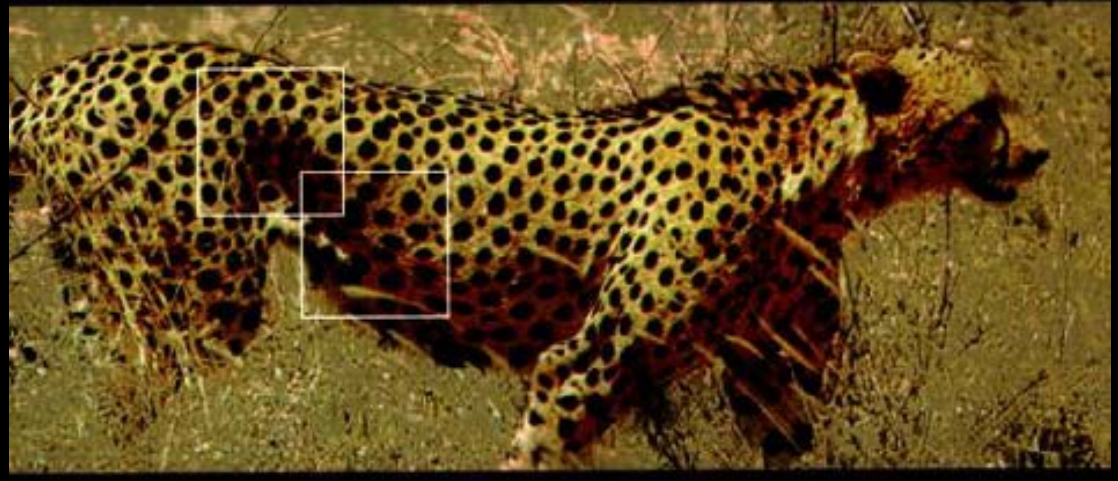


Statistical Pattern Recognition

Second Edition

Andrew Webb

Selekcja cech

- Wprowadzenie
- Metody selekcji cech
  - Miary niepodobieństwa
  - Algorytmy przeszukiwania
- Przykład zastosowania

Wprowadzenie

- Cel selekcji: dobór cech obiektu, na których opierać się będzie klasyfikacja
  - spośród *p* dostępnych pomiarów należy wybrać podzbiór *d* cech, które mają istotny wpływ na wyniki klasyfikacji lub na separowalność zbioru danych

Korzyści

- uproszczenie klasyfikatora (klątwa wymiarowości)
- zwiększenie jakości klasyfikacji i zdolności „uogólniania”
- pozbycie się mało istotnej informacji
- ułatwienie graficznej wizualizacji zbioru danych

Określenie kryterium jakości selekcji *J* prowadzi do zadań optymalizacji:

- ze zbioru $\mathcal{X}_d$ wszystkich podzbiorów o rozmiarze *d* zbioru *p* pomiarów, wybrać taki podzbiór $\tilde{X}_d$, dla którego

$$J\left(\tilde{X}_d\right) = \max_{X \in \mathcal{X}_d} J\left(X\right)$$

Oznaczenia

- Wektory średnich z próby

$$\mathbf{m}_i = \frac{1}{n_i} \sum_{j=1}^{n} z_{ij} \mathbf{x}_j$$

– średnia z próby dla klasy ω_i

gdzie $z_{ij} = \begin{cases} 1 & \text{dla} \quad \mathbf{x}_j \in \omega_i \\ 0 & \text{w przec. przyp} \end{cases}$ i $n_i = \sum_{j=1}^{n} z_{ij}$

$$\mathbf{m} = \sum_{i=1}^{C} \frac{n_i}{n} \mathbf{m}_i$$

– średnia dla całego zbioru

- Macierze kowariancji

$\boldsymbol{\Sigma}$ – macierz kowariancji dla całego zbioru

$\boldsymbol{\Sigma}_i$ – macierz kowariancji dla klasy ω_i

Estymatory największej wiarygodności:

$$\hat{\boldsymbol{\Sigma}} = \frac{1}{n}\sum_{j=1}^{n}\left(\mathbf{x}_j - \mathbf{m}\right)\left(\mathbf{x}_j - \mathbf{m}\right)^T$$

$$\hat{\boldsymbol{\Sigma}}_i = \frac{1}{n_i}\sum_{j=1}^{n} z_{ij}\left(\mathbf{x}_j - \mathbf{m}_i\right)\left(\mathbf{x}_j - \mathbf{m}_i\right)^T$$

Nieobciążony estymator kowariancji ma postać $\frac{n}{n-1}\hat{\boldsymbol{\Sigma}}$

- Macierze rozproszenia

$$\mathbf{S}_W = \sum_{i=1}^{C} \frac{n_i}{n} \hat{\mathbf{\Sigma}}_i$$

– macierz rozproszenia wewnątrz klas

$$\mathbf{S} = \frac{n}{n-C} \mathbf{S}_W$$

– nieobciążony estymator macierzy $\mathbf{S}_W$

$$\mathbf{S}_B = \sum_{i=1}^{C} \frac{n_i}{n} (\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^T$$

– macierz kowariancji między klasami

Zauważmy, że: $\mathbf{S}_B + \mathbf{S}_W = \hat{\mathbf{\Sigma}}$

Metody selekcji cech

- Sformułowanie problemu
  - Dane:
    - zbiór pomiarów *p* cech
    - kryterium jakości, oceniające separowalność zbioru
  - Szukane: najlepszy (w sensie przyjętego kryterium) podzbiór *d* cech
- Przestrzeń rozwiązań

  Liczba możliwych podzbiorów cech: $$n_d = \binom{p}{d} = \frac{p!}{(p-d)!d!}$$

Przykładowo, wybierając 10 cech spośród 25, mamy 3 268 760 możliwości.

- Metody oceny podzbioru cech
  - Metody filtrujące (*filters approaches*): rankingi cech oparte na statystykach
  - Inne metody niezależne od klasyfikatora: np. oparte o pokrycie (overlap) rozkładów danych lub o teorię informacji
  - Metody opakowujące (*wrapper aproach*): klasyfikator jest czarną skrzynką służącą do oceny cech na podstawie wyników klasyfikacji
  - Metody zagnieżdżające (*embedded aproach*): selekcja cech z jednoczesnym uczeniem klasyfikatora

- Miary niepodobieństwa

Jeżeli d_{rs} jest niepodobieństwem obiektów ***s*** i ***r***, to:

1) $d_{rs} \geq 0$ dla wszystkich ***r***, ***s***

2) $d_{rr} = 0$ dla każdego ***r***

3) $d_{rs} = d_{sr}$ dla wszystkich ***r***, ***s***

Jeżeli dodatkowo spełniony jest warunek:

4) $d_{rt} + d_{ts} \geq d_{rs}$ dla wszystkich ***r***, ***s***, ***t*** (nierówność trójkąta)

to miara niepodobieństwa jest metryką.

- Miary podobieństwa

Miarę niepodobieństwa d_{rs} można przekształcić na miarę podobieństwa s_{rs} , na przykład:

$$s_{rs} = \frac{1}{1 + d_{rs}}$$

lub

$$s_{rs} = c - d_{rs}$$, gdzie c jest stałą

- Miary niepodobieństwa pomiędzy wektorami

Wektory cech: $\boldsymbol{x} = [x_1, x_2, \ldots, x_p]^T$, $\boldsymbol{y} = [y_1, y_2, \ldots, y_p]^T$

- Odległość Euklidesowa $d = \sqrt{\sum_{i=1}^{p} (x_i - y_i)^2}$
- Odległość „Manhattan” $d = \sum_{i=1}^{p} |x_i - y_i|$
- Odległość Czebyszewa $d = \max_i |x_i - y_i|$
- Odległość Minkowskiego $d = \left\{ \sum_{i=1}^{p} (x_i - y_i)^m \right\}^{\frac{1}{m}}$, m - rząd
- Odległość Mahalanobisa $d = \sqrt{(\mathbf{x} - \mathbf{y}) \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \mathbf{y})}$

- Miary niepodobieństwa pomiędzy rozkładami

Miara niepodobieństwa $J(\omega_1, \omega_2)$ rozkładów klas spełnia warunki:

1) $J = 0$ dla identycznych rozkładów, tj. dla $p(x|\omega_1) = p(x|\omega_2)$

2) $J \geq 0$

3) J osiąga maksimum, gdy klasy są rozłączne, tj. gdy $p(x|\omega_1) = 0$ oraz $p(x|\omega_2) \neq 0$

- Empiryczne – średni stopień separowalności

Dla n_1 obiektów z klasy ω_1 i n_2 obiektów z klasy ω_2 :

$$J(\omega_1, \omega_2) = \frac{1}{n_1 n_2} \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} d(\mathbf{x}_i, \mathbf{y}_j)$$

- Miary niepodobieństwa pomiędzy rozkładami
  - Oparte o rozkłady warunkowe – dywergencja Kullbacka-Leiblera

$$J(\omega_1,\omega_2)=\int[p(\mathbf{x}\mid\omega_1)-p(\mathbf{x}\mid\omega_2)]\log\left(\frac{p(\mathbf{x}\mid\omega_1)}{p(\mathbf{x}\mid\omega_2)}\right)d\mathbf{x}$$

Dla rozkładów normalnych ze średnimi $\boldsymbol{\mu}_1$, $\boldsymbol{\mu}_2$ i macierzami kowariancji $\boldsymbol{\Sigma}_1$, $\boldsymbol{\Sigma}_2$ dywergencja przyjmuje postać:

$$J=\frac{1}{2}(\boldsymbol{\mu}_2-\boldsymbol{\mu}_1)^T\left(\boldsymbol{\Sigma}_1^{-1}+\boldsymbol{\Sigma}_2^{-1}\right)(\boldsymbol{\mu}_2-\boldsymbol{\mu}_1)+\mathrm{Tr}\left\{\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\Sigma}_2+\boldsymbol{\Sigma}_2^{-1}\boldsymbol{\Sigma}_1-2\mathbf{I}\right\}$$

Jeżeli macierze kowariancji są równe, tj. $\boldsymbol{\Sigma}_1=\boldsymbol{\Sigma}_2=\boldsymbol{\Sigma}$, to dywergencja ma postać identyczną z odległością Mahalanobisa:

$$J=(\boldsymbol{\mu}_2-\boldsymbol{\mu}_1)^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_2-\boldsymbol{\mu}_1)$$

- Miary niepodobieństwa pomiędzy rozkładami
  - Przypadek wieloklasowy:

$$J = \sum_{i=1}^{C} \sum_{j=1}^{C} p(\omega_i)\, p(\omega_j)\, J\left(\omega_i, \omega_j\right)$$

$$J = \sum_{i<j} p(\omega_i)\, p(\omega_j)\, J\left(\omega_i, \omega_j\right)$$

$$J = \max_{i,j(i \neq j)} J\left(\omega_i, \omega_j\right)$$

  - Miary niepodobieństwa mogą być wyznaczane w sposób rekurencyjny. Pozwala to ograniczyć nakłady obliczeniowe dla algorytmów przeszukiwania przestrzeni podzbiorów cech.

Algorytmy przeszukiwania przestrzeni podzbiorów cech

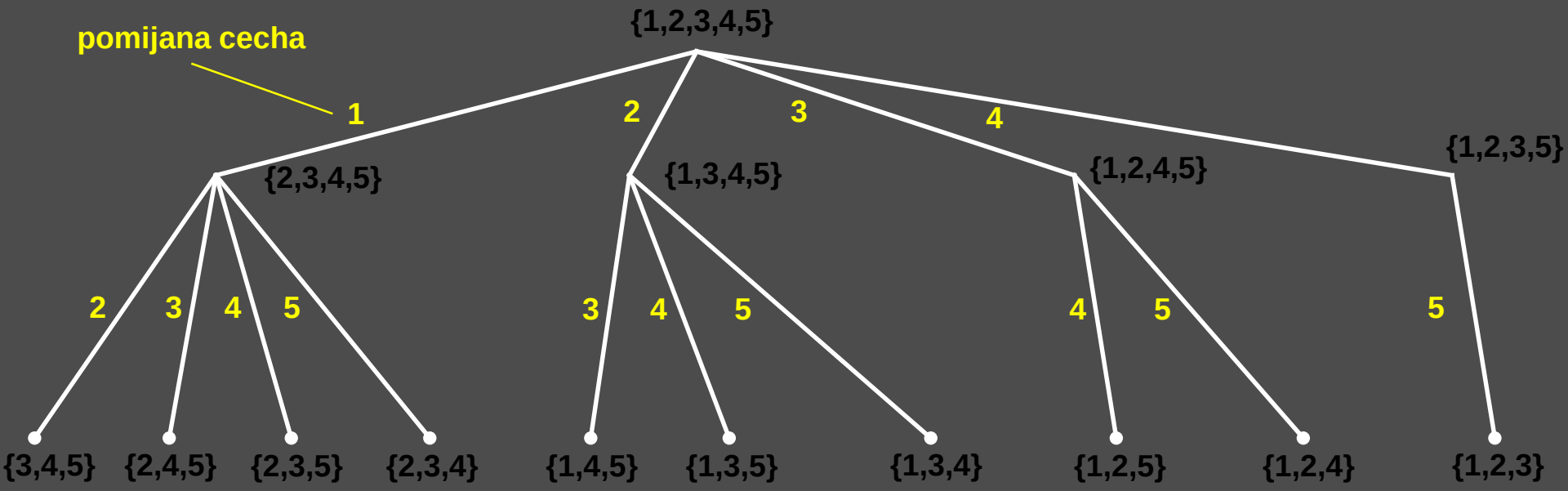
- Dziel i zwyciężaj (*branch and bound*)
- Metody heurystyczne (suboptymalne)
  - *N* najlepszych cech
  - SFS – *Sequential Forward Selection*
  - SBS – *Sequential Backward Selection*
  - Metoda „dodaj *l*, odrzuć *r*”
  - *Floating Search Methods*
- Metody randomizowane
  - Symulowane wyżarzanie
  - Algorytm genetyczny

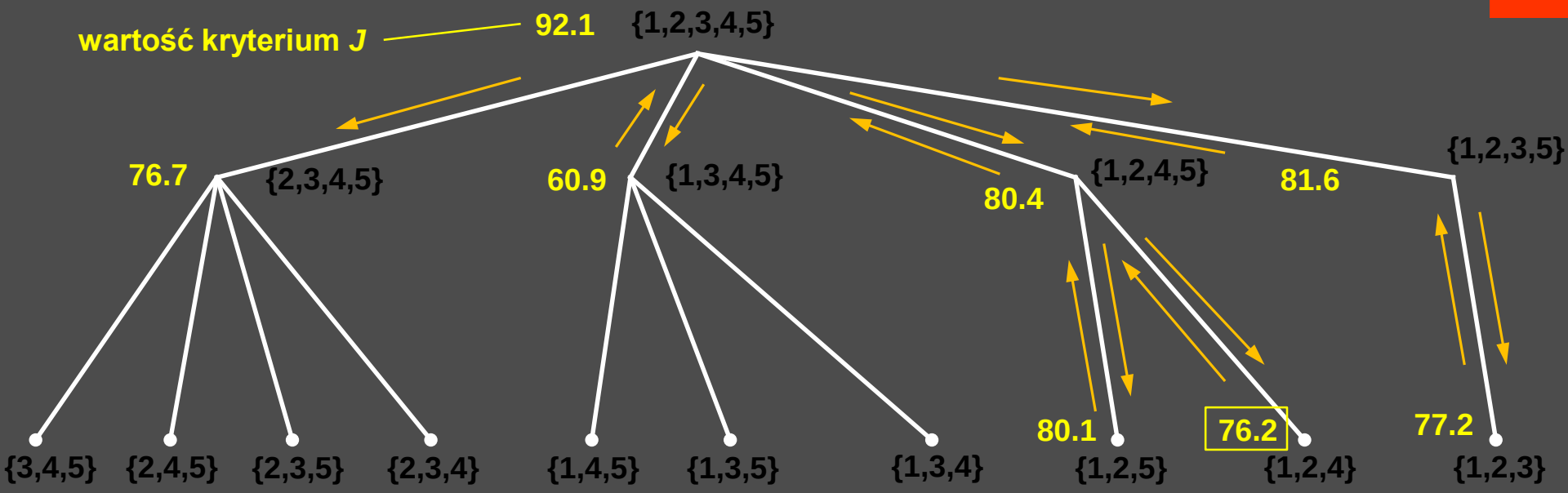
Dziel i zwyciężaj (*branch and bound*)

Dla dwóch podzbiorów cech zachodzi: $X \subset Y \Rightarrow J(X) < J(Y)$

Można tę własność wykorzystać do dokładnego przeszukiwania przestrzeni podzbiorów cech, bez konieczności sprawdzania wszystkich rozwiązań.

Przykład: wybór zestawu 3 najlepszych cech spośród 5-ciu.





- Przeszukiwanie drzewa rozpoczynamy od strony zawierającej mniej rozgałęzień
- Zapamiętujemy największą wartość kryterium z odwiedzonych liści
- Jeżeli wartość kryterium w odwiedzanym węźle jest mniejsza od zapamiętanej, odrzucamy tę gałąź drzewa
- Najczęściej stosowanym kryterium jest odległość Mahalanobisa, które dla dwóch klas przyjmuje postać:

$$J = (\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1)^T \left(\frac{\boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_2}{2} \right)^{-1} (\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1)$$

N najlepszych cech

1) Wyznaczenie wartości kryterium jakości dla każdej cechy z osobna
2) Uszeregowanie cech tak, aby:

$$J(x_1) \geq J(x_2) \geq \cdots \geq J(x_p)$$

3) Wybór *N* pierwszych cech

- Algorytm nie uwzględnia zależności pomiędzy cechami

SFS – *Sequential Forward Selection*

- *n*-ta iteracja algorytmu:

1) Dany jest podzbiór *k* cech – X_k (w kroku $n = 0$ podzbiór X_k jest pusty)

2) Dla każdej z $p - k$ pominiętych cech ξ_j wyznacz wartość kryterium

$$J_j = J\left(X_k \cup \xi_j\right)$$

3) Utwórz nowy podzbiór X_{k+1} dodając do podzbioru X_k tę cechę, dla której wartość kryterium J_j jest największa (krok w przód)

4) Zakończ algorytm, gdy *k* przekracza przyjęte maksimum lub gdy dodanie kolejnej cechy pogarsza wartość kryterium: $J(X_{k+1}) < J(X_k)$

- Cechy mogą być tylko dodawane, nie usuwane

GSFS – *Generalized Sequential Forward Selection*

- W pojedynczej iteracji algorytmu do zbioru X_k może być dodanych *l* cech spośród pozostałych *p – k*
- Liczba podzbiorów *l* cech – Y_l , jakie mogą być dodane do zbioru X_k jest równa

$$\binom{p-k}{l}$$

i dla nich wszystkich należy wyznaczyć wartość kryterium $J(X_k \cup Y_l)$

- Koszt obliczeniowy algorytmu GSFS jest większy niż SFS, ale w zamian uwzględnia on częściowo związki pomiędzy cechami

SBS – *Sequential Backward Selection*

- *n*-ta iteracja algorytmu:

1) Dany jest podzbiór *k* cech – X_k (w kroku *n=0* zawiera on wszystkie cechy)

2) Dla każdej z *k* cech ξ_j z bieżącego podzbioru X_k wyznacz wartość kryterium $J_j = J\left(X_k \setminus \xi_j\right)$

3) Utwórz nowy podzbiór X_{k+1} usuwając z podzbioru X_k tę cechę, dla której wartość kryterium J_j jest największa (krok w tył)

4) Zakończ algorytm, gdy *k* przekracza przyjęte minimum.

- Cechy mogą być tylko usuwane, nie dodawane
- Nakład obliczeniowy jest większy niż w SFS, gdyż starujemy od większych podzbiorów

GSBS – *Generalized Sequential Backward Selection*

- W pojedynczej iteracji algorytmu ze zbioru X_k może być usuniętych *r* cech spośród *k*
- Liczba podzbiorów *r* cech – Y_r , jakie mogą być usunięte zbioru X_k jest równa

$$\binom{k}{r}$$

i dla nich wszystkich należy wyznaczyć wartość kryterium $J(X_k \setminus Y_r)$

- Koszt obliczeniowy algorytmu GSBS jest większy niż SBS, ale w zamian uwzględnia on częściowo związki pomiędzy cechami

Metoda „dodaj *l*, odrzuć *r*”

- W pojedynczej iteracji algorytmu do podzbioru X_k może być dodanych *l* cech spośród pozostałych *p* – *k* oraz usuniętych *r* cech spośród *k* (*l* kroków w przód, *k* kroków w tył)
- Jeżeli *l* > *r* , to algorytm startuje z pustego podzbioru X_k
- Jeżeli *l* < *r* , to algorytm startuje z podzbioru X_k zawierającego wszystkie *p* cech

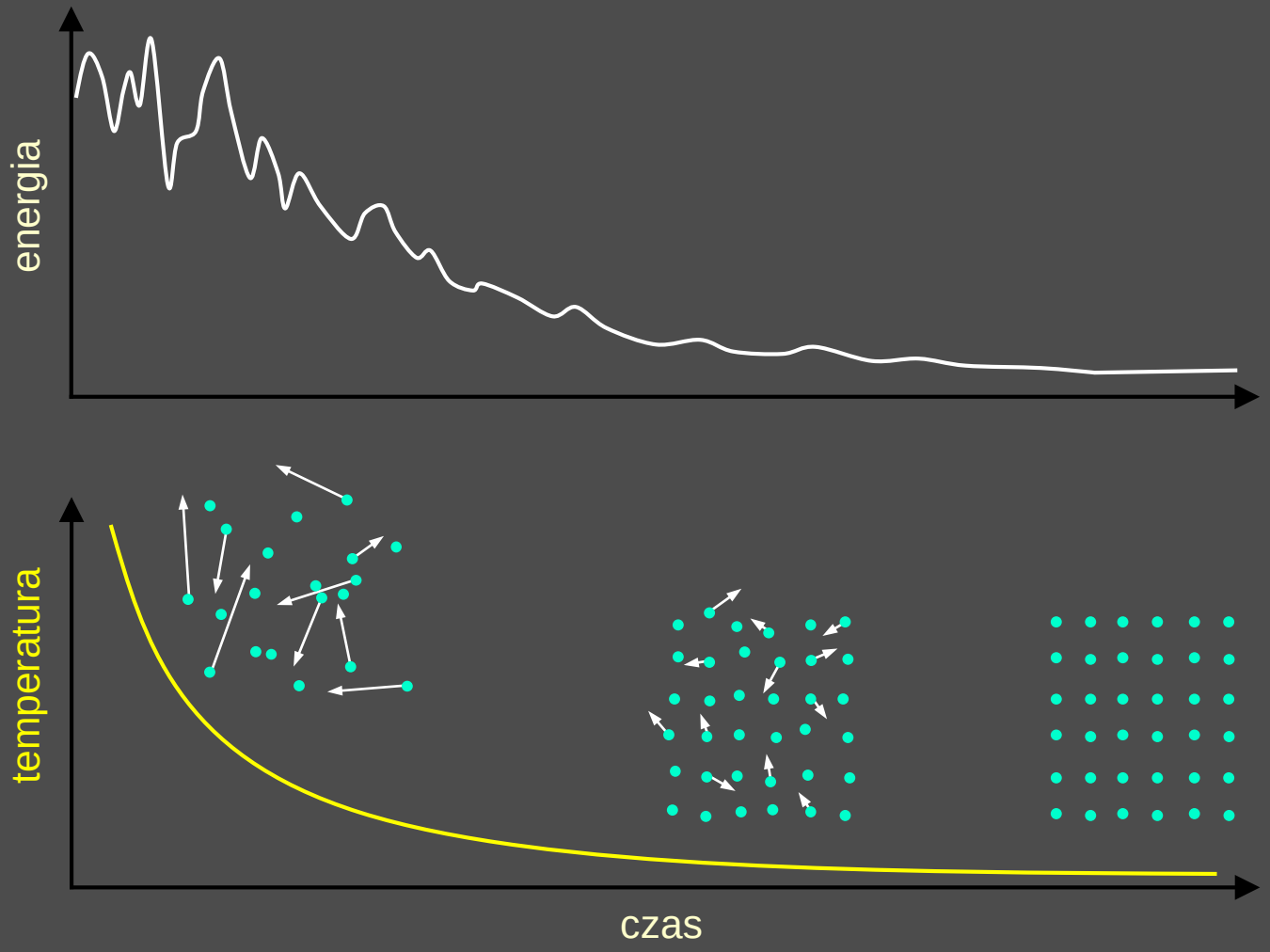
Floating Search Methods

- Wartości *l* oraz *r* mogą być zmieniane w każdej iteracji algorytmu w sposób automatyczny, co prowadzi do algorytmów:
  - SFFS – *Sequential Forward Floating Selection*
  - SBFS – *Sequential Backward Floating Selection*
- Algorytm SFFS: w każdej iteracji, po wykonaniu kroku w przód, algorytm wykonuje sekwencję kroków w tył tak długo, jak kolejno generowane podzbiory są lepiej oceniane od poprzednich
- Algorytm SBFS: w każdej iteracji, po wykonaniu kroku w tył, algorytm wykonuje sekwencję kroków w przód tak długo, jak kolejno generowane podzbiory są lepiej oceniane od poprzednich

Symulowane wyżarzanie (*Simulated Annealing*)

- Inspiracją algorytmu jest proces schładzania pewnych materiałów, podczas którego minimalizowana jest energia związana z konfiguracją cząstek
- W dużych temperaturach cząstki szybko zmieniają swoje położenie, przy niższych temperaturach ruch cząstek ulega spowolnieniu
- Obniżanie temperatury z odpowiednim tempem prowadzi do uzyskania struktury krystalicznej odpowiadającej minimalnej energii

Symulowane wyżarzanie



Symulowane wyżarzanie

- Algorytm rozpoczyna działanie od losowo wybranego podzbioru cech i wysokiej „temperatury”
- W kolejnych iteracjach temperatura obniża się. Ponadto do podzbioru cech wprowadzane są przypadkowe zmiany.
- Wielkość zmian wprowadzanych do podzbioru cech zależy od temperatury: im wyższa temperatura, tym większe zmiany mogą wystąpić
- Jeżeli zmieniony podzbiór cech jest oceniony jako lepszy, zmiana zostaje zaakceptowana
- Jeżeli zmieniony podzbiór cech jest oceniony jako gorszy, zmiana zostaje zaakceptowana z prawdopodobieństwem zależnym od temperatury : im wyższa temperatura, tym wyższe prawdopodobieństwo akceptacji

Symulowane wyżarzanie

- Podzbiór cech może zostać zakodowany w postaci ciągu bitów. Przykładowo, zestaw cech { x_1, x_2, x_5, x_7 } spośród dziesięciu zakodujemy jako 1100101000
- Zmiana zestawu cech jest realizowana przez zmianę wartości bitów
- Przy wysokiej temperaturze jednoczesnej zmianie może ulec wiele bitów
- Jeżeli podzbiór cech *X* można przekształcić w *X* ' w jednym kroku algorytmu, to *X* ' nazywamy sąsiadem *X*
- Rozmiar sąsiedztwa zależy od temperatury: im wyższa temperatura, tym większe sąsiedztwo
- Funkcja celu *Q* ocenia podzbiór cech. Na jej wartość wpływ może mieć kryterium *J* oraz liczba cech. Algorytm poszukuje maksimum funkcji *Q*

Symulowane wyżarzanie - algorytm

1) Wprowadź: X (początkowy podzbiór cech), $Q(X)$ (funkcja celu), $S(X, T)$ (funkcja sąsiedztwa), T_0 (temperatura początkowa), T_N (temperatura końcowa), t (współczynnik zmniejszania temperatury),
2) $X_{best} \leftarrow X, \quad T \leftarrow T_0$
3) Dopóki $T \geq T_0$, wykonuj:
4) $X_{new} \leftarrow S(X_{best}, T)$
5) $\Delta Q \leftarrow Q(X_{best}) - Q(X_{new})$
6) Jeżeli $\Delta Q < 0$ to $X_{best} \leftarrow X_{new}$
6) Jeżeli $\Delta Q \geq 0$ to
7) $X_{best} \leftarrow X_{new}$ z prawdopodobieństwem równym $\exp\left(-\frac{\Delta Q}{T}\right)$
8) $T \leftarrow t \cdot T$

Symulowane wyżarzanie

- Prawdopodobieństwo akceptacji (krok 7 algorytmu) i sposób obniżania temperatury (krok 8 algorytmu) można dobierać dowolnie (krok 8 algorytmu)

- Wysoka temperatura – optymalizacja globalna (niedokładna)
- Niska temperatura – optymalizacja lokalna (dokładna)

Algorytm genetyczny

- Inspiracją algorytmu genetycznego jest proces doboru naturalnego
- Skrajnie upraszczając, w procesie doboru naturalnego „lepsze” geny z większym prawdopodobieństwem dostają się do kolejnych pokoleń osobników
- W algorytmie genetycznym „doborowi naturalnemu” podlegają zakodowane rozwiązania problemu optymalizacyjnego
- Rozwiązania oceniane są przy użyciu tzw. funkcji przystosowania F, która jest monotoniczną funkcją kryterium jakości J.
- Populacja nowych rozwiązań wyznaczana jest na podstawie poprzedniej przy użyciu operatorów selekcji, krzyżowania i mutacji.

Algorytm genetyczny

- Identycznie jak w przypadku symulowanego wyżarzania, rozwiązanie (czyli podzbiór cech) może zostać zakodowane w postaci ciągu bitów
- Rozwiązania są wybierane do tzw. puli rodzicielskiej w drodze losowania ze zwracaniem z uwzględnieniem wartości funkcji przystosowania (operator selekcji)
- Rozwiązania z puli rodzicielskiej są dobierane w pary i krzyżowane (operator krzyżowania), dodając do populacji nowe rozwiązania
- Nowe rozwiązania podlegają losowym modyfikacjom (operator mutacji).

Algorytm genetyczny

1) Wprowadź: q (rozmiar populacji rodzicielskiej), P (początkowa populacja rozwiązań), *Fitness*(X) (funkcja przystosowania), *Selection* (F,P) (operator selekcji), *Crossover (P)* (operator krzyżowania), *Mutation* (X) (operator mutacji), N (liczba iteracji)
2) $n \leftarrow 0$, $X_{best} \leftarrow$ najlepsze rozwiązanie z populacji P
3) Dopóki $n \leq N$, wykonuj:
4) $P_{parents} \leftarrow$ *Selection* (*Fitness* ,P, q)
5) $X \leftarrow$ najlepsze rozwiązanie z populacji $P_{parents}$
6) Jeżeli *Fitness* (X) > *Fitness* (X_{best}) to $X_{best} \leftarrow X$
7) $P_{childrens} \leftarrow$ *Crossover* ($P_{parents}$)
8) Dla wszystkich rozwiązań w $P_{childrens}$ wykonuj
9) $X_{childrens} \leftarrow$ *Mutation* ($X_{childrens}$)
10) $P \leftarrow P_{childrens}$, $n \leftarrow n + 1$

Algorytm genetyczny

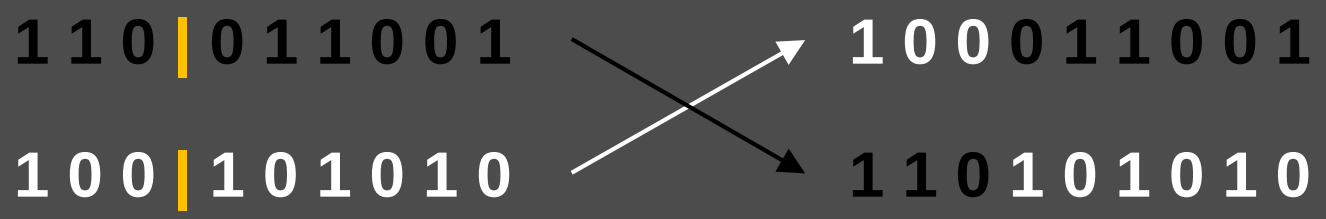
- Opracowano wiele operatorów selekcji. Najprostszy z nich, tj. selekcja proporcjonalna polega na wylosowaniu ze zwracaniem z całej populacji q rozwiązań, przy czym prawdopodobieństwo wylosowania rozwiązania o wartości funkcji przystosowania F_i jest równe

$$\frac{F_i}{\sum_i F_i}$$

- Operator mutacji w najprostszym przypadku polega na losowej zamianie wartości każdego bitu rozwiązania z niewielkim prawdopodobieństwem p_{mut}

Algorytm genetyczny

- Istnieje wiele operatorów krzyżowania. Najprostszy z nich, tj. krzyżowanie jednopunktowe polega na:
  - wylosowaniu z populacji pary rozwiązań
  - losowym wyborze punktu krzyżowania
  - złożenie nowych rozwiązań z obu części każdego z rozwiązań rodzicielskich



Przykład zastosowania

Selekcja cech w systemie rozpoznawania powikłań cukrzycowych

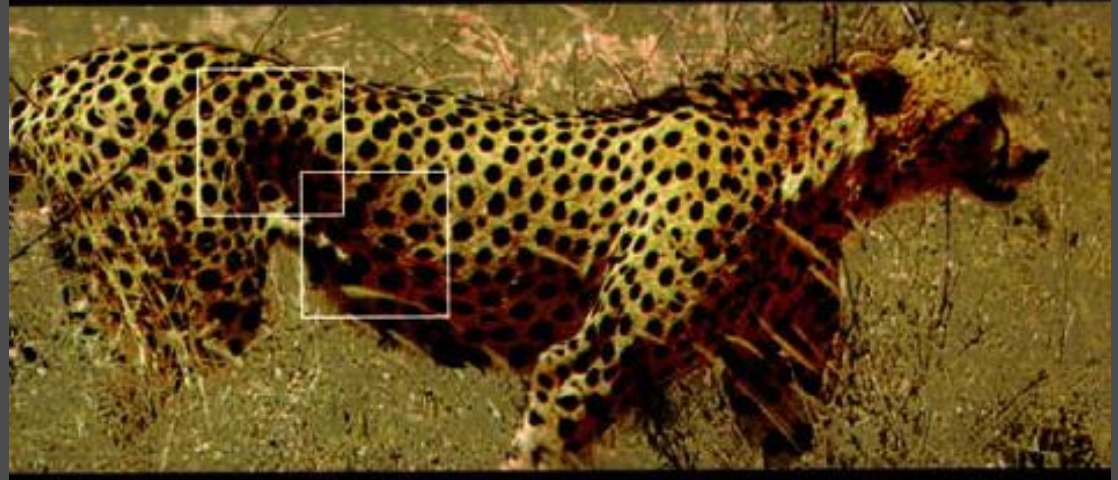
- Cel pracy: selekcja cech na potrzeby predykcji wystąpienia nefropatii (niewydolność nerek)
- Pacjenci:
  - 4321 dorosłych osób z cukrzycą typu 2,
  - Korea, Seul, *Samsung Medical Center*
  - Liczba cech: 184

Feature index	Feature[a]
1–4	Onset age, diabetes duration, age, sex
5–13	White blood cell count (WBC)
14–22	Hemoglobin
23–31	Platelet count
32–40	Serum cholesterol level
41–49	Serum aspartate aminotransferase (AST) level
50–58	Serum alanine aminotransferase (ALT) level
59–67	Serum alkaline phosphatase (ALP) level
68–76	Blood urea nitrogen (BUN)
77–85	Creatinine
86–94	Uric acid
95–103	Na^+
104–112	K^+
113–121	Serum triglycerides level
122–130	High density lipoprotein cholesterol (HDL-C) level
131–139	Low density lipoprotein cholesterol (LDL-C) level
140–148	Glycosylated hemoglobin (HbA_{1C})
149–157	Microalbumin
158–166	Systolic blood pressure (sBP)
167–175	Diastolic blood pressure (dBP)
176–184	Body mass index (BMI)

[a] Each feature set except for 1–4 has 11 features: slope, mean, variance, maximum, minimum, *K*, EST (estimated value on the date of prediction), initial value and latest value.

- Rozwiązanie problemu
  - Wykorzystanie miar niepodobieństwa między wektorami
  - Różne odmiany metod przeszukiwania wstecz (*Backward Selection*)
  - Ocena podzbioru cech za pomocą klasyfikatorów SVM (*Support Vector Machines*)

- Rezultaty
  - Liczba cech po selekcji: 39, pozwoliła na odseparowanie klas nefropatia/ brak nefropatii w 98 %
  - Predykcja wystąpienia objawów nefropatii na 2-3 miesiące przed postawieniem diagnozy przez lekarza



Statistical Pattern Recognition

Second Edition

Andrew Webb