Wykład 1: Wykład 1: Analiza Składowych Głównych (PCA)

1.1 Cel metody

Głównym celem Analizy Składowych Głównych (ang. *Principal Component Analysis*) jest redukcja wymiaru danych (zmniejszenie liczby zmiennych) przy jednoczesnym zachowaniu jak największej ilości informacji.

- **Informacja** w PCA jest utożsamiana z **wariancją** (zmiennością) danych.
- Metoda polega na transformacji układu zmiennych pierwotnych $X_1, \dots, X_p$ w nowy układ zmiennych nieskorelowanych $Z_1, \dots, Z_p$ (składowych głównych).

1.2 Definicja pierwszej składowej głównej (Z_1)

Pierwsza składowa główna to taka kombinacja liniowa zmiennych pierwotnych, która posiada **maksymalną wariancję**.

$$Z_1 = a_{11}X_1 + a_{12}X_2 + \cdots + a_{1p}X_p$$

Gdzie wektor wag $a_1 = [a_{11}, \dots, a_{1p}]^T$ jest wyznaczany tak, aby zmaksymalizować $Var(Z_1)$.

1.2.1 Warunek normalizacyjny (Haczyk egzaminacyjny)

Aby zadanie maksymalizacji wariancji miało skończone rozwiązanie, konieczne jest nałożenie ograniczenia na wagi:

$$\sum_{j=1}^{p} a_{1j}^2 = 1 \quad \text{(lub wektorowo } a_1^T a_1 = 1\text{)}$$

Bez tego warunku wariancję można by zwiększać w nieskończoność, po prostu skalując wagi.

1.3 Rozwiązanie matematyczne i Wartości Własne

Problem maksymalizacji wariancji przy warunku normalizacyjnym rozwiązuje się metodą mnożników Lagrange'a.

- **Wektor wag** a_1 okazuje się być **wektorem własnym** macierzy kowariancji (lub korelacji) odpowiadającym jej **największej wartości własnej** (λ_1).
- **Wariancja** Z_1 jest równa tej wartości własnej: $Var(Z_1) = \lambda_1$.
- Kolejne składowe ($Z_2, \dots$) odpowiadają kolejnym (malejącym) wartościom własnym.

1.4 Własności składowych głównych

1. Składowe są uporządkowane malejąco względem niesionej informacji (wariancji):

$$Var(Z_1) \geq Var(Z_2) \geq \cdots \geq Var(Z_p)$$

2. Składowe są wzajemnie **nieskorelowane** (ortogonalne): $Cov(Z_i, Z_j) = 0$ dla $i \neq j$.
3. **Wariancja całkowita** układu jest zachowana:

$$\text{Wariancja Całkowita} = \sum_{i=1}^{p} Var(X_i) = \sum_{i=1}^{p} Var(Z_i) = \sum_{i=1}^{p} \lambda_i = Tr(R)$$

Dla zmiennych standaryzowanych (macierz korelacji) wariancja całkowita wynosi p (liczbę zmiennych).

1.5 Kryteria wyboru liczby składowych

Decyzja o tym, ile składowych ($k < p$) zachować w ostatecznym modelu:

- **Kryterium Kaisera:** Wybieramy tylko te składowe, dla których wartość własna $\lambda_i > 1$ (dla macierzy korelacji). Logika: składowa musi wyjaśniać więcej niż jedna pojedyncza zmienna pierwotna.
- **Kryterium Cattella (Wykres osypiska):** Szukamy punktu „łokcia" na wykresie wartości własnych. Odrzucamy te, które leżą na płaskim „osypisku" (wyjaśniają szum).
- **Kryterium procentowe:** Wybieramy tyle składowych, aby wyjaśniły łącznie np. 75% lub 80% wariancji całkowitej.

Wykład 2: Wykład 2: Założenia PCA i Wstęp do Analizy Czynnikowej (FA)

2.1 Weryfikacja zasadności stosowania PCA

Przed wykonaniem analizy należy sprawdzić, czy zmienne są ze sobą skorelowane (jeśli nie są, redukcja wymiaru nie ma sensu).

1. **Macierz korelacji:** Współczynniki powinny być istotnie różne od zera (zalecane $|r| > 0.3$).
2. **Test sferyczności Bartletta:**

- **Hipoteza zerowa (H_0):** Macierz korelacji jest macierzą jednostkową ($R = I$). Oznacza to brak korelacji między zmiennymi.
- **Cel:** Chcemy **odrzucić** H_0 (czyli wykazać, że korelacje istnieją).
- Statystyka testowa opiera się na wyznaczniku macierzy korelacji ($\ln |R|$).

3. **Współczynnik KMO (Kaiser-Meyer-Olkin):**
   - Mierzy adekwatność doboru zmiennych (Measure of Sampling Adequacy).
   - Porównuje kwadraty korelacji (r_{ij}^2) z kwadratami korelacji cząstkowych ($\hat{r}_{ij}^2$).
   - **Interpretacja:** Wartości z przedziału $[0, 1]$.
     - $KMO > 0.5$: Analiza akceptowalna.
     - $KMO > 0.7$: Analiza dobra.

2.2 Analiza Czynnikowa (FA) – Istota metody

Analiza czynnikowa (Factor Analysis) różni się od PCA podejściem do wariancji.

- **PCA** analizuje **wariancję całkowitą** (wszystkie informacje w zmiennych, łącznie z szumem). Na przekątnej macierzy korelacji są jedynki.
- **FA** analizuje tylko **wariancję wspólną** (część zmienności wynikającą z powiązań między zmiennymi). Zakłada istnienie ukrytych (latentnych) przyczyn – **czynników**.

2.3 Model matematyczny FA

Zmienna obserwowalna X_i jest sumą liniową czynników wspólnych i czynnika swoistego:

$$X_i = a_{i1}F_1 + a_{i2}F_2 + \cdots + a_{ik}F_k + \varepsilon_i$$

gdzie:

- F_j – **Czynniki wspólne** (oddziałują na wiele zmiennych). Są standaryzowane ($E = 0, Var = 1$) i nieskorelowane ze sobą.
- ε_i – **Czynnik swoisty** (unikalny dla zmiennej X_i). Zawiera specyficzną cechę zmiennej oraz błąd pomiaru.
- a_{ij} – **Ładunki czynnikowe** (Factor Loadings). Są to współczynniki korelacji między zmienną X_i a czynnikiem F_j.

2.4 Dekompozycja wariancji (Kluczowy wzór!)

Dla zmiennych standaryzowanych (gdzie $Var(X_i) = 1$), wariancję dzielimy na dwie części:

$$1 = h_i^2 + \psi_i$$

- h_i^2 – **Zasoby zmienności wspólnej (Communality):** Część wariancji wyjaśniona przez czynniki wspólne.

$$h_i^2 = \sum_{j=1}^{k} a_{ij}^2$$

- ψ_i – **Wariancja swoista (Specific Variance):** Część wariancji, której czynniki nie wyjaśniają.

2.5 Interpretacja

- **Kwadrat ładunku (a_{ij}^2):** Mówi, jaki procent wariancji zmiennej X_i wyjaśnia czynnik F_j.
- **Wariancja czynnika F_j:** Suma kwadratów ładunków w kolumnie ($V_j = \sum_{i=1}^{p} a_{ij}^2$). Mówi o „sile" danego czynnika.

Wykład 3: Wykład 3: Analiza Czynnikowa (Metody i Selekcja)

3.1 Kluczowy problem: Szacowanie zmienności wspólnej

Aby przeprowadzić Analizę Czynnikową, musimy najpierw znać wartości na przekątnej zredukowanej macierzy korelacji ($\tilde{R}$), czyli zasoby zmienności wspólnej (h_i^2). Ponieważ ich nie znamy (to błędne koło), musimy je oszacować.

3.1.1 Metody szacowania h_i^2

1. **Metoda triad (Dla jednego czynnika):** Jeśli model zakłada istnienie jednego dominującego czynnika, h_i^2 można oszacować, biorąc trójki zmiennych silnie skorelowanych.

$$h_i^2 = \frac{r_{ij}^* r_{ik}^*}{r_{jk}}$$

   gdzie r^* to korelacje o największych modułach.
2. **Metoda najsilniejszej korelacji:** Proste przybliżenie: $h_i^2 = \max_{j \neq i} |r_{ij}|$.

3. **Metoda korelacji wielorakiej (Najważniejsza):** Traktujemy zmienną X_i jako zmienną objaśnianą przez wszystkie pozostałe $p-1$ zmiennych.
$$h_i^2 = R^2_{i \cdot 1 \dots p} = 1 - \frac{|R|}{R_{ii}}$$
gdzie R_{ii} to dopełnienie algebraiczne elementu na przekątnej. Jest to dolne ograniczenie prawdziwej zmienności wspólnej.

3.2 Metody wyodrębniania czynników (Obliczania ładunków)

Gdy mamy już oszacowaną macierz $\tilde{R}$ (z h_i^2 na przekątnej), szukamy ładunków czynnikowych.

3.2.1 1. Metoda Centroidalna

Geometryczna interpretacja: Pierwsza oś układu przechodzi przez środek ciężkości (centroid) pęku wektorów reprezentujących zmienne.
- **Wzór na ładunek 1. czynnika:**
$$a_{i1} = \frac{\sum_j r_{ij}}{\sqrt{\sum_i \sum_j r_{ij}}} = \frac{T_i}{\sqrt{T}}$$
(Suma korelacji zmiennej i przez pierwiastek z sumy wszystkich korelacji).
- **Problem „znoszenia się" (Odbicie znaków):** Jeśli w macierzy reszt (po wyodrębnieniu czynnika) sumy korelacji są bliskie zera, stosuje się procedurę **odbicia znaków**.
  - Zmieniamy znak wektorów tak, by suma korelacji była maksymalna dodatnia.
  - Znak ostatecznego ładunku zależy od liczby wykonanych odbić (parzysta = bez zmian, nieparzysta = zmiana znaku).

3.2.2 2. Metoda osi głównych

Jest to odpowiednik metody PCA, ale stosowany na zredukowanej macierzy korelacji $\tilde{R}$ (z h_i^2 na przekątnej, a nie jedynkami).
- Rozwiązujemy równanie charakterystyczne $|\tilde{R} - \lambda I| = 0$.
- Ładunki to przeskalowane współrzędne wektorów własnych.
- Wariancja wyjaśniona przez czynnik F_j:
$$V_j = \sum_{i=1}^{p} a_{ij}^2$$
Udział procentowy w wariancji całkowitej: $\frac{V_j}{p} \cdot 100\%$.

3.3 Kryteria wyboru liczby czynników

Decyzja, ile czynników (k) pozostawić w modelu, jest kluczowa dla interpretacji.
- **Kryterium Kaisera:** Bierzemy tylko czynniki, dla których wartość własna $\lambda_j > 1$. (Logika: czynnik musi wyjaśniać więcej niż jedna zmienna standaryzowana).
- **Kryterium Cattella (Osypiska):** Analiza wykresu wartości własnych. Szukamy punktu, gdzie wykres przestaje gwałtownie spadać i zaczyna się „wypłaszczać" (osypisko). Punkt ten oddziela istotne czynniki od szumu.
- **Kryterium procentowe:** Wybieramy tyle czynników, by łącznie wyjaśniały np. 75-80% wariancji całkowitej zmiennych.
- **Kryterium „połowy":** Liczba czynników nie powinna przekraczać połowy liczby zmiennych ($k \leq p/2$).

Wykład 4: Wykład 4: Analiza Korespondencji (CA) – Część 1

4.1 Cel metody

Analiza korespondencji (Correspondence Analysis) służy do badania współwystępowania kategorii dwóch lub więcej zmiennych **jakościowych** (nominalnych lub porządkowych). Pozwala na graficzną prezentację zależności w przestrzeni o niskim wymiarze (mapa percepcyjna).

4.2 Tablica kontyngencji i podstawowe miary

Dane wejściowe to tablica krzyżowa (kontyngencji) N o wymiarach $r \times c$, gdzie n_{ij} to liczebność wystąpień pary kategorii (x_i, y_j).

4.2.1 Test niezależności χ^2 (Chi-kwadrat)

Zanim zaczniemy analizę, musimy sprawdzić, czy cechy są w ogóle zależne.
- **Hipoteza H_0:** Cechy są niezależne.
- **Liczebności oczekiwane (e_{ij}):** $e_{ij} = \frac{n_{i\cdot} \cdot n_{\cdot j}}{n}$ (iloczyn sum brzegowych przez ogólną liczebność).
- **Statystyka testowa:**

$$\chi^2 = \sum_{i=1}^{r} \sum_{j=1}^{c} \frac{(n_{ij} - e_{ij})^2}{e_{ij}}$$

- **Stopnie swobody:** $df = (r-1)(c-1)$.

4.3 Miary siły związku (Haczyk egzaminacyjny)

Samo χ^2 zależy od liczebności próby n, więc nie jest dobrą miarą siły związku. Stosuje się współczynniki unormowane:
1. **Współczynnik V-Craméra:** Najpopularniejszy, przyjmuje wartości z przedziału $[0, 1]$.

$$V = \sqrt{\frac{\chi^2}{n \cdot \min(r-1, c-1)}}$$

2. **Współczynnik T-Czuprowa:**

$$T = \sqrt{\frac{\chi^2}{n\sqrt{(r-1)(c-1)}}}$$

3. **Współczynnik kontyngencji C-Pearsona:**

$$C = \sqrt{\frac{\chi^2}{\chi^2 + n}}$$

4.4 Kluczowe pojęcia w CA

- **Macierz korespondencji (P):** Tablica liczebności względnych ($p_{ij} = n_{ij}/n$).
- **Masy wierszowe (r) i kolumnowe (c):** Sumy brzegowe macierzy P. Odpowiadają one prawdopodobieństwom wystąpienia danej kategorii (ważność kategorii).
- **Profile wierszowe/kolumnowe:** Rozkłady warunkowe. Np. profil wiersza to wiersz macierzy P podzielony przez jego masę (p_{ij}/r_i). Analiza CA to tak naprawdę badanie różnic między tymi profilami.

4.5 Inercja całkowita (λ_{total})

Jest to miara całkowitej zmienności (zależności) w tabeli, analogiczna do wariancji w PCA.

$$\lambda_{total} = \frac{\chi^2}{n} = \sum_{i} \sum_{j} \frac{(p_{ij} - r_i c_j)^2}{r_i c_j}$$

Inercja całkowita jest sumą wartości własnych (inercji głównych) uzyskanych z rozkładu macierzy:

$$\lambda_{total} = \sum_{k} \lambda_k$$

Każda wartość λ_k mówi, jaką część zależności wyjaśnia k-ty wymiar mapy percepcyjnej.

Wykład 5: Wykład 5: Analiza Korespondencji (cz. 2) i Wstęp do Grupowania

5.1 Wybór wymiaru przestrzeni rzutowania w CA

Podobnie jak w PCA, musimy zdecydować, ile wymiarów (osi) uwzględnić na mapie percepcyjnej.
- **Kryterium procentowe:** Sumujemy inercje główne (λ_k) pierwszych osi. Jeśli np. $\frac{\lambda_1+\lambda_2}{\lambda_{total}} > 75\%$, to rzut 2D jest wiarygodny.
- **Kryterium łokcia:** Analiza wykresu osypiska wartości własnych.
- **Kryterium Greenacre'a:** Wybieramy te osie, dla których $\lambda_k > \frac{1}{q}$, gdzie q to liczba zmiennych.

5.2 Interpretacja Mapy Percepcyjnej (Kluczowe dla zadań)

Na wykresie znajdują się punkty reprezentujące kategorie obu cech (wiersze i kolumny).
- **Bliskość środka układu (0,0):** Kategoria ma profil zbliżony do profilu **przeciętnego** (średniego). Nie wnosi wiele do różnicowania danych.
- **Odległość od środka:** Im dalej punkt leży od początku układu, tym silniejszy jest jego wkład w zależność (większa inercja).
- **Bliskość dwóch punktów tej samej cechy:** Kategorie są do siebie podobne (mają podobne profile).
- **Bliskość punktów różnych cech:** Oznacza silne powiązanie (częste współwystępowanie) tych kategorii.
- **Kąt (Cosinus):** Mały kąt między wektorem punktu a osią główną oznacza, że ta oś dobrze wyjaśnia zmienność tej kategorii.

5.2.1 Efekt Guttmana (Efekt podkowy)

Jeśli na wykresie punkty układają się w charakterystyczny kształt paraboli (podkowy), oznacza to, że istnieje **jedna dominująca oś** (zmienność jest w istocie jednowymiarowa), a druga oś jest jej funkcją nieliniową. Często można wtedy zredukować analizę do 1 wymiaru.

5.3 Wstęp do Analizy Skupień (Clustering)

Celem jest podział zbioru obiektów na rozłączne grupy (skupienia, klastry) tak, aby:
- Obiekty wewnątrz grupy były do siebie **jak najbardziej podobne** (homogeniczność).
- Obiekty z różnych grup były od siebie **jak najbardziej różne** (heterogeniczność).

5.4 Miary odległości (niepodobieństwa)

Aby grupować, musimy mierzyć odległość między obiektami x i y.
1. **Odległość Euklidesowa:** $d(x,y) = \sqrt{\sum(x_i - y_i)^2}$. (Najbardziej intuicyjna, „linia prosta").
2. **Odległość Miejska (Manhattan):** $d(x,y) = \sum |x_i - y_i|$. (Suma różnic po współrzędnych, jak poruszanie się po mieście z przecznicami).
3. **Odległość Czebyszewa:** $d(x,y) = \max_i |x_i - y_i|$. (Maksymalna różnica na dowolnym wymiarze).
4. **Odległość Minkowskiego:** Uogólnienie powyższych: $d(x,y) = (\sum |x_i - y_i|^p)^{1/p}$.
5. **Odległość Mahalanobisa:** Uwzględnia korelacje między zmiennymi (kowariancję). Usuwa wpływ skali i korelacji.

5.4.1 Odległości dla danych binarnych

Dla zmiennych 0-1 tworzymy tablicę dopasowań (a: 1-1, b: 1-0, c: 0-1, d: 0-0).
- **Odległość Hamminga:** Liczba niezgodnych bitów ($b + c$).
- **Współczynnik Jaccarda (dla atrybutów asymetrycznych):** Ignoruje dopasowania „0-0" (d).

$$d_{Jaccard} = 1 - \frac{a}{a+b+c}$$

Stosujemy, gdy „1" jest rzadkie i ważne, a „0" powszechne i mało informacyjne.

Wykład 6: Wykład 6: Metody Grupowania (Clustering) i Text Mining

6.1 Podobieństwo dokumentów tekstowych

W analizie tekstu (Text Mining) standardowa odległość euklidesowa często się nie sprawdza (ze względu na rzadkość macierzy). Stosuje się inne miary:
1. **Miara cosinusowa:** Bazuje na kącie między wektorami częstości słów.

$$D(X_i, X_j) = 1 - \cos(\alpha) = 1 - \frac{X_i \circ X_j}{|X_i| \cdot |X_j|}$$

   Wartości z przedziału $[0, 1]$. Wynik bliski 0 oznacza dokumenty bardzo podobne (kąt bliski 0).
2. **Odległość Tanimoto:** Rozszerzenie miary Jaccarda dla danych ciągłych.

$$D(X_i, X_j) = 1 - \frac{X_i \circ X_j}{|X_i|^2 + |X_j|^2 - X_i \circ X_j}$$

3. **Odległość Levensteina (edycyjna):** Dla napisów. Jest to minimalna liczba operacji prostych (wstawienie, usunięcie, zamiana znaku) potrzebnych do przekształcenia jednego łańcucha w drugi.

6.2 Klasyfikacja metod grupowania

- **Metody hierarchiczne:** Tworzą strukturę drzewiastą (dendrogram). Nie wymagają podania liczby grup na starcie.
  - **Aglomeracyjne („bottom-up"):** Na początku każdy obiekt to osobne skupienie. W kolejnych krokach łączymy najbardziej podobne pary aż do uzyskania 1 grupy.
  - **Podziałowe („top-down"):** Na początku 1 wielka grupa. W kolejnych krokach dzielimy ją na mniejsze.
- **Metody niehierarchiczne (podziałowe/partycyjne):** Np. k-średnich. Wymagają podania k. Tworzą płaski podział. Obiekty mogą zmieniać przynależność w trakcie działania algorytmu.

6.3 Metody łączenia skupień (Linkage)

W metodach hierarchicznych musimy mierzyć odległość między dwiema *grupami* (a nie tylko punktami).

- **Metoda najbliższego sąsiada (Single Linkage):** Odległość to minimum odległości par obiektów (d_{min}). Ma tendencję do tworzenia łańcuchów (efekt „chaining").
- **Metoda najdalszego sąsiada (Complete Linkage):** Odległość to maksimum (d_{max}). Tworzy zwarte skupienia, wrażliwa na odstające.
- **Metoda środków ciężkości (Centroid):** Odległość między centroidami (średnimi) grup.
- **Metoda Warda (Kluczowa!):** Nie minimalizuje odległości, lecz **przyrost wariancji wewnątrzskupieniowej** (ESS). Na każdym kroku łączy te grupy, których połączenie najmniej zwiększy „bałagan" wewnątrz klastrów.

6.4 Gdzie uciąć dendrogram? (Decyzja o liczbie grup)

- **Metoda wizualna:** W miejscu, gdzie następuje największy „skok" odległości wiązania (pionowa linia na dendrogramie jest najdłuższa).
- **Reguła Mojeny:** Kryterium statystyczne.

$$d_{i+1} > \bar{d} + k \cdot s_d$$

Ucinamy, gdy odległość łączenia przekroczy średnią ($\bar{d}$) plus k odchyleń standardowych (s_d) z poprzednich łączeń. Standardowo $k \in [2.75, 3.5]$.

6.5 Algorytm k-średnich (k-means)

1. Wybierz losowo k punktów jako początkowe centra (centroidy).
2. Przypisz każdy obiekt do **najbliższego** centrum.
3. Oblicz nowe centra (średnie arytmetyczne obiektów w każdej grupie).
4. Powtarzaj kroki 2-3 aż centra przestaną się zmieniać (algorytm zbiegnie).

Wada: Wrażliwy na dobór początkowych punktów (może utknąć w minimum lokalnym) oraz na wartości odstające.

Wykład 7: Wykład 7: Klasyfikacja – Drzewa Decyzyjne (cz. 1)

7.1 Wprowadzenie do klasyfikacji

Celem klasyfikacji jest zbudowanie modelu (klasyfikatora) na podstawie danych historycznych (**zbiór treningowy**), który pozwoli przypisać nowy obiekt do jednej ze zdefiniowanych klas (wartości **atrybutu decyzyjnego**).

7.2 Struktura danych

- **Zbiór treningowy** D**:** Zbiór obiektów o znanych klasach.
- **Atrybuty warunkowe** $(A_1, \ldots, A_s)$**:** Cechy opisujące obiekt (zmienne niezależne).
- **Atrybut decyzyjny** (C)**:** Zmienna kategoryczna wskazująca klasę obiektu (np. $C_1, \ldots, C_m$).

7.3 Budowa drzewa decyzyjnego

Drzewo składa się z korzenia, węzłów (testów na atrybutach) i liści (decyzji). Kluczowym problemem jest wybór atrybutu, który najlepiej dzieli zbiór danych w danym węźle. Algorytmy takie jak ID3 czy C4.5 wykorzystują w tym celu pojęcie **Entropii**.

7.4 Entropia Shannona (H)

Jest to miara nieuporządkowania (chaosu) w zbiorze danych. Im wyższa entropia, tym trudniej przewidzieć klasę obiektu.

- **Wzór na entropię zbioru** D**:**

$$H(D) = -\sum_{i=1}^{m} p_i \log_2 p_i$$

gdzie p_i to prawdopodobieństwo (częstość względna $\frac{n_i}{n}$) wystąpienia i-tej klasy decyzyjnej.

- **Własności:**
  - $H(D) = 0$: Zbiór jest jednorodny (wszystkie obiekty należą do tej samej klasy).
  - $H(D)$ jest maksymalne, gdy liczebności wszystkich klas są równe.

7.5 Entropia atrybutu ($H(A)$)

Mierzy średnie nieuporządkowanie po podziale zbioru D według wartości atrybutu A. Jest to średnia ważona entropii podgrup.

$$H(A) = \sum_{j=1}^{r} \frac{n_j}{n} H(D_j)$$

gdzie:
- D_j to podzbiór obiektów, dla których atrybut A przyjmuje wartość a_j.
- $\frac{n_j}{n}$ to waga (udział liczebności podgrupy w całości).
- $H(D_j)$ to entropia wewnątrz tej podgrupy.

7.6 Zysk Informacyjny (Information Gain)

Jest to miara przyrostu informacji (redukcji niepewności) uzyskanego dzięki podziałowi zbioru według atrybutu A.

$$Gain(A) = H(D) - H(A)$$

Algorytm wyboru atrybutu:
1. Oblicz entropię całego zbioru $H(D)$.
2. Dla każdego atrybutu A_k oblicz entropię po podziale $H(A_k)$.
3. Oblicz zysk $Gain(A_k)$.
4. Wybierz ten atrybut, który daje **maksymalny zysk** (czyli minimalną entropię po podziale).

Wykład 8: Wykład 8: Klasyfikacja – Drzewa (Gini) i Bayes

8.1 Indeks Giniego (Miara nieczystości)

Alternatywą dla Entropii w budowie drzew decyzyjnych jest Indeks Giniego. Jest on stosowany w popularnych algorytmach takich jak **CART** (Classification and Regression Trees) oraz **SPRINT**.
- **Definicja:** Dla zbioru D indeks Giniego wynosi:

$$Gini(D) = 1 - \sum_{i=1}^{m} p_i^2$$

  gdzie p_i to prawdopodobieństwo wystąpienia i-tej klasy.
- **Interpretacja:**
  - $Gini(D) = 0$: Zbiór idealnie czysty (wszystkie elementy jednej klasy).
  - $Gini(D)$ rośnie wraz ze wzrostem różnorodności klas (maksymalny dla równego rozkładu).

8.2 Kryterium podziału (Indeks podziału)

Algorytmy oparte na Ginim (np. CART) tworzą zawsze **drzewa binarne** (każdy węzeł ma dokładnie 2 gałęzie). Szukamy takiego podziału atrybutu, który minimalizuje średnią ważoną nieczystości.
- **Indeks podziału (Split Index):** Jeśli atrybut A dzieli zbiór D na dwa podzbiory D_1 i D_2:

$$Gini_{split}(D) = \frac{|D_1|}{|D|} Gini(D_1) + \frac{|D_2|}{|D|} Gini(D_2)$$

- **Zysk Giniego:** Wybieramy podział, który maksymalizuje redukcję nieczystości:

$$Gain_{Gini}(A) = Gini(D) - Gini_{split}(D)$$

  Jest to równoważne minimalizacji $Gini_{split}$.

8.3 Klasyfikacja Bayesowska

Podejście probabilistyczne, oparte na Twierdzeniu Bayesa. Celem jest obliczenie prawdopodobieństwa, że obiekt X należy do klasy C_i pod warunkiem zaobserwowania jego cech.

$$P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(X)}$$

- $P(C_i|X)$ – Prawdopodobieństwo a posteriori (szukane).
- $P(X|C_i)$ – Prawdopodobieństwo wiarygodności (likelihood): jak prawdopodobne jest wystąpienie cech X w klasie C_i.
- $P(C_i)$ – Prawdopodobieństwo a priori (częstość występowania klasy w bazie).

Decyzję podejmujemy zgodnie z zasadą **MAP (Maximum A Posteriori)**: wybieramy klasę o najwyższym $P(C_i|X)$.

8.4 „Naiwny" Klasyfikator Bayesowski (Naive Bayes)

Obliczenie $P(X|C_i)$ jest trudne, gdy X składa się z wielu atrybutów $(A_1, \ldots, A_s)$, ze względu na ogromną liczbę kombinacji.

- **Założenie „naiwności":** Zakładamy, że atrybuty są **warunkowo niezależne** względem klasy decyzyjnej.
- **Wzór uproszczony:**

$$P(X|C_i) = P(A_1, \ldots, A_s|C_i) = \prod_{j=1}^{s} P(A_j = x_j|C_i)$$

  Zamiast jednego wielkiego prawdopodobieństwa, mnożymy prawdopodobieństwa pojedynczych cech (łatwe do policzenia z bazy).
- **Ostateczna reguła decyzyjna:** Przypisz obiekt do klasy C_i, która maksymalizuje wartość:

$$P(C_i) \cdot \prod_{j=1}^{s} P(A_j = x_j|C_i)$$

Wykład 9: Wykład 9: Klasyfikacja (cz. 3) i Wstęp do Asocjacji

9.1 Problem „częstości zero" (Zero Frequency Problem)

W „naiwnym" klasyfikatorze Bayesowskim prawdopodobieństwo $P(X|C_i)$ jest iloczynem prawdopodobieństw cząstkowych $\prod P(A_j|C_i)$.

- **Problem:** Jeśli w zbiorze treningowym dla danej klasy C_i ani razu nie wystąpiła konkretna wartość atrybutu A_j (czyli $n_{ij} = 0$), to estymowane prawdopodobieństwo wynosi 0.
- **Skutek:** Zero zeruje cały iloczyn, co **bezpowrotnie eliminuje** klasę C_i jako kandydata, niezależnie od wartości innych atrybutów.

9.1.1 Rozwiązanie: Wygładzanie Laplace'a (Laplace Smoothing)

Aby uniknąć zer, dodajemy „sztuczne" zliczenia do licznika i mianownika.

$$P(A_j = x_j|C = C_i) = \frac{n_{ij} + \lambda}{n_i + \lambda \cdot m_j}$$

gdzie:

- n_{ij} – liczba przypadków z cechą x_j w klasie C_i.
- n_i – liczebność klasy C_i.
- m_j – liczba wszystkich możliwych wartości atrybutu A_j.
- λ – współczynnik wygładzania (zazwyczaj $\lambda = 1$, wtedy jest to klasyczny estymator Laplace'a).

9.2 Metoda k-najbliższych sąsiadów (k-NN)

Metoda ta różni się fundamentalnie od drzew czy Bayesa.

- **Leniwe uczenie (Lazy Learning):** Algorytm nie buduje modelu (funkcji) w fazie uczenia. Zapamiętuje cały zbiór treningowy. Klasyfikacja odbywa się w momencie zapytania (tryb „on-line").
- **Algorytm:**
  1. Oblicz odległość (np. euklidesową) nowego obiektu od wszystkich obiektów w bazie.
  2. Wybierz k obiektów o najmniejszej odległości (sąsiedzi).
  3. Przypisz obiekt do klasy, która jest najliczniej reprezentowana wśród sąsiadów (głosowanie większościowe).
- Wymaga atrybutów ilościowych (aby liczyć odległość) i jest kosztowna obliczeniowo przy dużych bazach.

9.3 Reguły Asocjacyjne (Analiza Koszykowa)

Celem jest odkrywanie współwystępowania obiektów w transakcjach (np. „Klienci kupujący chleb i masło, kupują też mleko"). Reguła ma postać implikacji: $X \to Y$ (gdzie X, Y to zbiory produktów).

9.3.1 Kluczowe miary oceny reguł (Pewniaki egzaminacyjne!)

1. **Wsparcie (Support):** Prawdopodobieństwo, że w losowej transakcji wystąpią **jednocześnie** produkty ze zbioru X i Y.
$$supp(X \to Y) = P(X \cup Y) = \frac{\text{Liczba transakcji zawierających } X \cup Y}{\text{Liczba wszystkich transakcji}}$$
2. **Ufność (Confidence):** Prawdopodobieństwo warunkowe, że wystąpi Y, pod warunkiem że wystąpiło X. Mówi o „sile" reguły.
$$conf(X \to Y) = P(Y|X) = \frac{supp(X \cup Y)}{supp(X)}$$

9.3.2 Reguły silne

Regułę nazywamy **silną**, jeśli spełnia minimalne progi określone przez użytkownika:

$$supp \geq min\_supp \quad \text{oraz} \quad conf \geq min\_conf$$

Wykład 10: Wykład 10: Algorytmy Asocjacji – Apriori i FP-Growth

10.1 Ogólny schemat odkrywania reguł

Proces ten składa się z dwóch kroków:
1. **Znalezienie zbiorów częstych:** Wyznaczenie wszystkich podzbiorów produktów, których wsparcie przekracza próg *min_ supp*. Jest to etap najbardziej kosztowny obliczeniowo (2^k możliwych podzbiorów).
2. **Generowanie reguł:** Z każdego zbioru częstego X generuje się reguły postaci $A \to (X \setminus A)$, które spełniają warunek *min_ conf*. Ten etap jest obliczeniowo prosty.

10.2 Zasada Antymonotoniczności (Apriori Property)

Jest to fundament algorytmu Apriori, pozwalający ograniczyć przestrzeń poszukiwań.
- **Twierdzenie:** Jeśli zbiór X jest częsty, to **każdy** jego podzbiór musi być częsty.
- **Wniosek (Przycinanie):** Jeśli jakikolwiek podzbiór zbioru X jest **rzadki** (nieczęsty), to cały zbiór X też musi być rzadki. Dzięki temu możemy od razu odrzucić takie zbiory bez liczenia ich wsparcia.

10.3 Algorytm Apriori (Metoda pozioma)

Działa iteracyjnie, generując kandydatów o coraz większej liczności ($k \to k+1$).
1. Znajdź 1-elementowe zbiory częste (L_1).
2. **Złączanie (Join):** Połącz zbiory z L_{k-1}, aby utworzyć kandydatów C_k (kandydaci o długości k).
3. **Przycinanie (Prune):** Usuń z C_k tych kandydatów, którzy zawierają rzadkie podzbiory długości $k-1$ (na podstawie zasady antymonotoniczności).
4. **Weryfikacja:** Przeskanuj bazę danych, aby policzyć rzeczywiste wsparcie kandydatów i wyłonić L_k.

Wady: Wymaga wielokrotnego skanowania całej bazy danych (tyle razy, ile wynosi długość najdłuższego wzorca) i generuje ogromną liczbę kandydatów.

10.4 Algorytm FP-Growth (Frequent Pattern Growth)

Metoda ta eliminuje konieczność generowania kandydatów. Opiera się na skompresowanej strukturze danych zwanej **FP-drzewem** (FP-Tree).

10.4.1 Budowa FP-Drzewa

1. Przeskanuj bazę raz, aby znaleźć częstości produktów. Posortuj je malejąco (L-list).
2. Przeskanuj bazę drugi raz. Każdą transakcję posortuj zgodnie z listą L i wstaw do drzewa jako ścieżkę. Jeśli ścieżka (prefiks) już istnieje, zwiększ liczniki węzłów (współdzielenie prefiksów).
3. Utwórz **Header Table** – tabelę łączącą wszystkie wystąpienia tego samego produktu w drzewie.

10.4.2 Eksploracja (Metoda „Dziel i rządź")

Algorytm analizuje drzewo od dołu (od najrzadszych elementów w Header Table):
1. Dla ustalonego elementu (sufiksu) znajdź wszystkie ścieżki prowadzące do niego w górę drzewa (**Warunkowa baza wzorców**).
2. Zbuduj z tych ścieżek **Warunkowe FP-Drzewo**.
3. Rekurencyjnie powtarzaj proces dla tego warunkowego drzewa, znajdując wzorce częste.

Zalety: Tylko 2 skany bazy danych, brak generowania kandydatów, znacznie szybszy od Apriori dla dużych baz.

Wykład 11: Wykład 11: Eclat i Zbiory Domknięte

11.1 Algorytm Eclat (Eksploracja Pionowa)

W przeciwieństwie do Apriori i FP-Growth, które pracują na formacie poziomym (Transakcja → lista produktów), Eclat (Equivalence Class Transformation) wykorzystuje **format pionowy** danych.

- **Struktura danych:** Każdemu produktowi przypisana jest lista identyfikatorów transakcji (**TID-list**), w których on występuje.
- **Liczenie wsparcia:** Wsparcie dla zestawu produktów oblicza się poprzez **przecięcie (intersekcję)** ich list TID.

$$supp(XY) = |TID(X) \cap TID(Y)|$$

- **Zaleta:** Szybkie obliczenia na listach, brak konieczności skanowania całej bazy dla sprawdzenia kandydata.
- **Wada:** „Wąskim gardłem" algorytmu jest generowanie zbiorów 1- i 2-elementowych (listy są wtedy bardzo długie). Często łączy się go z Apriori na początkowym etapie.

11.2 Kompresja wyników: Zbiory Domknięte i Maksymalne

Aby zredukować ogromną liczbę generowanych reguł (które często są nadmiarowe), wprowadza się pojęcia zbiorów skompresowanych.

11.2.1 Zbiór Domknięty (Closed Itemset)

Zbiór częsty X jest **domknięty**, jeżeli nie istnieje żaden jego nadzbiór Y (zbiór zawierający X), który miałby **takie samo wsparcie** jak X.

- Oznacza to, że X jest „największym" zbiorem występującym w danej liczbie transakcji.
- **Kompresja bezstratna:** Zbiory domknięte pozwalają na odtworzenie dokładnego wsparcia wszystkich swoich podzbiorów (z własności antymonotoniczności, podzbiór musi mieć wsparcie co najmniej takie samo, a jeśli nie jest domknięty – to identyczne jak jego domknięcie).

11.2.2 Zbiór Maksymalny (Maximal Itemset)

Zbiór częsty X jest **maksymalny**, jeżeli żaden jego nadzbiór nie jest **częsty** (czyli dodanie jakiegokolwiek elementu powoduje spadek wsparcia poniżej progu min_supp).

- **Kompresja stratna:** Zbiory maksymalne dają największą redukcję liczby wyników, ale tracimy informację o dokładnych wartościach wsparcia ich podzbiorów (wiemy tylko, że są częste).

Relacja zawierania:

$$\text{Zbiory Maksymalne} \subseteq \text{Zbiory Domknięte} \subseteq \text{Wszystkie Zbiory Częste}$$

11.3 Algorytm CLOSET+ (Szukanie zbiorów domkniętych)

Jest to algorytm dedykowany do znajdowania domkniętych zbiorów częstych.

- Wykorzystuje strukturę **FP-drzewa** (podobnie jak FP-Growth).
- **Technika hybrydowa:**
  - Dla danych **gęstych** stosuje metodę projekcji „od dołu do góry" (bottom-up).
  - Dla danych **rzadkich** stosuje metodę „od góry do dołu" (top-down).
- **Mechanizmy optymalizacji:**
  - **Łączenie elementów:** Jeśli element Y pojawia się w każdej transakcji co X, to można je połączyć.
  - **Przycinanie podzbiorów:** Jeśli zbiór X jest podzbiorem już znalezionego zbioru domkniętego Y o tym samym wsparciu, to X i jego potomkowie nie mogą być nowymi zbiorami domkniętymi (można uciąć gałąź).

Wykład 12: Wykład 12: Max-Miner i Wzorce Sekwencji

12.1 Algorytm Max-Miner

Jest to algorytm zaprojektowany specjalnie do efektywnego znajdowania **maksymalnych** zbiorów częstych (bez generowania wszystkich zbiorów częstych).

- **Struktura:** Wykorzystuje strukturę **SE-drzewa** (Set-Enumeration Tree). Każdy węzeł reprezentuje grupę kandydującą g.
- **Definicje w węźle:**
  - **Głowa** $h(g)$**:** Elementy, które na pewno znajdują się w danym zbiorze (prefiks).
  - **Ogon** $t(g)$**:** Elementy, które mogą zostać dodane do głowy (rozszerzenia).
- **Strategia „Look-ahead" (Wybieganie w przyszłość):** To kluczowa optymalizacja. Zanim algorytm zacznie tworzyć podzbiory, sprawdza wsparcie dla sumy wszystkich elementów w grupie ($h(g) \cup t(g)$).

- **Reguła odcinania:** Jeśli zbiór $h(g) \cup t(g)$ jest częsty, to jest on **maksymalnym zbiorem częstym** w tej gałęzi. Algorytm od razu go zwraca i nie musi sprawdzać żadnych podzbiorów (ucinamy całą gałąź). Oszczędność czasu jest ogromna.

12.2 Odkrywanie Wzorców Sekwencji (Sequential Pattern Mining)

W przeciwieństwie do klasycznych reguł asocjacyjnych, tutaj istotna jest **kolejność zdarzeń (czas)**.
- **Dane wejściowe:** Baza sekwencji. Każdy obiekt (klient) ma przypisaną uporządkowaną listę transakcji (zestawów zakupów) w czasie: $\langle T_1, T_2, \ldots, T_n \rangle$.
- **Definicja Podsekwencji (Subsequence):** Sekwencja S zawiera się w sekwencji Q, jeśli elementy S można „zmapować" na elementy Q z zachowaniem kolejności.
- **Ważne:** Elementy nie muszą występować bezpośrednio po sobie (luki są dozwolone).
  – Przykład: Sekwencja $\langle (A), (C) \rangle$ zawiera się w $\langle (A, B), (D), (C, E) \rangle$, bo A było wcześniej niż C.
- **Wsparcie sekwencji:** Procent sekwencji wejściowych (klientów), które zawierają daną podsekwencję.

12.3 Algorytm AprioriAll (GSP)

Jest to adaptacja algorytmu Apriori do danych sekwencyjnych.
1. **Sortowanie:** Baza jest sortowana według ID klienta i czasu transakcji.
2. **L-Itemsets (Zbiory częste):** Najpierw znajdujemy klasyczne zbiory częste wewnątrz pojedynczych transakcji (ignorując czas).
3. **Transformacja:** Zamieniamy oryginalne transakcje na zbiory znalezione w kroku 2 (upraszczamy dane).
4. **Sekwencjonowanie (Hash Tree):**
   - Łączymy sekwencje długości $k-1$ w kandydatów długości k.
   - **Warunek łączenia:** Dwie sekwencje można połączyć, jeśli po usunięciu pierwszego elementu z pierwszej i ostatniego z drugiej otrzymamy **tę samą** podsekwencję (wymóg spójności czasowej).

Przykład łączenia: $\langle (1, 2), (3) \rangle$ i $\langle (2), (3, 4) \rangle$ można połączyć w $\langle (1, 2), (3, 4) \rangle$, ponieważ środek $\langle (2), (3) \rangle$ jest wspólny.

Wykład 13: Suplement: Detale „Dla pewności"

13.1 Rodzaje Rotacji w Analizie Czynnikowej (FA)

W Wykładzie 3 wspomniano o rotacji, aby ułatwić interpretację. Warto znać nazwy najpopularniejszych metod (na wypadek pytania „wymień rodzaje"):
- **Rotacja Varimax:** Najpopularniejsza. Dąży do tego, aby ładunki były bliskie 1 lub 0 (maksymalizuje wariancję kwadratów ładunków w kolumnach). Jest to rotacja **ortogonalna** (zachowuje kąty proste między czynnikami – czynniki pozostają nieskorelowane).
- **Rotacja Oblimin/Promax:** Rotacje **ukośne**. Dopuszczają korelację między czynnikami, co czasem lepiej oddaje rzeczywistość, ale trudniej się interpretuje.

13.2 Ocena jakości klasyfikacji (Macierz Pomyłek)

W wykładach 7-9 mówiliśmy o budowie modelu. Na egzaminie mogą zapytać, jak ocenić ten model. Jeśli nie było tego wprost na slajdach, to jest to „wiedza domyślna" z tego przedmiotu:
- **Macierz Pomyłek (Confusion Matrix):** Tabela 2×2 (dla 2 klas), zawierająca:
  – TP (True Positive) – poprawnie wykryte pozytywne.
  – TN (True Negative) – poprawnie wykryte negatywne.
  – FP (False Positive) – fałszywy alarm.
  – FN (False Negative) – przeoczenie.
- **Dokładność (Accuracy):** $\frac{TP+TN}{TP+TN+FP+FN}$ (Ogólna skuteczność).
- **Błąd (Error Rate):** $1 - Accuracy$.

13.3 Szczegół do metody k-średnich (Wykład 6)

Pamiętaj o **inicjalizacji**: Algorytm jest wrażliwy na początkowy (losowy) wybór środków.
- Jeśli pechowo wylosujemy środki, algorytm może utknąć w **minimum lokalnym** (nie znajdzie najlepszego podziału).
- Rozwiązanie w praktyce: Uruchamia się algorytm kilkukrotnie i wybiera wynik z najmniejszym błędem wewnątrzgrupowym.