Reguły asocjacyjne w analizie danych

K. Król
M. K. Karpiński

Uniwersytet Wrocławski
Instytut Informatyki
SPRAWOZDANIE Z PROJEKTU

Wrocław, dnia 24 czerwca 2012 r.

Spis treści

1 Wstęp

Dokument ten został sporządzony jako sprawozdanie do projektu z przedmiotu: Eksploracja Danych. Projekt ma na celu zbadanie danych ze świata rzeczywistego w poszukiwaniu ciekawych reguł asocjacyjnych z nimi związanych.

Kolejne rozdziały niniejszego sprawozdania zawierają opis sporządzonych eksperymentów, począwszy od znalezionych danych i wykorzystanego oprogramowania, poprzez testy właściwe, kończąc krótkim podsumowaniem wykonanego zadania.

2 Dane

Dane wykorzystane w projekcie zostały pobrane ze strony `http://repository.seasr.org/Datasets/UCI/arff/vote.arff`. Dane te zawierają głosy każdego z kongresmenów U.S. House of Representatives dotyczących 16 istotnych problemów publicznych w roku 1984. Docelowo ten zbiór danych wykorzystywano do klasyfikacji ludzi na demokratów/republikanów w zależności od stosunku do podanych problemów:

1. handicapped-infants (wsparcie dla upośledzonych noworodków)
2. water-project-cost-sharing (podział kosztów dla gospodarki wodnej)
3. adoption-of-the-budget-resolution (akceptacja budżetu (rok 1984))
4. physician-fee-freeze (wstrzymanie wypłat lekarzom)
5. el-salvador-aid (pomoc dla kraju El Salvador)
6. religious-groups-in-schools (grupy religijne w szkołach)
7. anti-satellite-test-ban (zakaz prób militarnych w kwesti rakiet przeciwsatelitarnych)
8. aid-to-nicaraguan-contras (wsparcie dla partyzantów w Nikaragui)
9. mx-missile (budowa pocisku MX-Missile)
10. immigration (wsparcie dla imigrantów)
11. synfuels-corporation-cutback (zmniejszenie wydatków w Synfuels Corp.)
12. education-spending (wydawanie pieniędzy na edukację)

13. superfund-right-to-sue (prawo do pozwania urzędów państwowych w sprawie zwrotu kosztów za czyszczenie zanieczyszczonych terenów prywatnych)

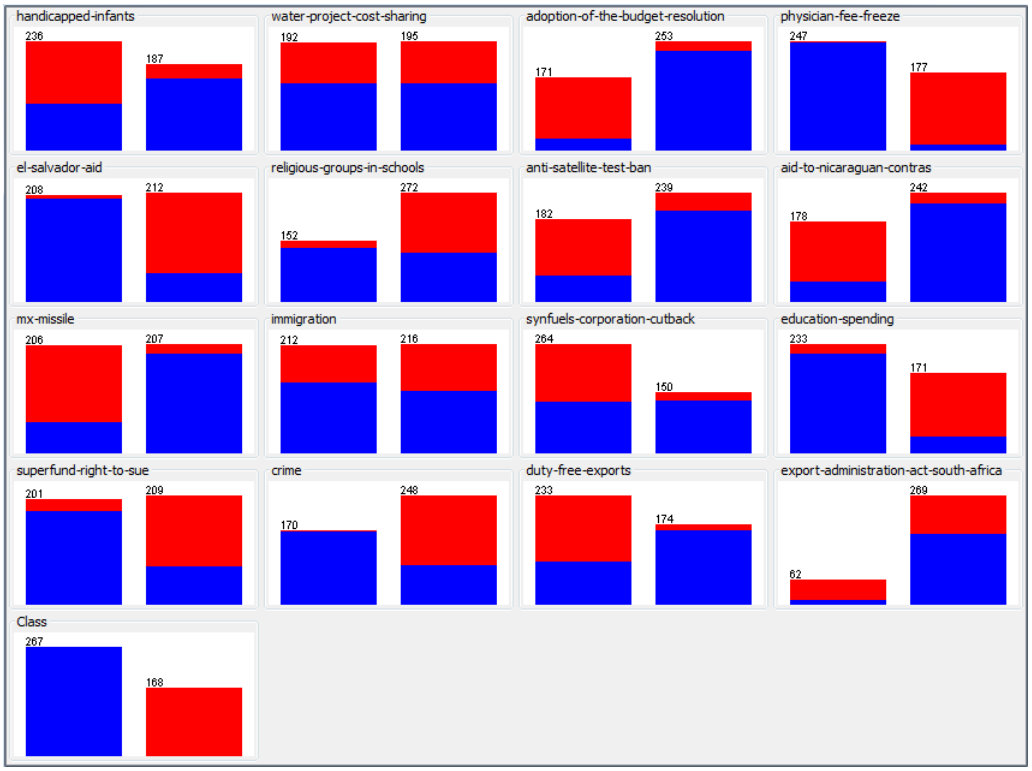
14. crime (walka z przestępszością)

15. duty-free-exports (bezcłowy eksport)

16. export-administration-act-south-africa (zakaz eksportu materiałów militarnych do południowej afryki)

Liczba demokratów i republikanów kształtowała się następująco (demokraci - kolor niebieski, republikanie - kolor czerwony):

Głosy typu 'voted for', 'paired for', 'announced for' zostały sklasyfikowane jako głosy 'yes', 'voted against', 'paired against', 'announced against' jako głosy 'no', a głosy 'voted present', 'voted present to avoid conflict of interest' i brak głosu zostały sklasyfikowane jako 'unknown' (oznaczone przez '?'). Poniżej widać tabele, na których oznaczone są głosy za i przeciw z podziałem na demokratów i republikanów (lewa kolumna - głosy 'no', prawa kolumna - głosy 'yes'):



Jak już wspomniano, danych tych użyto do klasyfikacji głosujących. Można jednak spojrzeć na te dane inaczej i spróbować wyszukiwać pewnych tendencji w głosowaniu na niektóre problemy. Chcemy dokonać tego poprzez wydobycie ciekawych reguł asocjacyjnych związanych z powyższymi danymi.

3 Oprogramowanie

Do wykonanie eksperymentów posłużyliśmy się programem wspomagającym eksplorację danych WEKA dostępnym na stronie: `http://www.cs.waikato.ac.nz/ml/weka/`.

4 Algorytmy

Wykorzystaliśmy niektóre algorytmy dostępne w programie WEKA:

1. Apriori
2. FPGrowth
3. Tertius

4.1 Apriori

Algorytm apriori był szeroko omawiany na wykładzie, dlatego też pominiemy jego opis w niniejszej pracy.

4.2 FPGrowth

Algorytm FPGrowth korzysta z oszczędnej (pod względem pamięciowym) struktury FP-drzewa. Definicja FP-drzewa:

- ukorzeniony, etykietowany graf acykliczny
- korzeń posiada etykietę 'null', pozostałe wierzchołki reprezentują 1-elementowe zbiory częste
- każdy wierzchołek zawiera ponadto liczbę transakcji wspierających dany zbiór częsty

Algorytm opiera się na dwóch krokach:

1. Kompresja bazy danych D i przekształcenie do FP-drzewa
2. Eksploracja FP-drzewa

W 1 kroku znajduje się najpierw wszystkie 1-elementowe zbiory częste w bazie D. Następnie, każdą tranksakcję $T_i \in D$ zamienia się na skompresowaną transakcję $T_{r_i} \in D$ usuwając z T_i wszystkie elementy, które nie są częste. W ostatniej fazie, skompresowane transkacje są sortowane malejąco po wartości wsparcia. Następnie konstruowane jest FP-drzewo:

1. utwórz korzeń z etykietą 'null'
2. Dla każdej transakcji T_{r_i} utwórz ścieżkę w FP-drzewie. Transakcje o wspólnym prefiksie, współdzielą istniejące ścieżki. W przypadku wystąpienia różnicy, ścieżka jest rozdzielana. Ostatni węzeł ścieżki zawiera liczbę transakcji wspierających zbiór elementów reprezentowany przez całą ścieżkę.

Ponadto przechowuje się 'tablicę nagłówków elementów', która to jest tablicą wskaźników na poszczególne elementy w drzewie, co przyspiesza i ułatwia przeszukiwanie drzewa.

W 2 kroku, następuje eksploracja uprzednio przygotowanego FP-drzewa. Porces ten opiera się na obserwacji, że dla każdego 1-elementowego zbioru częstego α, wszystkie częste nadzbiory zbioru α są reprezentowane przez ścieżkę zawierającą ścieżkę dla α. Zatem eksploracja drzewa przebiega następująco:

1. Dla każdego 1-elementowego zbioru częestego α znajdujemy wszystkie ścieżki w drzewie, które kończą się wierzchołkiem α

2. Ścieżką prefiksową wzorca α nazwiemy ścieżkę, która kończy się wierzchołkiem α. Zbiór wszystkich ścieżek prefiksowych wzorca tworzy warunkową bazę wzorca. Na podstawie warunkowej bazy wzorca tworzymy warunkowe FP-drzewo wzorca α: $Tree - \alpha$.

3. Następnie wywołujemy się rekurencyjnie dla drzewa $Tree - \alpha$ w celu znalezienia wszystkich zbiorów częstych zawierających α.

Konstrukcja FP-drzewa wymaga $O(|D|)$ czasu, natomiast 2. krok wymaga $O(|\text{tablica nagłówków}|^2 \times \text{głębokość FP-drzewa})$. W odróżnieniu od *Apriori*, *FPGrowth* nie wymaga załadowania do pamięci całej bazy danych.

4.3 Tertius

Algorytm Tertius buduje reguły z par atrybut-wartość z danych treningowych oraz porządkuje je według tego, jak bardzo są wiarygodne - czyli ile razy reguła uznana jest za prawdziwą w danych treningowych.

Reguła składa się z ciała (body) i głowy (head). Ciało zawiera warunki (literały) potrzebne do spełnienia danej reguły i może składać się z dowolnej liczby literałów. Głowa zawiera zdarzenie, które występuje, gdy reguła jest prawdziwa. Podczas wyszukiwania reguł, Tertius rozpoczyna od pustej reguły - takiej, która zawiera puste ciało i pustą głowę. Reguła jest następnie udoskonalana poprzez dodawanie par atrybut-wartość w kolejności, w jakiej pojawiają się w zbiorze danych. Gdy to nastąpi, algorytm zlicza ile razy reguła jest spełniona (zarówno ciało i głowa są prawdziwe) oraz ile razy reguła daje wynik false-positive (gdy ciało jest prawdą, ale głowa jest nieprawdziwa).

Jedną z wad algorytmu Tertius jest jego względnie długi czas pracy, który w dużej mierze zależy od liczby literałów w regułach. Zwiększenie dozwolonej liczby literałów zwiększa czas pracy wykładniczo. Mając A atrybutów przyjmujących średnio V różnych wartości, chcąc wyszukiwać reguły zawierające literały długości maksymalnie n, liczba możliwych reguł jest rzędu $(AV)^n$.

5 Analiza Danych

Poniżej znajduje się podsumowanie 7-miu (z ok. 50-ciu) testów wykonanych różnymi algorytmami z różnym stopniem parametryzacji. Po wykonaniu testów liczba znalezionych reguł była zbyt duża, aby w czytelnej formie

zaprezentować je w niniejszej pracy. Dlatego pod każdym z testów znajduje się krótkie podsumowanie wraz kilkoma ręcznie wybranymi, ciekawymi (według autorów) regułami asocjacyjnymi.

Poniżej znajduje się opis parametrów każdego z algorytmów.

1. Apriori
    - N : liczba najlepszych (według metryki) zwracanych reguł
    - T : metryka (0=confidence; 1=lift; 2=leverage)
    - C : minimalna wartość metryki
    - D : delta. Wartość zmniejszająca min-support z każdą iteracją algorytmu
    - U : górna granica min-support
    - M : dolna granica min-support
2. FPGrowth
    - oprócz parametrów opisanych powyżej, dodano:
    - I : maksymalna liczba elementów w zbiorach częstych (-1=inf.)
3. Tertius
    - K : liczba najlepszych zwracanych reguł
    - F : minimalna wiarygodność reguł
    - N : szum. maksymalna proporcja instancji niespełniających danej reguły
    - L : maksymalna liczba literałów w regule
    - G : gdzie negacja może występować w regułach (0=brak; 1=w ciele; 2=w głowie; 3=w obu)

Wybór przedstawionych reguł jest podyktowany przez najwyższe wartości metryki w danym teście. Dane reguły mówią o tendencji wyboru dla pewnych ważnych sytuacji społecznych. Wiedza ta zakrawa o takie sfery jak psychika i moralność człowieka, dlatego aby poprawnie przeanalizować przedstawione wyniki potrzebna jest wiedza z zakresu psychologii, socjologii i filozofii, której autorzy niniejszej pracy niestety nie posiadają. Potrafimy jedynie dokonać suchej analizy danych, wskazując mocne powiązania pomiędzy pewnymi problemami życia publicznego, jednak interpretacje tych wyników pozostawiamy ekspertom.

5.1 Apriori 1

- Parametry: -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1
- Metryka: confidence
- Ciekawsze reguły:
  - (conf.=1.00) adoption-of-the-budget-resolution=y physician-fee-freeze=n $->$ Class=democrat
  - (conf.=1.00) physician-fee-freeze=n aid-to-nicaraguan-contras=y $->$ Class=democrat
  - (conf.=0.98) el-salvador-aid=n aid-to-nicaraguan-contras=y $->$ Class=democrat
- Wniosek: standardowy zestaw parametrów polecany przez program nie daje spodziewanych rezultatów. Otrzymaliśmy same reguły klasyfikujące demokratów. Nie mniej jednak z otrzymanego zestawu reguł można wydobyć jakąś wiedzę: demokraci mają podobne zdanie na temat niektórych problemów społecznych. Republikanie bardziej różnią się w swoich poglądach.

5.2 Apriori 2

- Parametry: -N 10 -T 1 -C 1.1 -D 0.05 -U 1.0 -M 0.1
- Metryka: lift
- Ciekawsze reguły:
  - (lift=1.67) physician-fee-freeze=n $->$ Class=democrat
  - (lift=1.67) Class=democrat $->$ physician-fee-freeze=n
  - (lift=1.63) adoption-of-the-budget-resolution=y $->$ physician-fee-freeze=n
  - (lift=1.63) physician-fee-freeze=n $->$ adoption-of-the-budget-resolution=y
- Wniosek: zmiana metryki i lekka zmiana parametrów zaowocowała dwoma nowymi wnioskami: pierwszy znów dotyczy klasyfikacji. Zauważmy, że dwie pierwsze reguły tworzą równoważność. Dostajemy więc mocną zależność między problemem physician-fee-freeze a byciem demokratą. Drugi wniosek jest dla nas bardziej interesujący. Kolejne

dwie reguły również tworzą równoważność, co daje nam mocną zależność między problemami adoption-of-the-budget-resolution a physician-fee-freeze.

5.3 Apriori 3

- Parametry: -N 20 -T 2 -C 0.1 -D 0.1 -U 1.0 -M 0.2
- Metryka: leverage
- Ciekawsze reguły:
  - (lev.=0.21) anti-satellite-test-ban=y physician-fee-freeze=n aid-to-nicaraguan-contras=y $->$ el-salvador-aid=n
  - (lev.=0.18) el-salvador-aid=n $->$ physician-fee-freeze=n anti-satellite-test-ban=y aid-to-nicaraguan-contras=y
- Wniosek: zwiększenie liczby poszukiwanych reguł oraz kolejna zmiana metryki dały nam ciekawsze wyniki.

5.4 FPGrowth 1

- Parametry: -P 2 -I -1 -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1
- Metryka: confidence
- Ciekawsze reguły:
  - (conf.=0.99) el-salvador-aid=y, Class=republican: $->$ physician-fee-freeze=y
  - (conf.=0.98) crime=y, Class=republican: $->$ physician-fee-freeze=y
- Wniosek: zmiana algorytmu wyznaczyła nam zmianę opcji politycznej. Algorytm FPGrowth preferuje reguły związanie z republikanami.

5.5 FPGrowth 2

- Parametry: -P 2 -I -1 -N 10 -T 1 -C 0.9 -D 0.05 -U 1.0 -M 0.1
- Metryka: lift
- Ciekawsze reguły:
  - (lift=1.58) crime=y $->$ religious-groups-in-schools=y

  - (lift=1.58) adoption-of-the-budget-resolution=y − > aid-to-nicaraguan-contras=y
  - (lift=1.53) aid-to-nicaraguan-contras=y − > adoption-of-the-budget-resolution=y
  - (lift=1.53) el-salvador-aid=y − > religious-groups-in-schools=y
- Wniosek: z metryką lift nasz algorytm wyzbył się reguł posiadających klasyfikator. Baza wiedzy poszerza się o nowe reguły.

5.6 FPGrowth 3

- Parametry: -P 2 -I -1 -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1
- Metryka: conviction
- Ciekawsze reguły:
  - (conv.=4.96) superfund-right-to-sue=y − > crime=y
  - (conv.=3.62) mx-missile=y − > adoption-of-the-budget-resolution=y
  - (conv.=3.53) religious-groups-in-schools=y, crime=y − > el-salvador-aid=y
  - (conv.=3.30) mx-missile=y − > anti-satellite-test-ban=y
- Wniosek: zmiana parametrów oraz metryka conviction daje nam jeszcze więcej reguł.

5.7 Tertius 1

- Parametry: -K 10 -F 0.5 -C 0.5 -N 1.0 -L 5 -G 0
- Ciekawsze reguły:
  - education-spending = n − > crime = n
  - duty-free-exports = n − > aid-to-nicaraguan-contras = n
  - el-salvador-aid = y and crime = y − > adoption-of-the-budget-resolution = n or physician-fee-freeze = y
  - aid-to-nicaraguan-contras = y and Class = democrat − > physician-fee-freeze = n or mx-missile = y
- Wniosek: algorytm Tertius daje więcej możliwości niż poprzednie algorytmy. Zamiast zbiorów, po obu stronach implikacji znajdują się wyrażenia boolowskie, co czyni wyszukane reguły bardziej interesującymi.

6 Podsumowanie

Algorytmy wyszukujące reguły asocjacyjnie niewątpliwie przydają się do wydobywania wiedzy ze zgromadzonych danych. W wykorzystanym zbiorze zauważyliśmy dużą różnorodność znalezionych reguł w zależności od uruchomionego algorytmu. Dlatego oczywistym wnioskiem (a także radą na przyszłość) jest: wykorzystanie kilku algorytmów na jednym zbiorze danych jest dobrym sposobem na uzyskanie różnorodnej wiedzy.

Kolejną obserwacją, jaką wyciągnęliśmy z przeprowadzonych eksperymentów jest brak niektórych atrybutów w wynikowych regułach. Dla przykładu: w żadnej z reguł nie pojawia się problem imigrantów (mimo wielokrotnych powtórzeń testów). Nie oznacza to, że nie możemy wyciągnąć z tego żadnej wiedzy. Wręcz przeciwnie - oznacza to, że nastawienie do problemu imigrantów jest niezależne od innych problemów i opcji politycznej, do której dana osoba należy.