielone na dane trenujące oraz testowe w stosunku 9 do 1.

Dane dla każdego z obszarów (tract) są przechowywane w klasie *DBData*. Zawiera ona id danego obszaru, zbiór predykatów opisujących go oraz wartość klasyfikacji.

Następnie dla danych testowych zostaje zbudowany zbiór predykatów przestrzennych w funkcji *getDataFromDB*. Pobierane są dane dane dotyczące obecności stacji metra oraz typów ulic, które przecinają się z minimalnym prostokątem ograniczającym (minimal bounding box) dany obszar.

Dla zbioru predykatów zostaje zastosowany algorytm RELIEF. Funkcja *reliefAlg* zwraca tablicę wag dla predykatów. Następnie do predykatów, których waga jest powyżej zadanej progowej wartości zostają dołączone predykaty zbudowane na podstawie tabel *nyc_census_tracts* oraz *nyc_census_sociodata* w funkcji *getNonSpacialPredicates*. Dotyczą one między innymi: stosunkowej przynależności rasowej mieszkańców danego obszaru, ich edukacji czy dochodów. W przypadku klasyfikacji bezpieczeństwa obszaru dodane są też dane dotyczące środków transportu wykorzystywanych przez pracowników na danym obszarze.

Posiadając zbudowany ostateczny zbiór predykatów dla każdego klasyfikowanego obiektu zbudowane zostaje drzewo decyzyjne. W tym celu skorzystano z implementacji z pakietu *sklearn*. Bazuje on na algorytmie CART. Wynikowe drzewo decyzyjne zostaje zapisane do pliku w formacie dot. Drzewo ma maksymalną głębokość 5, ponieważ większe wartości nie dawały lepszych rezultatów.

Następnie otrzymany model jest testowany na zbiorze testowym. Funkcja *runTest* zwraca: współczynnik precyzji, współczynnik odzysku oraz współczynnik dokładności.

4. Wyniki testów i wnioski

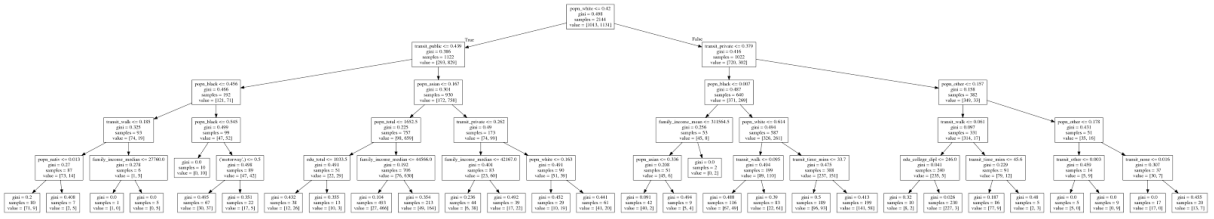
Czas klasyfikacji bezpieczeństwa danego obszaru w zależności od progu przyjętego przez algorytm *RELIEF*:

próg	liczba predykatów spełniających warunek	czas klasyfikacji [s]	dokładność
-3000	21	0.2921	0.5454
20	10	0.2448	0.5454
50	5	0.2338	0.6875
100	4	0.20	0.8333
200	1	0.1736	0.8666

Obserwacje pokrywają się z danymi z artykułu opisującego oryginalną implementację. Zwiększenie progu powoduje, że waga większej liczby predykatów przekracza wartość progową. W konsekwencji prowadzi to do wydłużenia czasu klasyfikacji (więcej zapytań przestrzennych do bazy danych). Rośnie również jakość klasyfikacji, ponieważ odrzucane są predykaty, które nie wnoszą informacji o przynależności próbki do klasy.

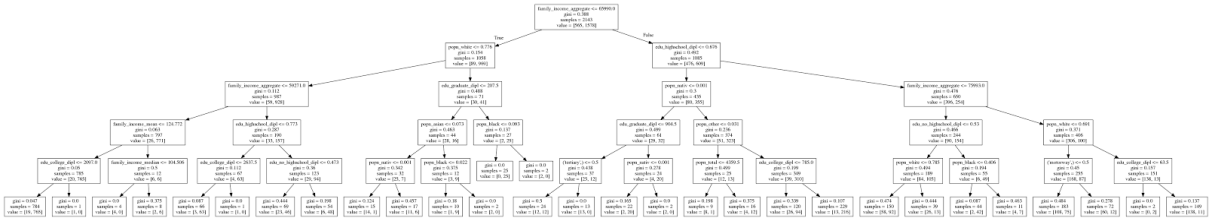
a. Dla klasyfikacji bezpieczeństwa danego obszaru. Dane uśrednione dla 10 testów przy progu algorytmu RELIEF wynoszącym 100.
   - Precyzja: 0.7127838827838828
   - Odzysk: 0.7561344211344212
   - Dokładność: 0.7272727272727273

   Wynikowe drzewo decyzyjne.



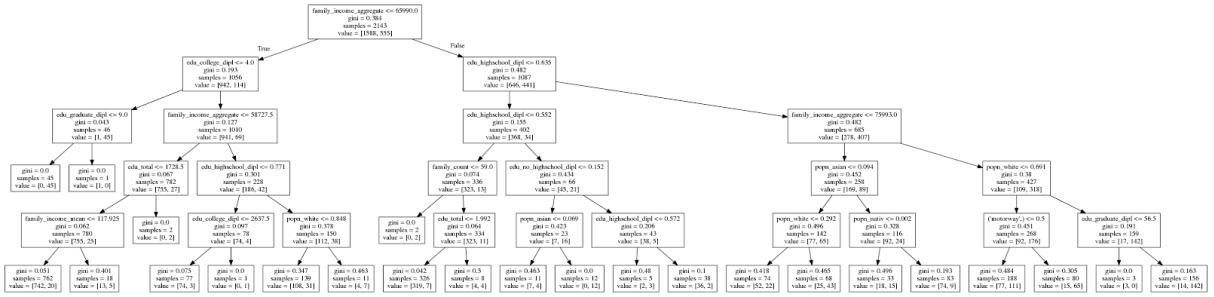
b. Dla klasyfikacji dominacji transportu **publicznego** dla progu algorytmu RELIEF wynoszącego 100. Średnie wartości dla 10 testów:
   - Precyzja: 0.8650656288156288
   - Odzysk: 0.8704669762641899
   - Dokładność: 0.8045454545454545

   Wynikowe drzewo decyzyjne:



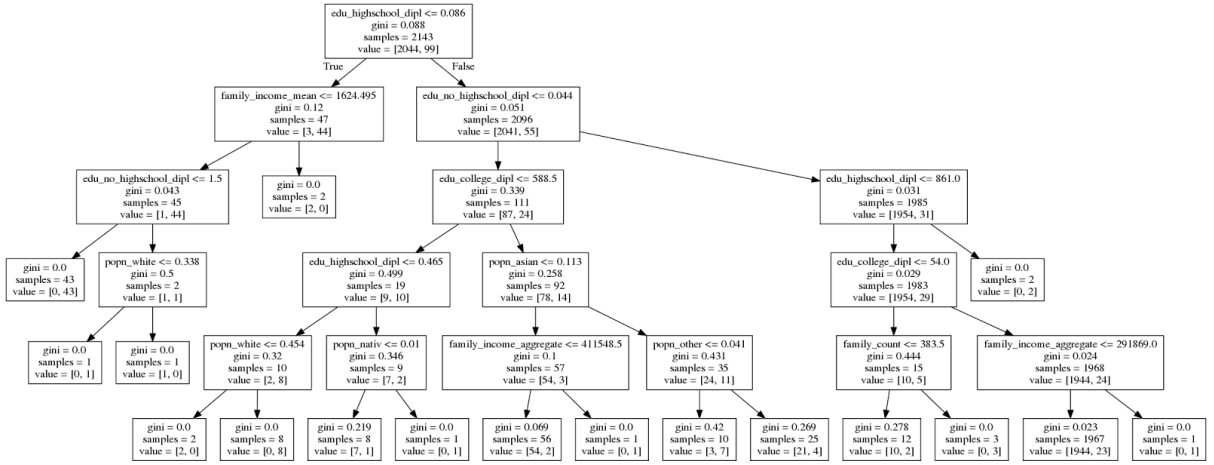
c. Dla klasyfikacji dominacji transportu **prywatnego** dla progu algorytmu RELIEF wynoszącego 100. Średnie wartości dla 10 testów:
   - Precyzja: 0.7111111111111111
   - Odzysk: 0.5952380952380952
   - Dokładność: 0.8181818181818182

   Wynikowe drzewo decyzyjne:



d. Dla klasyfikacji dominacji transportu **pieszego** dla progu algorytmu RELIEF wynoszącego 100. Średnie wartości dla 10 testów:
   - Precyzja: 0.6666666666666666
   - Odzysk: 0.5
   - Dokładność: 0.9848484848484849

   Wynikowe drzewo decyzyjne:



Dla wykorzystanych danych klasyfikacja daje przeciętne wyniki. Dokładność wynosi około 80%. Liczba próbek (2166) jest dość mała, przez co wynik jest mocno zależny od tego, jak rozlosowane zostaną próbki pomiędzy zbiorami trenującymi i testowymi.

Algorytm zachowuje się zgodnie z wnioskami przedstawionymi w oryginalnej publikacji. Zastosowanie dwustopniowego podejścia do budowy zbioru predykatów daje korzystne wyniki pod względem czasu przetwarzania. Niestety, dla przykładowego zbioru danych wykorzystanego w powyższej implementacji, ostateczne wyniki klasyfikacji nie są zadowalające.