# Statystyczne metody przetwarzania danych

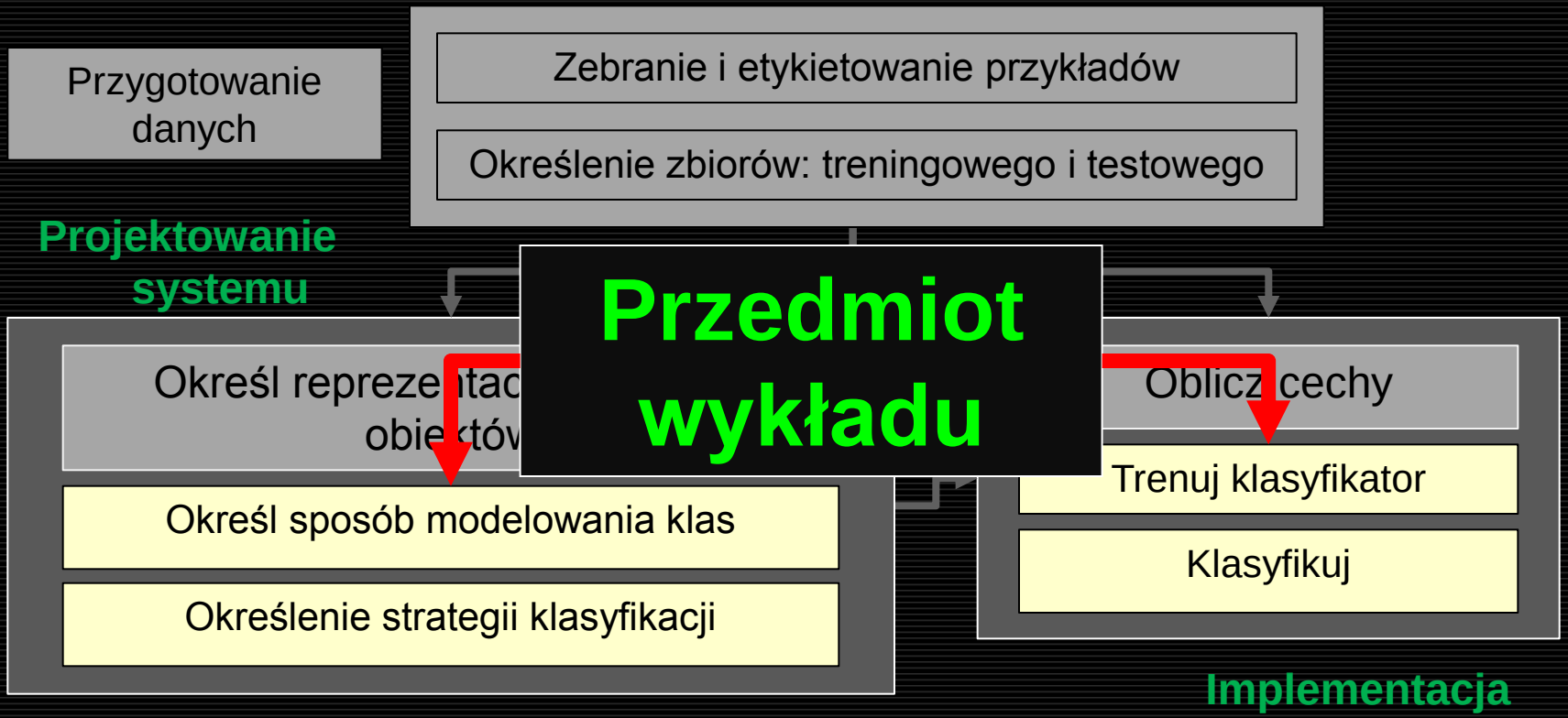
## Klasyfikacja minimalnoodległościowa

*Krzysztof Ślot*

*Instytut Informatyki Stosowanej*
*Politechnika Łódzka*

# Wprowadzenie

- **Etapy procedury rozpoznawania**
  - Przetwarzanie wstępne, obejmuje ekstrakcję obiektu z tla
  - Projektowanie systemu rozpoznawania
  - Klasyfikacja, wykorzystująca wybrane cechy i zbudowane modele klas



# Klasyfikacja danych

- **Przypisanie próbce etykiety klasy**
  - Próbka jest reprezentowana przez wektor cech (w odpowiedniej przestrzeni cech)
  - Modele klas muszą być zbudowane w oparciu o przykłady (definicje klas są nieznane)
  - Przykłady mogą nie posiadać etykiety klasy (klasyfikacja nienadzorowana)

Dane $\mathbf{x}_i, \quad C = \{C_\alpha, C_\beta ... C_\xi\}$ wyznacz: $\nu : \mathbf{x}_i \in C_\nu$

- **Podstawa klasyfikacji**
  - Maksymalizacja podbieństwa między próbką a klasą (minimalizacja różnicy)
  - Posiadanie określonych właściwości

- **Strategia postępowania**
  - Zbuduj ilościowe modele klas – oceń przynależność próbki

# Klasyfikacja danych

- **Strategie klasyfikacji**
  - Ocena podobieństwa:
    - Klasyfikacja minimalnoodległościowa
    - Klasyfikacja probabilistyczna
  - Posiadanie określonych właściwości
    - Klasyfikacja przy użyciu powierzchni decyzyjnych

- **Etapy budowy klasyfikatora**
  - Trening
    - Budowa modeli klas i estymacja ich parametrów (zbiór treningowy)
  - Testowanie
    - Ocena skuteczności klasyfikacji przy użyciu próbek zbioru testowego

- **Proces klasyfikacji (rozpoznawania)**
  - Wyznaczanie przynależności nieznanej próbki dokonana przy użyciu zbudowanego klasyfikatora

# Klasyfikacja minimalno-odległościowa

- **Podstawy**
  - Próbki są punktami w przestrzeni metrycznej
  - Podobieństwo oceniane przez określanie odległości próbki i klasy
  - Zwycięża klasa najbliższa

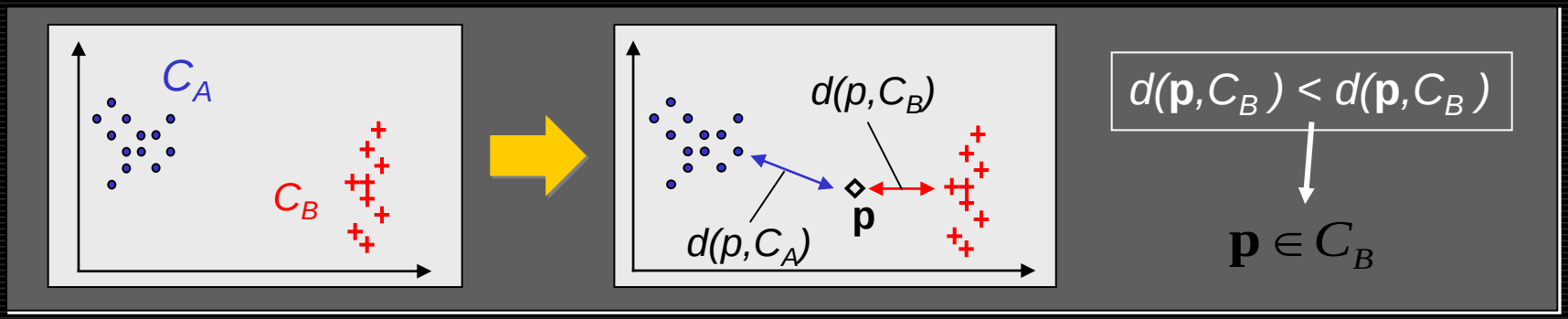
- **Strategie klasyfikacji minimalnoodległościowej**
  - Metoda najbliższego sąsiada (Nearest-Neighbor - NN)
  - Metoda najbliższej średniej (Nearest-Mean - NM)
  - Klasyfikacja k-NN
  - Klasyfikacja k-NM

- **Problemy budowy klasyfikatora**
  - Postać modelu klasy i trening modelu
  - Definicja odległości między próbką a klasą
  - Algorytm wyboru zwycięskiej klasy

# Klasyfikacja NN

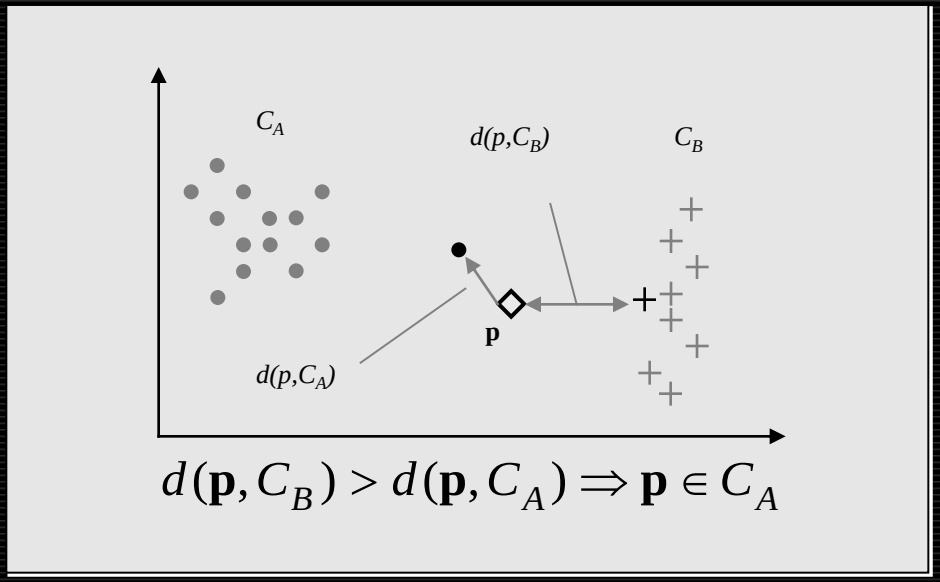
- **Definicja komponentów metody**
  - Model (prototyp) klasy: zapamiętane wszystkie próbki zbioru treningowego (brak procedury uczenia klasyfikatora)
  - Odległość próbki od klasy: najmniejsza z odległości między próbką a elementami klasy

$$k = \arg\left(\min_i \{d(\mathbf{p}, C_i)\}\right), \qquad d(\mathbf{p}, C_i) = \min_j d(\mathbf{p}, C_i^j)$$



# Klasyfikacja NN

- **Właściwości**
  - Prostota koncepcyjna
  - Brak procedury uczenia klasyfikatora
  - Kosztowny obliczeniowo proces klasyfikacji
  - Wymagana duża pamięć do składowania modeli klas
  - Wrażliwość na ‘złe’ przykłady (nieuchronnie obecne w dużych zbiorach)

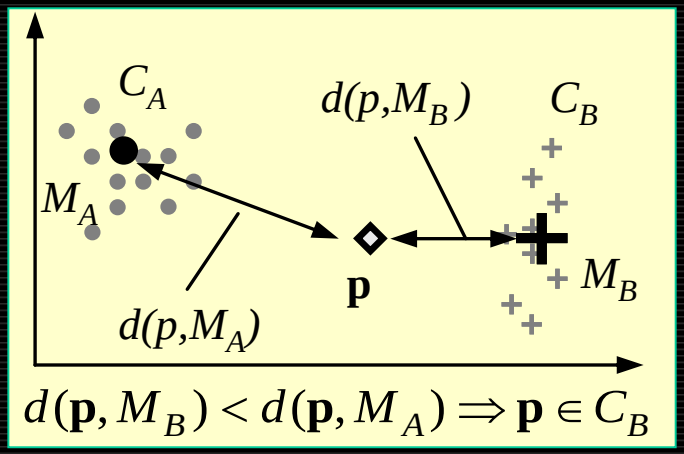


# Klasyfikacja NM

- **Definicja komponentów metody**
  - Model klasy: podstawowe właściwości statystyczne zbioru próbek– średnia, macierz kowariancji (prosty trening)
  - Odległość próbki od klasy: odległość próbki do wartości średniej/ odległość Machalobobisa (wyrażona w jednostkach odchylenia)

$$d(\mathbf{p}, C_i) = d(\mathbf{p}, M_i)$$

$$k = \arg\left(\min_i \{d(\mathbf{p}, M_i)\}\right), \qquad M_i = \frac{1}{N_i}\sum_{j=1}^{N_i} C_i^j$$



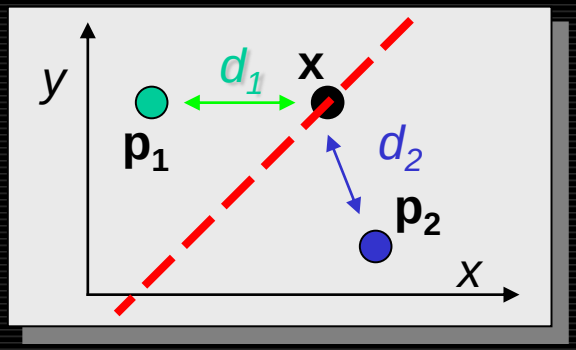

Odległość uwzględniająca rozrzuty (Machalonobisa)

$$d(\mathbf{p}, C_i) = d(\mathbf{p}, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$$

$$d(\mathbf{p}, C_i) = \sqrt{(\mathbf{p} - \boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}^{-1} (\mathbf{p} - \boldsymbol{\mu}_i)}$$

# Klasyfikacja NM

- **Właściwości**
  - Prosty trening
  - Szybka klasyfikacja
  - Mała wrażliwość na błędne przykłady (efekt uśrednienia)
  - Małe zasoby wymagane do zapamiętania modeli klas
  - Niejawne założenie Gaussowskiego modelu klasy: klasyfikacja liniowa




$$d(\mathbf{x},\mathbf{p}_1) = \| \mathbf{x}-\mathbf{p}_1 \| = \mathbf{x}^T\mathbf{x} - 2\mathbf{x}^T\mathbf{p}_1 + \mathbf{p}_1^T\mathbf{p}_1$$

$$d(\mathbf{x},\mathbf{p}_2) = \| \mathbf{x}-\mathbf{p}_2 \| = \mathbf{x}^T\mathbf{x} - 2\mathbf{x}^T\mathbf{p}_2 + \mathbf{p}_2^T\mathbf{p}_2$$

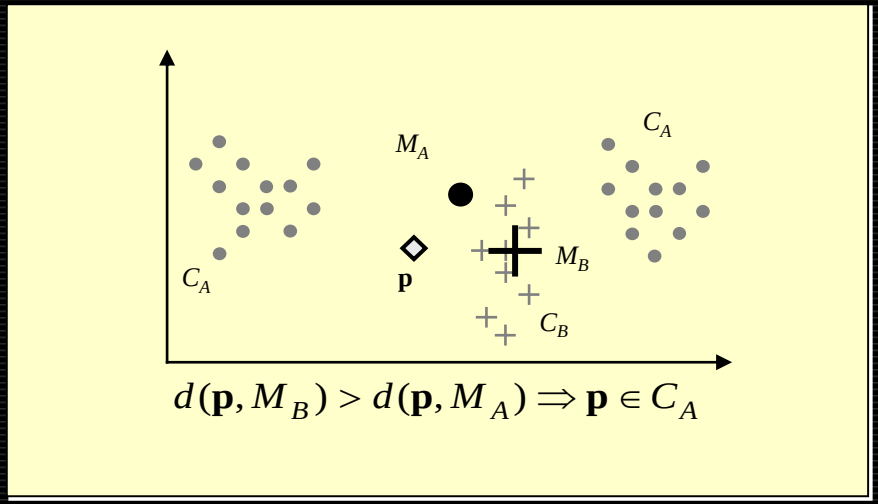
$$d(\mathbf{x},\mathbf{p}_1) = d(\mathbf{x},\mathbf{p}_2) \Rightarrow 2\mathbf{x}^T(\mathbf{p}_2-\mathbf{p}_1) + \mathbf{p}_1^T\mathbf{p}_1 - \mathbf{p}_2^T\mathbf{p}_2 = 0$$

- **NM – klasyfikator liniowy**

$$\mathbf{x}^T\Delta\mathbf{p} + C = 0$$

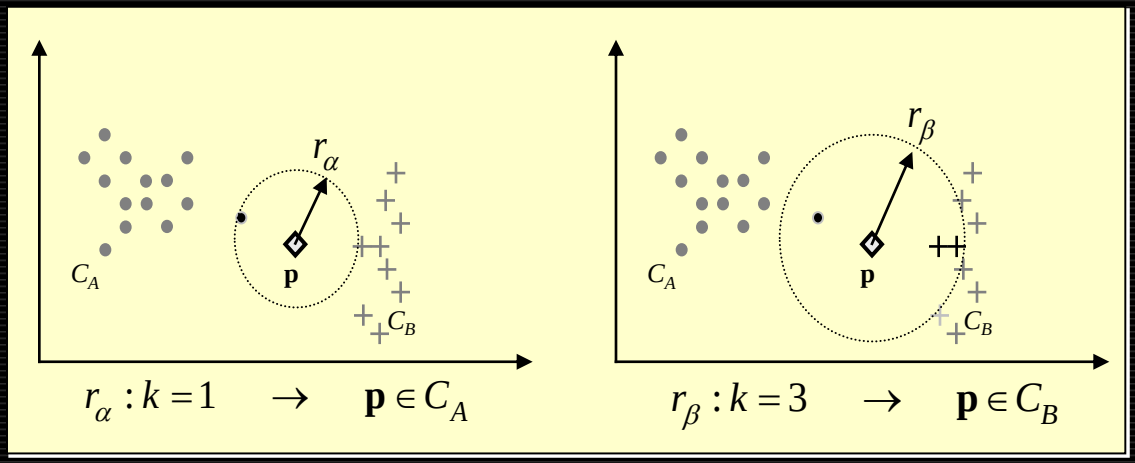
# Klasyfikacja NM

- **Rozkłady wielomodalne**
  - Rozkład jednomodalny: problemy trywialne
  - Rzeczywiste problemy rozpoznawania: wielomodalna reprezentacja klasy
  - NM – błędna klasyfikacja



# Metoda k-NN

- **Definicja komponentów metody**
  - Model klasy : zapamiętane wszystkie próbki zbioru treningowego (brak procedury uczenia klasyfikatora)
  - Odległość próbki od klasy: klasa najliczniej reprezentowana wśród k-zwycięzców (k-najbliższych próbek)
  - Parametr modelu: k – wartość optymalna parametru musi być określona w fazie treningu



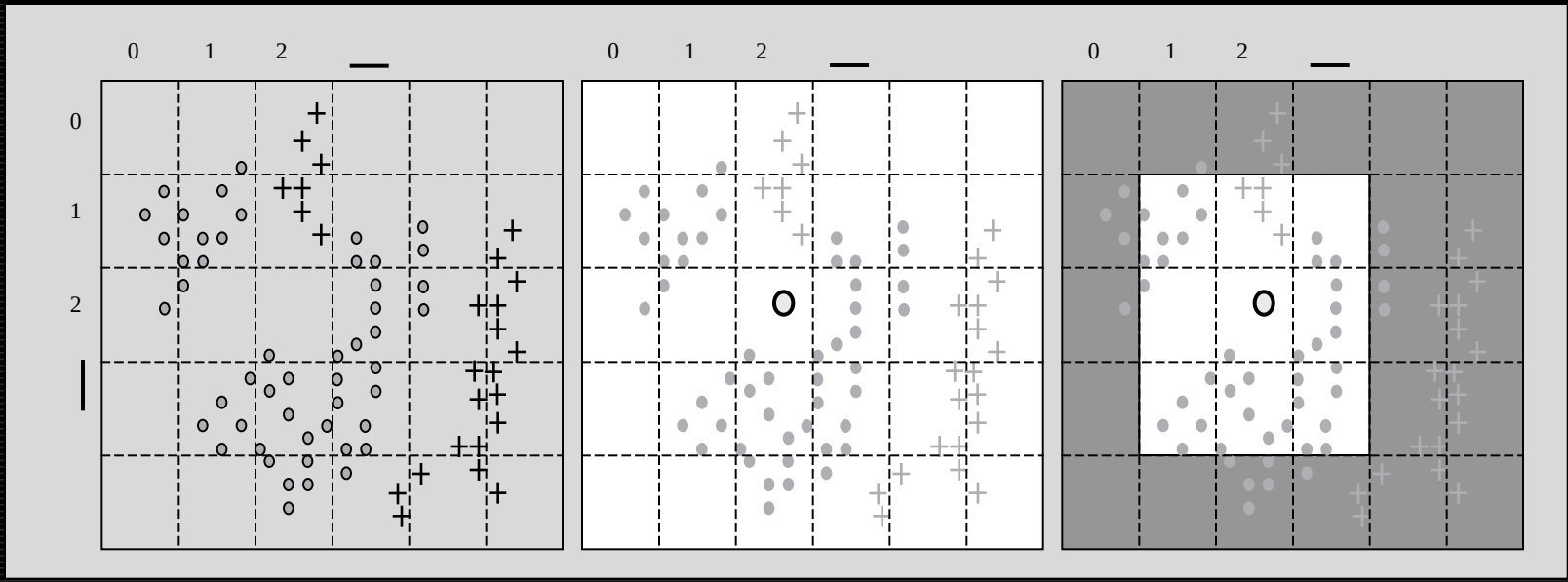
# Metoda k-NN

- **Właściwości**
  - Prosty trening (wybór k dającego najlepszą skuteczność rozpoznawania na zbiorze treningowym)
  - Arbitralne kształty powierzchni separujących klasy: możliwość rozwiązania problemów separowalnych nieliniowo (trudnych)
  - Mała wrażliwość na błędne przykłady (tym mniejsza im większe k)
  - Złożoność obliczeniowa
  - Duża zajętość pamięci przez modele klas

- **Przyśpieszanie metody k-NN**
  - Motywacja prac: duża skuteczność metody
  - Sposób realizacji: indeksowanie próbek i odpowiednie zawężanie zbioru testowanych kandydatów
  - Metody: grupowanie próbek, drzewa k-wymiarowe

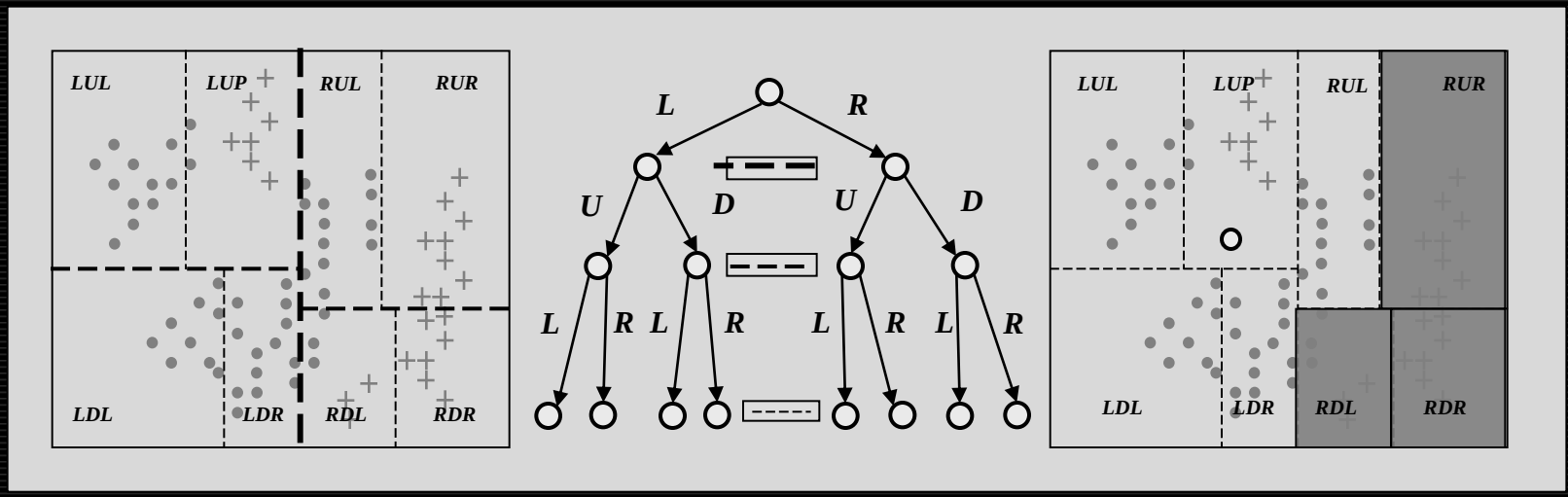
# Metoda k-NN

- **Grupowanie**
  - Zgrubna kwantyzacja przestrzeni cech (hipersześciany)
  - Etykieta próbki: indeks hipersześcianu
  - Krok 1 klasyfikacji: określ indeks hipersześcianu zawierającego sprawdzaną próbkę
  - Obliczaj odległości tylko do prototypów zawartych wewnątrz znalezionego hipersześcianu i jego sąsiadów



# Metoda k-NN

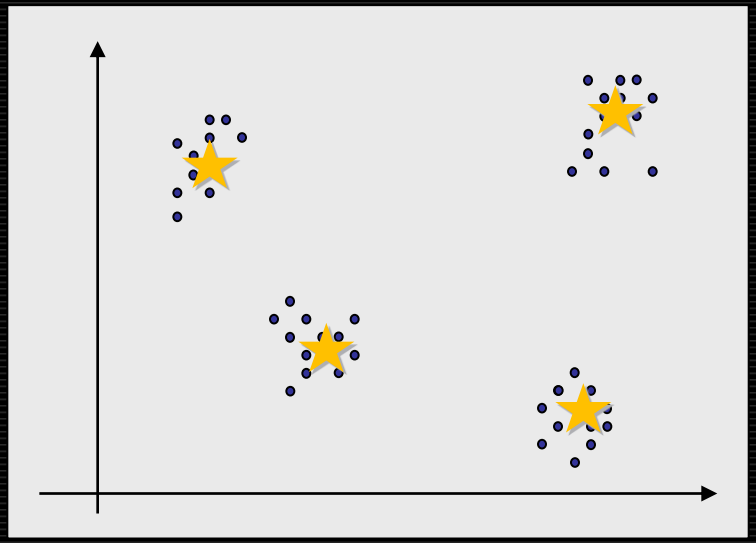
- **Drzewa k-wymiarowe**
  - Zgrubna kwantyzacja przestrzeni cech (adaptacyjna)
  - Określanie hiperpłaszczyzn dzielących zbiory na równe części (głębokość procedury podziału: k)
  - Przypisywanie prototypom etykiet obszarów
  - Sprawdzanie odległości tylko dla próbek z obszarów przyległych




- **Właściwości**
  - Efektywność obliczeniowa

# Metoda k-NM

- **Definicja komponentów metody**
  - Model klasy: podstawowe właściwości statystyczne modów klasy
  - Odległość próbki od klasy: odległość do najbliższego modu
  - Parametr klasyfikacji: k



Prototyp klasy: zbiór modów reprezentowanych przez parametry statystyczne

$$k = \arg\left( \min_i \left\{ d(\mathbf{p}, M_i^j) \right\} \right), \qquad M_i^j = \frac{1}{N_i} \sum_{j=1}^{N_i} C_i^j,\ j = 1 \ldots m$$

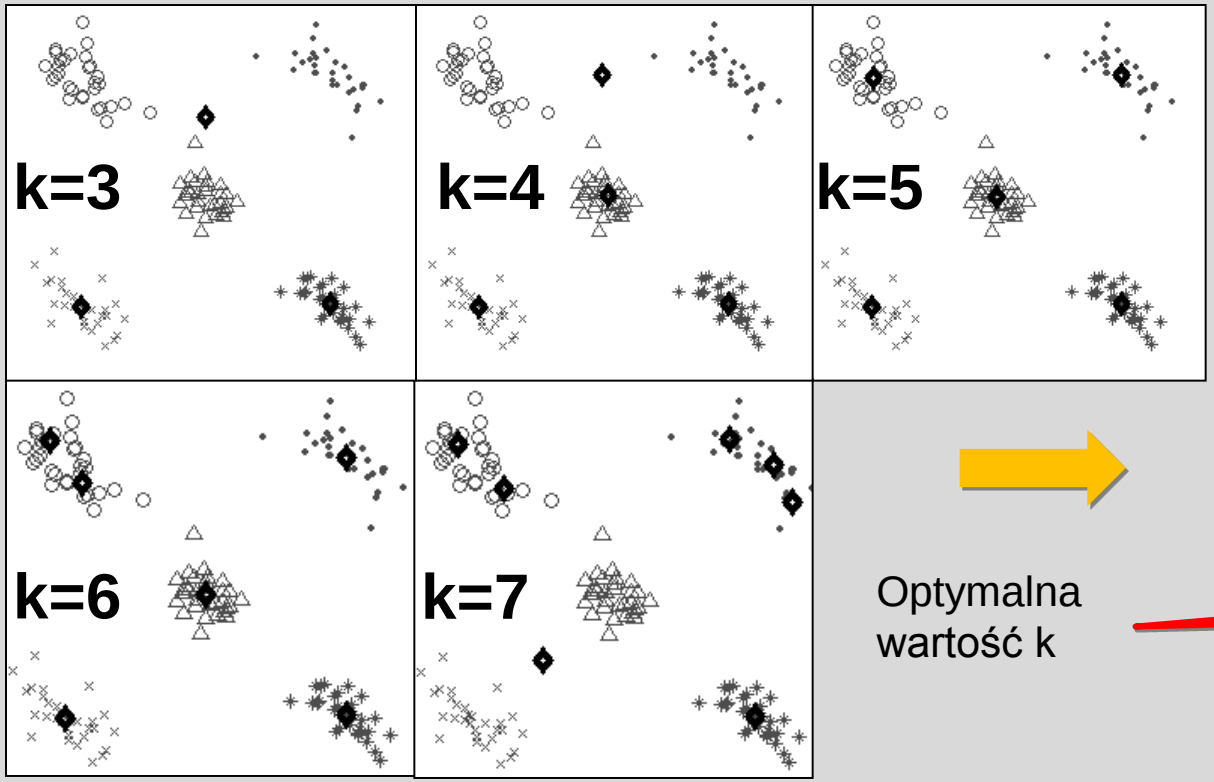
# Metoda k-NM

- **Trening klasyfikatora**
  - Określenie modów dla każdej z klas
  - Liczba modów zwykle nieznana z góry (musi być odkryta przez procedurę)

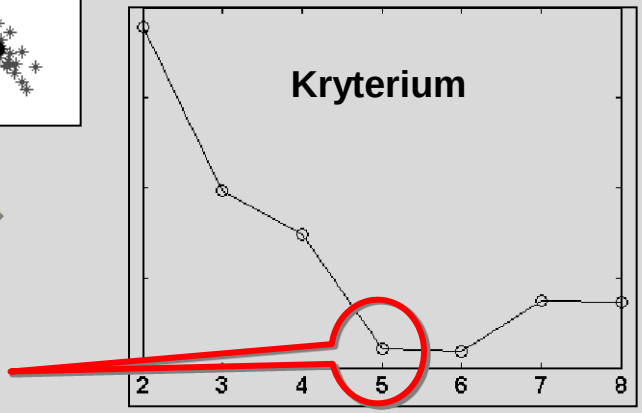
- **Algorytm k-średnich**
  - Iteracyjne powtarzanie dwóch faz:
    - Przypisanie próbki do modu (kandydata)
    - Reestymacja położeń modów
  - Do uzyskania zbieżności
  - Kryterium: średnia odległość próbek od modów
  - Powtarzanie procesu dla kolejnych wartości k, wybór k optymalnego

- **Przykład**
  - Założenie: k=2, początkowe parametry modów m1=(0,1) m2 = (1,0)
  - Próbki treningowe: (0,2), (1,1), (2,0),(3,5),(4,4),(5,3)

# Metoda k-NM




Optymalna wartość k




- **Właściwości**
  - Umiarkowanie złożony trening
  - Arbitralne powierzchnie decyzyjne –rozwiązywanie trudnych problemów
  - Mała wrażliwość na złe przykłady, szybka klasyfikacja, małe zasoby

# Gaussian Mixture Models (GMM)

- **Cechy**
  - Rozwinięcie k-NM (można traktować w kategoriach probabilistycznych)
  - Lepsze modelowanie modów (oprócz wartości średniej – informacje o rozrzucie)
  - Mody są reprezentowane funkcjami Gaussa

- **Trening**
  - Algorytm EM - Expectation Maximization (analogiczny do algorytmu k-średnich: dwie naprzemienne fazy)

- **Właściwości metody**
  - Jedna z najskuteczniejszych obecnie metod