Indukcja reguł z użyciem algorytmu Apriori

Joanna Kołodziejczyk

1 Wprowadzenie - cel zajęć

W trakcie zajęć użyty zostanie algorytm Apriori do przeprowadzenia analizy koszykowej (ang. Market Basket Analysis). Jest to technika używana przez dużych detalistów do odkrywania związków pomiędzy kupowanymi przedmiotami. Związki wykrywa się poprzez szukanie kombinacji przedmiotów, które często występują razem w transakcjach. Uzyskujemy w ten sposób wiedzę o zachowaniach klientów.

1. Wejściem do systemu jest zbiór transakcji.
2. Wynikiem jest reguła w postaci:

$$if(przedmiot_1)and(przedmiot_2)and\ldots$$
$$and(przdmiot_n)then(przedmiot_k)$$

2 Reguły asocjacyjne

Algorytm Apriori generuje reguły asocjacyjne dla podanego zbioru danych. Reguła zakłada, że jeśli wystąpi element A, to pewnym prawdopodobieństwem występuje również element B. Przykład zbioru danych:
W tabeli 1 widzimy siedem transakcji ze sklepu spożywczego. Każda transakcja pokazuje przedmioty zakupione w tej transakcji. Możemy reprezentować przedmioty jako zestaw przedmiotów w następujący sposób:

Tablica 1: Transakcje

$t1$	$\{pomidor, ogorek, feta\}$
$t2$	$\{pomidor, szynka\}$
$t3$	$\{chleb, szynka\}$
$t4$	$\{pomidor, ogorek, szynka\}$
$t5$	$\{pomidor, ogorek, cebula, szynka, feta\}$
$t6$	$\{ogorek, cebula, feta\}$
$t7$	$\{ogorek, feta, cebula\}$

$$I = \{i_1, i_2, \ldots, i_k\} \tag{1}$$

co odpowiada:

$$I = \{pomidor, ogorek, feta, szynka, chleb, cebula\} \tag{2}$$

Transakcja jest reprezentowana przez następujące wyrażenie:

$$T = \{t_1, t_2, \ldots, t_n\} \tag{3}$$

Na przykład:

$$t_1 = \{pomidor, ogorek, feta\} \tag{4}$$

Reguła asocjacyjna jest zdefiniowana jako implikacja:

$$X \Rightarrow Y,$$

where $X \subset I, Y \subset I$ and $X \cap Y = 0$, na przykład:

$$\{pomidor, ogorek\} \Rightarrow \{feta\} \tag{5}$$

2.1 Wsparcie - (support)

Wsparcie wskazuje, jak często dany zestaw danych pojawia się w zbiorze transakcji (uczącym).

$$\text{supp}(X \Rightarrow Y) = \frac{|X \cup Y|}{n} \tag{6}$$

Innymi słowy, jest to liczba transakcji zawierających X i Y podzielona przez łączną liczbę transakcji. Reguły nie są użyteczne jeżeli wartość wsparcia jest niska. Poniżej różne przykłady wsparcia wyliczone z transakcji ze sklepu spożywczego z tabeli 1.

1. $\text{supp}(pomidor \Rightarrow ogorek) = \frac{3}{7} = 43\%$
2. $\text{supp}(ogorek \Rightarrow feta) = \frac{4}{7} = 57\%$
3. $\text{supp}(\{pomidor, ogorek\} \Rightarrow \{feta\}) = \frac{2}{7} = 28\%$

2.2 Zaufanie (confidence)

Dla reguły $X \Rightarrow Y$ zaufanie to odsetek w jakim Y jest kupowany z X. Wskazuje jak często reguła jest prawdziwa.

$$\text{conf}(X \Rightarrow Y) = \frac{\text{supp}(X \cup Y)}{\text{supp}(X)} \tag{7}$$

Na przykład, reguła $\text{supp}(pomidor \Rightarrow ogorek) = \frac{3/7}{4/7} = 3/4$ ma zaufanie na poziomie $3/4$, co oznacza, że dla 75% transakcji zawierających pomidory reguła jest prawidłowa (75% razy klient kupując pomidory, kupi również ogórek). Jeszcze trzy przykłady:

1. $\text{conf}(ogorek \Rightarrow feta) = \frac{4/7}{5/7} = \frac{4}{5}(80\%)$
2. $\text{conf}\{pomidor\} \Rightarrow \{feta\}) = \frac{2/7}{4/7} = 50\%$
3. $\text{conf}(\{pomidor, ogorek\} \Rightarrow \{feta\}) = \frac{2/7}{3/7} = 66\%$

2.3 Wzrost (lift)

Wzrost w regule jest stosunkiem obserwowanego do oczekiwanego wsparcia, jeśli X i Y były niezależne, i jest definiowane jako:

$$\text{lift}(X \Rightarrow Y) = \frac{\text{supp}(X \cup Y)}{\text{supp}(X) \cdot \text{supp}(Y)} \tag{8}$$

Większe wartości wzrostu wskazują na silniejsze kojarzenie. Oto kilka przykładów:

1. $\text{lift}(pomidor \Rightarrow ogorek) = \frac{3/7}{(4/7)(5/7)} = 1.05$
2. $\text{lift}(ogorek \Rightarrow feta) = \frac{4/7}{(5/7)(4/7)} = 1.4$
3. $\text{lift}(pomidor \Rightarrow feta) = \frac{2/7}{(4/7)(4/7)} = 0.875$
4. $\text{lift}(pomidor, ogorek \Rightarrow feta) = \frac{3/7}{(4/7)(5/7)} = 1.17$

2.4 Przekonanie (conviction)

Przekonanie o regule definiujemy jako:

$$\text{conv}(X \Rightarrow Y) = \frac{1 - \text{supp}(Y)}{1 - \text{conf}(X \Rightarrow Y)} \tag{9}$$

f
Przekonanie można interpretować jako stosunek oczekiwanej częstotliwości występowania X bez Y jeżeli X i Y były niezależne, podzielony przez obserwowaną częstość występowania błędnych prognoz (predykcji). Wysoka wartość współczynnika oznacza, że konsekwencja (then) zależy w dużym stopniu od przesłanki (if). Oto kilka przykładów:

1. $\text{conv}(pomidor \Rightarrow ogorek) = \frac{1-5/7}{1-3/4} = 1.14$
2. $\text{conv}(ogorek \Rightarrow feta) = \frac{1-4/7}{1-4/5} = 2.14$
3. $\text{conv}(pomidor \Rightarrow feta) = \frac{1-4/7}{1-1/2} = 0.86$
4. $\text{conv}(\{pomidor, ogorek\} \Rightarrow ogorek) = \frac{1-4/7}{1-2/3} = 1.28$

2.5 Algorytm

W przypadku dużych zbiorów danych, w setkach tysięcy transakcji mogą znajdować się setki pozycji. Algorytm Apriori próbuje wyodrębnić reguły dla każdej możliwej kombinacji przedmiotów. Na przykład dla przedmiot 1 i przedmiot 2, przedmiot 1 i przedmiot 3, przedmiot 1 i przedmiot 4, a następnie przedmiot 2 i przedmiot 3, przedmiot 2 i przedmiot 4, a następnie kombinacji przedmiotów np. przedmiot 1, przedmiot 2 i przedmiot 3; podobnie przedmiot 1, przedmiot 2 i przedmiot 4 itd.
Jak widać z powyższego przykładu, proces ten może być bardzo powolny ze względu na liczbę kombinacji. Aby przyspieszyć ten proces, musimy wykonać następujące kroki:

1. Ustalić minimalną wartość wsparcia i zaufania. Oznacza to, że interesujące jest tylko znalezienie reguł dla przedmiotów, które mają pewne domyślny poziom istnienia (wsparcie) i mają minimalną wartość dla współwystępowania z innymi przedmiotami (zaufanie).

2. Wyodrębnić wszystkie podzbiory, które mają wyższą wartość wsparcia niż minimalny próg.

3. Wybrać wszystkie reguły z podzbiorów o wartości zaufania wyższej niż minimalny próg.

4. Uporządkować reguły w kolejności malejącej po parametrze Lift.

3 Zadanie

Do zadania wykorzystany zostanie język R i środowisko R-studio. Konieczne jest załadowanie bibliotek.

Listing 1: Konieczne biblioteki

```
library(arules)
library(arulesViz)
library(tidyverse)
library(gridExtra)
```

3.1 Wczytanie zbioru treningowego

Do zadania należy wykorzystać zbiór store_data.csv udostępniony na stronie `wikizmsi.zut.edu.pl`. Należy wczytać do R-studio plik i przyjrzeć się jego strukturze. Wczytanie danych należy wykonać z pomocą metody `read.transactions()` z pakietu `arules`:

Listing 2: Wczytanie danych

```
# czytanie danych do zmiennej trans
trans <- read.transactions("Store_data.csv", format="basket", sep=",", rm.
    duplicates=TRUE)
```

Do określenia cech wczytanego zbioru sprawdź następujące polecenia:

Listing 3: Zapoznanie się z danymi

```
# Wyświetlenie informacji o obiekcie typu trans
trans
# Podsumowanie
summary(trans))
```

3.2 Analiza zbioru treningowego

Przed zastosowaniem algorytmu Apriori na zbiorze uczącym, wykonane zostanie kilka wizualizacji, aby dowiedzieć się czegoś więcej o transakcjach. Na przykład, możemy wygenerować histogramy, które pozwolą prezentują dystrybucję produktów.

Listing 4: Zapoznanie się z danymi

```
# Bezwzględna częstotliwość transakcji licznik
itemFrequencyPlot(trans, topN=15, type="absolute", col="lightgreen",xlab="
    Przedmiot",ylab="Częstosliwość (bezwzględna)", main="Histogram czę
    stotliwości zakupów")

# Względna częstotliwość transakcji procent wszystkich transakcji
itemFrequencyPlot(trans, topN=15, type="relative", col="lightcyan", xlab="
    Przedmiot", ylab="Częstosliwość (względna)", main="Względna Histogram
    częstotliwości zakupów")
```

`itemFrequencyPlot()` pozwala na pokazanie wartości bezwzględnych lub względnych. Jeśli jest to wartość bezwzględna, wykreśla się częstotliwości numeryczne każdego elementu niezależnie. Jeśli względne, wykreślone zostanie ile razy elementy pojawiły się w porównaniu do innych.

3.3 Algorytm Apriori

3.3.1 Wybór wsparcia i zaufania

Pierwszym krokiem do uzyskania zbioru reguł asocjacyjnych jest określenie optymalnych progów wsparcia i zaufania. Jeśli ustawimy te wartości zbyt nisko, wówczas wykonanie algorytmu potrwa dłużej i powstanie wiele reguł (większość z nich nie będzie użyteczna). W takim razie, jakie wartości wybrać? Można wypróbować różne wartości wsparcia i zaufania i ocenić je graficznie.

Listing 5: Wybór poziomów wsparcia i zaufania

```
# Wsparcie i zaufanie wybór wartości do testowania
supportLevels <- c(0.1, 0.05, 0.01, 0.005)
confidenceLevels <- c(0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1)

# Wektory dla różnych poziomów wsparcia i 9 wartości zaufania
rules_sup10 <- integer(length=9)
rules_sup5 <- integer(length=9)
rules_sup1 <- integer(length=9)
rules_sup0.5 <- integer(length=9)

# Apriori z poziomem wsparcia 10%
```

```
for (i in 1:length(confidenceLevels)) {
    rules_sup10[i] <- length(apriori(trans,
    parameter=list(sup=supportLevels[1],
    conf=confidenceLevels[i],
    target="rules")))
}

# Apriori z poziomem wsparcia 5%
for (i in 1:length(confidenceLevels)){
    rules_sup5[i] <- length(apriori(trans,
    parameter=list(sup=supportLevels[2],
    conf=confidenceLevels[i], target="rules")))
}

# Apriori z poziomem wsparcia 1%
for (i in 1:length(confidenceLevels)){
    rules_sup1[i] <- length(apriori(trans,
    parameter=list(sup=supportLevels[3],
    conf=confidenceLevels[i], target="rules")))
}

# Apriori z poziomem wsparcia 0.5%
for (i in 1:length(confidenceLevels)){
    rules_sup0.5[i] <- length(apriori(trans,
    parameter=list(sup=supportLevels[4],
    conf=confidenceLevels[i], target="rules")))
}
```

Na wykresach zwizualizowany zostanie liczba generowanych reguł z poziomem wsparcia 10%, 5%, 1% i 0,5%.

Listing 6: Wykresy dla różnych poziomów wsparcia

```
# Liczba znalezionych reguł z poziomem wsparcia wynoszącym 10%
plot1 <- qplot(confidenceLevels, rules_sup10, geom=c("point", "line"),
               xlab="Zaufanie", ylab="Liczba znalezionych reguł",
               main="Apriori z poziomem wsparcia 10%") + theme_bw()

# Liczba znalezionych reguł z poziomem wsparcia wynoszącym 5%
plot2 <- qplot(confidenceLevels, rules_sup5, geom=c("point", "line"),
               xlab="Zaufanie", ylab="Liczba znalezionych reguł",
               main="Apriori z poziomem wsparcia 5%") + theme_bw()

# Liczba znalezionych reguł z poziomem wsparcia wynoszącym 1%
plot3 <- qplot(confidenceLevels, rules_sup1, geom=c("point", "line"),
               xlab="Zaufanie", ylab="Liczba znalezionych reguł",
               main="Apriori z poziomem wsparcia 1%") + + theme_bw()

# Liczba znalezionych reguł z poziomem wsparcia wynoszącym 0.5%
plot4 <- qplot(confidenceLevels, rules_sup0.5, geom=c("point", "line"),
               xlab="Zaufanie", ylab="Liczba znalezionych reguł",
               main="Apriori z poziomem wsparcia 0.5%") + + theme_bw()

# Wykres zbiorczy
grid.arrange(plot1, plot2, plot3, plot4, ncol=2)
```

3.3.2 Analiza wyników i reguł dla wybranych parametrów

Należy popatrzeć na liczbę reguł w każdym poziomie wsparcia 10%, 5%, 1%, 0,5%. Dokonać analizy ile reguł i na jakim poziomie zaufania otrzymuje się liczbę sensowną reguł interesujących. Na tej podstawie zdecydować jakie parametry są obiecujące i wygenerować reguły algorytmem Apriori. Następnie wyświetlić reguły.

Listing 7: Przykład uruchomienia algorytmu celem analizy reguł.

```
# Apriori uruchomione dla wybranego poziomu wsparcia i zaufania
mysupp =
myconf =
rules <- apriori(trans, parameter=list(sup=mysupp,
                                       conf=myconf, target="rules"))

# Pokazanie reguł asocjacyjnych
inspect(rules)
inspectDT(rules)
```

3.3.3 Wizualizacje reguł

Wykorzystać pakiet `arulesViz` do stworzenia wizualizacji. Zacznijmy od prostego wykresu punktowego z różnymi miarami na osiach (lift i support) oraz trzecią miarą (zaufanie) reprezentowane przez kolor.

Listing 8: Różne wizualizacje

```
# Scatter plot
plot(rules, measure=c("support","lift"), shading="confidence")

# Graf dla reguł
plot(rules, method="graph")

# Graf dla reguł uporządkowany
plot(rules, method="graph", control=list(layout=igraph::in_circle()))

# Pogrupowane reguły w postaci tablicowej
plot(rules, method="grouped")
```

4 Sprawozdanie

Wykonać sprawozdanie zawierające:

1. Obliczenia czterech miar (supp, conf, lift i conv) dla reguły $\{feta, ogorek\} \Rightarrow \{pomidor\}$ używając transakcji w sklepie spożywczym (tabela 1 niniejszej instrukcji).

2. Pobrać zbiór transakcji BreadBasket_DMS.csv. Przekształcić na format basket.

3. Opisać zbiór danych BreadBasket_DMS.csv. (podać liczbę transakcji, najliczniejsze elementy itp., itd.)

4. Na podstawie badania z Listingu 5 wybrać zdroworozsądkowo poziom ufności i wsparcia do kolejnego zadania.

5. Podać jeden zestaw reguł wygenerowanych algorytmem Apriori przy poziomach ocenionych w zadaniu w pkt poprzednim oraz ich słowną interpretację. Trzeba wybrać taki, który wydaje się być ciekawy i łatwy do zastosowania (reguły są czytelne dla człowieka).

6. Wybrać arbitralnie poziomem wsparcia i zaufania, tak by otrzymać większą liczbę reguł niż w poprzednim punkcie (zadanie z Listing 7). Wykonać wizualizację reguł i interpretację słowną.

7. Jakie są konsekwencje obniżenia poziomu wsparcia (supp)? Dlaczego?

8. Jakie są konsekwencje podwyższenia poziom zaufania (conf)? Dlaczego?

Sprawozdanie należy podpiąć na Teams. Sprawozdanie może być przygotowane w formie notatki w R-studio (format .Rmd)
Czas wykonania zadania: tydzień.