

POLITECHNIKA ŁÓDZKA

WYDZIAŁ FIZYKI TECHNICZNEJ, INFORMATYKI
I MATEMATYKI STOSOWANEJ

Kierunek: Matematyka Stosowana

Specjalność: Analiza Danych w Biznesie i Logistyce

Matematyczne modele wykorzystywane w systemach rekomendacji.

Anita Kudaj

Nr albumu: 220020

Praca magisterska

napisana w Instytucie Matematyki Politechniki Łódzkiej

Promotor: dr, mgr inż. Piotr Kowalski

ŁÓDŹ, ?????

Spis treści

1	Wstęp	3
2	Preliminaria	4
2.1	Elementy rachunku prawdopodobieństwa i statystyki	4
2.2	Elementy algebry liniowej	5
3	Eksploracja danych w systemach rekomendujących	9
3.1	Preprocesing danych	9
3.1.1	Miary podobieństwa	9
3.1.2	Redukcja wymiaru	11
3.2	Metody eksploracji danych	16
3.2.1	k - Najbliższych Sąsiadów	17
3.2.2	k-Średnich	18
3.2.3	drzewa decyzyjne-?	20
3.2.4	regresja wielowymiarowa - logistyczna - ?	20
3.2.5	Sztuczna Sieć Neuronowa	20
3.2.6	Maszyna Wektorów Nośnych	21
3.3	Szacowanie Błędów Obliczeń	24
3.3.1	Ocena Dokładności Metody	24
3.3.2	Ocena Precyzji Użyteczności	25
3.3.3	Ocena Rankingów	26
4	Modele tworzenia rekomendacji	28
4.1	Systemy rekomendujące oparte na treści (Content-based recommender systems):	29
4.1.1	Algorytm	29
4.1.2	Wygenerowanie profilu produktu	31
4.1.3	Wygenerowanie profilu użytkownika	33
4.1.4	Rozpoznanie cech produktu odpowiednich dla użytkownika	33
4.2	Filtrowanie kolaboratywne (Collaborative filtering)	33

4.2.1	Przykład	34
4.2.2	Filtrowanie kolaboratywne oparte na użytkowniku	34
4.2.3	Filtrowanie kolaboratywne oparte na elementach	36
4.3	Systemy rekomendujące kontekstowe (Context – aware recommender systems):	38
4.3.1	Algorytm	38
4.4	Hybrydowe systemy rekomendujące ():	39
4.5	Systemy rekomendujące oparte na modelach():	39
5	Eksperymenty / część praktyczne	40
6	Podsumowanie	41

Rozdział 1

Wstęp

Rozdział 2

Preliminaria

2.1 Elementy rachunku prawdopodobieństwa i statystyki

Niech F oznacza σ - algebrę podzbiorów z przestrzeni Ω oraz niech X oznacza funkcję rzeczywistą określoną na przestrzeni Ω , to znaczy:

$$X : \Omega \longrightarrow \mathbb{R}.$$

Definicja 1 Zmienna losowa

Zmienną losową nazywamy funkcję X , która jest F - mierzalna, to znaczy jeżeli dla każdego $a \in \mathbb{R}$ zachodzi:

$$\{\omega : X(\omega) < a\} = X^{-1}((-\infty, a)) \in F,$$

gdzie X^{-1} jest operacją przeciwobrazu zbioru przez funkcję X .

Definicja 2 Kowariancja

Kowariancją zmiennych losowych X, Y nazywamy liczbę:

$$\text{Cov}(X, Y) = E[(X - E(X))(Y - E(Y))],$$

gdzie $E(X)$ oznacza wartość oczekiwaną zmiennej losowej X .

Definicja 3 Odchylenie standardowe

Wariancją zmiennej losowej X nazywamy liczbę:

$$\text{Var}(X) = E[(X - E(X))^2],$$

jeżeli po prawej stronie równości wartość oczekiwana istnieje.

Odchyleniem standardowym zmiennej losowej X nazywamy liczbę:

$$\sigma(X) = \sqrt{\text{Var}(X)}.$$

Definicja 4 **Współczynnik korelacja**

Współczynnikiem korelacji nazywamy charakterystykę ilościową stopnia zależności dwóch zmiennych losowych X i Y zdefiniowaną następująco:

$$\rho(x, Y) = \frac{\text{Cov}(X, Y)}{\sigma(X)\sigma(Y)}.$$

W statystyce do przedstawienia poziomu zależności liniowej między dwoma zmiennymi losowymi używamy Współczynnika Korelacji Pearsona.

Definicja 5 **Współczynnik Korelacji Pearsona**

Niech $X, Y \in \mathbb{R}^n$ będą zmiennymi losowymi o rozkładach ciągłych oraz niech x_k, y_k , gdzie $k \in \{1, \dots, n\}$ oznaczają wartości prób losowych tych zmiennych.

Przez \bar{x} i \bar{y} oznaczmy:

$$\begin{aligned}\bar{x} &= \frac{1}{n} \sum_{k=1}^n x_k, \\ \bar{y} &= \frac{1}{n} \sum_{k=1}^n y_k.\end{aligned}$$

Wówczas Współczynnikiem Korelacji Pearsona nazywamy:

$$\rho(X, Y) = \frac{\sum_{k=1}^n (x_k - \bar{x})(y_k - \bar{y})}{\sqrt{\sum_{k=1}^n (x_k - \bar{x})^2} \sqrt{\sum_{k=1}^n (y_k - \bar{y})^2}}.$$

2.2 Elementy algebry liniowej

Definicja 6 **Iloczyn skalarny**

Niech U oznacza przestrzeń liniową nad ciałem \mathbb{R} . Iloczynem skalarnym nazywamy formę dwuliniową

$$d : U \times U \longrightarrow \mathbb{R},$$

gdy:

- dla każdego $u \in U$ zachodzi:

$$d(u, u) \geq 0$$

- jest symetryczna, oznacza to, że dla dowolnych $u, v \in U$ zachodzi:

$$d(u, v) = d(v, u)$$

- $(u, u) = 0 \Leftrightarrow u = \Theta_U$

Przestrzeń liniową U nad ciałem liczb rzeczywistych z iloczynem skalarnym $d : U \times U \longrightarrow \mathbb{R}$ nazywamy przestrzenią euklidesową.

Definicja 7 Miara odległości

Miarą odległości (funkcją odległości) nazywamy rzeczywistą funkcję d , która dla każdego $x, y, z \in \mathbb{R}^n$ spełnia warunki:

- $d(x, y) \geq 0$
- $d(x, y) = 0 \Leftrightarrow x = y$
- $d(x, y) = d(y, x)$
- $d(x, y) + d(y, z) \geq d(x, z)$

Definicja 8 Odległość euklidesowa

Niech $(x_1, x_2, \dots, x_n) \in \mathbb{R}^n$. Normą x nazywamy:

$$||x|| = \sqrt{x_1^2 + \dots + x_n^2}.$$

Jeśli $x, y \in \mathbb{R}^n$ to liczbę: $||x - y||$ nazywamy odległością euklidesową punktów x i y , gdzie:

$$(x - y) = (x_1 - y_1, \dots, x_n - y_n).$$

Definicja 9 Odległość kosinusowa

Niech $x, y \in \mathbb{R}^n$ (x, y są n -wymiarowymi wektorami). Odległością kosinusową nazywamy:

$$d(x, y) = 1 - \text{sim}(x, y),$$

gdzie $\text{sim}(x, y)$ to współczynnik podobieństwa wektorów x i y :

$$\text{sim}(x, y) = \frac{x \cdot y}{||x|| ||y||},$$

zatem

$$\text{sim}(x, y) = \frac{\sum_{k=1}^n x_k y_k}{\sqrt{\sum_{k=1}^n x_k^2} \sqrt{\sum_{k=1}^n y_k^2}}.$$

Definicja 10 Macierz transponowana

Niech dana będzie macierz $A \in M_{m,n}(K)$ o n kolumnach i m wierszach:

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}.$$

Macierzą transponowaną nazywamy macierz $A^T \in M_{n,m}(K)$:

$$A^T = \begin{bmatrix} a_{11} & a_{21} & \cdots & a_{m1} \\ a_{12} & a_{22} & \cdots & a_{m2} \\ \vdots & \vdots & \ddots & \vdots \\ a_{1n} & a_{2n} & \cdots & a_{mn} \end{bmatrix}.$$

Twierdzenie 1 Własności transpozycji macierzy

Niech A i B będą macierzami o współczynnikach z K oraz niech posiadają tyle kolumn i wierszy, że operacje występujące powyżej są określone. Niech $\lambda \in K$.

Zachodzą następujące równości:

- $(A^T)^T = A$,
- $(A + B)^T = A^T + B^T$,
- $(\lambda A)^T = \lambda A^T$,
- $(AB)^T = B^T A^T$.

Definicja 11 Macierz ortogonalna

Macierzą ortogonalną nazywamy macierz kwadratową $A \in M_{n,n}(\mathbb{R})$ spełniającą nierówność:

$$A^T \cdot A = A \cdot A^T = I_n,$$

gdzie A^T oznacza macierz transponowaną względem A , natomiast I_n oznacza macierz jednostkową.

Definicja 12 Macierz diagonalna

Macierzą diagonalną nazywamy macierz $A \in M_{n,n}(K)$ taką, że:

$$A = (a_{ij}) \text{ dla } i, j = 1, 2, \dots, n$$

oraz

$$a_{ij} = 0 \text{ gdy } i \neq j.$$

Definicja 13 Macierz nieosobliwa

Macierzą nieosobliwą nazywamy macierz kwadratową $A \in M_{n,n}(K)$, jeżeli istnieje macierz $B \in M_{n,n}(K)$ taka, że:

$$A \cdot B = B \cdot A = I_n,$$

gdzie I_n jest macierzą jednostkową.

Definicja 14 Macierz osobliwa

Macierzą osobliwą nazywamy macierz kwadratową $A \in M_{n,n}(K)$, której wyznacznik jest równy zero.

Definicja 15 Rozkład wartości osobliwych

Rozkładem wartości osobliwych nad \mathbb{R} nazywamy przedstawienie macierzy A w postaci:

$$A = U \Sigma V^T,$$

gdzie U i V są macierzami ortogonalnymi. Natomiast Σ jest macierzą diagonalną, która zawiera kolejne wartości osobliwe.

Definicja 16 Macierz odwrotna**Definicja 17 Ślad macierzy**

Śladem macierzy $A \in M_{n,n}(K)$ nazywamy wielkość:

$$\text{tr}(A) = \sum_{i=1}^n a_{ii} = a_{11} + a_{22} + \cdots + a_{nn}.$$

Definicja 18 Rząd macierzy

Rzędem macierzy nazywamy liczbę wiodących jedynek w dowolnej postaci zredukowanej macierzy. Rząd macierzy A oznaczamy symbolem $\text{rz}A$ lub $R(A)$.

Rząd macierzy wyznacza maksymalną liczbę wektorów niezależnych, które są kolumnami lub wierszami tej macierzy.

Definicja 19 Zakres macierzy**Definicja 20 Jądro macierzy****Definicja 21 Przestrzeń generowana przez układ wektorów**

Przestrzenią generowaną przez układ wektorów X nazywamy zbiór wszystkich kombinacji liniowych skończonych podukładów X .

Oznaczamy ją symbolem $\text{span}(X)$ i definiujemy ją następująco:

$$\text{span}(X) := \left\{ \sum_{i=1}^n a_i x_i : (\forall i \in [1;n] \cap \mathbb{N}) a_i \in \mathbb{K} \wedge (\forall i \in [1;n] \cap \mathbb{N}) x_i \in X \wedge n \in \mathbb{N} \right\},$$

gdzie $X = (x_i)_{i=1}^n$ jest układem wektorów przestrzeni wektorowej V nad ciałem \mathbb{K} .

Rozdział 3

Eksploracja danych w systemach rekomendujących

Większość systemów rekomendujących opiera swój rdzeń na algorytmach, które możemy rozumieć jako konkretne przypadki technik eksploracji danych.

Proces eksploracji danych składa się zazwyczaj z trzech kroków:

1. Preprocessing Danych,
2. Analiza Danych,
3. Interpretacja Wyników.

W tym rozdziale zostaną przeanalizowane najważniejsze i najczęściej używane w regułach rekomendujących metody. Zaczniemy od miar podobieństw i redukcji wymiaru. W kolejnym etapie spojrzymy na metody klasyfikacji, grupowania i regresji, aby zakończyć interpretacją wyników i oceną błędów obliczeń.

3.1 Preprocessing danych

Przed przystąpieniem do kroku analizy dane wymagają przygotowania: wyczyszczenia, przefiltrowania, transformacji. Dopiero tak przygotowane dane mogą zostać poddane zadaniom uczenia maszynowego. W tej sekcji zostaną przedstawione problemy, które spotkamy przy tworzeniu reguł rekomendujących.

3.1.1 Miary podobieństwa

W systemach rekomendujących, jak filtrowanie kolaboratywne bardzo częstym podejściem jest używanie metod klasyfikacji i grupowania. Metody te bardzo ściśle opierają się na obliczaniu podobieństwach i odległości.

Najprostszym i jednocześnie najczęściej używanym podejściem jest tu **Odległość Euklidesowa**:

$$d(x, y) = \sqrt{\sum_{k=1}^n (x_k - y_k)^2},$$

gdzie n oznacza liczbę atrybutów elementów x i y przy czym x_k oznacza k -ty atrybut elementu x .

Innym przykładem jest **Odległość Minkowskiego**, która jest uogólnioną wersją Odległości Euklidesowej:

$$d(x, y) = (\sum_{k=1}^n |x_k - y_k|^r)^{\frac{1}{r}}.$$

W zależności od wartości stopnia odległości r Odległość Minkowskiego przyjmuje konkretne nazwy:

- $r = 1$ - Odległość Manhatan (norma L_1),
- $r = 2$ - wspomniana wcześniej Odległość Euklidesowa,
- $r \rightarrow \infty$ - supremum (norma L_{max} , norma L_∞).

Kolejnym podejściem, gdzie poszczególne elementy są postrzegane jako n - wymiarowe wektory, a podobieństwo między nimi jest obliczane na podstawie kątów, który tworzą jest **podobieństwo kosinusów**:

$$\cos(x_u, x_v) = \frac{x_u^T x_v}{\|x_u\| \|x_v\|},$$

gdzie x_u i x_v oznaczają wektory preferencji użytkowników u oraz v .

Inną miarą jest **Korelacja Pearsona**, zdefiniowana następująco:

$$\rho(u, v) = \frac{\sum_{i \in I_{uv}} (r_{ui} - \bar{r}_u)(r_{vi} - \bar{r}_v)}{\sqrt{\sum_{i \in I_{uv}} (r_{ui} - \bar{r}_u)^2 \sum_{i \in I_{uv}} (r_{vi} - \bar{r}_v)^2}}$$

I_{uv} oznacza w tym przypadku zbiór elementów, które zostały ocenione przez użytkownika u i użytkownika v , r_{ui} natomiast ocenę elementu i przez użytkownika u .

Indeks Jaccarda (Współczynnik podobieństwa Jaccarda) to kolejny wskaźnik opisujący podobieństwo. Jeżeli oznaczymy przez A i B wektory to:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}.$$

Z racji tego, że użytkowników i elementy możemy przedstawić za pomocą wektorów łatwo zastosować współczynnik Jaccarda do obliczania podobieństwa.

3.1.2 Redukcja wymiaru

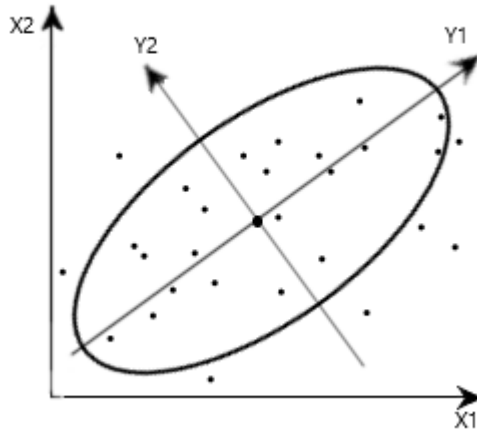
Wraz ze wzrostem ilości obserwacji rośnie ich dokładność. Warto zauważyć, że tym samym rośnie stopień komplikacji w interpretacji otrzymanych wyników. Zbyt duża ilość zmiennych, które opisują obserwacje powoduje wzrost prawdopodobieństwa, że zmienne te są ze sobą skorelowane, a informacje wnoszone przez część zmiennych są redundantne. Podobne zjawiska możemy dostrzec także w regułach rekomendujących. Proces redukcji wymiaru pozwala przezwyciężyć ten problem poprzez transformację przestrzeni danych do przestrzeni o mniejszej liczbie wymiarów. W poniższym rozdziale przyjrzymy się dwóm najczęściej wybieranym algorytmom redukcji wymiarów w kontekście reguł rekomendujących. Są to Analiza Głównych Składowych oraz Rozkład Według Wartości Osobliwych.

Analiza Głównych Składowych

Definicja 22 *Analiza Głównych Składowych (ang. **Principal Component Analysis (PCA)**)*

Analizę głównych składowych nazywamy procedurę statystyczną, która polega na ortogonalnej transformacji układu badanych zmiennych X w zbiór nowych zmiennych Y . W rzeczywistość zmienne Y są kombinacją liniową zmiennych X .

Niech dane będą dwie zmienne X_1, X_2 oraz n pomiarów, które oznaczmy (X_{i1}, X_{i2}) dla $i \in 1, 2, 3, \dots, n$. Pomiarów przedstawmy na układzie współrzędnych w formie diagramu korelacyjnego. Możemy zauważyć, że wzdłuż jednej osi dane są bardziej rozproszone - jest to pierwsza główna składowa Y_1 . Druga biegnąca pod kątem prostym do pierwszej oś, wyznacza kierunek drugiej składowej - Y_2 . Osie X_1, X_2 są transformowane przez przesunięcie środka do nowego punktu $(\overline{X_1}, \overline{X_2})$, a następnie obrócone w ten sposób, by otrzymać współrzędne Y_1 i Y_2 głównych składowych.



Rysunek 3.1: Diagram korelacyjny wyodrębnienia głównych składowych.

Punktem wyjścia rozważanego algorytmu jest macierz kowariancji lub macierz korelacji utworzone ze zbioru wejściowego. Zawierają one informację niezbędną do wyznaczenia głównych składowych.

W przypadku użycia macierzy kowariancji największy wpływ na wynik przeprowadzanego algorytmu mają zmienne o największej wariancji. Stąd też rozwiązanie to świetnie sprawdzi się w przypadku, gdy rozważane zmienne mają porównywalne wielkości:

$$A = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1n} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{n1} & \sigma_{n2} & \cdots & \sigma_{nn} \end{bmatrix},$$

gdzie:

- σ_{ii} - wariancja zmiennej X_i ,
- $\sigma_{ij} = Cov(X_i, X_j)$ - kowariancja między zmiennymi X_i i X_j .

W przypadku, gdy zmienne różnią się znacznie w wyrażonych jednostkach, lub gdy nie są proporcjonalne zastosujemy macierz korelacji:

$$B = \begin{bmatrix} 1 & \rho_{12} & \cdots & \rho_{1n} \\ \rho_{21} & 1 & \cdots & \rho_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{n1} & \rho_{n2} & \cdots & 1 \end{bmatrix},$$

gdzie:

- $\rho_{ij} = \frac{\text{Cov}(X_i, X_j)}{\sigma_i \sigma_j}$ - współczynnik korelacji X_i i X_j .

Własności głównych składowych:

- są kombinacją liniową pierwotnych zmiennych,
- względem siebie są ortogonalne,
- suma wariancji pierwotnych zmiennych jest równa sumie wariancji zmiennych składowych.

Dla wspomnianej transformacji ortogonalnej, w przypadku gdy znamy wektory własne macierzy istnieje bezpośrednie równanie.

Twierdzenie 2 Niech W będzie macierzą wektorów własnych przekształcenia XX^T uporządkowaną malejąco. Wtedy transformacja PCA jest dana wzorem $Y = W^T X$.

Dowód 1 Bez utraty ogólności założmy, że X ma średnią zero.

Niech $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$ będą wartościami własnymi przekształcenia XX^T oraz v_i odpowiadającą bazą ortogonalną. Założmy, że $\omega = \sum_{i=1}^n a_i v_i$ maksymalizuje wariancję oraz $\omega \neq v_i$.

Wtedy:

$$\begin{aligned}\omega_1 &= \operatorname{argmax}_{\|\omega\|=1} \omega^T X X^T \omega = \operatorname{argmax}_{\|\omega\|=1} (\sum a_i v_i)^T X X^T (\sum a_i v_i) \\ &= \operatorname{argmax}_{\|\omega\|=1} (\sum a_i v_i)^T (\sum a_i \lambda_i v_i) = \operatorname{argmax}_{\|\omega\|=1} (\sum a_i)^2.\end{aligned}$$

Stąd ω_1 maksymalizuje wariancję wtedy i tylko wtedy, gdy $\omega_1 = v_1$.

Znajdąc $k-1$ głównych składowych k -tą znajdziemy postępując z transformacją:

$$X_{k-1} = X - \sum_{i=1}^{k-1} \omega_i \omega_i^T X.$$

Podstawiając X_{k-1} otrzymujemy:

$$\omega_k = \operatorname{argmax}_{\|\omega\|=1} \omega^T X_{k-1} X_{k-1}^T \omega.$$

□

Twierdzenie przedstawia użyteczną interpretację algorytmu PCA, jednakże nie jest wydajną metodą przy odnajdowaniu wektorów własnych. Poniżej zostanie przedstawiony proces odnalezienia transformacji.

Założmy, że średnia X to zero. Naszym celem jest znaleźć transformację ortogonalną W taką, że WX jest losowym wektorem z parami niezależnych komponentów:

$$\text{cov}(WX) = \mathbb{E}[WX(WX)^T] = \mathbb{E}[WXX^TW^T] = W\mathbb{E}[XX^T]W^T = W\text{cov}(X)W^{(-1)}$$

Dopóki $\text{cov}(X)$ jest macierzą o wartościach dodatnich istnieje W , które spełnia powyższe równanie.

PCA jest klasyczną metodą, która pozwala odnaleźć zależności w zbiorach danych o wielu wymiarach. W wyniku tej metody otrzymujemy uporządkowaną listę cech wyznaczoną dla największych wartości wariancji przy jednocześnie najmniejszym błędzie kwadratowym (podrozdział 3.3.1). W ten sposób spośród wielu cech opisujących nasze elementy jesteśmy w stanie wybrać te najbardziej dla nas istotne w procesie rekomendacji.

Rozkład Według Wartości Osobliwych (ang. Singular Value Decomposition (SVD))

Definicja 23 *Rozkład Według Wartości Osobliwych*

Rozkładem według wartości osobliwych $m \times n$ - wymiarowej macierzy \mathbb{X} , gdzie $m \geq n$ oraz $r \in \mathbb{N}$ jest rzędem macierzy A nazywamy rozkład:

$$\mathbb{X} = \mathbb{U} \Sigma \mathbb{V}^T,$$

gdzie:

- \mathbb{U} jest macierzą ortogonalną $m \times m$ - wymiarową,
- Σ jest macierzą diagonalną, $m \times n$ - wymiarową o nieujemnych wartościach, $\Sigma = \text{diag}(d_1, d_2, \dots, d_n)$, $n \in \mathbb{N}$ taką, że $d_i > 0$ dla $1 \leq i \leq r$ i $d_i = 0$ dla $i \geq r + 1$,
- \mathbb{V} są macierzą ortogonalną $n \times n$ - wymiarową.

Definicja 24 *Norma Frobeniusa*

Normą Frobeniusa nazywamy:

$$\|A\|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2} = \sqrt{\text{tr} A^T A},$$

gdzie A jest macierzą $m \times n$ - wymiarową. A^T jest sprzężeniem macierzy, a $\text{tr}(A)$ śladem macierzy A .

Twierdzenie 3 *Następstwa SVN*

Niech rozkład według wartości osobliwych macierzy A będzie dany wzorem

$$\mathbb{X} = \mathbb{U} \Sigma \mathbb{V}^T$$

gdzie $U = [u_1, u_2, \dots, u_m]$, $V = [v_1, v_2, \dots, v_n]$ oraz

$$d_1 \geq d_2 \geq \dots \geq d_r > d_{r+1} = \dots = d_n = 0.$$

$R(A)$ i $N(A)$ oznaczają zakres i jądro macierzy.

Wtedy:

1. właściwości rzędu macierzy: $\text{rank}(A) = r$, $N(A) = \text{span}\{v_{r+1}, \dots, v_n\}$, $R(A) = \text{span}\{u_1, u_2, \dots, u_r\}$,
2. $A = \sum_{i=1}^r u_i \cdot d_i \cdot v_i^T$,
3. $\|A\|_F^2 = d_1^2 + \dots + d_r^2$ i $\|A\|_2^2 = d_1^2$.

Twierdzenie 4 Twierdzenie Eckart - Younga

Niech X będzie macierzą $m \times n$ - wymiarową z rozkładem według wartości osobliwych $\mathbb{X} = U \Sigma V^T$, $r \in \mathbb{N} = \text{rank}(A)$ niech będzie rzędem macierzy i $r \leq p = \min(m, n)$.

Zdefiniujmy:

$$A_k = \sum_{i=1}^k u_i \cdot d_i \cdot v_i^T,$$

wtedy

$$\min_{\text{rank}(B)=k} \|A - B\|_F^2 = \|A - A_k\|_F^2 = d_{k+1}^2 + \dots + d_p^2.$$

Dowód 2 Niech $A \in \mathbb{R}^{m \times n}$ będzie macierzą o wartościach rzeczywistych, gdzie $m \geq n$.

Załóżmy, że

$$\mathbb{X} = U \Sigma V^T$$

jest rozkładem według wartości osobliwych macierzy A .

Chcemy pokazać, że najlepszym przybliżeniem macierzy A w normie Frobeniusa (oznaczamy $\|\cdot\|_F$) jest

$$A_k = \sum_{i=1}^k u_i \cdot d_i \cdot v_i^T,$$

gdzie u_i i v_i oznaczają odpowiednio i -te kolumny macierzy U i V .

Zauważmy, że

$$\|A - A_k\|_F^2 = \|\sum_{i=k+1}^n u_i \cdot d_i \cdot v_i^T\|_F^2 = \sum_{i=k+1}^n d_i^2.$$

Stąd należy udowodnić, że $B_k = XY^T$, gdzie X i Y są macierzami oraz

$$\|A - A_k\|_F^2 = \sum_{i=k+1}^n d_i^2 = \|A - B_k\|_F^2.$$

Z nierówności trójkąta, jeżeli $A = A' + A''$ wtedy $d_1(A) \leq d_1(A') + d_1(A'')$.

Przez A'_k i A''_k oznaczmy przybliżenia SVD odpowiednio macierzy A' i A'' .

Stąd dla każdego $i, j \geq 1$

$$d_i(A') + d_j(A'') = d_1(A' - A'_{i-1}) + d_1(A'' - A''_{j-1}) \geq d_1(A - A'_{i-1} - A''_{j-1}) \geq d_1(A - A_{i+j-2})(\text{rank}(A'_{i-1} + A''_{j-1})) \geq d_{i+j-1}(A),$$

gdy $\text{rank}(A'_{i-1} + A''_{j-1}) \leq \text{rank}(A_{i+j-2})$.

Jeżeli

$$d_{k+1}(B_k) = 0,$$

kiedy $A' = A - B_k$ i $A'' = B_k$ wnioskujemy, że dla $i \geq 1, j = k + 1$

$$d_i(A - B_k) \geq d_{k+1}(A).$$

Stąd:

$$\|A - B_k\|_F^2 = \sum_{i=1}^n d_i(A - B_k)^2 \geq \sum_{k+1}^n d_i(A)^2 = \|A - A_k\|_F^2.$$

□

Zawsze jest możliwe dokonać dekompozycji macierzy A do postaci $\mathbb{X} = \mathbb{U} \Sigma \mathbb{V}^T$.

Zakładamy, że naszą macierzą wejściową jest macierz danych A $m \times n$ -wymiarowa odzwierciedlająca m elementów oraz n cech dla każdego z nich. Po przeprowadzeniu rozkładu według wartości osobliwych otrzymamy:

- macierz U o wymiarach $m \times r$ prezentującą m elementów i r kategorii,
- macierz diagonalną Σ o wymiarach $r \times r$ prezentującą istotność poszczególnych kategorii,
- macierz V o wymiarach $n \times r$ prezentującą n cech i r kategorii.

Macierz Σ zawiera pojedyncze wartości uporządkowane zawsze w porządku malejącym. Macierz U jest tu interpretowana jako macierz podobieństwa elementów i kategorii, macierz V natomiast jako macierz podobieństwa cech i kategorii.

SVD jest powszechnie używane w celu odkrycia relacji występujących między użytkownikami i produktami.

3.2 Metody eksploracji danych

Termin eksploracja danych jest często używany jako określenie procesu odkrywania wiedzy z danych. Coraz częściej jednak terminem "proces odkrywania wiedzy" określamy cały proces pracy z danymi, natomiast termin "eksploracja danych" odnosi się do etapu odkrywania pewnego rodzaju reguł.

Jak podaje w jednym ze swoich artykułów "Eksploracja danych: problemy i rozwiązania" Tadeusz Morzy metody eksploracji można podzielić na sześć klas:

- Odkrywanie asocjacji,
- Klastrowanie,
- Odkrywanie wzorców sekwencji,
- Odkrywanie klasyfikacji,
- Odkrywanie podobieństw w przebiegach czasowych,
- Wykrywanie zmian i odchyłeń.

W tym podrozdziale skupimy się tylko na tych, które są najczęściej stosowane w regułach rekomendujących.

3.2.1 k - Najbliższych Sąsiadów

Algorytm k-najbliższych sąsiadów (k-NN) jest najczęściej używanym algorytmem klasyfikacji. Przyporządkowanie nowych elementów zostaje przeprowadzone na podstawie porównania obserwacji z k najbardziej podobnymi jej obiektami ze zbioru treningowego. Podstawowa idea algorytmu mówi, że jeżeli nowy rekord znajduje się w pewnym otoczeniu, to na podstawie k - najbliższych mu obserwacji zostanie przyporządkowana do niego etykieta (klasa), której pojawienie się w rozważanym zbiorze jest najliczniejsze.

Niech q będzie punktem dla którego chcemy odnaleźć jego klasę l .

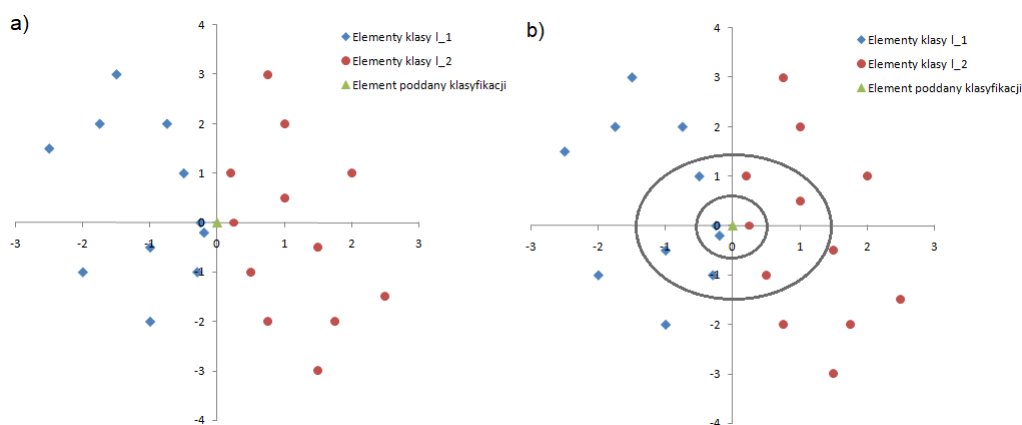
$X = \{\{x_1, l_1\}, \dots, \{x_n, l_n\}\}$ niech będzie zbiorem treningowym, gdzie x_j jest j -tym elementem zbioru, natomiast l_j etykietką klasy do której zbiór należy, $j \in \{1, \dots, n\}$.

W metodzie k-NN zostaje wybrany podzbiór

$$Y = \{\{y_1, l_1\}, \dots, \{y_k, l_k\}\}, k \in \{1, \dots, n\}$$

taki, że $Y \in X$ oraz $\sum_1^k d(q, y_k)$ jest minimalna. Y zawiera więc k punktów z X , które leżą najbliżej rozważanego punktu q . Następnie do punktu q zostaje przyporządkowana klasa taka, że:

$$l = f(\{l_1, \dots, l_k\}).$$



Rysunek 3.2: Metoda k - Najbliższych Sąsiadów.

Na powyższym rysunku widzimy przykładowe zastosowanie algorytmu k-NN. W części a) przedstawiony został zbiór treningowy z podziałem na dwie klasy (rąby, koła) oraz punkt, który będziemy chcieli przyporządkować do jednej z nich (trójkąt). W części b) przedstawiono dwa koła jedno prezentuje najbliższe sąsiedztwo dla $k = 3$, drugie dla $k = 9$. W obu przypadkach nowy punkt (trójkąt) zostanie przyporządkowany do klasy l_1 . Warto również zauważyć, że znajduje się on na granicy dwóch klastrów.

Jak wyżej opisana metoda przyporządkowuje wybranemu rekordowi najbardziej mu podobne w przypadku, gdy dane są więcej niż dwuwymiarowe?

Wykorzystywane są tutaj wcześniej opisane miary odległości z których najczęściej stosowaną jest odległość euklidesowa.

Najtrudniejszym zadaniem przy przeprowadzaniu algorytmu k-NN jest wybór k . Jeżeli k będzie zbyt małe - klasyfikator stanie się bardzo wrażliwy, jeżeli jednak k będzie zbyt duże sąsiedztwo może zawierać zbyt dużo punktów z innych klas. Rozważając przykład z Rysunku 3.2 łatwo zauważyć, że nawet mała zmiana w obserwacjach zbioru treningowego może doprowadzić do zmiany wyniku.

3.2.2 k-Średnich

Algorytm k-średnich jest prostym i zarazem efektywnym algorytmem grupowania.

Głównym celem algorytmu jest podział pewnego zbioru X :

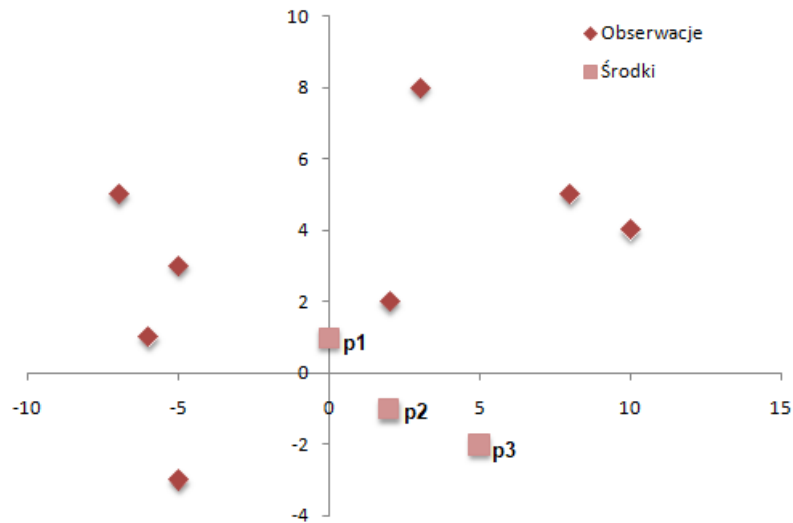
$$X = \{x_i = (x_{i1}, \dots, x_{id}) : i \in \{1, \dots, N\}\},$$

gdzie x_i jest d -wymiarowym wektorem cech opisującym obiekt na podzbiory.

W wyniku grupowania n - elementowego zbioru X na k podgrup jest macierz podziału

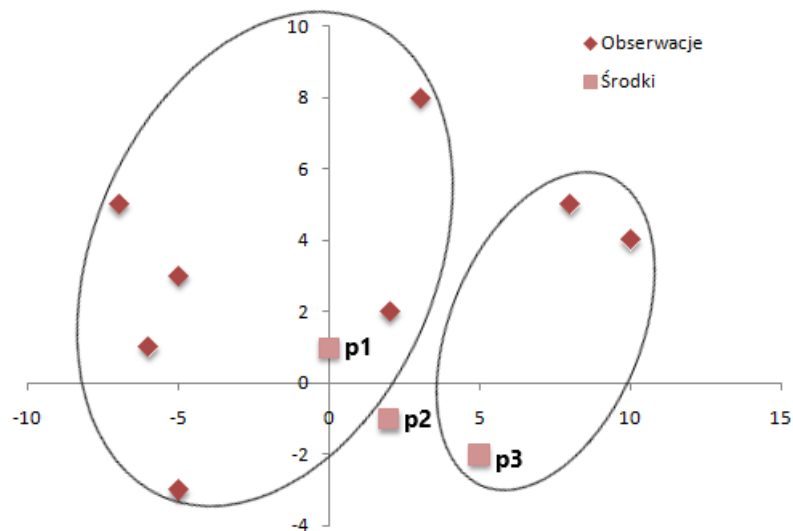
A o wymiarach $k \times n$. Każdy z elementów tej macierzy a_{ik} oznacza stopień w jakim wektor x_k przynależy do grupy.

Na wstępie algorytmu ustalamy wartość parametru k jako liczbę grup, które zostaną wyodrębnione. Wybieramy k reprezentantów, które stanowią prototypy grup.



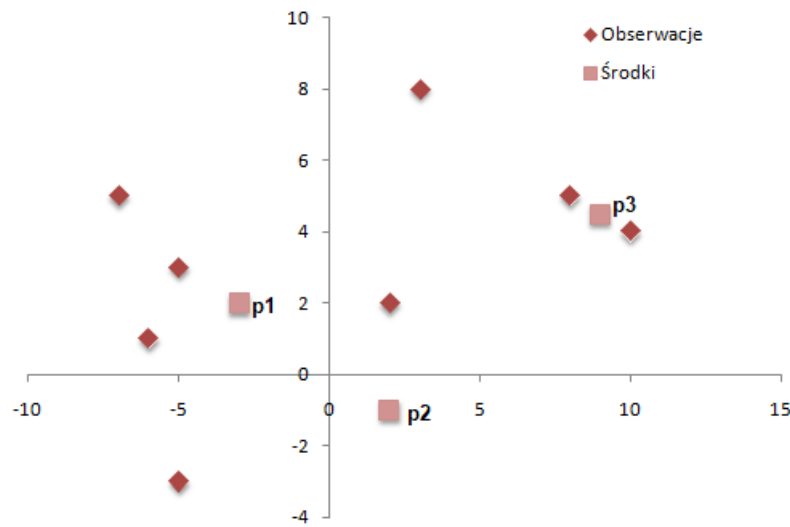
Rysunek 3.3: Metoda k-średnich. Wybór początkowych środków.

W powyższym przykładzie (Rysunek 3.3) wybranymi środkami są punkty p1, p2, p3. Kolejnym krokiem jest przypisanie każdego z elementów do najbliższej mu grupy.



Rysunek 3.4: Metoda k-średnich. Przypisanie elementów do grup.

Dla każdej z tak ustalonych grup obliczamy średnią arytmetyczną współrzędnych, które staną się kolejnymi środkami.



Rysunek 3.5: Metoda k-średnich. Wybór nowych środków.

Kroki te są wykonywane do momentu występowania migracji między obiektami.

W algorytmie k-średnich liczba grup pozostaje więc niezmienną, zmienna jest tylko przynależność do grup.

W metodzie tej poszukiwanie optymalnego podziału odpowiada wyznaczaniu takich grup, które minimalizują następującą funkcję:

$$J(u, A) = \sum_{i=1}^k \sum_{k=1}^N b_{ki} d^2(u_i, x_k),$$

gdzie

- $d(u, x)$ oznacza odległość elementu x od grupy wyznaczonej przez środek u ,
- N to liczebność zbioru X ,
- A oznacza macierz podziału.

3.2.3 drzewa decyzyjne-?

3.2.4 regresja wielowymiarowa - logistyczna - ?

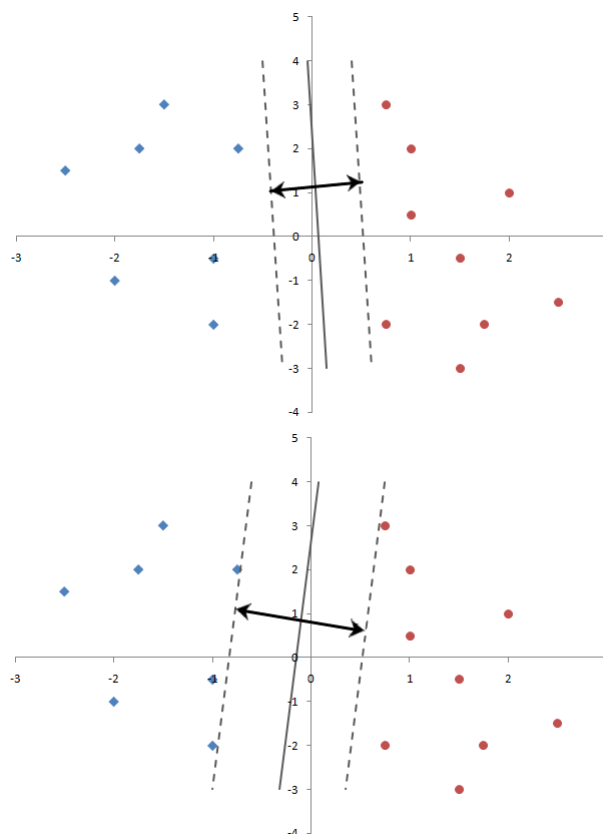
3.2.5 Sztuczna Sieć Neuronowa

Sztuczna Sieć Neuronowa jest klasyfikatorem opartym na systemie działania ludzkiego mózgu. Składa się z neuronów, które są połączone ze sobą synapsami. Te proste

elementy połączone są w sieć, która posiada możliwość uczenia się problemów klasyfikacyjnych.

3.2.6 Maszyna Wektorów Nośnych

Maszyny Wektorów Nośnych (ang. Support Vector Machines (SVM)) jest algorytmem uczenia maszynowego używanym zarówno w przypadku zadań klasyfikacji jak i regresji. Głównym celem algorytmu jest znalezienie w przestrzeni n - wymiarowej hiperpłaszczyzny, która wyraźnie sklasyfikuje obserwacje i podzieli na dwie klasy maksymalizując odległość między obserwacjami. Podejście to pozwala uniknąć błędnego przyporządkowania marginalnych obserwacji w przyszłości.



Rysunek 3.6: Maszyna Wektorów Nośnych - maksymalizacja odległości między klasami.

Aby formalnie zdefiniować Maszynę Wektorów Nośnych zaczniemy od zdefiniowania problemu klasyfikacji.

Definicja 25 *Klasyfikacja*

Niech w przestrzeni Ω znajdują się wektory danych $x_i \in \mathbb{R}^n$, $i \in \{1, \dots, n\}$. P niech

będzie próbą uczącą.

Wtedy:

$$P = \{(x_i, c_i) : x_i \in \mathbb{R}^n, c_i \in \{1, -1\}\}_{i=1}^N.$$

Należy znaleźć klasyfikator dzielący całą przestrzeń Ω na dwie klasy $\{-1, 1\}$, który pozwoli na przyporządkowywać nowego elementu do klasy. Szukamy funkcji $f(x)$ - granicy między klasami.

Definicja 26 *Liniowa Separowalność*

Dwie klasy nazywamy liniowo separowalnymi, jeżeli istnieje hiperpłaszczyzna $f(x)$ postaci:

$$f(x) = w^T x + b$$

taka, że:

$$\begin{cases} f(x_i) > 0, & x_i \in \{1\} \\ f(x_i) < 0, & x_i \in \{-1\}. \end{cases}$$

Liniową separowalność definiujemy więc za pomocą funkcji:

$$\langle w, x \rangle + b = 0,$$

gdzie w, b są parametrami spełniającymi założenia modelu:

$$f(x) = \begin{cases} 1, & \text{gdy } \langle w, x \rangle + b > 0 \\ -1, & \text{gdy } \langle w, x \rangle + b < 0. \end{cases}$$

Parametry w i b wyznaczamy w taki sposób, by odległość między płaszczyznami

$$\begin{aligned} b_{i1} \langle w, x \rangle + b &= 1, \\ b_{i2} \langle w, x \rangle + b &= -1 \end{aligned}$$

była maksymalna.

Zatem

$$f(x) = \begin{cases} 1, & \text{gdy } \langle w, x \rangle + b \geq 1 \\ -1, & \text{gdy } \langle w, x \rangle + b \leq -1, \end{cases}$$

oraz szukana odległość wyznaczana jest przez maksymalizowanie funkcji

$$\text{Margin (odległość)} = \frac{2}{\|w\|^2}.$$

Jest to fakt równoważny minimalizowaniu funkcji odwrotnej do powyższej

$$L(W) = \frac{\|w\|^2}{2},$$

przy ograniczeniach

$$y_i(\langle w, x \rangle + b) \geq 1.$$

Problem ten nazywamy problemem optymalizacji kwadratowej i rozważamy przy użyciu mnożników Lagrange'a. Minimalizujemy więc funkcję Lagrange'a:

$$L(w, b, \alpha) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^n \alpha_i (y_i(\langle w, x \rangle + b) - 1),$$

gdzie $\alpha_i \geq 0$ dla $i \in \{1, \dots, n\}$ to mnożniki Lagrange'a spełniające ograniczenia Karusha-Kuhna-Tuckera:

- $\alpha_i \geq 0$,
- $\alpha_i (y_i(\langle w, x \rangle + b) - 1) = 0$.

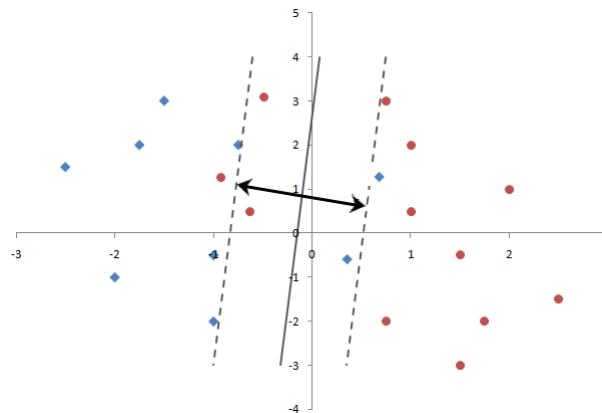
Możemy zauważyć, że w konsekwencji mnożniki Lagrange'a są niezerowe tylko dla wektorów nośnych. Liczba parametrów do oszacowania pozostaje jednak nadal zbyt duża. Częstym podejściem w tym przypadku jest przejście na postać dualną zadania optymalizacyjnego. Maksymalizujemy tu funkcję:

$$\sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j (x_i \times x_j),$$

przy ograniczeniach:

- $\alpha_i \geq 0$,
- $\forall_i \sum_{i=1}^n \alpha_i y_i = 0$.

Jeżeli elementy nie są jednak separowalne liniowo:



Rysunek 3.7: Maszyna Wektorów Nośnych - elementy nieseparowalne liniowo.

wprowadzamy klasyfikator **Soft Margin SVM** oraz zmienne osłabiające (pomocnicze).

W tym przypadku należy poddać minimalizacji funkcję:

$$L(w) = \frac{\|w\|^2}{2} + C \sum_{i=1}^n \varepsilon_i,$$

gdzie parametr C ocenia starty związane z każdym błędnie zaklasyfikowanym punktem. Funkcja $f(x)$ jest natomiast dana wzorem:

$$f(x) = \begin{cases} 1, & \text{gdzie } \langle w, x \rangle + b \geq 1 - \varepsilon \\ -1, & \text{gdzie } \langle w, x \rangle + b \leq -1 + \varepsilon. \end{cases}$$

Interpretacja rozwiązań:

- $\varepsilon_n = 0$ i $\alpha_i = 0$ - poprawne zaklasyfikowanie obserwacji, obserwacja leży poza marginesem,
- $\varepsilon_n = 0$ i $0 < \alpha_i < C$ - poprawne zaklasyfikowanie obserwacji, obserwacja leży na jednej z płaszczyzn ograniczających margines,
- $0 < \varepsilon_n < 1$ i $\alpha_i = C$ - poprawne zaklasyfikowanie obserwacji, obserwacja leży wewnątrz marginesu,
- $\varepsilon_n \geq 1$ i $\alpha_i = C$ - niepoprawne zaklasyfikowanie obserwacji.

3.3 Szacowanie Błędów Obliczeń

3.3.1 Ocena Dokładności Metody

Kolejna kwestia często poruszana w regułach rekomendujących to dokładność przewidywań. Jak oceniamy, czy uzyskany przez nas wynik jest wystarczająco dokładny?

Założmy, że dla użytkownika u i elementu i ze zbioru testowego P dostarczamy predykcję $\widehat{r_{(u,i)}}$. Aby ocenić jakość wyniku należy porównać ją ze znaną wartością $r_{(u,i)}$.

Najczęściej używanymi miarami dokładności modelu są:

- Średni Błąd (Mean Error)

$$ME = \frac{1}{|P|} \sum_{(u,i) \in P} (\widehat{r_{(u,i)}} - r_{(u,i)})$$

- Średni Błąd Bezwzględny (Mean Absolute Error)

$$MAD = \frac{1}{|P|} \sum_{(u,i) \in P} |\widehat{r_{(u,i)}} - r_{(u,i)}|$$

- Średni Błąd Kwadratowy (Mean Squared Error)

$$MSE = \frac{1}{|P|} \sum_{(u,i) \in P} (\widehat{r_{(u,i)}} - r_{(u,i)})^2$$

Funkcja kwadratowa jest funkcją monotoniczną co pozwala na dość częste zastępowanie średniego błędu kwadratowego przez średnią kwadratową błędów (**Root Mean Squared Error (RMSE)**).

$$RMSE = \sqrt{MSE}$$

Normalized RMSE (NRMSE) oraz **Normalized MAE (NMAE)** są znormalizowanymi, przez użycie zakresu wartości $r_{max} - r_{min}$, wersjami błędów RMSE i MAE. Kolejnym rodzajem powszechnie używanego błędu jest **Average RMSE**, który pozwala na użycie sum ważonych. Niech $w_i > 0$ będzie wagą dla elementu i i $\sum w_i = 1$.

$$ARMSE = \sqrt{\sum_{(u,i) \in P} w_i (\widehat{r_{(u,i)}} - r_{(u,i)})^2}$$

3.3.2 Ocena Precyzji Użyteczności

W regułach rekomendujących częstymi są przypadki, gdzie systemy nie będą przewidywać konkretnych preferencji użytkownika, jak oceny ale samo prawdopodobieństwo podjęcia przez użytkownika akcji.

Oceniając taki model wybieramy przykładowego użytkownika i ukrywamy kilka wybranych przez niego elementów aby spróbować przewidzieć zestaw elementów jakie użytkownik będzie chciał wybrać. W poniższej tabeli zostały zaprezentowane rezultaty jakie są możliwe do otrzymania przy przeprowadzaniu takiego testu:

	Zarekomendowane	Niezarekomendowane
Wybrane przez użytkownika	prawda-pozytywnie (pp)	fałsz-negatywnie (fn)
Niewybrane przez użytkownika	fałsz-pozytywnie (fp)	prawda-negatywnie (pn)

Zakładamy, że elementy nieużyte przez użytkownika nie zostałyby użyte również w przypadku gdy zostałyby zarekomendowane. Założenie to może okazać się jednak błędne, ponieważ w wyniku rekomendacji możemy otrzymać interesujące propozycje, których użytkownik nie brał pod uwagę. Jest to powód dlaczego wynik fp jest bardzo często przeszacowany.

W celu porównania algorytmów możemy posłużyć się następującymi statystykami:

$$\begin{aligned} \text{Precyzja (ang. Precision)} &= \frac{|pp|}{|pp|+|fp|}, \\ \text{Wskaźnik pp (ang. Recall / True Positive Rate)} &= \frac{|pp|}{|pp|+|fn|}, \\ \text{Wskaźnik fp (ang. False Positive Rate)} &= \frac{|fp|}{|fp|+|pn|}. \end{aligned}$$

Często jednak jest tu obserwowana zależność, która pokazuje, że przy wydłużaniu się listy rekomendacji rośnie wartość *wskaźnika pp* i jednocześnie maleje *precyzja*. Dla stałej długości listy rekomendacji porównanie precyzji algorytmów jest jak najbardziej miarodajne. Niemniej jednak porównanie precyzji, gdzie długości list rekomendacji są różne jest często bardzo trudne. Zostaje zatem wyznaczona krzywa przedstawiająca kompromis między *wskaźnikiem pp* i *precyzją*, lub między *wskaźnikiem pp* i *wskaźnikiem fp*. Formalnie pierwsza krzywa jest nazywana **krzywą precyzji (ang. precision - recall curve)** natomiast druga **krzywą ROC (ang. Receiver Operating Characteristic)**.

3.3.3 Ocena Rankingów

W poprzedniej sekcji została omówiona metoda wyboru odpowiedniego modelu rekomendującego. Każdy z takich modeli kończy się przedstawieniem pewnej listy rekomendowanych elementów. Często jednak również kolejność elementów jest bardzo ważna i ma duży wpływ na wybory użytkowników.

W tej części zostaną zaprezentowane metody służące do oceny otrzymanych na podstawie modelu rankingów i pomagające zapewnić odpowiedni porządek rekomendowanych elementów. Jeżeli elementy posiadają oceny (np. użytkowników) intuicyjnym jest stworzyć ranking przez uporządkowanie tych ocen w malejący sposób. W przypadku gdy jednak nie mamy takich danych lub nie jest odpowiednie tworzenie takiego rankingu użyjemy **znormalizowanej miary opartej na odległości (ang. Normalized Distance-based Performance Measure (NDPM))**:

Niech r_{ui} będzie rankingiem odniesienia i \hat{r}_{ui} rankingiem stworzonym przez wybrany system rekomendującym n_u elementów i użytkownikowi u .

Ponadto niech:

$$\begin{aligned} C^+ &= \sum_{i < j} \text{sgn}(r_{ui} - r_{uj}) \text{sgn}(\hat{r}_{ui} - \hat{r}_{uj}) \\ C^- &= \sum_{i < j} \text{sgn}^2(r_{ui} - r_{uj}) \text{sgn}(\hat{r}_{uj} - \hat{r}_{ui}) \\ C^u &= \sum_{i < j} \text{sgn}^2(r_{ui} - r_{uj}) \\ C^s &= \sum_{i < j} \text{sgn}(\hat{r}_{ui} - \hat{r}_{uj}) \\ C^0 &= C^u - (C^+ - C^-), \text{ gdzie} \end{aligned}$$

C^u jest liczbą par dla których ranking referencyjny okazuje się lepszą możliwością,

C^+ i C^- to liczba par z poprawną i niepoprawną kolejnością,

C^0 jest liczbą par dla których ranking referencyjny nie jest wiążący, kiedy ranking systemu rekomendującego jest.

Ostatecznie NDPM definiujemy w następujący sposób:

$$NDPM = \frac{C^- + 0.5C^0}{C^u}$$

Powiązania w rankingu referencyjnym pojawiają się naturalnie, kiedy znamy preferencje użytkownika (np. przydziela on oceny). Czasami jednak rankingi są bardziej specyficzne (np. kiedy dajemy użytkownikowi wybór między dwoma elementami). Wtedy też system rekomendujący nie powinien tworzyć rankingu klasyfikując jedne elementy wyżej niż inne. W takich przypadkach przychodzą nam z pomocą

miara Kendall's τ :

$$\tau = \frac{C^+ - C^-}{\sqrt{C^u} \sqrt{C^s}}$$

oraz **miara Spearman's ρ :**

$$\rho = \frac{1}{n_u} \frac{\sum_i (r_{iu} - \bar{r})(\hat{r}_{iu} - \bar{\hat{r}})}{\sigma(r)\sigma(\hat{r})},$$

gdzie \bar{r} i $\bar{\hat{r}}$ oznaczają średnie, natomiast $\sigma(r)$ i $\sigma(\hat{r})$ odchylenia standardowe.

Rozdział 4

Modele tworzenia rekomendacji

Systemy rekomendujące mogą zostać podzielone na systemy oparte na kontekście i systemy oparte na użytkownikach. W pierwszym przypadku eksploracja danych jest dokonywana pod względem wektorów cech rozważanych elementów. Główne założenie mówi, że jeżeli użytkownik wybrał przedmiot A w przeszłości oraz przedmiot B jest podobna do A, to użytkownik będzie skłonny wybrać również przedmiot B. W systemach opartych na użytkownikach badaniu podlegają natomiast zależności, które występują między produktami, a użytkownikami. Idea rozważana pod tym hasłem mówi, że jeżeli użytkownicy A i B wykazują podobieństwo oraz użytkownik A zaopiniuje pewien przedmiot, którego użytkownik B jeszcze nie ocenił, to prawdopodobnie opinia użytkownika B będzie podobna do opinii użytkownika A.

Założmy, że mamy zbiór sześciu książek $\{k_1, k_2, k_3, k_4, k_5, k_6\}$, oraz zbiór sześciu czytelników $\{c_1, c_2, c_3, c_4, c_5, c_6\}$.

Poniższa tabela przedstawia oceny jakie czytelnicy przyporządkowali po przeczytaniu poszczególnym pozycjom. Znak '?' oznacza, że czytelnik danej książki nie czytał.

Czytelnicy		c_1	c_2	c_3	c_4	c_5	c_6
Książki	k_1	6	3	?	6	4	?
	k_2	?	6	6	5	6	?
	k_3	7	7	8	7	8	9
	k_4	8	10	10	7	6	8
	k_5	9	6	6	6	6	?
	k_6	5	7	7	5	4	2

Niech:

- n_u oznacza liczbę użytkowników, w tym przypadku $n_u = 6$,

- n_m oznacza liczbę elementów, w tym przypadku liczbę książek, $n_m = 6$.

Dodatkowo wprowadźmy parametry:

- $r(i, j)$, gdzie $r(i, j) = 1$, jeżeli użytkownik j ocenił książkę i ,
- $y(i, j)$ - ocena jaką użytkownik j przydzielił książce i , $y(i, j) \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ (ocenę definiujemy tylko wtedy gdy $r(i, j) = 1$).

Celem jest na podstawie wskaźników $r(i, j)$, $y(i, j)$ przewidzieć brakujące oceny.

4.1 Systemy rekomendujące oparte na treści (Content-based recommender systems):

Rozważmy sytuację w której szukamy przyjemnej lektury na zimowy wieczór. Zaczynamy od przeglądania tytułów wśród interesujących nas gatunków. Następnie spojrzymy zapewne na oceny jakie inni czytelnicy wydali na temat danej pozycji. Analizując wartość uzyskanych ocen ostatecznie podejmujemy na ich podstawie najbardziej odpowiednią dla nas decyzję.

4.1.1 Algorytm

Metody rozwiązywania podobnych problemów często spotykane są w matematyce. Jednym z przykładów są systemy oparte na treści, które wyróżnia ukierunkowanie na spersonalizowany poziom użytkownika oraz treść produktu. Głównym celem tej metody jest scharakteryzowanie wektora cech, które użytkownik ceni, a następnie zasugerowanie mu produktów, które te cechy posiadają. Wspomniana metoda opiera się na obliczaniu podobieństw oraz wykorzystuje zadania uczenia maszynowego, takie jak klasyfikacja.

W typie rekomendacji opartym na treści stworzenie rekomendacji i wygenerowanie listy elementów, które mogą być odpowiednie użytkownikowi możemy przedstawić w trzech krokach:

1. Wygenerowanie profilu produktu
2. Wygenerowanie profilu użytkownika
3. Rozpoznanie cech produktu odpowiednich dla użytkownika

Przykład

Niech poniższa tabela określa w jakim stopniu (skala 0-1) każda z książek reprezentuje cechy gatunków g_1, g_2 .

Gatunki		g_1	g_2
Książki	k_1	0,9	0
	k_2	1	0,01
	k_3	0,99	0
	k_4	0,1	1
	k_5	0	0,9
	k_6	0,8	0,3

Dodatkowo zakładamy, że istnieje gatunek g_0 , którego cechy reprezentują wszystkie książki oraz dla każdej z książek $g_0 = 1$.

Zatem $g^{(1)} = \begin{bmatrix} 1 \\ 0,9 \\ 0 \end{bmatrix}$ jest wektorem cech odpowiadającym książce k_1 , $g^{(2)} = \begin{bmatrix} 1 \\ 1 \\ 0,01 \end{bmatrix}$ jest wektorem cech odpowiadającym książce k_2 . Analogicznie możemy wyznaczyć podobne wektory dla każdej z książek.

Należy jednak zaznaczyć, że liczba wybranych przez nas cech to $n = 2$.

Dla każdego użytkownika j wyznaczamy wektor parametrów $\Theta^{(j)} \in \mathbb{R}^3$ (w ogólnej postaci $\Theta^{(j)} \in \mathbb{R}^{n+1}$), który przedstawia (w skali 0-5) preferencje użytkownika dotyczące poszczególnych gatunków. I tak odpowiednio $\Theta^{(1)}$ to wektor preferencji czytelnika c_1 , $\Theta^{(2)}$ wektor preferencji użytkownika c_2 , $\Theta^{(3)}$ wektor preferencji użytkownika c_3 i $\Theta^{(4)}$ wektor preferencji użytkownika c_4 , $\Theta^{(5)}$ wektor preferencji użytkownika c_5 i $\Theta^{(6)}$ wektor preferencji użytkownika c_6 .

Aby przewidzieć ocenę jaką użytkownik j wystawiłby książce i po jej przeczytaniu użyjemy równania:

$$(\Theta^{(j)})^T g^{(i)}.$$

Obliczmy ocenę jaką książce k_3 wystawiłby użytkownik c_1 przy założeniu, że wektor preferencji użytkownika c_1 jest postaci $\Theta^{(1)} = \begin{bmatrix} 0 \\ 5 \\ 0 \end{bmatrix}$. Użytkownik ten preferuje więc książki gatunku g_1 , gdy książki gatunków g_0 i g_2 są dla niego nieatrakcyjne.

Zatem:

$$(\Theta^{(1)})^T g^{(3)} = [0, 5, 0] \begin{bmatrix} 1 \\ 0,99 \\ 0 \end{bmatrix} = 0 \cdot 1 + 5 \cdot 0,99 + 0 \cdot 0 = 4,95.$$

Przewidywaną oceną jest zatem 4,95. Po przeprowadzeniu podobnych obliczeń dla wszystkich wcześniej nieznanymi ocen możemy zarekomendować naszemu użytkownikowi nową lekturę.

4.1.2 Wygenerowanie profilu produktu

W większość systemów rekomendacji opartych na treści można zauważyć użycie prostych modeli wyszukiwujących. Jednym z najbardziej popularnych jest Model Przestrzeni Wektorowej (*ang. Vector Space Model*) z algorytmem TF-IDF (*ang. TF – term frequency, IDF – inverse document frequency*). Jest to przestrzenna forma reprezentacji dokumentów, gdzie dokument i jest reprezentowany przez wektor w przestrzeni n -wymiarowej x_i , a każdy z n wymiarów stanowi rozważaną cechę produktu. Dla danych w formie dokumentów tekstowych cechami, które pomagają scharakteryzować temat dokumentu, np. artykułu na stronie internetowej są słowa. Dla każdego ze słów zostaje obliczona wartość funkcji TFIDF (została ona dokładnie omówiona poniżej). W końcowym etapie słowa, które otrzymały najwyższe wyniki zostają uznane za charakteryzujące rozważany dokument.

Formalnie rzecz ujmując każdy z dokumentów jest przedstawiony za pomocą wektora wag, gdzie waga w odpowiedni sposób wyraża zależność między dokumentem, a badanym terminem.

Niech $K = \{k_1, k_2, \dots, k_n\}, n \in \mathbf{N}$ będzie zestawem dokumentów / analizowanych przedmiotów. Natomiast $G = \{g_1, g_2, \dots, g_n\}, n \in \mathbf{N}$ zestawem cech rozważanych w przedmiotach. Każdy z dokumentów $k_j, j \in \{1, \dots, n\}$ jest reprezentowany jako wektor w przestrzeni wektorowej n -wymiarowej. Zatem $k_j = [w_{1j}, w_{2j}, \dots, w_{nj}]$, gdzie w_{kj} jest wagą dla cechy g_k w dokumencie k_j .

Do generowania profilu produktu używany jest wspomniany wcześniej algorytm TFIDF pozwalający policzyć względną ważność powiązania cechy z przedmiotem. Zakładamy, że:

- rzadkie cechy są równie istotne jak częste (założenie IDF),
- kilkukrotne wystąpienie terminu w rozważanym dokumencie jest równie istotne jak pojedyncze (założenie TF),

- długość dokumentu (filmu, książki) nie ma znaczenia (założenie normalizacji).

Zatem jeżeli termin występuje często w konkretnym przedmiocie rozważań (TF) i równocześnie rzadko w pozostałych elementach zbioru (IDF) ma większe prawdopodobieństwo stać się jedną z istotnych cech rozważanych w temacie. Ponadto normalizacja wektorów wag pozwala zrównoważyć wartość wyników i umożliwia ich porównywanie w dalszej analizie.

Powyższe założenia odzwierciedla funkcja TFIDF:

$$TFIDF(t_k, d_j) = TF(g_k, k_j) * IDF,$$

gdzie:

- $TF(g_k, k_j)$ (macierz *term frequency*) przedstawia odniesienie każdego z podanych terminów do każdego z badanych elementów:

$$TF(g_k, k_j) = \frac{f_{k,j}}{\max_z f_{z,j}},$$

$\max_z f_{z,j}$ - maksymalna w odniesieniu do wszystkich cech c_z , które pojawiły się w dokumencie p_j częstotliwość wystąpień ($f_{z,j}$)

- IDF (*inverse document frequency*) wyraża się formułą:

$$IDF = \log \frac{N}{n_k}$$

N - całkowita liczba dokumentów w zbiorze,

n_k - liczba dokumentów w których cecha c_k wystąpiła przynajmniej raz.

Ponadto w związku z założeniem o normalizacji wagi, które zostały uzyskane w wyniku $TFIDF(t_k, d_j)$ poddane zostaną metodzie transformacji kosinusowej:

$$w_{k,j} = \frac{TFIDF(t_k, d_j)}{\sqrt{\sum_{i=1}^{|T|} TFIDF(t_i, d_j)^2}}.$$

Tak wygenerowany został zbiór słów, które są reprezentacją danego dokumentu oraz jego tematu. Ostatnim krokiem przed zarekomendowaniem pozycji czytelnikom jest wyestymowanie podobieństw występujących między poszczególnymi dokumentami. Naturalnymi miarami odległości, które można tu wykorzystać są:

- współczynnik Jaccarda mierzący odległość między zbiorami słów,
- podobieństwa kosinusów mierzące odległość między wektorami:

$$\text{sim}(k_i, k_j) = \frac{\sum_{i=k} w_{k,i} \cdot w_{k,j}}{\sqrt{\sum_{k=1} w_{k,i}^2} \cdot \sqrt{\sum_{k=1} w_{k,j}^2}}.$$

Przy estymowaniu odległości kosinusów między dwoma dokumentami elementami wektora są słowa, które zostały wybrane na podstawie wcześniej przeprowadzonego algorytmu. W wektorze tym wartość 1 oznacza, że słowo pojawiło się w zbiorze opisującym dokument, natomiast 0 oznacza sytuację przeciwną.

4.1.3 Wygenerowanie profilu użytkownika

Przy generowaniu wektora opisującego rozważane dokumenty należy uwzględnić komponenty opisujące preferencje użytkowników. Dla każdego użytkownika u przedstawiamy jego preferencje w postaci wektora x_u , gdzie użytkownik pozycjonuje element i poprzez wektor cech x_i .

4.1.4 Rozpoznanie cech produktu odpowiednich dla użytkownika

Proces rekomendacji bazuje na dopasowaniu cech profilu użytkownika i wartości opisujących treść obiektu. Rezultatem jest stwierdzenie czy rozważany kandydat jest zainteresowany analizowanym przedmiotem. Zainteresowanie użytkownika danym przedmiotem można wyestymować używając podobieństwa cosinusów. Rozważmy dokument, gdzie użytkownik wyraża zainteresowanie większością cech w wektorze x_i opisującym dokument i . Otrzymamy cosinus kąta między użytkownikiem i dokumentem będący dodatnim ułamkiem. Oznacza to, że kąt będzie bliski 0° a tym samym odległość między wektorami będzie mała. W przypadku przeciwnym otrzymamy cosinus bliski zeru lub mniejszy od zera, oznacza to kąt należący do przedziału $(90^\circ; 180^\circ)$ oraz małe podobieństwo między rozważanymi wektorami.

4.2 Filtrowanie kolaboratywne (Collaborative filtering)

Idea rozważana pod tym hasłem mówi, że jeżeli użytkownicy A i B wykazują podobieństwo oraz użytkownik A zaopiniuje pewien przedmiot, którego użytkownik B jeszcze nie ocenił, to prawdopodobnie opinia użytkownika B będzie podobna do opinii użytkownika A. Wyróżniamy dwa podstawowe typy filtrowania kolaboratywnego:

- filtrowanie kolaboratywne oparte na użytkowniku (ang. user-based)
- filtrowanie kolaboratywne oparte na elementach (ang. item-based)

Cechą wspólną dla obu powyższych metod jest fakt, że oceny jednych użytkowników są podstawą do tworzenia rekomendacji dla innych.

Podejście kolaboratywne omija niektóre ograniczenia występujące w metodach kontekstowych. Rekomendację są dokonywane tylko na podstawie ocen. Dzięki temu systemowi możemy również dokonywać rekomendacji na przestrzeni kilku kontekstów, co pozwala uniknąć nam konkretne wyspecjalizowanej przestrzeni obecnej w metodach kontekstowych.

4.2.1 Przykład

Założmy więc przypadek gdy nie posiadamy informacji dotyczących gatunków książek ani informacji o preferencjach użytkowników.

Wiemy tylko jakie oceny (z zakresu 1-10) wystawili użytkownicy książkom, które przeczytali:

Czytelnicy		c_1	c_2	c_3	c_4	c_5	c_6
Książki	k_1	6	3	?	6	4	?
	k_2	?	6	6	5	6	?
	k_3	7	7	8	7	8	9
	k_4	8	10	10	7	6	8
	k_5	9	6	6	6	6	?
	k_6	5	7	7	5	4	2

W takim przypadku, aby odnaleźć brakujące oceny i dokonać rekomendacji możemy użyć metody filtrowania kolaboratywnego opartego na użytkownikach lub filtrowania kolaboratywnego opartego na elementach.

4.2.2 Filtrowanie kolaboratywne oparte na użytkowniku

Algorytm

Stworzenie rekomendacji opartej na użytkowniku wykonamy w kolejnych krokach:

1. Znalezienie podobieństwa między czytelnikami opierającego się na informacji o przeczytanych książkach. Najczęstszymi stosowanymi podejściami do obliczania szukanego podobieństwa są Metryka Euklidesowa i Współczynnik Korelacji Pearsona.
2. Wyestymowanie ocen, które czytelnicy (w szczególności osoba dla której tworzymy rekomendację) mogliby wystawić dla nieprzeczytanych książek z rozważanego zbioru.

W celu dokładniejszego zrozumienia tego typu filtrowania został przedstawiony poniższy przykład.

Przykład

Chcąc obliczyć podobieństwo między użytkownikiem c_2 i c_3 wybierzmy książki, które przeczytali obaj użytkownicy. W tym przypadku są to: k_2, k_3, k_4, k_5, k_6 . Na tej podstawie tworzymy odpowiednio wektory ocen użytkowników $o_{2,3}^{(2)}$ i $o_{2,3}^{(3)}$.

Zatem

$$o_{2,3}^{(2)} = \begin{bmatrix} 6 \\ 7 \\ 10 \\ 6 \\ 7 \end{bmatrix} \text{ oraz } o_{2,3}^{(3)} = \begin{bmatrix} 6 \\ 8 \\ 10 \\ 6 \\ 7 \end{bmatrix}.$$

Posługując się odległością euklidesową

$$d_e(x, y) = \sqrt{\sum_{i=1}^n |x_i - y_i|^2}$$

obliczamy odległość między użytkownikami c_2 i c_3 :

$$d_e(o_{2,3}^{(2)}, o_{2,3}^{(3)}) = \sqrt{(6-6)^2 + (7-8)^2 + (10-10)^2 + (6-6)^2 + (7-7)^2} = \sqrt{1} = 1.$$

Postępując w podobny sposób dla każdej z par użytkowników otrzymamy następującą macierz odległości:

Czytelnicy	c_1	c_2	c_3	c_4	c_5	c_6
c_1	0	5,099	4,243	3	4,359	3,606
c_2	5,099	0	1	4,796	5,196	5,745
c_3	4,243	1	0	3,873	5	5,477
c_4	3	4,796	3,873	0	2,828	3,742
c_5	4,359	5,196	5	2,828	0	3
c_6	3,606	5,745	5,477	3,742	3	0

W celu uzyskania wartości z przedziału $[0, 1]$ poddajemy macierz normalizacji ($\max\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\} - \min\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$):

Czytelnicy	c_1	c_2	c_3	c_4	c_5	c_6
c_1	0	0,5099	0,4243	0,3	0,4359	0,3606
c_2	0,5099	0	0,1	0,4796	0,5196	0,5745
c_3	0,4243	0,1	0	0,3873	0,5	0,5477
c_4	0,3	0,4796	0,3873	0	0,2828	0,3742
c_5	0,4359	0,5196	0,5	0,2828	0	0,3
c_6	0,3606	0,5745	0,5477	0,3742	0,3	0

Im większa odległość między czytelnikami tym mniejsze jest podobieństwo jakie między nimi występuje. Zakładając, że największa wartość prawdopodobieństwa to 1 macierz podobieństwa przyjmuje wartości:

Czytelnicy	c_1	c_2	c_3	c_4	c_5	c_6
c_1	1	0,4901	0,5757	0,7	0,5641	0,6394
c_2	0,4901	1	0,9	0,5204	0,4804	0,4255
c_3	0,5757	0,9	1	0,6127	0,5	0,4523
c_4	0,7	0,5204	0,6127	1	0,7172	0,6258
c_5	0,5641	0,4804	0,5	0,7172	1	0,7
c_6	0,6394	0,4255	0,4523	0,6258	0,7	1

Bazując na przedstawionych podobieństwach wyestymujemy ocenę jaką użytkownik c_1 zaproponuje dla książki k_2 . W poniższym równaniu obliczymy średnią ważoną ocen wykorzystując wartość podobieństwa między użytkownikiem c_1 , a pozostałymi użytkownikami, którzy wcześniej ocenili książkę:

$$\frac{0,4901 \cdot 6 + 0,5757 \cdot 6 + 0,7 \cdot 5 + 0,5641 \cdot 6}{0,4901 + 0,5757 + 0,7 + 0,5641} = 5,7$$

Na podstawie metody filtrowania kolaboratywnego opartej na użytkownikach wnioskujemy, że użytkownik c_1 wystawiłby książce k_2 ocenę 5,7.

Postępując w analogiczny sposób przewidzimy wszystkie brakujące oceny :

Czytelnicy		c_1	c_2	c_3	c_4	c_5	c_6
Książki	k_1	6	3	4,57	6	4	4,88
	k_2	5,7	6	6	5	6	5,72
	k_3	7	7	8	7	8	9
	k_4	8	10	10	7	6	8
	k_5	9	6	6	6	6	6,67
	k_6	5	7	7	5	4	2

na których podstawie możemy dokonać rekomendacji dla naszych użytkowników.

Możemy wnioskować, że dla użytkownika c_3 książka k_1 prawdopodobnie nie będzie zbyt atrakcyjna. Użytkownik c_6 , natomiast, z chęcią przeczyta książkę k_5 .

4.2.3 Filtrowanie kolaboratywne oparte na elementach

W przypadku filtrowania kolaboratywnego opartego na elementach wartości podobieństwa między użytkownikami zostaje zastąpiona przez wartości podobieństwa między elementami.

W tym przypadku założenie mówi, że jeżeli użytkownik wybrał przedmiot A w przeszłości oraz przedmiot B jest podobna do A, to użytkownik będzie skłonny wybrać również przedmiot B.

Algorytm

Podobnie jak w przypadku opartym na użytkowniku, w tym również należy wykonać dwa kroki.

1. Pierwszym etapem jest znalezienie podobieństw występujących między elementami. Najczęstszą miarą podobieństwa w tym przypadku jest podobieństwo kosinusów. Miara ta wyraża podobieństwo między n-wymiarowymi wektorami poprzez kąt między nimi w przestrzeni wektorowej. Wraz ze wzrostem wartości kąta rośnie podobieństwo.
2. Następnie, na podstawie wydanych przez użytkownika ocen, należy wyestymować noty dla elementów przez niego nieocenionych.

Przykład

W tym przypadku wektory odzwierciedlają zbiory ocen wystawione poszczególnym książkom przez użytkowników.

Wektor mają zatem postać:

$$o^{(k_1)} = \begin{bmatrix} 6 \\ 3 \\ 6 \\ 4 \end{bmatrix}, o^{(k_2)} = \begin{bmatrix} 6 \\ 6 \\ 5 \\ 6 \end{bmatrix}, o^{(k_3)} = \begin{bmatrix} 7 \\ 7 \\ 8 \\ 7 \\ 8 \\ 9 \end{bmatrix}, o^{(k_4)} = \begin{bmatrix} 8 \\ 10 \\ 10 \\ 7 \\ 6 \\ 8 \end{bmatrix}, o^{(k_5)} = \begin{bmatrix} 9 \\ 6 \\ 6 \\ 6 \\ 6 \end{bmatrix}, o^{(k_6)} = \begin{bmatrix} 5 \\ 7 \\ 7 \\ 5 \\ 4 \\ 2 \end{bmatrix}$$

Warto zauważyć, że wektory są różnej długości.

Aby obliczyć podobieństwo między książkami k_1 i k_2 wyznaczmy wektory ocen w których uwzględnimy przypadki, gdzie jeden użytkownik ocenił obie pozycje.

Zatem:

$$\overline{o^{(k_1)}} = \begin{bmatrix} 3 \\ 6 \\ 4 \end{bmatrix}, \overline{o^{(k_2)}} = \begin{bmatrix} 6 \\ 5 \\ 6 \end{bmatrix}.$$

Następnie używając wzoru na podobieństwo kosinusowe

$$\text{sim}(\vec{a}, \vec{b}) = \frac{\vec{a} \cdot \vec{b}}{|\vec{a}| |\vec{b}|}$$

obliczamy podobieństwo między wybranymi książkami

$$\text{sim}(k_1, k_2) = \frac{\overline{o^{(k_1)}} \cdot \overline{o^{(k_2)}}}{|\overline{o^{(k_1)}}| |\overline{o^{(k_2)}}|} = \frac{3 \cdot 6 + 6 \cdot 5 + 4 \cdot 6}{\sqrt{6^2 + 3^2 + 6^2 + 4^2} \sqrt{6^2 + 6^2 + 5^2 + 6^2}} = 0,6339.$$

Postępując w analogiczny sposób otrzymamy macierz podobieństwa elementów:

Czytelnicy	k_1	k_2	k_3	k_4	k_5	k_6
k_1	1	0,6339	0,7372	0,7195	0,8935	0,7599
k_2	0,6339	1	0,7951	0,8150	0,7977	0,8898
k_3	0,7372	0,7951	1	0,9780	0,8586	0,9200
k_4	0,7195	0,8150	0,9780	1	0,8860	0,9681
k_5	0,8935	0,7977	0,8586	0,8860	1	0,9413
k_6	0,7599	0,8898	0,9200	0,9681	0,9413	1

Wyestymujemy teraz ocenę jaką użytkownik c_6 zaproponuje dla książki k_2 . Ponownie obliczymy średnią ważoną ocen, tym razem, wykorzystując wartość podobieństwa między książką k_1 , a książkami ocenionymi wcześniej przez użytkownika oraz oceny jakie nadał on tym pozycjom:

$$\frac{(0,7951 \cdot 9 + 0,8150 \cdot 8 + 0,8898 \cdot 2)}{(0,7951 + 0,8150 + 0,8898)} = 6,16.$$

Na podstawie przeprowadzonych obliczeń zakładamy, że ocena jaką wystawiłby po przeczytaniu użytkownik c_6 książce k_2 to 6,16.

Powtarzając powyższe obliczenia dla każdej z pozycji wcześniej nieocenionej przez wybranego użytkownika otrzymamy wszystkie brakujące opinie. Następnie bazując na zdobytych danych z łatwością odnajdziemy pozycję najbardziej odpowiednią do zarekomendowania użytkownikowi.

4.3 Systemy rekomendujące kontekstowe (Context – aware recommender systems):

4.3.1 Algorytm

Systemy rekomendujące kontekstowe są systemami rekomendującymi opartymi na treści w których zostaje uwzględniony dodatkowy wymiar zwany kontekstem.

Definicja 27 (*Kontekst*)

Kontekstem w rozumieniu eksploracji danych nazywamy obecny stan użytkownika. Autorzy książki "Recommender Systems Handbook" definiują kontekst jako wydarzenie charakteryzujące etap życia użytkownika i wpływające na jego preferencje, status. Pod pojęciem tym kryje się nie tylko miejsce, czas, pogoda, dzień, ale także fakt, że użytkownik spędza czas samotnie lub w gronie innych osób, narodziny dziecka, zmiana pracy, małżeństwo. Wiedza na temat kontekstowych informacji pozwala zbudować wzorce i algorytmy w odniesieniu do konkretnych, istotnych danych.

Przykładem, który dobrze obrazuje podejście kontekstowe w tworzeniu rekomendacji są biura podróży, które w tworzeniu ofert uwzględniają sezon, miejsca, czas, sytuację finansową klienta oraz czas, kiedy oferta zostaje przedstawiona.

Warto zauważyć również, że uprzednio opisane metody opierały się głównie na rozważaniu problemów dwu-wymiarowych. W tym podejściu, przez dodanie nowego wymiaru, jakim jest kontekst, zaczynamy rozważać problemy trój-wymiarowe:

$$R: \text{Użytkownik} \times \text{element} \times \text{kontekst} \Rightarrow \text{Rekomendacja}$$

W modelu kontekstowym rekomendacje są generowane w dwóch krokach:

1. Metody systemów rekomendujących opartych na treści służące do wygenerowania listy rekomendacji bazującej na preferencjach użytkownika.
2. Odfiltrowanie rekomendacji, które odpowiadają przyjętemu kontekstowi.
Wyróżniamy tutaj dwa warianty. W pierwszym etapie filtrowania zostaje dokonany na końcu, natomiast w drugim filtrowanie jest etapem wstępnym do tworzenia rekomendacji.

Filtrowanie jako etap wstępny (ang. Pre-Filtering)

W tym podejściu informacje kontekstowe używane są do odfiltrowania najbardziej istotnych informacji i skonstruowania dwuwymiarowego zbioru danych. Bardzo dużą zaletą tego podejścia jest możliwość implementacji w kolejnym kroku wcześniej opisanych metod rekomendacji.

Filtrowanie jako etap końcowy (ang. Post-Filtering)

Informacje o kontekście są ignorowane w wejściowych danych, a rekomendacja dokonywana jest na całym zbiorze. To w następnym kroku lista rekomendacji stworzona dla użytkownika jest zawężana przez uwzględnienie kontekstu.

4.4 Hybrydowe systemy rekomendujące ():

(Czy potrzebny mi jeszcze ten podrozdział?)

4.5 Systemy rekomendujące oparte na modelach():

(Czy potrzebny mi jeszcze ten podrozdział?)

Rozdział 5

Eksperymenty / część praktyczne

Rozdział 6

Podsumowanie