

# Metody selekcji cech

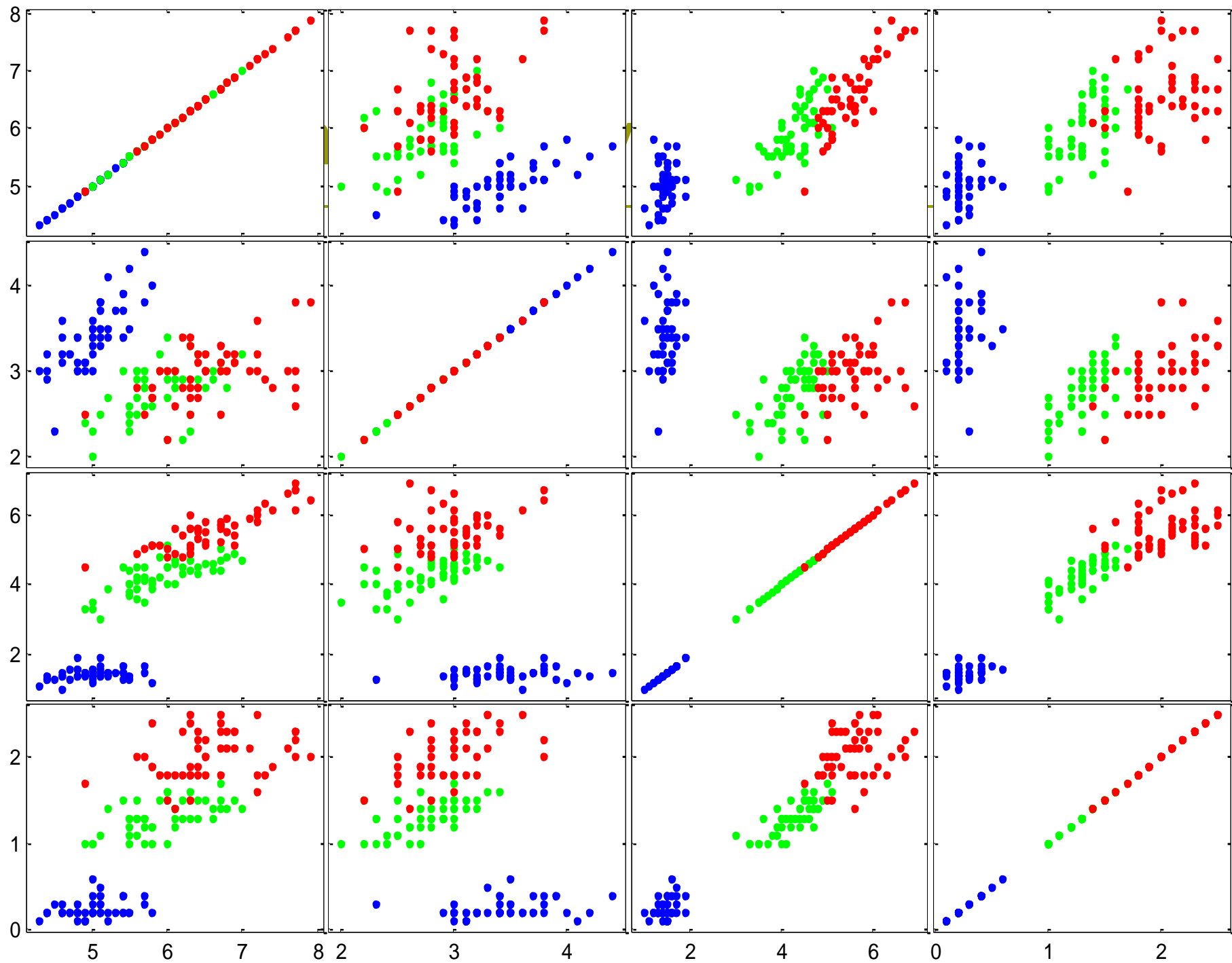


# A po co to

---

- Często mamy do dyspozycji dane w postaci zbioru cech lecz nie wiemy które z tych cech będą dla nas istotne.
- W zbiorze cech mogą wystąpić cechy redundantne – niosące identyczną informację jak istniejące już cechy
- Cel – wybranie ze zbioru dostępnych cech tych które nas „interesują”

Interesujące cechy to takie, których kombinacja pozwala na możliwie najlepszą klasyfikację lub regresję!



# Podział metod selekcji cech

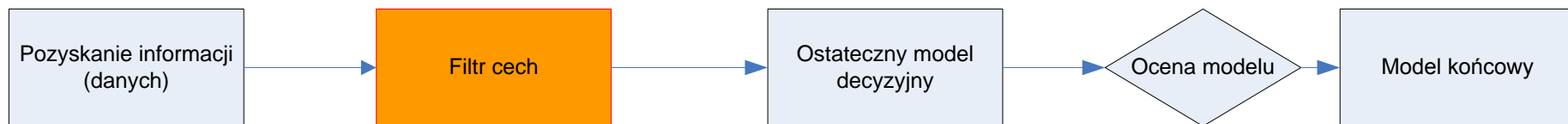
---

- Ze względu na charakter problemu
  - Nadzorowane
  - Nienadzorowane
- Ze względu na relację z innymi algorytmami nadrzędnymi
  - Filtry
  - Wrappery (opakowane)
  - Frapery – kombinacja filtrów i Wrapperów
  - Metody wbudowane

# Filtry cech

---

- Filtry cech to taka grupa metod, która autonomicznie podejmuje decyzję, które z cech będą istotne dla późniejszego procesu uczenia. Decyzja ta podejmowana jest na podstawie niezależnego od klasyfikatora współczynnika takiego jak „informacja wzajemna” lub „dywergencja Kullbacka–Leiblera”



# Informacja wzajemna i dywergencja Kullbacka–Leiblera

---

## □ Informacja wzajemna (ang. Mutual Information)

- Dla zmiennych o rozkładach dyskretnych  $p(x)$  i  $q(y)$   
$$I(X; Y) = \sum_{y \in Y} \sum_{x \in X} p(x, y) \log \left( \frac{p(x, y)}{p_1(x) p_2(y)} \right).$$

- Dla zmiennych o rozkładach ciągłych  $p(x)$  i  $q(y)$   
$$I(X; Y) = \int_Y \int_X p(x, y) \log \left( \frac{p(x, y)}{p_1(x) p_2(y)} \right) dx dy.$$

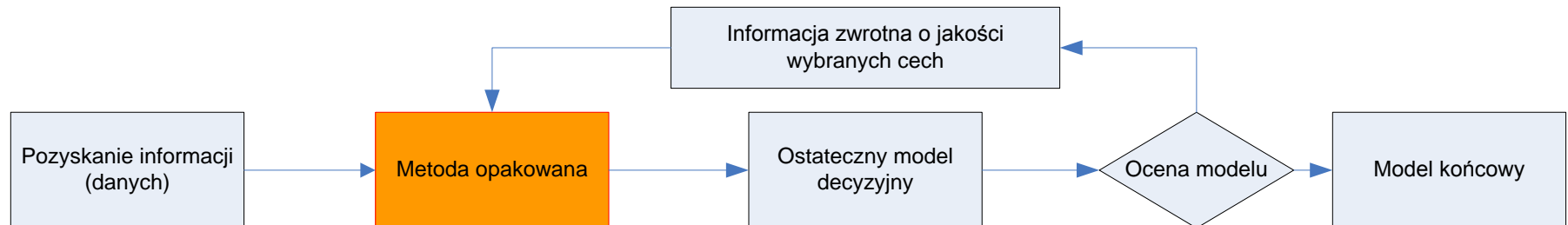
## □ Dywergencja Kullbacka–Leiblera (ang. Kullback–Leibler divergence)

- Dla zmiennych o rozkładach dyskretnych  $p(x)$  i  $q(x)$   
$$D_{\text{KL}}(P \parallel Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)}$$

$$D_{\text{KL}}(P \parallel Q) = \int_{-\infty}^{\infty} p(x) \log \frac{p(x)}{q(x)} dx.$$

# Metody opakowane

- Metody opakowane to grupa metod w której występuje sprzężenie zwrotne pomiędzy elementem decyzyjnym (np.. Siecią neuronową) a algorytmem selekcji cech. Dzięki temu podzbiór cech optymalizowany jest pod kątem konkretnego klasyfikatora



# Metody filtrów

---

## □ Zalety

- Uniwersalność – uzyskany podzbiór cech jest niezależny od klasyfikatora, dzięki czemu teoretycznie możemy użyć dowolny klasyfikator
- W problemach medycznych jak analiza DNA zależy nam na znalezieniu genów odpowiedzialnych za pewne cechy, nie chcemy by wynik był zależny od użytej sieci neuronowej
- Szybkość – jesteśmy niezależni od metody klasyfikacyjnej dzięki czemu złożoność obliczeniowa nie wpływa na szybkość i wydajność tego algorytmu
- Uniwersalność - algorytm tego typu może być wykorzystany do każdego problemu klasyfikacyjnego

## □ Wady

- Konieczność estymacji wielowymiarowych rozkładów prawdopodobieństwa



# Metody opakowane

---

## □ Zalety

- Wybrany podzbiór cech jest dostosowany do wymagań lub charakteru algorytmu decyzyjnego (sieci neuronowej itp)
- Większa dokładność niż metod filtrów
- Uniwersalność - algorytm tego typu może być wykorzystany do każdego problemu klasyfikacyjnego

## □ Wady

- Często większa złożoność obliczeniowa

# Kombinacje filtrów i metod opakowanych

---

- Wykorzystuje się algorytm filtrów do selekcji cech, jednakże parametry filtru dostraja się na podstawie metody opakowującej.
- Właściwości
  - Szybkość
  - Często większa dokładność niż metod filtrów, lecz mniejsza niż metod opakowanych
  - Uniwersalność - algorytm tego typu może być wykorzystany do każdego problemu klasyfikacyjnego

# Metody wbudowane

---

- Metody wbudowane to taka grupa algorytmów które wykorzystują pewne cechy algorytmów uczenia dokonując automatycznej selekcji cech na etapie uczenia sieci neuronowej lub innego algorytmu decyzyjnego
- Właściwości
  - Szybkość – selekcja cech realizowana jest podczas procesu uczenia dzięki czemu nie musimy dokonywać żadnych dodatkowych obliczeń
  - Dokładność – metody te są zaprojektowane pod kątem konkretnego algorytmu
  - Brak uniwersalności – metody te można wykorzystywać jedynie dla danego algorytmu

# Algorytmy przeszukiwania



# Selekcja w przód

---

- Startuje z pustego zbioru cech i stopniowo zwiększ jego rozmiar aż do osiągnięcia maksimum funkcji oceniającej (np.. MI lub max. dokładność sieci neuronowej)
- Złożoność obliczeniowa:  $n^2$

```
Require: f
f' ← ∅
n ← numof(f)
repeat
  chk ← true
  for i = 1 ... n do
    tacc ← oceń(f' ∪ fi)
    if tacc > acc then
      acc ← tacc
      j ← i
      chk ← false
    end if
  end for
  if not chk then
    f' ← f' ∪ fi
    f ← f / fi
  end if
  n ← numof(f)
until chk ∨ n = 0
return f'
```

---

# Selekcja w tył

- Startuje z pełnego zbioru cech i stopniowo usuwa z niego najmniej użyteczne cechy aż do osiągnięcia maksimum funkcji oceniającej (np.. MI lub max. dokładność sieci neuronowej)
- Złożoność obliczeniowa:  $n^2$

```
Require: f
f' ← f
n ← numof(f)
repeat
  chk ← true
  for i = 1 ... n do
    tacc ← oceń(f'/fi)
    if tacc > acc then
      acc ← tacc
      j ← i
      chk ← false
    end if
  end for
  if not chk then
    f' ← f'/fi
    f ← f/fi
  end if
  n ← numof(f)
until chk ∨ n = 0
return f'
```

# Inne

---

- Selekcja z wykorzystaniem algorytmów genetycznych
- Selekcja typu „i” w przód „j” w tył i odwrotna
  - Dadajemy algorytmem selekcji w przód „i” nowych cech po czym z rezultatu usuwamy „j” najmniej użytecznych. Warunek  $i > j$
  - Z całego podzbioru usuwamy „i” najmniej istotnych cech, po czym dodajemy „j” nowych. Warunek  $i > j$

# Metody rankingowe





# Metody rankingowe

---

- Metody rankingowe są bardzo wydajnymi (szybkimi) metodami selekcji cech
- Stosowane są jako filtry – wówczas należy podać liczbę wybranych cech jako wejście algorytmu
- Stosowane jako frappers (kombinacja metod filtrów i opakowanych) – wówczas liczba wybranych cech optymalizowana jest przez algorytm decyzyjny (np.. Sieć neuronową)
- Wady brak stabilności

# Metody rankingowe - algorytm

---

```
Require:  $f$  {wejściowy zbiór cech}
Require:  $J(\cdot)$  {funkcja rankingowa}
  for  $i = 1 \dots n$  do
     $a_i \leftarrow J(f_i)$  {ocień i-tą cechę}
  end for
 $f \leftarrow \text{sort}(f, a)$  {sortuj cechy wg. istotności}
 $acc \leftarrow 0$ 
 $f^a \leftarrow \emptyset$ 
  for  $j = 1 \dots n$  do
     $f^a = f^a \cup f_i$  {dodaj do podzbioru  $f^a$  nową cechę}
     $tacc \leftarrow \text{ocień}(f^a)$  {ocień podzbiór cech  $f^a$  }
    if  $tacc > acc$  then
       $acc \leftarrow tacc$  {jeżeli nowy podzbiór jest lepszy od poprzedniego }
       $f' \leftarrow f^a$  {zapamiętaj wynik aktualny podzbiór}
    end if
  end for
return  $f'$ 
```

# Współczynniki rankingowe

---

- Znormalizowany zysk informacji (ang. Normalized information gain) lub asymetryczny współczynnik zależności (ang. asymmetric dependency coefficient, ADC)

$$ADC(C, f) = \frac{MI(C, f)}{H(C)}$$

- znormalizowany względny zysk informacyjny (ang. normalized gain ratio)

$$U_S(C, f) = \frac{MI(C, f)}{H(f)}$$

- kryterium *DML*

$$D_{ML}(f_i, C) = H(f_i|C) + H(C|f_i)$$

- *Gdzie:*

$H(c)$  – entropia klasy

$H(f)$  – entropia cechy

$MI(c, f)$  – informacja wzajemna

*Pomiędzy cechą i klasą*

$$H(C) = -\sum_{i=1}^c p(C_i) \lg_2 p(C_i)$$

$$H(f) = -\sum_x p(f = x) \lg_2 p(f = x)$$

$$MI(C, f) = -H(C, f) + H(C) + H(f)$$